TARK 2013

Theoretical Aspects of Rationality and Knowledge
Proceedings of the 14th Conference – Chennai, India

Edited by Burkhard C. Schipper

Proceedings of the 14th Conference on

Theoretical Aspects of Rationality and Knowledge

TARK 2013

January 7 to 9, 2013

Institute of Mathematical Sciences, Chennai

Edited by Burkhard C. Schipper

ISBN number 978-0-615-74716-3

Burkhard C. Schipper
University of California, Davis
Department of Economics
One Shields Avenue
Davis, CA 95616
USA

Table of Contents

Foreword

Ten years ago, I had the privilege to attend my first TARK conference. It was the 9th TARK conference held at Indiana University, Bloomington. At that time I was a doctoral student of economics interested in modeling subjective perceptions of players in games. I presented a paper on unawareness. My first TARK conference was an educational experience similar to an infant discovering language. I was "bathing in the sound" of TARK; see my cover design of the proceedings. This year, the sound of TARK will be mixed with the sound of India. For the first time in TARK's history, the conference is held in India. We are extremely grateful for the hospitality and support of the Institute of Mathematical Sciences, Chennai. Chennai is famous for its Indian Music Festival, and I hope we get a flavor of it. The conference is held in winter (January 7 - 9, 2013); another first time for TARK.

TARK conferences are truly interdisciplinary bringing together researchers from a wide variety of fields, including Artificial Intelligence, Cryptography, Distributed Computing, Economics and Game Theory, Linguistics, Philosophy, and Psychology, in order to further our understanding of interdisciplinary issues involving reasoning about rationality and knowledge. This year we had 64 submissions out of which 18 were accepted as contributed talks and 8 as poster presentations for the program. I am very grateful for working with the other 16 members of the multidisciplinary program committee: Samson Abramsky (Oxford University), Thomas Agotnes (Universitetet i Bergen), Hans van Ditmarsch (University of Sevilla), Amanda Friedenberg (Arizona State University), Aviad Heifetz (The Open University of Israel), Jérôme Lang (CNRS and Universite Paris-Dauphine), Fenrong Liu (Tsinghua University, Beijing), Larry Moss (Indiana University, Bloomington), Antonio Penta (University of Wisconsin-Madison), Andres Perea (Maastricht University), R. Ramanujam (Institute of Mathematical Sciences, Chennai), Oliver Roy (Ludwig-Maximilians Universität München), Marciano Siniscalchi (Northwestern University), Giacomo Sillari (Scuola Normale Superiore, Pisa), Nobuyuki Suzuki (Shizuoka University), and Jonathan Zvesper (London, UK). I thank them for their timely and careful reviews and the interesting discussions about the submissions. We hope that we found a "good" trade-off between minimizing false rejections and false acceptances.

This TARK we have the pleasure of listening to three eminent invited speakers: Pierpaolo Battigalli (Bocconi University), Fangzhen Lin (Hong Kong University of Science and Technology), and Rineke Verbrugge (University of Groningen). Pierpaolo is the leading researcher in dynamic epistemic game theory. Fangzhen will tell us how we can discover the theorems of our future TARK papers by computer. Rineke will finally bring an empirical component to TARK. If we are serious about analyzing reasoning about knowledge and rationality, we ought to study how humans really reason.

TARK 2013 colocates with Fifth Indian Conference on Logic and its Applications (ICLA 2013). We are very grateful to the people heading ICLA - especially Kamal Lodaya and R. Ramanujam - for the collaboration and coordination between TARK and ICLA. It is common knowledge that the actual work involved with a conference rests with the local organizing committee. We thank Sujata Ghosh (Indian Statistical Institute, Chennai), Kamal Lodaya (Institute of Mathematical Sciences, Chennai), R. Ramanujam (Institute of Mathematical Sciences, Chennai), and S. P. Suresh (Chennai Mathematical Institute) for their hard work. I am extremely grateful to the chair of the local organizing committee, R. Ramanujam, who put TARK into action and made things really happen leaving to me the pleasant part.

We would like also to thank the people behind the EasyChair conference system. As the name suggests, Easychair makes it easy to chair the program committee handling submissions, reviews, and emails free of charge.

Last but not least, I thank Joe Halpern, the founder and chair of the TARK conference series. Without his admirable energy, enthusiasm, curiosity, and wide intellectual breath, TARK

wouldn't exist, and I would have never been exposed to the sound of TARK and learned its meaning.

Burkhard C. Schipper
University of California, Davis
Program Chair TARK 2013

Invited Talks

Dynamic Epistemic Game Theory

Pierpaolo Battigalli
Department of Economics, Department of Decision Sciences, and IGIER
Università Bocconi
Via Roegten, 20136 Milano, Italy
pierpaolo.battigalli@unibocconi.it

TARK 2013, Chennai, India. 2013, Chennai, India

Knowledge Representation and Computer-Aided Theorem Discovery

Fangzhen Lin
Department of Computer Science
Hong Kong University of Science and Technology
Clearwater Bay, Hong Kong
flin@cs.ust.hk

ABSTRACT

Using examples from game theory, social choice theory, and software engineering, I will talk about how knowledge representation formalisms can be used to help discover interesting and useful theorems.

TARK 2013, Chennai, India. 2013, Chennai, India

Logic in the Lab

Rineke Verbrugge
Institute of Artificial Intelligence
University of Groningen
P.O. Box 407
9700 AK Groningen,The Netherlands
L.C.Verbrugge@rug.nl

Categories and Subject Descriptors

F.4.1 [**Mathematical Logic**]: Modal Logic

General Terms

Theory

Keywords

higher-order social cognition, theory of mind, epistemic logic, game theory, cognitive science

1. THEORY OF MIND

As humans, we live in a remarkably complex social environment. One cognitive tool which helps us manage all this complexity is our *theory of mind*, the ability to reason about the mental states of others. By deducing what other people want, feel and think, we can understand their actions, predict how our actions will influence them, and decide how we should behave to be successful. Theory of mind is the cognitive capacity to understand and predict external behavior of others and oneself by attributing internal mental states, such as knowledge, beliefs, and intentions [17]. This is thought to be the pinnacle of social cognition. A heated debate is going: Do very smart animals, such as chimpanzees and ravens, have any theory of mind? [3, 19].

Especially important in intelligent interaction is higher-order theory of mind, an agent's ability to model recursively mental states of other agents, including the other's model of the first agent's mental state, and so forth. More precisely, zero-order theory of mind concerns world facts, whereas $k+1$-order reasoning models k-order reasoning of the other agent or oneself. For example, "Bob knows that Alice knows that he wrote a novel under pseudonym" ($K_{Bob}K_{Alice}p$) is a second-order attribution. It is commonly accepted that animals other than human beings do *not* use second- and higher-order theory of mind.

Several formal theories well-known to the TARK audience are suited to represent higher-order theory of mind in intelligent interaction, for example, epistemic logic, dynamic epistemic logic, and epistemic game theory [15, 6, 22, 16]. However, in epistemic logic, unlimited rationality is usually taken for granted. Agents are assumed to be *logically omniscient*: they know all logical truths. The epistemic language allows reasoning on any modal depth and presupposes that agents can immediately decide whether a formula like $K_{Ann}\neg K_{Bob}K_{Ann}K_{Carol}\neg K_{Ann}\ w_{Ann}$ is true in a given possible world. This is clearly not the case for all people [23]. Similarly, people often do not act according to the game-theoretic assumption of common knowledge of rationality [4]. In particular, several researchers have found that both children and adults have difficulties when applying second-order theory of mind in game situations [10, 7]. But how do people really reason about others' mental states?

TARK 2013, Chennai, India. 2013, Chennai, India

2. EXPERIMENTS

In our lab, we have performed several experiments with subjects applying second-order theory of mind in simple dynamic games. It turned out that we could facilitate their correct and fast performance a lot, for example, by providing step-wise training, by introducing a visual presentation that is easy to understand, and by prompting subjects to think about what their opponent would do [12, 13]. With the help of these cues, the subjects made the best possible decision in more than 90% of the game items. From what the subjects told us, however, we got the impression that even if they made the correct decisions, they did not reason exactly according to the game theory textbook. By a follow-up experiment with an eye-tracker, we concluded that indeed, most experimental subjects did not apply backward induction from the start, but tried to get by with forward reasoning as much as possible [14].

Formal methods are very useful for designing experiments and interpreting the results. As an example, Stenning and Van Lambalgen [18] provide an interesting analysis of the difficulties that autistic children have in ascribing false beliefs to another person, if they themselves know the true facts. As another example, one can investigate the computational complexity of the tasks that experimental subjects have been set [11]. Currently, we are investigating the complexity of several instances of backward induction and comparing them with subjects' behavior in terms of reaction times, decisions, and eye movements.

3. COGNITIVE MODELS

In order to understand how people really reason and solve problems, it has proven fruitful in cognitive psychology to use computational cognitive models implemented in a cognitive architecture such as ACT-R, which has been validated in hundreds of experiments [1]. It is also possible to use such computational models when investigating how people reason about other people's knowledge, beliefs and plans. One way to do this is to make an ACT-R model in which different reasoning strategies, such as backward reasoning and forward

reasoning, 'compete' with one another and the model learns by experience which reasoning strategy efficiently provides effective decisions [9, 8]. The main advantage of using computational cognitive models is that one can formulate very precise predictions and see whether the simulations match results of new experiments in the lab.

This is just what we did in the case of the controversy about smart birds: Elske van der Vaart constructed a computational cognitive model of birds' smart social behavior. It turned out that this 'virtual bird', equipped with sophisticated memory based on the theory behind ACT-R [2], and reacting to the stress of being observed, performed similarly to the real birds in several experiments [20, 21]. In the literature, the birds' behavior is often thought to exemplify a form of perspective-taking: "I want to prevent that the other bird knows where I've hidden my worms" [5]. We made some precise predictions that can help settle the disputes between 'theory of mind' versus 'simple behavioral rules', and that are currently being investigated in the lab.

Acknowledgments

I would like to thank the Netherlands Organization for Scientific Research (NWO) for Vici grant NWO 227-80-001, *Cognitive systems in interaction: Logical and computational models of higher-order social cognition.*

4. REFERENCES

[1] J. Anderson. *How Can the Human Mind Occur in the Physical Universe?* Oxford University Press, New York (NY), 2007.

[2] J. R. Anderson and L. J. Schooler. Reflections of the environment in memory. *Psychological Science*, 2(6):396–408, 1991.

[3] J. Call and M. Tomasello. Does the chimpanzee have a theory of mind? 30 years later. *Trends in Cognitive Sciences*, 12:187–192, 2008.

[4] C. Camerer. *Behavioral Game Theory: Experiments in Strategic Interaction.* Princeton University Press, Princeton (NJ), 2003.

[5] J. Dally, N. Emery, and N. Clayton. Food-caching western scrub-jays keep track of who was watching when. *Science*, 312:1662–1665, 2006.

[6] R. Fagin, J. Halpern, Y. Moses, and M. Vardi. *Reasoning about Knowledge.* MIT Press, Cambridge, MA, 1995. Second edition 2003.

[7] L. Flobbe, R. Verbrugge, P. Hendriks, and I. Krämer. Children's application of theory of mind in reasoning and language. *Journal of Logic, Language and Information*, 17:417–442, 2008. Special issue on formal models for real people, edited by M. Counihan.

[8] S. Ghosh and B. Meijering. On combining cognitive and formal modeling: A case study involving strategic reasoning. In J. van Eijck and R. Verbrugge, editors, *Proceedings of the Workshop on Reasoning About Other Minds (RAOM 2011)*, volume 751, pages 79–92. CEUR Workshop Proceedings, 2011.

[9] S. Ghosh, B. Meijering, and R. Verbrugge. Logic meets cognition: Empirical reasoning in games. In *Proceedings of the 3rd International Workshop on Logics for Resource Bounded Agents (LRBA 2010), in 3rd Multi-Agent Logics, Languages, and Organisations Federated Workshops, MALLOW'10, CEUR Workshop Proceedings*, volume 627, pages 15–34, 2010.

[10] T. Hedden and J. Zhang. What do you think I think you think? Strategic reasoning in matrix games. *Cognition*, 85:1–36, 2002.

[11] A. Isaac, J. Szymanik, and R. Verbrugge. Logic and complexity in cognitive science. In *Logical and Informational Dynamics: Johan van Benthem*, Trends in Logic: Outstanding Contributions, Berlin, 2013, to appear. Springer.

[12] B. Meijering, L. v. Maanen, H. v. Rijn, and R. Verbrugge. The facilitative effect of context on second-order social reasoning. In *Proceedings of the 32nd Annual Meeting of the Cognitive Science Society*, pages 1423–1428, Philadelphia, PA, 2010. Cognitive Science Society.

[13] B. Meijering, H. v. Rijn, N. Taatgen, and R. Verbrugge. I do know what you think I think: Second-order theory of mind in strategic games is not that difficult. In *Proceedings of the 33nd Annual Meeting of the Cognitive Science Society*, pages 2486–2491, Austin, TX, 2011. Cognitive Science Society.

[14] B. Meijering, H. van Rijn, N. Taatgen, and R. Verbrugge. What eye movements can tell about theory of mind in a strategic game. *PLoS ONE*, In press.

[15] J.-J. C. Meyer and W. van der Hoek. *Epistemic Logic for AI and Theoretical Computer Science.* Cambridge University Press, Cambridge, 1995.

[16] A. Perea. *Epistemic Game Theory.* Cambridge University Press, Cambridge, 2012.

[17] D. Premack and G. Woodruff. Does the chimpanzee have a theory of mind? *Behavioral and Brain Sciences*, 4:515–526, 1978.

[18] K. Stenning and M. van Lambalgen. *Human Reasoning and Cognitive Science.* MIT Press, Cambridge (MA), 2008.

[19] E. van der Vaart and C. Hemelrijk. 'Theory of mind' in animals: Ways to make progress. *Synthese*, pages 1–20, 2012.

[20] E. van der Vaart, R. Verbrugge, and C. Hemelrijk. Corvid caching: Insights from a cognitive model. *Journal of Experimental Psychology: Animal Behavior Processes*, 37(3):330, 2011.

[21] E. van der Vaart, R. Verbrugge, and C. Hemelrijk. Corvid re-caching without 'theory of mind': A model. *PloS ONE*, 7(3):e32904, 2012.

[22] H. van Ditmarsch, W. van der Hoek, and B. Kooi. *Dynamic Epistemic Logic*, volume 337 of *Synthese Library Series.* Springer Verlag, Berlin, 2007.

[23] R. Verbrugge. Logic and social cognition: The facts matter, and so do computational models. *Journal of Philosophical Logic*, 38(6):649–680, 2009.

Contributed Talks

Utility-based Decision-making in Distributed Systems Modelling

[Extended Abstract]

Gabrielle Anderson* University of Aberdeen Scotland, UK

Matthew Collinson† University of Aberdeen Scotland, UK

David Pym‡ University of Aberdeen Scotland, UK

ABSTRACT

We consider a calculus of resources and processes as a basis for modelling decision-making in multi-agent systems. The calculus represents the regulation of agents' choices using utility functions that take account of context. Associated with the calculus is a (Hennessy–Milner-style) context-sensitive modal logic of state. As an application, we show how a notion of 'trust domain' can be defined for multi-agent systems.

1. INTRODUCTION

Mathematical modelling is a key tool in designing and reasoning about the complex systems of systems upon which the world depends. For modelling complex information processing systems, including both logical and physical components, the classical theory of distributed systems — see, for example, [13] for an elegant account — provides a suitable conceptual basis [11] for a modelling discipline. Executable modelling languages are important supporting tools, providing methods for simulating — using both Monte Carlo and what-if methods — systems that are too complex for useful analytical solvable descriptions. Techniques such as model checking can be applied in sufficiently constrained circumstances [11].

In this paper, we show how a compositional mathematical systems modelling theory — which is grounded in process algebra [28, 29] and logical resource semantics [32, 34, 35, 11], which has been developed in detail by some of us elsewhere [12, 9, 10, 11], and which is supported by an execution engine and a model checker [11] — can be extended to include an account of decision-making by agents as they execute within models.

Our approach introduces into the account of processes a notion of utility that associates values to the agents' choices. Our addition of utility is formulated so as to support a key measure of decision-making in multi-agent systems: agents make their decisions in the context provided by the other agents that are executed within the model, so that different decision paths occur in different contexts. As is usual in the process-algebraic approach to modelling, the language of processes is associated with a logic of state, in the sense of Hennessy and Milner [11], in which propositional assertions describe properties of the states of the model. In this paper, these logical judgements are also context-dependent.

In Section 2, we provide a brief introduction to our background modelling theory [11] and in Section 3 we explain our utility-theoretic approach to modelling decision-making. In Section 4, we explain our contextual process calculus with utility and, in Section 5, we explain its associated logic. We conclude, in Section 6, with (a sketch of) an application of our ideas to the concept of a 'trust domain'. We provide an extended derivation of the example used in this paper in Appendix A, and full proofs of all claims in [2].

2. SYSTEMS MODELLING BACKGROUND

While the notion of process has been explored in some detail by the semantics community, concepts like resource have usually been treated as second class ([30] is a partial exception). From the point-of-view of a theorist, there are many advantages in doing this. We have taken the opposite view [12, 9, 10, 11]: we explore what can be gained by developing an approach in which the structures present in modelling languages are given a rigorous treatment as first-class citizens in a theory. In particular, we ensure that each component — locations, resources, and processes — is handled compositionally. These key structural components are considered, drawing upon distributed systems theory (e.g., [13]), as follows:

Location: Places are connected by (directed) links. Locations may be abstracted and refined provided the connectivity of the links and the placement of resources is respected. Mathematically, the axioms for locations [9] are satisfied by various graphical structures, including simple directed graphs and hyper-graphs, as well as various topological constructions [12, 9, 10, 11];

Resource: The notion of resource captures the components of the system that are manipulated by its processes (see below). Resources include things like the components used by a production line, the tools on a production line, computer memory, system operating staff, or system users, as well as money. Conceptually, the axioms of resources are that they can be combined and compared. Mathematically, we model this notion using *(partial commutative) resource monoids* [32, 12, 9, 10, 11]. That is, structures $\mathbf{R} = (\mathbf{R}, \sqsubseteq, \circ, e)$ with carrier set $\mathbf{R}$, preorder $\sqsubseteq$, and par-

*Email: g.a.anderson@abdn.ac.uk

†Email: matthew.collinson@abdn.ac.uk

‡Email: d.j.pym@abdn.ac.uk

TARK 2013, Chennai, India.

tial binary composition $\circ$ with unit e, and which satisfies the *bifunctoriality condition*: $R \sqsubseteq R'$ and $S \sqsubseteq S'$ and $R \circ S$ is defined implies $R' \circ S'$ is defined and $R \circ S \sqsubseteq R' \circ S'$, for all $R, S, R', S' \in \mathbf{R}$. In this paper, the order $\sqsubseteq$ is always taken to be equality. Let $\mathbf{R}$ be a given resource monoid;

Process: The notion of process captures the (operational) dynamics of the system. Processes manipulate resources in order to deliver the system's intended services. Mathematically, we use algebraic representation of processes based on the ideas in [28], integrated with the notions of resource and location [12, 9, 10, 11].

Let Act be a commutative monoid of *actions*, with multiplication written as juxtaposition and unit 1. Let $a, b \in$ Act, etc. The execution of models based on these concepts, as formulated in [12, 9, 10, 11], is described by a transition system with a basic structural operational semantics judgement [33, 28] of the form

$$L, R, E \xrightarrow{a} L', R', E',$$

which is read as 'the occurrence of the action a evolves the process E, relative to resources R at locations L, to become the process E', which then evolves relative to resources R' at locations L''.

The meaning of this judgement is given by a structural operational semantics [33, 28]. The basic case, also know as 'action prefix', is the rule

$$\frac{}{L, R, a : E \xrightarrow{a} L', R', E'} \quad \mu(L, R, a) = (L', R').$$

Here μ is a 'modification' function from locations, resources, and actions (assumed to form a monoid) to locations and resources that describes the evolution of the system when an action occurs. Neglecting locations for now, partial function $\mu : \text{Act} \times \mathbf{R} \to \mathbf{R}$ is a *modification* if it satisfies the following conditions for all a, b, R, S: $\mu(1, R) = R$; if $R \circ S$ and $\mu(a, R) \circ \mu(b, S)$ are defined, then $\mu(ab, R \circ S) = \mu(a, R) \circ \mu(b, S)$.

There are also rules giving the semantics to combinators (which together form a a complete set for enumerating r.e. graphs) for concurrent composition, choice, and hiding — similar to restriction in SCCS and other process algebras (e.g., [28, 29] — as well for recursion. For example, the rule for synchronous concurrent composition of processes is

$$\frac{L, R, E \xrightarrow{a} L', R', E' \qquad M, S, F \xrightarrow{b} M', S', F'}{L \cdot M, R \circ S, E \times F \xrightarrow{ab} L' \cdot M', R' \circ S', E' \times F'},$$

where we presume, in addition to the evident monoidal compositions of actions and resources, a composition on locations. The rules for the other combinators, with suitable coherence conditions on the modification functions, follow similar patterns [11]. Note that our choice of a synchronous calculus retains the ability to model asynchrony [28, 29, 14] (this doesn't work the other way round).

Associated with this transition semantics is a modal logic, given in the sense of Hennessy and Milner [19], with satisfaction relation $L, R, E \models \phi$ read as 'property ϕ holds of the process E executing with resources R at locations L'. In developing our subsequent theory, we will, for brevity and simplicity, suppress locations, working with a calculus and associated logic of resources and processes, based on judgements of the form $R, E \xrightarrow{a} R', E'$ and $R, E \models \phi$, respectively. Whilst this simplification represents some loss of generality (see [11] for more detail), the essential ideas are not significantly affected (and, indeed, some aspects of location can be coded within resource). In our discussion of the concept of a trust domain, in Section 6, the intuitive need for location, be it 'logical' or 'physical', is apparent and we revisit the concept of location there.

The logic includes — in addition to the usual additive connectives, quantifiers, and modalities that are familiar from Hennessy-Milner logic — multiplicative connectives, quantifiers, and modalities [12, 9, 10, 11]. For example, dropping locations for brevity, multiplicative conjunction is defined by the logical decomposition of the system, as follows:

$$R, E \models \phi_1 * \phi_2 \quad \text{iff} \quad \begin{array}{l}\text{there are } R_1, R_2 \text{ and } E_1, E_2 \text{ s.t.} \\ R = R_1 \circ R_2,\ E \sim E_1 \times E_2, \text{ and} \\ R_1, E_1 \models \phi_1 \text{ and } R_2, E_2 \models \phi_2.\end{array}$$

Here $\sim$ is bisimulation, explained in detail for this set-up in [12, 9, 10, 11], where treatments of location can also be found.

The action modalities work just as in Hennessy–Milner logic. For example,

$$R, E \models \langle a \rangle \phi \quad \text{iff} \quad \begin{array}{l}\text{for some } R', E' \text{ s.t. } R, E \xrightarrow{a} R', E' \\ \text{and } R', E' \models \phi.\end{array}$$

The multiplicative version of this rule would permit the evolution by action a to employ additional resource; that is, for some S, R', and E, $R \circ S, E \xrightarrow{a} R', E'$. Details and theoretical development may be found in [12, 9, 10, 11]. Although the basic logic of bunched implications, with intuitionistic additives and multiplicatives, is decidable [17], its counterpart with classical additives, widely known as 'Boolean BI' and the basis of Separation Logic [37], is not [6, 18]. We conjecture that the undecidability of Boolean BI implies the undecidability of the logic presented here.

In addition to the structural components of models, we consider also the environment within which a system exists:

Environment: All systems exist within an external environment, which is typically treated as a source of events that are incident upon the system rather than being explicitly stated. Mathematically, environments are represented stochastically, using probability distributions that are sampled in order to provide such events [12, 9, 10, 11].

The modal logic discussed above can also be extended to the stochastic world, but an account of this is beyond our present needs and scope. Related work can be found in [38]. Logical reasoning about distributed systems has also been studied by Barwise and Seligman [4]. We provide full proofs of all claims made in this paper in the accompanying technical report [2].

3. MODELLING DECISION-MAKING

One use of the process component within our models is to represent agents within a system as they explore their worlds, making decisions between the choices that are available to them. In the set-up described so far, as presented in [12, 9, 10, 11], we have considered (again, suppressing location, for brevity) an operational rule for choice of the form

$$\frac{R, E_i \xrightarrow{a} R', E'}{R, \sum_{i \in I} E_i \xrightarrow{a} R', E'},$$

where I is an indexing set. This rule is understood, in the style of structural operational semantics [33], read from con-

clusion to premisses, as follows: the sum of the processes can evolve by an action a, to become R', E' if one of its summands can evolve by the action a to become R', E'. In other words, there is a set of possible evolutions each element of which leads to the evolution of the sum.

Models of distributed systems often capture situations in which an agent or group of agents is exploring a world and interacting with it and themselves. In such cases, a process containing choices of the kind discussed represents the choices made by agents as they evolve. It is therefore useful to extend our modelling theory to provide an account of agents' decision-making.

Our approach, applying our methodology of incorporating representations of system modelling components as first-class citizens, is to model agents' decision-making by developing a utility-carrying version of the location-resource-process calculus sketched above. It is possible to encode some notions of location in the resource component; for brevity and simplicity, we follow this approach to location for the remainder of this paper. We return, in particular, to the strengths of using location in modelling when we consider trust domains in Section 6.

The key idea is to replace the simple choice combinator described above with the *utility-dependent choice* (or simply *sum*) $\sum_{i\in I}{}_u\, E_i$, in which an agent has a choice between alternatives E_i, and its preference is codified by the utility $u \in U$. The operational rules of the calculus ensure that this preference takes into account the wider context in which the choice is embedded. For example, if the choice $R, a:E +_u b:F$ (here we use an infix notation) occurs within a wider context $R \circ S, (a : E +_u b : F) \times G$, then preference will be determined by the utility calculations

$$u(R \circ S, a:E \times G) \quad \text{and} \quad u(R \circ S, b:F \times G). \qquad (1)$$

If the former is greater than the latter, the $a : E$ option will be chosen. An occurrence of the same choice $R, a:E +_u b:F$ within a different context, such as $R \circ T, (a:E +_u b:F) \times H$ with $G \neq H$ or $S \neq T$, will have different utility calculations, and may result in $b : F$ being chosen instead.

Many process calculi include a form of prioritized sum, for example [41]. In prioritized sums, say $w \cdot a : E + w' \cdot b : F$, with $w > w'$, the option $a : E$ is always chosen in any context in which both a and b are available (which they may not be, because of restriction operations). In contrast, our utility-based sum can make different choices in different contexts even when the same options are available.

In our set-up, resource-process contexts correspond to the semantical notion of world. Utility functions are simply given, being aspects of agents described in the model by the modeller. A *preference order* $\succeq_u$ on worlds is induced, in the usual way, by: $R, E \succeq_u S, F$ iff $u(R, E) \geq u(S, F)$, for all R, E and S, F. This means that the mathematical structures we are considering appear to be similar to those used to give a semantics for dynamic logics of preference [42], but in a special case where processes are used to give a very richly descriptive dynamics.

The use of a utility restricts the ordinal preference relations that can be represented (in the usual way [25]), but it is a trivial step to replace the utility comparisons in our calculus with more general relational comparisons, if so required.

Our utility calculations are not required to be determined by the system dynamics described by the transition system. At a sum, an agent makes a decision as to its own (partial) control (i.e., process) choice knowing its context, but not using information about the resulting future evolution of the system. Even if an agent may face a sequence of decisions, we are not forced to model it as undertaking a traditional, rational, multi-stage decision process. The decoupling of choice from dynamics should also make it possible to incorporate expected utility for probabilistic choices [41] in a natural fashion. Traditional decision theory usually treats decision situations in a flat, atomic way, and is not concerned with similar choice-points in different contexts. Prior works that address this issue include [21, 16].

Process calculi in which contexts are treated as first-class citizens include [8, 40, 7, 39]. Logics of propositions in context have also been extensively studied, for example [27]. Points of contact between decision theory and process calculus include [15, 31, 3]. Our approach differs from these in having an explicit utility-based choice constructor, which, in particular, which takes into account the wider context. The importance of the combination of utility and process in reasoning about trust has also been recognized in [3].

4. A PROCESS ALGEBRA WITH UTILITY

In process calculi such as the one sketched in Section 1, the behaviour of a composite process, such as $E_1 \times E_2$, is usually defined in terms of the behaviours of its sub-processes, with the reductions of E_1 and E_2 being independent of each other. In our calculus, however, choices take account of the context in which sub-processes are reduced, so that the reductions of E_1 and E_2 may not be independent.

To see how this works, consider the example used in the utility expressions (1) in Section 3. The process $a : A +_u b : B$ reduces taking account of its context G. We annotate the context in which a process is reduced on the underside of the reduction arrow (e.g., $R, a : A +_u b : B \xrightarrow[S,[\,]\times G]{}{}^{a} R', A$), where $[\,]$ denotes the hole into which $a : A +_u b : B$ may be substituted to regain the complete system $(a : A +_u b : B) \times G$, and S are the resources allocated to G. Note also that any choices in $[\,] \times G$ may depend on what process is substituted into the hole $[\,]$. We therefore annotate the process that is substituted, into the process being reduced, on top of the reduction arrow; for example, $S, [\,] \times G \xrightarrow{R,a:A+_u b:B}{}^{b} S', [\,] \times G'$.

So, the key judgement of the reduction relation for processes with utility is of the form

$$C \xrightarrow[C_1]{C_2}{}^{a} C', \qquad (2)$$

which denotes how a context C, that exists in a system that can be decomposed as $C_1(C(C_2))$, reduces. We refer to C as the *(primary) context*, C_1 as the *outer context*, and C_2 as the *substituted*, or *inner context*. Intuitively this denotes the reduction of one part, C, of an entire system, $C_1(C(C_2))$. In order to reason compositionally we wish to be able to describe the reduction of C independently and structurally. As choices can take account of context, this is not possible. The choices in C, however, only make use of the definition of C_1 and C_2, and disregard their structure. Hence we do not need to reason over the structure of C_1 and C_2, as we do with C, only to record their definitions which can then be referred to at choice points. They are therefore annotated

on the reduction arrow for reference, but are not reduced in said relation.

We now describe the theoretical set-up in detail. Assume a set U of symbols, called *formal utilities*, with a distinguished element 0_U, called the *neutral utility*. *Processes* are generated by the grammar:

$$E ::= \mathbf{1} \mid [\,] \mid a : E \mid \sum_{i \in I}{}_u\, E_i \mid E \times E.$$

These are really process contexts: the term $[\,]$ is a *hole* into which other processes may be substituted. For this work, it turns out to be convenient to develop contexts as first-class citizens rather than merely meta-theoretic tools.

The *choice* $\sum_{i \in I}{}_u\, E_i$ is new: it describes situations in which an agent has a choice between alternatives E_i indexed by a $i \in I$, and its preference (in a larger context) is codified by the utility $u \in U$. The infix operator $E +_u F$ may be used for binary sums, and the subscript u may be dropped when $u = 0_U$. The *zero* process $\mathbf{0}$ is defined to be the sum indexed by the empty set and the neutral utility. The zero process, *unit* process $\mathbf{1}$, and *synchronous products* $E \times F$ are well-known in process calculus, as are *prefixes* $a : E$, where $a \in \mathrm{Act}$.

A process E is *well-formed* if it contains at most one hole and that hole is not guarded by action prefixes. The process E is *closed* if it has no holes and *open* otherwise. Let $PCont$ be the set of all well-formed processes, $PCCont$ be the set of all closed well formed processes, and $POCont$ be the set of all open well-formed processes.

Let $\mathbf{R}$ be a resource monoid and μ be a fixed modification function, as defined in Section 2. Define the products of sets $Cont = \mathbf{R} \times PCont$, $CCont = \mathbf{R} \times PCCont$ and $OCont = \mathbf{R} \times POCont$. The letter C is reserved for contexts. Define $C_\emptyset = e, [\,]$. Brackets will be freely used to disambiguate both processes and contexts. For $C = R, E$, the notational abuses $C \times F = R, (E \times F)$ and $C +_u F = R, (E +_u F)$ will sometimes be used. Substitution in processes, $E(F)$, replaces all occurrences of $[\,]$ in E with F; for example, $(([\,] +_u E) \times G)(F) = (F +_u E) \times G$. Substitution of contexts $C_1(C_2)$, where $C_1 = R, E$ and $C_2 = S, F$, is defined as follows: if E is open, then $C_1(C_2) = R \circ S, E(F)$, where $E(F)$ is process substitution; if E is closed, then $C_1(C_2) = C_1$.

We assume that, for each formal utility $u \in U$, there is an associated, real-valued *utility function* $u : Cont \longrightarrow \mathbb{R}$ [24] that fixes an interpretation for each formal symbol $u \in U$. The identically zero function is associated with 0_U. Henceforth, we do not distinguish between formal utilities and their utility functions.

The operational semantics of our process-utility calculus is given in Figure 1. The side-condition $(S\sum)$ is that $C_3 = C_1((e, \sum_{I \setminus j}{}_u\, E_i(C_2)) +_u [\,])$ and $\forall i \in I . u(C_1(R, E_i(C_2))) \leq u(C_1(R, E_j(C_2)))$. The side-condition $(S\times)$ is that $C_3 = C_1((S, F(C_2)) \times [\,])$ and $C_4 = C_1((R, E(C_2)) \times [\,])$.

The unit process always ticks, effecting no change. The prefix process evolves via its head action. The hole rule is a technical one used to terminate reduction derivations of open contexts. The sum process $\sum_I{}_u\, E_i$ represents a preference-based choice by the agent: it follows the behaviour of any of its constituent E_j which has at least as high a value ascribed by its utility u as any other option E_i for $i \in I$. The

$$\frac{}{R, \mathbf{1} \xrightarrow[C_1]{C_2}{}^{1} R, \mathbf{1}} \quad \text{(Tick)}$$

$$\frac{}{R, a : E \xrightarrow[C_1]{C_2}{}^{a} \mu(a, R), E} \quad \text{(Prefix)}$$

$$\frac{C_2 \xrightarrow[C_1]{(e,\mathbf{1})^a} C_2'}{e, [\,] \xrightarrow[C_1]{C_2}{}^{1} e, [\,]} \quad \text{(Hole)}$$

$$(S\textstyle\sum) \qquad \frac{R, E_j \xrightarrow[C_3]{C_2}{}^{a} S, F}{R, \sum_I{}_u\, E_i \xrightarrow[C_1]{C_2}{}^{a} S, F} \quad \text{(Sum)}$$

$$(S\times) \qquad \frac{R, E \xrightarrow[C_3]{C_2}{}^{a} R', E' \qquad S, F \xrightarrow[C_4]{C_2}{}^{b} S', F'}{R \circ S, E \times F \xrightarrow[C_1]{C_2}{}^{ab} R' \circ S', E' \times F'} \quad \text{(Prod)}$$

Figure 1: Operational Semantics

first special case of the sum is for the zero process $\mathbf{0}$, which never evolves. The second special case is where $u = 0_U$ and the sum becomes an ordinary non-deterministic sum: in this case, the utility is irrelevant, and the sum may evolve as any of its component processes. The product evolves two processes synchronously in parallel, according to the decomposition of the associated resources. An important feature of this system is that contextual information about conclusions is propagated up to premisses. In the product case, information about each premiss is propagated up from the conclusion to the other premiss, so that derivations of transitions occur in context.

To demonstrate how contextual decisions can be utilized in modelling, we give a simple example (inspired by [5]. Consider a banker who has a presentation (for a client, that includes confidential business data) on a USB drive. The banker may chose to access the drive or not, depending on the situation. The banker is modelled as a process

$$Banker = present : Banker' +_{u_B} idle_B : Banker', \quad (3)$$

where u_B represents its preferences. The banker may be willing to access the presentation when visiting a client, on the assumption that the client's network is firewalled, so making the document safe from attack. In order to do so, however, the banker must be given access to a computer by the client. The client is modelled as

$$Client = logIn : Client' +_{0_U} idle_C : Client', \quad (4)$$

which, for simplicity, makes a non-deterministic choice between logging the guest in and idling. The interaction between the banker and the client is a form of joint access control, in which the banker cannot show the presentation without having been logged in, and the client cannot see the presentation unless the banker accesses it. Here we can show

that the principals co-operate to access the presentation:

$$R, Client \times Banker \xrightarrow{logIn, present} S, Client' \times Banker'. \quad (5)$$

If the banker's utility is u_B, then we have

$$u_B(R, Client \times (idle_B : Banker')) \leq u_B(R, Client \times (present : Banker')). \quad (6)$$

In a different situation — here, a different context — the banker may make a different decision. The banker may use a home computer, compromised by an attacker, who wants to steal the presentation, but cannot unless the banker accesses it from the USB stick. The attacker is modelled as

$$Attacker = steal : Attacker' +_{0_U} idle_A : Attacker'. \quad (7)$$

In this situation, the banker prefers to idle than to work on the presentation. As, in order for the attacker to steal the presentation the banker must access it, and the banker choses not to, then the attacker must also idle, so that

$$R, Attacker \times Banker \xrightarrow{idle_A, idle_B} S, Attacker' \times Banker'. \quad (8)$$

Here the banker's utility yields

$$u_B(R, Attacker \times (present : Banker')) \leq u_B(R, Attacker \times (idle_B : Banker')). \quad (9)$$

A more detailed exposition of these examples is presented in Appendix A.

A fundamental aspect of process calculus is the ability to reason equationally about behavioural equivalence of processes [28]. We now adapt these notions to suit the calculus above, which incorporates ideas from [12, 9, 11].

The *bisimilarity (or bisimulation) relation* $\sim \subseteq PCont \times PCont$ is the largest binary relation such that, if $E \sim F$, then $\forall a \in \text{Act}$, $\forall R, R', S, T \in \mathbf{R}$, and for all $G, H, I, J \in PCont$ with $G \sim I$ and $H \sim J$, then

1. $\forall E' \in PCont$, if $R, E \xrightarrow[S,G]{T,H}{}^{a} R', E'$ then $\exists F'$ such that $R, F \xrightarrow[S,I]{T,J}{}^{a} R', F'$ and $E' \sim F'$, and
2. $\forall F' \in PCont$, if $R, F \xrightarrow[S,I]{T,J}{}^{a} R', F'$ then $\exists E'$ such that $R, E \xrightarrow[S,G]{T,H}{}^{a} R', E'$ and $E' \sim F'$.

The union of any set of relations that satisfy these two conditions also satisfies these conditions, so the largest such relation is well-defined. Define $\sim \subseteq Cont \times Cont$ by: if $E \sim F$ then $R, E \sim R, F$ for all $R \in \mathbf{R}$ and $E, F \in Cont$.

Definition 1. *A utility, u,* respects bisimilarity *if, for all $C_1, C_2 \in Cont$, $C_1 \sim C_2$ implies $u(C_1) = u(C_2)$.*

That is, behaviourally equivalent (bisimilar) states are required to be indistinguishable by u. The set U of utilities respects bisimilarity if every $u \in U$ respects bisimilarity. Henceforth utilities are assumed to respect bisimilarity. We can show that if bisimilar contexts are substituted into each other, then the result is bisimilar:

Proposition 1. *If $E \sim G$ and $F \sim H$, then $E(F) \sim G(H)$.*

We can then prove a key property for reasoning compositionally.

Theorem 1 (Bisimulation Congruence). *The relation $\sim$ is a congruence. It is reflexive, symmetric and transitive, and for all a, E, F, G with $E \sim F$, and all families $(E_i)_{i \in I}$, $(F_{i \in I})_I$ with $E_i \sim F_i$ for all $i \in I$, $a : E \sim a : F$, $E \times G \sim F \times G$, and $\sum_{i \in I}{}_u E_i \sim \sum_{i \in I}{}_u F_i$.*

Proof. Symmetry, reflexivity, and transitivity are straightforward. Prefixed processes $a : E$ and $a : F$ can only reduce via an a action to E and F, which are bisimilar.

Consider the choices $E +_u G$ and $F +_u G$, in (bisimilar) outer contexts C_1 and C_2, with (bisimilar) inner contexts C_3 and C_4, and the case where $R, E +_u G \xrightarrow[C_1]{C_2}{}^{a} S, E'$. By the (Sum) rule we know that $u(C_1(R, G(C_2))) \leq u(C_1(R, E(C_2)))$ and that $R, E \xrightarrow[C_5]{C_2}{}^{a} S, E'$, where $C_5 = C_1((e, G(C_2)) +_u [])$. Let $C_6 = C_3((e, G)(C_4) +_u [])$; we can show that $C_5 \sim C_6$ (by Proposition 1), and hence that $R, F \xrightarrow[C_6]{C_4}{}^{a} S, F'$. By Proposition 1 we know that $C_1(R, E(C_2)) \sim C_3(R, F(C_4))$. Using Proposition 1, and the fact that utility functions respect bisimilarity we can show that $u(C_3(R, G(C_4))) \leq u(C_3(R, F(C_4)))$, and hence that $R, F +_u G \xrightarrow[C_3]{C_4}{}^{a} S, F'$.

The product case follows from the fact that the contexts in which each sub-process reduces, such as $C_1((S, G(C_2)) \times [])$ for E, is bisimilar to the context in which the counterpart reduces, for example $C_3((S, G(C_4)) \times [])$ for F, by Proposition 1. □

In order to reason equationally about processes, it is also useful to establish various algebraic properties concerning parallel composition and choice. We derive these below, for our calculus. In order to do so, we make some additional definitions concerning utility functions.

Definition 2. *The set of utilities, U, is* (algebraically) accordant *it respects bisimilarity and, for all $u, v \in U$, all $C, C_1, C_2, C_3, C_4 \in Cont$, all $E, F, G \in PCont$, and $R \in \mathbf{R}$,*

1. *$u(C(R, F)) \leq u(C(R, E))$ and $u(C(R, G)) \leq u(C(R, E))$ if and only if $u(C(R, F +_v G)) \leq u(C(R, E))$,*
2. *$u(C(R, F)) \leq u(C(R, E))$ and $u(C(R, G)) \leq u(C(R, E))$ if and only if $u(C(R, G)) \leq u(C(R, E +_u F))$.*
3. *for all R, E, $u(C(R, 0)) \leq u(C(R, E))$, and*
4. *for all $C_1 \sim C_3, C_2 \sim C_4, R, E, F, G$, $u(C_1(R, E \times G +_u F \times G(C_2))) = u(C_3(R, (E +_u F) \times G(C_4)))$.*

We use the binary version of sum here in order to aid comprehension, but finite choices between sets of processes work straightforwardly. Any real-valued function defined on the quotient $Cont/\sim$ defines a utility that respects bisimilarity.

Proposition 2 (Algebraic Properties). *If U is accordant, then:*

1	$E +_u F$	$\sim$	$F +_u E$
2	$E +_u (F +_u G)$	$\sim$	$(E +_u F) +_u G$
3	$E +_u \mathbf{0}$	$\sim$	E
4	$E \times \mathbf{0}$	$\sim$	$\mathbf{0}$
5	$E \times \mathbf{1}$	$\sim$	E
6	$E \times F$	$\sim$	$F \times E$
7	$E \times (F \times G)$	$\sim$	$(E \times F) \times G$
8	$(E +_u F) \times G$	$\sim$	$E \times G +_u F \times G$.

PROOF. The interesting cases are: associativity of choice (2) as it uses Definition 2.1 and 2.2, the unit of choice (3) as it uses Definition 2.3, and distributivity of product over choice (8), which uses Definition 2.4. The others are as usual, and do not make use of the accordance properties.

As a representative of the interesting properties we prove the associativity of choice. Consider the choices $E +_u (F +_u G)$ and $(E +_u F) +_u G$, in (bisimilar) outer contexts C_1 and C_2, with (bisimilar) inner contexts C_3 and C_4, and the case where $R, E +_u (F +_u G) \xrightarrow[C_1]{C_2}{}^{a} S, E'$. By the (SUM) rule we know that $u(C_1(R, F +_u G(C_2))) \leq u(C_1(R, E(C_2)))$. By the accordance properties (Def. 2.1) we then know that $u(C_1(R, F$
$(C_2))) \leq u(C_1(R, E(C_2)))$ and $u(C_1(R, G(C_2))) \leq u(C_1(R,$ $E(C_2)))$.

Let $C_5 = C_1((e, F +_u G(C_2)) +_u [\,])$ and $C_6 = C_3((e, F +_u$ $G(C_4)) +_u [\,])$; using Proposition 1 we can show that these two contexts are bisimilar. By the (SUM) rule we know that, as $R, E +_u (F +_u G) \xrightarrow[C_1]{C_2}{}^{a} S, E'$, then $R, E \xrightarrow[C_5]{C_2}{}^{a} S, E'$. So, by the definition of bisimulation, we have that $R, E \xrightarrow[C_6]{C_4}{}^{a} S, E'$.

As utility respects bisimilarity (Definition 1) we have that $u(C_3(R, G(C_4))) = u(C_1(R, G(C_2))) \leq u(C_1(R, E +_u F(C_2)))$ $= u(C_3(R, E +_u F(C_4)))$, and that $u(C_3(R, F(C_4))) = u(C_1$ $(R, F(C_2))) \leq u(C_1(R, E(C_2))) = u(C_3(R, E(C_4)))$. Let $C_7 = C_3((e, G(C_4)) +_u [\,])$. By (SUM) we can then show that $R, E +_u F \xrightarrow[C_7]{C_4}{}^{a} S, E'$, and finally that $R, (E +_u F) +_u$ $G \xrightarrow[C_3]{C_4}{}^{a} S, E'$. □

Future work includes extending the calculus to include probabilistic choice [41] and expected utility [24]. It would also be interesting to consider whether the (pre)sheaf-theoretic semantics considered by Winskel [23] can be adapted to our calculus.

5. A PROCESS LOGIC WITH UTILITY

We now introduce a modal logic of system properties. The semantics is given using a satisfaction relation

$$C \vDash_{C'} \phi,$$

where C is a closed context, C' is a context and ϕ is a formula of a (Hennessy–Milner-style) modal logic of processes: this may be read 'the *primary context* C satisfies ϕ in the *surrounding context* C'' (cf. (2)). The context C may satisfy different logical propositions, perhaps even negations of each other, when placed in different surrounding contexts; an example of this is below. Context-sensitive logics have been studied by other authors [26, 4]. The structural nature of processes and resources provides a semantic framework in which such logics seem particularly natural.

The propositions of the logic are defined by the grammar

$$\begin{aligned} \phi ::= \quad & p \mid \bot \mid \top \mid \neg\phi \mid \phi \wedge \phi \mid \phi \vee \phi \mid \phi \to \phi \mid \\ & \langle a \rangle \phi \mid [a]\phi \mid I \mid \phi * \phi \mid \phi \mathrel{-\!\!*} \phi, \end{aligned}$$

where p ranges over atomic propositions, and a over actions. The symbols for propositions for *truth*, *falsehood*, *negation* and *(additive) conjunction*, *disjunction*, and *implication* are standard. The (*additive*) *modal connectives* are $\langle a \rangle$ and $[a]$. The connectives I, $*$, and $-\!\!*$ are the *multiplicative unit*, *conjunction*, and *implication*, respectively.

A *valuation*, $\mathcal{V}$, is a function that maps each atomic proposition to a $\sim$-closed set of closed contexts. The satisfaction relation is specified in Figure 2.

In the interpretation of atoms, the surrounding context is wrapped around the primary context, and the valuation of the atom consulted to see if it contains this compound context. This is what makes our logic context-sensitive. $\top$, $\bot$, $\neg$, $\wedge$, $\vee$, and $\to$ are all interpreted (essentially) classically.

The interpretation of the multiplicatives connectives here is similar to that for the logic MBI in [12]. Recall also the comments on $*$ in Section 1. The semantics of $*$ in Figure 2 is slightly modified, because of the way that contextual information is propagated upwards from conclusion to premisses in the product rule of the operational semantics.

The standard interpretation of Hennessy–Milner logics uses the relation specified by the operational semantics as a Kripke structure to support the modal connectives. In our work, the operational semantics is more complex: a context occurs, and reduces alongside an outer context. Hence when we consider whether $C_1 \vDash_{C_2} \langle a \rangle \phi$ holds, we have to consider whether there are reductions of the form $C_1 \xrightarrow[C_2]{C_\emptyset}{}^{a} C_1'$ *and* $C_2 \xrightarrow[C_\emptyset]{C_1}{}^{b} C_2'$ such that $C_1' \vDash_{C_2'} \phi$. The occurrences of the empty context ensure that no extraneous contextual information is introduced into the reductions of interest. The $[a]$ modality is interpreted similarly.

Recall the example of the banker who decides which actions to take in different contexts (3-9). In a situation that consists of a client (context C_C), the banker chooses to access the presentation, but in a situation that consists of an attacker (context C_A) the banker chooses not to: that is,

$$\begin{aligned} R_B, Banker &\vDash_{C_C} \langle present \rangle \top \\ R_B, Banker &\vDash_{C_A} \neg \langle present \rangle \top. \end{aligned} \tag{10}$$

A derivation of these properties is in Appendix A. Hence, in different contexts the process satisfies different propositions that, moreover, would be inconsistent over the same context.

Behaviourally equivalent processes are also logically equivalent (they satisfy the same logical properties). This is half of the Hennessy–Milner property [19, 20].

THEOREM 2. *If $C_1 \vDash_{C_2} \phi$, $C_1 \sim C_3$, and $C_2 \sim C_4$, then $C_3 \vDash_{C_4} \phi$.*

PROOF. A standard argument, by induction over the definition of $C_1 \vDash_{C_2} \phi$, using Proposition 1 in the cases where contexts are extended. □

Hence, bisimilar processes can be used interchangeably within a larger system, without changing the logical properties of the larger system.

It is unclear whether a useful converse can be obtained. With restrictions on the available fragments of the logic, and a different equivalence relation, however, it is possible to obtain a converse. To this end, we introduce the *local equivalence relation* $\approx \subseteq (OCont \times Cont \times CCont) \times (OCont \times Cont \times CCont)$, the largest binary relation such that, if $C_1, A, D_1 \approx C_2, B, D_2$, $\forall a \in \text{Act}$ where $A = R, E$ and $B = S, F$ (with A' and B', etc., modifying them, as usual), then

1. $\forall C_1' \in OCont, A' \in Cont, D_1' \in CCont$, if $A \xrightarrow[C_1]{D_1}{}^{a} A'$ and $C_1 \xrightarrow[C_\emptyset]{A(D_1)}{}^{c} C_1'$ and $D_1 \xrightarrow[C_1(A)]{C_\emptyset}{}^{d} D_1'$ then $\exists C_2' \in$

$C \models_{C'} p$ iff $C'(C) \in \mathcal{V}(p)$

$C \models_{C'} \bot$ never

$C \models_{C'} \top$ always

$C \models_{C'} \neg\phi$ iff $C \not\models_{C'} \phi$

$C \models_{C'} \phi \wedge \psi$ iff $C \models_{C'} \phi$ and $C \models_{C'} \psi$

$C \models_{C'} \phi \vee \psi$ iff $C \models_{C'} \phi$ or $C \models_{C'} \psi$

$C \models_{C'} \phi \to \psi$ iff $C \models_{C'} \phi$ implies $C \models_{C'} \psi$

$C_1 \models_{C_2} \langle a \rangle \phi$ iff $\exists\, C_1', C_2', b$ such that if $C_1 \xrightarrow[C_2]{C_\emptyset}{}^{a} C_1'$ and $C_2 \xrightarrow[C_\emptyset]{C_1}{}^{b} C_2'$, then $C_1' \models_{C_2'} \phi$

$C_1 \models_{C_2} [a]\phi$ iff $\forall\, C_1', C_2', b$ such that if $C_1 \xrightarrow[C_2]{C_\emptyset}{}^{a} C_1'$ and $C_2 \xrightarrow[C_\emptyset]{C_1}{}^{b} C_2'$, then $C_1' \models_{C_2'} \phi$

$R, E \models_{C'} I$ iff $R = e$ and $E \sim \mathbf{1}$

$R, E \models_{C'} \phi * \psi$ iff $\exists\, S, T, F, G$ such that $R = S \circ T$, $E \sim F \times G$, and $S, F \models_{C'(T,[]\times G)} \phi$ and $T, G \models_{C'(S,F\times[])} \psi$

$R, E \models_{C'} \phi \mathrel{-\!\!*} \psi$ iff $\forall\, S, F$ such that $R \circ S$ is defined and $S, F \models_{C'} \phi$, $R \circ S, E \times F \models_{C'} \psi$

Figure 2: Interpretation of Propositional Formulae

$OCont, B' \in Cont, D_2' \in CCont$ such that $B \xrightarrow[C_2]{D_2}{}^{a} B'$ and $C_2 \xrightarrow[C_\emptyset]{B(D_2)}{}^{c} C_2'$ and $D_2 \xrightarrow[C_2(B)]{C_\emptyset}{}^{d} D_2'$ and $C_1', A', D_1' \approx C_2', B', D_2'$

2. $\forall C_2' \in OCont, B' \in Cont, D_2' \in CCont$, if $B \xrightarrow[C_2]{D_2}{}^{a} B'$ and $C_2 \xrightarrow[C_\emptyset]{B(D_2)}{}^{c} C_2'$ and $D_2 \xrightarrow[C_2(B)]{C_\emptyset}{}^{d} D_2'$ then $\exists C_1' \in OCont, A' \in Cont, D_1' \in CCont$ such that $A \xrightarrow[C_1]{D_1}{}^{a} A'$ and $C_1 \xrightarrow[C_\emptyset]{A(D_1)}{}^{c} C_1'$ and $D_1 \xrightarrow[C_1(A)]{C_\emptyset}{}^{d} D_1'$ and $C_1', A', D_1' \approx C_2', B', D_2'$

3. $R = S$.

The union of any set of relations that satisfy these two conditions also satisfies these conditions, so the largest such relation is well-defined. We define $C_1, A \approx C_2, B$ whenever $C_1, A, D \approx C_2, B, D$, for all D.

Fundamentally, this equivalence relation starts from the view that processes should be considered equivalent whenever they have the same behaviour given the same resources and context. The local equivalence relation fails to be a congruence, however, as it is not respected by the product constructor, $\times$, for processes [9]. Therefore, we do not have an analogue of Theorem 1 for local equivalence. (Note that, in [12, 9, 11], the equivalence corresponding to the equivalence $\sim$ taken here is referred to as the *global equivalence*.)

We can, however, obtain a version of the full Hennessy-Milner theorem, provided we restrict the logic to the fragment without $\mathrel{-\!\!*}$. The need for this restriction arises from the failure of the local equivalence to be a congruence, because the satisfaction relation for $\mathrel{-\!\!*}$ requires that two subsystems be combined using $\times$.

Consider the fragment of the logic that excludes $\mathrel{-\!\!*}$. Assume that all atomic propositions are values as sets of contexts that are also closed under $\approx$. Alter the I and $*$ clauses of the interpretation so that

$C \models_{C_1} I$ iff $C_1, C \approx C_1, (e, \mathbf{1})$

$C \models_{C_1} \phi * \psi$ iff $\exists\, S, T$ and F, G such that $C_1, C \approx C_1, (S \circ T, F \times G)$, and $S, F \models_{C_2} \phi$ and $T, G \models_{C_3} \psi$, where $C_2 = C'(T, [\,] \times G)$ and $C_3 = C'(S, F \times [\,])$.

Define two contexts (with accompanying outer contexts) to be logically equivalent if they satisfy exactly the same set of logical statements; that is, $C_1, A \equiv C_2, B$ if and only if, for all ϕ, $A \models_{C_1} \phi$ iff $B \models_{C_2} \phi$. The following version of Theorem 2 then holds:

THEOREM 3. *If $C_1, A, D_1 \approx C_2, B, D_2$, then $C_1, A(D_1) \equiv C_2, B(D_2)$.*

PROOF. Standard, by induction over the definition of $A(D_1) \models_{C_1} \phi$, using a forward-only analogous version of Proposition 1 for $\approx$, in the cases where contexts are extended. □

We can now also obtain a converse, for the local equivalence relation. Define a context to be *image finite* if it has finitely many immediate derivatives (for any given inner and outer contexts with which it reduces). We then have the following.

THEOREM 4. *If $C_1, A \equiv C_2, B$, then there exist A_1, D_1, B_1, D_2 such that $A = A_1(D_1)$, $B = B_1(D_2)$, and $C_1, A_1, D_1 \approx C_2, B_1, D_2$.*

PROOF. By contradiction. Take the finite set of contexts $\mathcal{C}$ that can be obtained through the reduction of $B_1(D_2)$ in outer context C_2 (with an empty inner context). If this set is empty, then we can show that $A_1(D_1) \models_{C_1} \langle a \rangle \top$ and $B_1(D_2) \not\models_{C_2} \langle a \rangle \top$, which contradicts the premiss that $C_1, A \equiv C_2, B$. If the set is non-empty, then we can construct characteristic formulae ϕ_i for each context in $\mathcal{C}$, such that the result of reducing $A_1(D_1)$ in C_1 satisfies the formula, but the result of reducing $B_1(D_2)$ in C_2 does not. We can combine these to show that $A_1(D_1) \models_{C_1} \langle a \rangle(\phi_1 \wedge ... \wedge \phi_n)$ and $B_1(D_2) \not\models_{C_2} \langle a \rangle(\phi_1 \wedge ... \wedge \phi_n)$, which again contradicts the premiss that $C_1, A \equiv C_2, B$. □

We remark that the usefulness of this result is limited by the failure of local equivalence to be a congruence. It is a strictly local reasoning tool.

Since each utility function u induces a preference relation $\preceq_u$ on closed contexts, the language could easily be enriched with preference modalities such as $\langle \preceq_u \rangle$ and $[\preceq_u]$ in the style of dynamic preference logic [42]. Consider the necessitated formula, $[\preceq_u]\phi$, which denotes that any context that is valued at least as much as the current context (in the outer context) satisfies property ϕ. Formally, this is interpreted as

$$C \vDash_{C'} [\preceq_u]\phi \quad \text{iff} \quad \begin{array}{l}\text{for all } C'',\ C'(C) \preceq_u C'(C'') \text{ implies} \\ C'' \vDash_{C'} \phi.\end{array}$$

These modalities can interact powerfully with existing structural operators.

In game-theoretic approaches to security, the notion of a level of security is important. That is, if a defender chooses to perform some defensive action, then no matter what a given attacker does, the defender is guaranteed to maintain at least a certain level of security. With preference modalities we can make statements relevant to security levels. For example, if a defensive measure d is in place, then every better state for an attacker (which would be chosen by the attacker) involves not attacking. To see this, consider the proposition

$$\phi \multimap [d][\preceq_v](\neg\langle a\rangle\top),$$

where the attacker is characterized by the property ϕ and has preference function v. The multiplicative implication operator permits us to reason about substitution within arbitrary contexts, and hence of the efficacy of defensive measures relative to an arbitrary (partially) described attacker.

The logic might also be enriched to handle expected utility [24]. Quantitative path-based logical properties of Markov Chains are studied in [22]: they can reason about complex notions, such as average utility with a given time discount, but do not provide compositionality results over model structures. A more extensive study of such extensions is future work.

6. TRUST DOMAINS

An agent, situated within a system that contains also other agents, may establish a part of the system, or a collection of other agents within the system, that it trusts. Similarly, a system's designer or manager might establish a collection of parts of the system such that, within any given part, the agents trust one another. We shall refer to such a part of the system, or such a collection of agents, as a 'trust domain'.

The term 'trust domain' is in use in range of settings, such as the Trusted Computing Project (`www.trustedcomputing.org.uk`), the Open Trusted Computing (OpenTC) consortium (`www.opentc.net`), and the 'Trust Domains' project (`www.hpl.hp.com/research/cloud_security/TrustDomains.pdf`). The literature on models of trust is very large and cannot be surveyed in this short article, but a good survey with a relevant perspective for us is [36].

In this section, we consider how the process-utility calculus might be used to characterize a notion of a 'trust domain'. Within a system model, with an agent is represented as a process, at any given point in the agent's execution, the process is associated with a location (which we suppress for now) within the system and has access to a collection of resources. That is, the agent has a state. As described above, the agent is also associated with a utility function. Here we interpret the utility function as a loss function, associating a *cost* $k_E(a_i)$ with each choice a_i that is made as a process executes, so that the trace σ of the process that describes agent E gives the total cost K of an agent E's execution:

$$K_E(\sigma) \quad = \quad \sum_{\sigma = a_1,\ldots,a_k} k_E(a_i). \qquad (11)$$

For now, we consider just finite traces.

The intended situation is depicted in Figure 6. Here the need for the concept of location should be apparent. Indeed, a logical or physical location would seem to be an essential component of the intended notion of domain. Informally, located agents manipulate their resource environments, but, in our formulation, they do so in contexts which characterize the extent to which they do so whilst maintaining a required logical property (intuitively, the 'trust' property) within a specified bound on cost. This approach stands in contrast to approaches in which constraints are expressed purely in terms of preferences, where impossible choices, that can be expressed logically in our setting, must represented by 'infinitely negative' utility. For brevity, we will not employ location explicitly, along the lines of the discussion in Section 2, instead trusting that the intuitions suggested in Figure 6 will make a sufficiently strong suggestion.

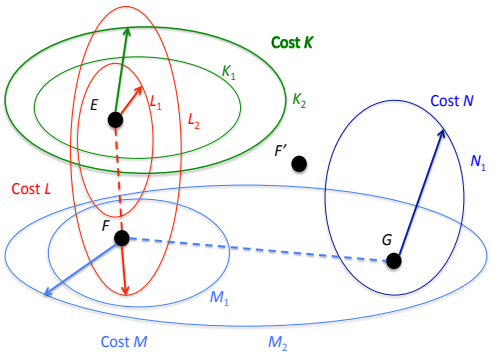


Figure 3: Iso-utilities and Trust Domains

Here the agent E may be given one of two different choices of cost (utility) function. If $K_E = K$, then F is not within E's trust domain at either the K_1 or K_2 levels. If, however, $K_E = L$, then F is within E's trust domain at the L_2, but not at the L_1 level. Agent F's cost function, M, includes agent G at the M_2 level, but not at the M_1 level ($M_1 \leq M_2$). F' is in no-one's domain at any of the given levels of utility.

The formal definition of a trust domain is set up, using the process-utility calculus, for an agent E together with a property ϕ required (by the agent or by the designer/manager) of the part of the system or collection of agents that is to be trusted, the agent's utility function K_E, which assigns values to choices made as the agent executes, and a bound K on the total cost of the trace, which characterizes the total acceptable cost to the agent in reaching or interacting with other parts of the system or other agents within it.

The trust domain is then constructed as a collection of contexts within which the agent may evolve whilst maintaining the properties by which it determines trust. Two properties are required to establish a viable definition. First,

a bound K on the cost that E is prepared to incur. Second, a propositional assertion ϕ about the state to which E can evolve within that cost constraint. So, if R is the resource initially associated with E, then we can define, building on (11),

$$\mathrm{TD}(E,\phi,K_E,K) = \Big\{ C \mid \text{there exist closed } F \text{ and trace } \sigma \text{ such that } R, E \xrightarrow[C]{(e,\mathbf{1})}{}^{\sigma} S, F \text{ and } S, F \models_{C'} \phi \text{ and } K_E(\sigma) \leq K \Big\}, \quad (12)$$

where the resource S is that which is derived from R by the the trace σ and $C \xrightarrow[C]{R,E}{}^{\sigma'} C'$, where σ' is the trace of context actions corresponding to σ.

Notice that the inner context is empty, so imposing no restriction on the evolutions considered. More generally, in further research, we could generalize the definition to consider trust domains for agents A with non-empty inner contexts, corresponding to a degree of under-specification. A richer treatment of logical satisfaction would then be needed.

Extending our banking example, we can give a simple sketch of how trust domains work. Consider a process

$$Banker \times Lawyer \times P, \quad (13)$$

where P is either *Attacker* or *Client*, and where we modify *Banker* and introduce *Lawyer*, as follows:

$$Banker = shareL : Banker' +_K notshareL : Banker'$$
$$Lawyer = 1 : (shareP : Lawyer' +_L notshareP : Lawyer').$$

In terms of Figure 6, *Banker* corresponds to A, *Lawyer* to B, and P to C. Letting *Banker*'s cost be L, *Lawyer*'s cost be M, and letting *shareL*, *shareC*, and *shareA*, etc., be the evident sharing (of data, say) actions with lawyer, client, and attacker, we obtain

$$L(shareL : Banker' \times 1 : Lawyer \times 1 : Attacker) \geq L(notshareL : Banker' \times 1 : Lawyer \times 1 : Attacker),$$

but

$$L(shareL : Banker' \times 1 : Lawyer \times 1 : Client) \leq L(notshareL : Banker' \times 1 : Lawyer \times 1 : Client),$$

and

$$M(1 : Banker' \times shareA : Lawyer \times 1 : Attacker) \geq M(1 : Banker' \times notshareA : Lawyer \times 1 : Attacker),$$

but

$$M(1 : Banker' \times shareC : Lawyer \times 1 : Client) \leq M(1 : Banker' \times notshareC : Lawyer \times 1 : Client),$$

and see that *Banker*'s trust domain for sharing will include *Lawyer* and *Client*, but not *Attacker*.

Here, the proposition ϕ in (12) would be something like ϕ_{Banker} = 'the bank retains a good credit rating while sharing data'. Different ϕ's give different domains.

We remark that work in the economics tradition would tend to code propositional constraints within utility, and that work in logic would tend to code utility constraints propositionally. In our setting, the structure provided by the Hennessy–Milner-style logic suggests it is natural to maintain the distinction between utility and logical properties. Further work from this section is to consider information flow [1, 4] between trust domains.

7. REFERENCES

[1] S. Abramsky and J. Vaananen. From IF to BI. *Synthese*, 167:2, 207–230, 2009

[2] G. Anderson, M. Collinson, D. Pym. Utility-based Decision-making in Distributed Systems Modelling [Technical Report], TR–ABDN–CS–12–04, Department of Computing Science, University of Aberdeen, UK. 2012. Available (30 November 2012) at: `http://homepages.abdn.ac.uk/d.j.pym/pages/AndersonCollinsonPym-TR.pdf`

[3] A. Arenas. Trust and security in collaborative systems. E-Science Centre, STFC Rutherford Appleton Lab., UK.

[4] J. Barwise and J. Seligman. *Information Flow: The Logic of Distributed Systems*. Cambridge University Press, 1997.

[5] A. Beautement, R. Coles, J. Griffin, C. Ioannidis, B. Monahan, D. P. C. Author), A. Sasse, and M. Wonham. Modelling the Human and Technological Costs and Benefits of USB Memory Stick Security. In M. E. Johnson, editor, *Managing Information Risk and the Economics of Security*, 141–163. Springer, 2008.

[6] J. Brotherston and M. Kanovich. Undecidability of propositional separation logic and its neighbours. In *Proc. LICS XXV*, 130-139, 2010.

[7] D. Bucur and M. Nielsen. Secure data flow in a calculus for context awareness. In *Concurrency, Graphs and Models*, 39–456, 2008.

[8] L. Cardelli and A. D. Gordon. Anytime, anywhere. modal logics for mobile ambients. In *Proc. 27th ACM Symp on Principles of Prog. Langs.*, 365–377, 2000.

[9] M. Collinson, B. Monahan, and D. Pym. A logical and computational theory of located resource. *Journal of Logic and Computation*, 19(b):1207–1244, 2009.

[10] M. Collinson, B. Monahan, and D. Pym. Semantics for structured systems modelling and simulation. In *Proc. Simutools 2010*. ACM Digital Library, ISBN 78-963-9799-87-5, 2010.

[11] M. Collinson, B. Monahan, and D. Pym. *A Discipline of Mathematical Systems Modelling*. College Publications, 2012.

[12] M. Collinson and D. Pym. Algebra and logic for resource-based systems modelling. *Mathematical Structures in Computer Science*, 19:959–1027, 2009.

[13] G. Coulouris, J. Dollimore, and T. Kindberg. *Distributed Systems: Concepts and Design*. Addison Wesley, 2000.

[14] R. de Simone. Higher-level synchronising devices in Meije-SCCS. *Theoret. Comp. Sci.*, 37:245–267, 1985.

[15] E. Eberbach. $-calculus bounded rationality = process algebra + anytime algorithms. In *Applic. Math.: Its Perspectives and Challenges*, 213–220. Narosa, 2001.

[16] A. Friedenberg and M. Meier. The context of the game. In *Proc. TARK XII*, 134–135, 2009.

[17] D. Galmiche, D. Méry, and D. Pym. The Semantics of **BI** and Resource Tableaux. *Mathematical Structures in Computer Science* 15, 1033–1088, 2005.

[18] D. Galmiche and D. Larchey-Wendling. The Undecidability of Boolean BI through Phase Semantics. In *Proc. LICS XXV*, 140-149, 2010.

[19] M. Hennessy and R. Milner. On Observing Nondeterminism and Concurrency. LNCS 85:299-309,

1980.

[20] M. Hennessy and R. Milner. Algebraic laws for nondeterminism and concurrency. *JACM*, 32(1):137–161, 1985.

[21] J. Horty and M. Pollack. Evaluating new options in the context of existing plans. *Artif. Intel.*, 127:199–220, 2001.

[22] W. Jamroga. A temporal logic for Markov chains. *Proc. AAMAS 2008*, 607–704, ISBN: 978-0-9817381-1-6, `http://dl.acm.org/citation.cfm?id=1402298.1402321`.

[23] A. Joyal, M. Nielsen, and G. Winskel. Bisimulation from open maps. *Information and Computation*, 127(2):164–185, 1996.

[24] R. Keeney and H. Raiffa. *Decisions with multiple objectives: Preferences and value tradeoffs*. Wiley, 1976.

[25] D. Kreps. *Notes on the Theory of Choice*. Underground Classics in Economics. Westview Press, 1988.

[26] J. McCarthy. Formalizing context. In *IJCAI*, 555–562, 1993.

[27] J. McCarthy and S. Buvač. Formalizing context (expanded notes). http://www-formal.stanford.edu/jmc/mccarthy-buvac-98/context.pdf.

[28] R. Milner. Calculi for synchrony and asynchrony. *Theoret. Comp. Sci.*, 25(3):267–310, 1983.

[29] R. Milner. *Communication and Concurrency*. Prentice Hall, New York, 1989.

[30] R. Milner. *The Space and Motion of Communicating Agents*. Cambridge University Press, 2009.

[31] M. Núñez and I. Rodríguez. PAMR: A process algebra for the management of resources in concurrent systems. In *Formal Techniques for Networked and Distributed Systems (FORTE)*, 169–184, 2001.

[32] P. O'Hearn and D. Pym. The logic of bunched implications. *Bulletin of Symbolic Logic*, 5(2):215–244, June 1999.

[33] G. Plotkin. A structural approach to operational semantics. Technical Report DAIMI FN-19, Department of Computer Science, Aarhus University, 1981.

[34] D. Pym. *The Semantics and Proof Theory of the Logic of Bunched Implications*, volume 26 of *Applied Logic Series*. Kluwer Academic Publishers, 2002. Errata and Remarks maintained at publisher's website and at: `http://homepages.abdn.ac.uk/d.j.pym/pages/BI-monograph-errata.pdf`.

[35] D. Pym, P. O'Hearn, and H. Yang. Possible Worlds and Resources: The Semantics of **BI**. *Theoret. Comp. Sci.*, 315(1):257–305, 2004.

[36] S. Ramchurn, D. Huynh, and N. Jennings. Trust in multi-agent systems. *The Knowledge Engineering Review*, 19(1):1–25, 2004.

[37] J. Reynolds. Separation Logic: A Logic for Shared Mutable Data Structures. *Proceedings of the Seventeenth Annual IEEE Symposium on Logic in Computer Science*, Copenhagen, Denmark, July 22-25, 2002. IEEE Computer Society Press, 55–74, 2002.

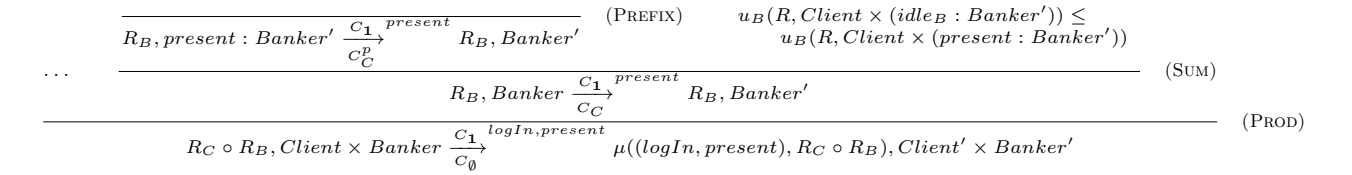
[38] J. Rutten, M. Kwiatkowska, G. Norman, and D. Parker. *Math. Tech. for Analyzing Conc. and Prob. Sys.* American Mathematical Society, 2004.

[39] P. Sewell. From rewrite rules to bisimulation congruences. *Theoretical Computer Science*, 274(1–2):183–230, 2002.

[40] F. Siewe, A. Cau, and H. Zedan. The calculus of context-aware ambients. *J. Comput. Syst. Sci.*, 77(4):597–620, 2011.

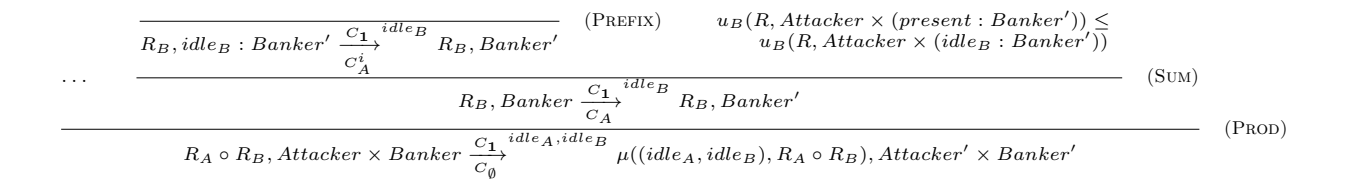
[41] C. Tofts. Processes with probability, priority and time. *Formal Aspects of Computing*, 6(5):536–564, 1994.

[42] J. van Benthem. For better or for worse: Dynamic logics of preference. *Theory and Dec. Lib. A*, 42:57–84, 2009.



where $C_{\mathbf{1}} = e, \mathbf{1}$, $C_C = R_C, Client \times [\,]$, $C_C^p = R_C, Client \times ([\,] +_{u_B} idle_B : Banker')$

Figure 4: Banker choice in Client context



where $C_{\mathbf{1}} = e, \mathbf{1}$, $C_A = R_A, Attacker \times [\,]$, $C_A^i = R_A, Attacker \times (present : Banker' +_{u_B} [\,])$

Figure 5: Banker choice in Attacker context

APPENDIX

A. DERIVATIONS OF EXAMPLES

We provide a detailed derivation of the examples introduced in Equations 3–10. This example consists of situationally dependent choices that make use of joint access control. In order to encode the joint access control, we make use of semaphore resources. We let Rs stand for sets of atomic resources, such as $Acnt$, USB, r_is, etc., and make use of the ρ notation [12, 11], defined by

$$\rho(a) = \min \{R \mid \mu(a, R) \downarrow\},$$

to denote which resources are required for the modification function to be defined for a given action:

$$\rho(logIn) = \{Acnt, r_1\} = R_C \qquad \rho(idle_C) = \{r_2\}$$
$$\rho(present) = \{USB, r_2\} = R_B \qquad \rho(idle_B) = \{r_1\}.$$

This ensures that the $idle_C$ and $present$ actions, and the $idle_B$ and $logIn$ actions, cannot co-occur in a reduction, as they require the same semaphore resources. We then define the modification function for each action:

$$\mu(logIn, R_C) = R_C \quad \mu(idle_C, \{r_2\}) = \{r_2\}$$
$$\mu(present, R_B) = R_B \quad \mu(idle_B, \{r_2\}) = \{r_2\}.$$

Let $R = \{Acnt, r_1, USB, r_2\}$. In order to denote our preference of giving the presentation over idling, in the presence of the client and the absence of the attacker, we define a portion of the banker's preference function as

$$u_B(R, Client \times (present : Banker')) = 0.7$$
$$u_B(R, Client \times (idle_B : Banker')) = 0.5.$$

We then have the reduction

$$R_C \circ R_B, Client \times Banker \xrightarrow[C_\emptyset]{C_{\mathbf{1}}}{}^{logIn, present} \mu((logIn, present), R_C \circ R_B), Client' \times Banker',$$

as derived in Figure 4. We also have the property

$$R_B, Banker \vDash_{C_C} \langle present \rangle \top.$$

This can be derived using the satisfaction relation in Figure 2, specifically the case for the diamond modality, as

$$R_B, Banker \xrightarrow[C_C]{C_{\mathbf{1}}}{}^{present} \mu((present), R_B), Banker',$$

by Figure 4.

In order to encode the joint access control we make use of semaphore resources, as defined which resources are required for the attacker's actions:

$$\rho(attack) = \{r_1\} = R_A \qquad \rho(idle_A) = \{r_2\}.$$

and define the modification function for the attackers actions

$$\mu(attack, R_A) = R_A \quad \mu(idle_A, \{r_c\}) = \{r_c\}$$

To express the banker's preference to idle in the presence of an attacker, we define a further portion of its preference function, and give a higher utility to idling in such a situation:

$$u_B(R, Attacker \times (present : Banker')) = 0.1$$
$$u_B(R, Attacker \times (idle_B : Banker')) = 0.2.$$

Here we have the reduction

$$R_A \circ R_B, Attacker \times Banker \xrightarrow[C_\emptyset]{C_{\mathbf{1}}}{}^{idle_A, idle_B} \mu((idle_A, idle_A), R_A \circ R_B), Client' \times Banker'$$

as derived in Figure 5. We also have the following property:

$$R_B, Banker \vDash_{C_A} \neg\langle present \rangle \top.$$

This can be derived using the satisfaction relation in Figure 2. By the diamond modality, as $R_B, Banker \not\xrightarrow[C_A]{C_{\mathbf{1}}}{}^{present}$, by Figure 4, we have that $R_b, Banker \not\vDash_{C_A} \langle present \rangle \top$. Then the property follows directly by the interpretation of negation.

On the Complexity of Dynamic Epistemic Logic *

Guillaume Aucher
University of Rennes 1 - INRIA
guillaume.aucher@irisa.fr

François Schwarzentruber
ENS Cachan - Brittany extension
francois.schwarzentruber@bretagne.ens-cachan.fr

ABSTRACT

Although Dynamic Epistemic Logic (DEL) is an influential logical framework for representing and reasoning about information change, little is known about the computational complexity of its associated decision problems. In fact, we only know that for public announcement logic, a fragment of DEL, the satisfiability problem and the model-checking problem are respectively PSPACE-complete and in P. We contribute to fill this gap by proving that for the DEL language with event models, the model-checking problem is, surprisingly, PSPACE-complete. Also, we prove that the satisfiability problem is NEXPTIME-complete. In doing so, we provide a sound and complete tableau method deciding the satisfiability problem.

Categories and Subject Descriptors

I.2.4 [**Knowledge representation formalisms and methods**]: Modal logic; F.1.3 [**Complexity measure and classes**]: Reducibility and completeness

General Terms

Theory

Keywords

Dynamic epistemic logic, computational complexity, model checking, satisfiability

1. INTRODUCTION

Research fields like distributed artificial intelligence, distributed computing and game theory all deal with groups of human or non-human agents which interact, exchange and receive information. The problems they address range from multi-agent planning and design of distributed protocols to strategic decision making in groups. In order to address appropriately and rigorously these problems, it is necessary to be able to provide formal means for representing and reasoning about such interactions and flows of information. The framework of Dynamic Epistemic Logic (DEL for short) is very well suited to this aim. Indeed, it is a logical framework where one can represent and reason about beliefs and knowledge change of multiple agents, and more generally about information change.

The theoretical work of the above mentioned research fields has already been applied to various practical problems stemming from telecommunication networks, World Wide Web, peer to peer networks, aircraft control systems, and so on... In general, in all applied contexts, the investigation of the algorithmic aspects of the formalisms employed plays an important role in determining whether and to what extent they can be applied. For this reason, the algorithmic aspects of DEL need to be studied.

To this aim, a preliminary step consists in studying the computational properties of its main associated decision problems, namely the model checking problem and the satisfiability problem. Indeed, it will indirectly provide algorithmic methods to solve these decision problems and give us a hint of whether and to what extent our methods can be applied. However, surprisingly little is known about the computational complexity of these problems. We only know that for public announcement logic, a fragment of DEL [Plaza, 1989], the model checking problem is in P and the satisfiability problem is PSPACE-complete. Here, we aim to fill this gap for the full language of DEL with event models.

DEL is built on top of epistemic logic. An epistemic model represents how a given set of agents perceive the actual world in terms of beliefs and knowledge about this world and about the other agents' beliefs. The insight of the DEL approach is that one can describe how an event is perceived by agents in a very similar way: an agent's perception of an event can also be described in terms of beliefs and knowledge. For example, at the battle of Waterloo, when marshal Blücher received the message of the duke of Wellington inviting him to join the attack at dawn against Napoleon, Wellington did not *know* at that very moment that Blücher was receiving his message, and Blücher *knew* it. This is a typical example of announcement which is not public. This led Baltag, Moss and Solecki to introduce the notion of *event model* [Baltag et al., 1998]. The definition of an event model, denoted $(\mathcal{M}', w')$, is very similar to the definition of an epistemic model. They also introduced a *product update*, which defines a new epistemic model representing the situation after the event. Then, they extended the epistemic language with dynamic operators $[\mathcal{M}', w']\varphi$ standing for 'φ holds after the occurrence of the event represented by $(\mathcal{M}', w')$'.

Using the so-called reduction axioms, it turns out that any formula with dynamic operator(s) can be translated to an equivalent epistemic formula without dynamic operator. As a first approximation, we could be tempted to

*An extended version of this article with full proofs can be found at the following url: `http://hal.inria.fr/docs/00/75/95/44/PDF/RR-8164.pdf`

TARK 2013, Chennai, India.

use these reduction axioms to reduce both the model checking problem and the satisfiability problem of DEL to the model checking problem and the satisfiability problem of epistemic logic, because optimal algorithmic methods already exist for these related problems. However, the reduction algorithm induced by the reduction axioms is exponential in the size of the input formula. Therefore, for the satisfiability problem, we only obtain an algorithm which is in EXPSPACE (because the satisfiability problem of epistemic logic is PSPACE-complete), and for the model checking problem, we only obtain an algorithm which is in EXPTIME (because the model checking problem of epistemic logic is in P). These algorithms are not optimal because, as we shall see, there exists an algorithm solving the satisfiability problem which is in NEXPTIME$\subseteq$ EXPSPACE and also an algorithm solving the model checking problem which is in PSPACE$\subseteq$ EXPTIME. Our algorithm for solving the satisfiability problem is based on a sound and complete tableau method which does not resort to the reduction axioms.

The paper is organized as follows. In Section 2, we recall the core of the DEL framework and the different variants of languages with event models which have been introduced in the literature. In Section 3, we prove that the model checking problem of DEL is PSPACE-complete, and in Section 4 we prove that the satisfiability problem is NEXPTIME-complete. In Section 5, we discuss related works and whether our results still hold when we extend the expressiveness of the language with common belief and 'star' iteration operators. We conclude in Section 6.

2. DYNAMIC EPISTEMIC LOGIC

Following the methodology of DEL, we split the exposition of the DEL logical framework into three subsections. In Section 2.1, we recall the syntax and semantics of the epistemic language. In Section 2.2, we define event models, and in Section 2.3, we define the product update. In Section 2.4, we recall the different languages that have been introduced in the DEL literature and we introduce our language $\mathcal{L}_{DEL}$.

2.1 Epistemic language

In the rest of the paper, ATM is a countable set of atomic propositions and AGT is a finite set of agents.

A (pointed) epistemic model $(\mathcal{M}, w)$ represents how the actual world represented by w is perceived by the agents. Intuitively, in this definition, vR_au means that in world v agent a considers that world u might be the actual world.

Definition 1 (Epistemic model).
An *epistemic model* is a tuple $\mathcal{M} = (W, R, V)$ where W is a non-empty set of possible worlds, R maps each agent $a \in AGT$ to a relation $R_a \subseteq W \times W$ and $V : ATM \to 2^W$ is a function called a valuation. We abusively write $w \in \mathcal{M}$ for $w \in W$ and we say that $(\mathcal{M}, w)$ is a *pointed epistemic model*. We also write $v \in R_a(w)$ for wR_av.

Then, we define the following epistemic language $\mathcal{L}_{EL}$. It can be used to state properties of epistemic models:

Definition 2 (Epistemic language).
The *language* $\mathcal{L}_{EL}$ of epistemic logic is defined as follows:

$$\mathcal{L}_{EL} : \varphi ::= p \mid \neg\varphi \mid (\varphi \wedge \varphi) \mid B_a\varphi$$

where p ranges over ATM and a ranges over AGT. A formula of $\mathcal{L}_{EL}$ is called an *epistemic formula*. The formula $\bot$ is an abbreviation for $p \wedge \neg p$, and $\top$ is an abbreviation for $\neg\bot$. The formula $\langle B_a\rangle\varphi$ is an abbreviation of $\neg B_a\neg\varphi$. The *size of a formula* $\varphi \in \mathcal{L}_{EL}$ is defined by induction as follows: $|p| = 1$; $|\neg\varphi| = 1+|\varphi|$; $|\varphi\wedge\psi| = 1+|\varphi|+|\psi|$; $|B_a\varphi| = 1+|\varphi|$.

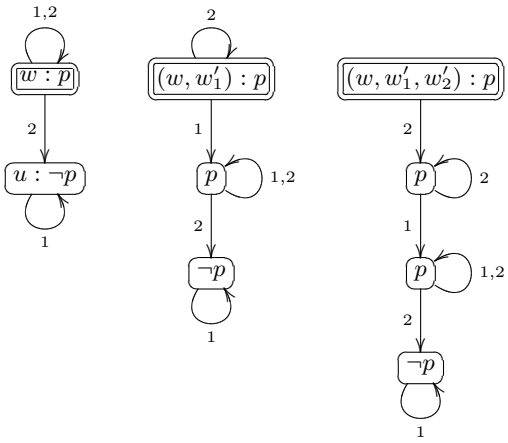


Figure 1: Pointed epistemic models $(\mathcal{M}, w)$ (*left*), $((\mathcal{M} \otimes \mathcal{M}'_1), (w, w'_1))$ (*center*) and $(\mathcal{M} \otimes \mathcal{M}'_1 \otimes \mathcal{M}'_2, (w, w'_1, w'_2))$ (*right*)

Intuitively, the formula $B_a\varphi$ reads as 'agent a believes that φ holds in the current situation'.

Definition 3 (Truth conditions).
Given an epistemic model $\mathcal{M} = (W, R, V)$ and a formula $\varphi \in \mathcal{L}_{EL}$, we define inductively the satisfaction relation $\models \subseteq W \times \mathcal{L}_{EL}$ as follows: for all $w \in W$,

$\mathcal{M}, w \models p$	iff	$w \in V(p)$
$\mathcal{M}, w \models \varphi \wedge \psi$	iff	$\mathcal{M}, w \models \varphi$ and $\mathcal{M}, w \models \psi$
$\mathcal{M}, w \models \neg\varphi$	iff	not $\mathcal{M}, w \models \varphi$
$\mathcal{M}, w \models B_a\varphi$	iff	for all $v \in R_a(w)$, we have $\mathcal{M}, v \models \varphi$

We write $\mathcal{M} \models \varphi$ when for all $w \in \mathcal{M}$, it holds that $\mathcal{M}, w \models \varphi$. Also, we write $\models \varphi$, and we say that φ is *valid*, when for all epistemic model $\mathcal{M}$, it holds that $\mathcal{M} \models \varphi$. Dually, we say that φ is *satisfiable* when $\neg\varphi$ is not valid.

Example 1. *Our running example is inspired by the coordinated attack problem from the distributed systems folklore [Fagin et al., 1995]. Our set of atomic propositions is $ATM = \{p\}$ and our set of agents is $AGT = \{1, 2\}$. Agent 1 is the duke of Wellington and agent 2 is marshal Blücher; p stands for 'Wellington wants to attack at dawn'. The initial situation is represented in Figure 1 by the pointed epistemic model $(\mathcal{M}, w) = (\{w, u\}, R_1 = \{(w, w), (u, u)\}, R_2 = \{(w, w), (w, u)\}, V(p) = \{w\})$. In this pointed epistemic model, it holds that $\mathcal{M}, w \models p \wedge B_1 p$: Wellington 'knows' that he wants to attack at dawn. It also holds that $\mathcal{M}, w \models \neg B_2 p$: Blücher does not 'know' that Wellington wants to attack at dawn; and $\mathcal{M}, w \models B_1\neg B_2 p$: Wellington 'knows' that Blücher does not 'know' that he wants to attack at dawn.*

2.2 Event model

A (pointed) event model $(\mathcal{M}', w')$ represents how the actual event represented by w' is perceived by the agents. Intuitively, in this definition, $u'R'_av'$ means that while the possible event represented by u' is occurring, agent a considers possible that the event represented by v' is in fact occurring.

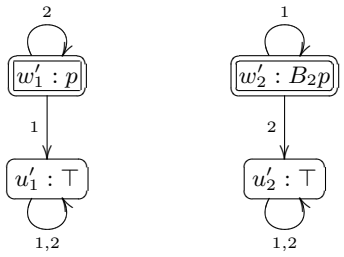


Figure 2: Pointed event models $(\mathcal{M}_1', w_1')$ (*left*) and $(\mathcal{M}_2', w_2')$ (*right*)

Definition 4 (Event model).
An *event model* is a tuple $\mathcal{M}' = (W', R', Pre)$ where W' is a non-empty and finite set of possible events, R' maps each agent $a \in AGT$ to a relation $R_a' \subseteq W' \times W'$ and $Pre : W' \to \mathcal{L}_{EL}$ is a function that maps each event to a precondition expressed in the epistemic language.

We abusively write $w' \in \mathcal{M}'$ for $w' \in W'$ and we say that $(\mathcal{M}', w')$ is a *pointed event model*. The *size of an event model* $\mathcal{M}' = (W', R', Pre)$ is noted $|\mathcal{M}'|$ and is defined as follows: $card(W') + \sum_{a \in AGT} card(R_a') + \sum_{w' \in W'} |Pre(w')|$.

Example 2. *In Figure 2 are represented two pointed event models. The first,* $(\mathcal{M}_1, w_1') = (\{w_1', u_1'\}, R_1 = \{(w_1', u_1'), (u_1', u_1')\}, R_2 = \{(w_1', w_1'), (u_1', u_1')\}, Pre, w_1')$ *where* $Pre(w_1') = p$ *and* $Pre(u_1') = \top$*, represents the event whereby Blücher receives the message of Wellington that he wants to attack at dawn. When this happens, Wellington believes that nothing happens and believes that this is even common knowledge. The second,* $(\mathcal{M}_2, w_2') = (\{w_2', u_2'\}, R_1 = \{(w_2', w_2'), (u_2', u_2')\}, R_2 = \{(w_2', u_2'), (u_2', u_2')\}, Pre, w_2')$*, where* $Pre(w_2') = B_2p$ *and* $Pre(u_2') = \top$*, represents the event whereby Wellington receives the message of Blücher telling him that he 'knows' that Wellington wants to attack at dawn.*

2.3 Product update

The following product update yields a new pointed epistemic model $\mathcal{M} \otimes \mathcal{M}', (w, w')$ representing how the new situation which was previously represented by $(\mathcal{M}, w)$ is perceived by the agents after the occurrence of the event represented by $(\mathcal{M}', w')$.

Definition 5 (Product update).
Let $\mathcal{M} = (W, R, V)$ be an epistemic model and let $\mathcal{M}' = (W', R', Pre)$ be an event model. The *product update of* $\mathcal{M}$ *by* $\mathcal{M}'$ is the epistemic model $\mathcal{M} \otimes \mathcal{M}' = (W'', R'', V'')$ defined as follows (p and a range over ATM and AGT respectively):

$$W'' = \{(w, w') \in W \times W' \mid \mathcal{M}, w \models Pre(u')\}$$
$$R_a'' = \{\langle (w, w'), (v, v') \rangle \in W'' \times W'' \mid wR_av \text{ and } w'R_a'v'\}$$
$$V''(p) = \{(w, w') \in W'' \mid w \in V(p)\}$$

Given a pointed epistemic model $(\mathcal{M}, w)$, and a pointed event model $(\mathcal{M}', w')$, we say that $(\mathcal{M}', w')$ is *executable* in $(\mathcal{M}, w)$ when $\mathcal{M}, w \models Pre(w')$. If $\mathcal{M}$ is an epistemic model and $\mathcal{M}_1', \ldots, \mathcal{M}_n'$ are event models, we abusively write $\mathcal{M} \otimes \mathcal{M}_1' \otimes \cdots \otimes \mathcal{M}_n'$ for $(\ldots((\mathcal{M} \otimes \mathcal{M}_1') \otimes \mathcal{M}_2') \otimes \ldots) \otimes \mathcal{M}_n'$ and $(w, w_1', \ldots, w_n')$ for $(\ldots((w, w_1'), w_2'), \ldots), w_n')$.

Example 3. *The pointed epistemic models* $((\mathcal{M} \otimes \mathcal{M}_1'), (w, w_1'))$ *and* $(\mathcal{M} \otimes \mathcal{M}_1' \otimes \mathcal{M}_2', (w, w_1', w_2'))$ *are represented in Figure 1. After Blücher receives the message of Wellington, Blücher 'knows' that Wellington wants to attack at dawn, but Wellington does not 'know' that Blücher 'knows' it:* $\mathcal{M} \otimes \mathcal{M}_1', (w, w_1') \models p \wedge B_2p \wedge \neg B_1B_2p$*. Likewise, after Wellington receives the message of Blücher telling him that he 'knows' that he wants to attack at dawn (*B_2p*), Wellington 'knows' that Blücher 'knows' that he wants to attack at dawn, but Blücher does not 'know' that Wellington 'knows' it:* $\mathcal{M} \otimes \mathcal{M}_1' \otimes \mathcal{M}_2', (w, w_1', w_2') \models p \wedge B_2p \wedge B_1B_2p \wedge \neg B_2B_1B_2p$*. Hence, in particular,* $\mathcal{M}, w \models \neg[\mathcal{M}_1', w_1'][\mathcal{M}_2', w_2']B_2B_1B_2p$.

2.4 Languages of DEL

In [Baltag et al., 1998], the language is defined as follows:

$$\varphi \ ::= \ p \mid \neg\varphi \mid (\varphi \wedge \varphi) \mid B_a\varphi \mid [\mathcal{M}', w']\varphi$$

where p ranges over ATM, a over AGT and $(\mathcal{M}', w')$ is any pointed and finite event model. The formula $\langle \mathcal{M}', w' \rangle \varphi$ is an abbreviation for $\neg[\mathcal{M}', w']\neg\varphi$.

Intuitively, $[\mathcal{M}', w']\varphi$ reads as 'φ will hold after the occurrence of the event represented by $(\mathcal{M}', w')$' and $\langle \mathcal{M}', w' \rangle \varphi$ reads as 'the event represented by $(\mathcal{M}', w')$ is executable in the current situation and φ will hold after its execution'.

However, note that in this definition, preconditions of event models are necessarily epistemic formulas. In [Baltag and Moss, 2004], another language is introduced which can deal with event models whose preconditions may involve formulas with event models. This language relies on the notion of *event signature* and the epistemic language is extended with a modality $[\Sigma, \varphi_1, \ldots, \varphi_n]\varphi$, where Σ is an event signature. The language of [Baltag and Moss, 2004] also includes PDL-like program constructions such as sequential composition, union and 'star' operation of event models (see Section 5 for a definition of these program constructions).

In [van Ditmarsch et al., 2007], preconditions can also be formulas involving event models, but only union of programs is allowed. It is therefore a fragment of the language of [Baltag and Moss, 2004] since it does not include sequential composition nor the 'star' operation. This will be our language in this paper.

Definition 6 ([van Ditmarsch et al., 2007]).
The language $\mathcal{L}_{DEL}$ is the union of the *formulas* $\varphi \in \mathcal{L}_{\otimes}^{stat}$ and the *events* (or *epistemic events*) $\pi \in \mathcal{L}_{\otimes}^{dyn}$ defined by the following rule:

$$\mathcal{L}_{\otimes}^{stat} : \varphi \ ::= p \mid \neg\varphi \mid (\varphi \wedge \varphi) \mid B_a\varphi \mid [\pi]\varphi$$
$$\mathcal{L}_{\otimes}^{dyn} : \pi \ ::= \mathcal{M}', w' \mid (\pi \cup \pi)$$

where p ranges over ATM, a over AGT and $(\mathcal{M}', w')$ is any pointed and finite event model such that for all $w' \in \mathcal{M}'$, $Pre(w')$ is a formula of $\mathcal{L}_{\otimes}^{stat}$ that has already been constructed in a previous stage of the inductively defined hierarchy.

The *size of* $\varphi \in \mathcal{L}_{DEL}$ is defined as for the epistemic language together with the induction case $|[\pi]\varphi| = 1 + |\pi| + |\varphi|$ where $|\mathcal{M}', w'| = |\mathcal{M}'|$, and $|\pi \cup \gamma| = 1 + |\pi| + |\gamma|$.

Definition 7 (Truth conditions).
Given an epistemic model $\mathcal{M} = (W, R, V)$ and a formula $\varphi \in \mathcal{L}_{DEL}$, we define inductively the satisfaction relation

$\models \subseteq W \times \mathcal{L}_{DEL}$ as follows:

$$\begin{array}{lll} \mathcal{M}, w \models [\mathcal{M}', w']\varphi & \text{iff} & \mathcal{M}, w \models Pre(w') \text{ implies} \\ & & \mathcal{M} \otimes \mathcal{M}', (w, w') \models \varphi \\ \mathcal{M}, w \models [\pi \cup \gamma]\varphi & \text{iff} & \mathcal{M}, w \models [\pi]\varphi \text{ and } \mathcal{M}, w \models [\gamma]\varphi. \end{array}$$

The other induction steps are identical to the induction steps of Definition 3.

The results in this paper are the same whether or not the formulas of the preconditions involve event models. However, the result of NEXPTIME-completeness of the satisfiability problem of Section 4 holds only if we consider union of event models as a program construction in the language.

3. MODEL CHECKING PROBLEM

The *model checking problem of* $\mathcal{L}_{DEL}$ is defined as follows:

> **Input**: a pointed epistemic model $(\mathcal{M}, w)$ and a formula $\varphi \in \mathcal{L}_{DEL}$;
>
> **Output**: yes iff $\mathcal{M}, w \models \varphi$.

Whereas the model checking problem with an epistemic formula of $\mathcal{L}_{EL}$ is in P, model checking with a formula of $\mathcal{L}_{DEL}$ is surprisingly PSPACE-complete. This shows that the addition of dynamic modalities with event models to $\mathcal{L}_{EL}$ increases tremendously the computational complexity of the model checking problem.

3.1 Upper bound

In Figure 3 is defined a deterministic algorithm `M-Check`$(w \quad \mathcal{M}'_1, w'_1; \ldots; \mathcal{M}'_i, w'_i \quad, \varphi)$ that checks whether we have $\mathcal{M} \otimes \mathcal{M}'_1 \otimes \ldots \mathcal{M}'_i, (w, w'_1, \ldots, w'_i) \models \varphi$, where $(\mathcal{M}, w)$ is a pointed epistemic model and for all $j \in \{1, \ldots, i\}$, $(\mathcal{M}'_j, w'_j)$ is a pointed event model. The precondition of a call to the function `M-Check`$(w \quad \mathcal{M}'_1, w'_1; \ldots; \mathcal{M}'_i, w'_i \quad, \varphi)$ is that $(w, w'_1, \ldots, w'_i) \in \mathcal{M} \otimes \mathcal{M}'_1 \otimes \ldots \mathcal{M}'_i$, that is, the sequence $(\mathcal{M}'_1, w'_1) \ldots, (\mathcal{M}'_i, w'_i)$ is executable in $(\mathcal{M}, w)$. In order to check whether $\mathcal{M}, w \models \varphi$, we just call `M-Check`$(w, \varphi)$.

THEOREM 1. *The model checking problem of* $\mathcal{L}_{DEL}$ *is in PSPACE.*

PROOF SKETCH. Termination and correction of the algorithm `M-Check` are easily proved over the size of the input defined by $|\mathcal{M}| + \sum_{k=1}^{i} |\mathcal{M}'_k| + |\varphi|$. As for complexity, the algorithm requires a polynomial amount of space in the size of the input. Indeed, as the size of the input is strictly decreasing at each recursive call, the number of recursive calls in the call stack is linear in the size of the input. Then, each of the current call requires a polynomial amount of space in the size of the input for storing the value of local variables: the most consuming case is $B_a\psi$ where we have to save all the current values of u, $u_1, \ldots, u_i$ in the loop **for**. □

3.2 Lower bound

We prove that the algorithm of the previous section is optimal. To do so, we provide a polynomial reduction of the *quantified Boolean formula satisfiability problem*, known to be PSPACE-complete [Papadimitriou, 1995, p. 455] to the model-checking problem of $\mathcal{L}_{DEL}$.

```
function M-Check(w  M'_1, w'_1; ...; M'_i, w'_i  φ)
  match (φ)
    case p:
      return w ∈ V(p);
    case ¬ψ:
      return not M-Check(w  M'_1, w'_1; ...; M'_i, w'_i  ψ);
    case ψ_1 ∧ ψ_2:
      return (M-Check(w  M'_1, w'_1; ...; M'_i, w'_i  ψ_1) and
        M-Check(w  M'_1, w'_1; ...; M'_i, w'_i  ψ_2));
    case B_a ψ:
      for u ∈ R_a(w)
      for u'_1 ∈ R'_a(w'_1)
      if M-Check(u, Pre(u'_1))
      :
      for u'_i ∈ R'_a(w'_i)
      if M-Check(u  M'_1, u'_1; ...; M'_{i-1}, u'_{i-1}  Pre(u'_i))
        if not M-Check(u  M'_1, u'_1; ...; M'_i, u'_i  ψ);
          return false ;
        endIf
      endIf endFor ... endIf endFor endFor
      return true ;
    case [M', w']ψ:
      if M-Check(w  M'_1, w'_1; ...; M'_i, w'_i  Pre(w'))
        return M-Check(w  M'_1, w'_1; ...; M'_i, w'_i; M', w'  ψ);
      endIf
      return true ;
    case [π ∪ γ]ψ:
      return (M-Check(w  M'_1, w'_1; ...; M'_i, w'_i  [π]ψ) and
        M-Check(w  M'_1, w'_1; ...; M'_i, w'_i  [γ]ψ));
  endMatch
endFunction
```

Figure 3: PSPACE algorithm for model checking

THEOREM 2. *The model checking problem of* $\mathcal{L}_{DEL}$ *is PSPACE-hard.*

PROOF. Without loss of generality, we only consider in this proof quantified Boolean formulas of the form $\forall p_1 \exists p_2 \forall p_3 \ldots \forall p_{2k-1} \exists p_{2k} \psi(p_1, \ldots p_{2k})$, where $\psi(p_1, \ldots, p_{2k})$ is a Boolean formula over the atomic propositions $p_1, \ldots, p_{2k}$. The formula $\forall p_1 \exists p_2 \forall p_3 \ldots \forall p_{2k-1} \exists p_{2k} \psi(p_1, \ldots p_{2k})$ is *satisfiable* iff for both truth values of the atomic proposition p_1 there is a truth value for the atomic proposition p_2 such that for both truth values of the atomic proposition p_3, and so on up to p_{2k}, the formula $\psi(p_1, \ldots p_{2k})$ is true in the overall truth assignment.

We can restrict ourselves to $\mathcal{L}_{DEL}$ where there is only one agent a. The *quantified Boolean formula satisfiability problem* is defined as follows:

> **Input**: a natural number k and a quantified Boolean formula $\varphi \triangleq \forall p_1 \exists p_2 \forall p_3 \ldots \forall p_{2k-1} \exists p_{2k} \psi(p_1, \ldots, p_{2k})$;
>
> **Output**: yes iff φ is satisfiable.

Let $\varphi = \forall p_1 \exists p_2 \forall p_3 \ldots \forall p_{2k-1} \exists p_{2k} \psi(p_1, \ldots p_{2k})$ be a quantified Boolean formula. We define a pointed epistemic model $(\mathcal{M}, w^0)$, $2k$ pointed event models $(\mathcal{M}'_1, w'^0_1), \ldots, (\mathcal{M}'_{2k}, w'^0_{2k})$, a pointed event model $\mathcal{M}'_\circlearrowleft, w'^0_\circlearrowleft$ and an epistemic formula ψ' that are computable in polynomial time in the size of φ such that:

$$\begin{array}{c} \varphi \text{ is satisfiable in quantified Boolean logic} \\ \text{iff} \\ \mathcal{M}, w^0 \models [\mathcal{M}'_1, w'^0_1 \cup \mathcal{M}'_\circlearrowleft, w'^0_\circlearrowleft]\langle \mathcal{M}'_2, w'^0_2 \cup \mathcal{M}'_\circlearrowleft, w'^0_\circlearrowleft \rangle \ldots \\ [\mathcal{M}'_{2k-1}, w'^0_{2k-1} \cup \mathcal{M}'_\circlearrowleft, w'^0_\circlearrowleft]\langle \mathcal{M}'_{2k}, w'^0_{2k} \cup \mathcal{M}'_\circlearrowleft, w'^0_\circlearrowleft \rangle \psi'. \end{array}$$

The corresponding instance of the model checking problem of $\mathcal{L}_{DEL}$ is computable in polynomial time in the size of φ. Now, let us describe $\mathcal{M}, w_0$, the event models $\mathcal{M}'_1, w'^0_1, \ldots, \mathcal{M}'_{2k}, w'^0_{2k}$, the event model $\mathcal{M}'_\circlearrowleft, w'^0_\circlearrowleft$ and ψ'.

- $\mathcal{M} = (W, R, V)$ is defined by:
  - $W = \{w^0, w^1, \ldots, w^{2k+1}\}$;
  - $R_a = \{(w^j, w^{j+1} \mid j \in \{0, \ldots, 2k\}\}$;
  - and $V(p) = \emptyset$ for all $p \in ATM$
- For all $i \in \{1, \ldots, 2k\}$, $\mathcal{M}'_i = (W'_i, R'_i, Pre_i)$ is defined by:
  - $W'_i = \{w'^0_i, w'^1_i, \ldots, w'^i_i, w'^\circlearrowleft_i\}$
  - $R'_{i\,a} = \{(w'^j_i, w'^{j+1}_i) \mid j \in \{0, \ldots, i-1\}\} \cup \{(w'^0_i, w'^\circlearrowleft_i), (w'^\circlearrowleft_i, w'^\circlearrowleft_i)\}$
  - and $Pre_i(u') = \top$ for all $u' \in W'_i$
- $\mathcal{M}'_\circlearrowleft, w'^0_\circlearrowleft = (W'_\circlearrowleft, R'_\circlearrowleft, Pre_\circlearrowleft)$ is defined by:
  - $W'_\circlearrowleft = \{w'^0_\circlearrowleft\}$
  - $R'_{\circlearrowleft\,a} = \{(w'^0_\circlearrowleft, w'^0_\circlearrowleft)\}$
  - $Pre_\circlearrowleft(w'^0_\circlearrowleft) = \top$
- $\psi' = \psi(p_1 \leftarrow \langle B_a\rangle B_a\bot, \ldots, p_{2k} \leftarrow (\langle B_a\rangle)^{2k} B_a\bot)$, that is, ψ' is the formula ψ where all p_i occurrences are substituted by $(\langle B_a\rangle)^i B_a\bot$.[1]

The semantics is simulated in the following way. The proposition p_i is interpreted as the presence of a chain of length exactly i from the root of a given epistemic model. That is why in ψ', the proposition p_i is substituted by $(\langle B_a\rangle)^i B_a\bot$, which is true in the root of the final epistemic model iff there exists a chain of length i in that model.

Note that updating an epistemic model where there is a chain of length $2k+1$ by $\mathcal{M}'_i, w'^0_i$ where $i \in \{1, \ldots, 2k\}$:

- preserves the presence or absence of any chain of length $j \neq i$; in particular, it always preserves the presence of the chain of length $2k+1$;
- adds a chain of length i, that is p_i becomes true;

Note also that updating an epistemic model where there is a chain of length $2k+1$ by $\mathcal{M}'_\circlearrowleft, w'^0_\circlearrowleft$ preserves the presence or absence of any chain. So, it will keep p_i false if it was already false and it will keep any p_i true if it was already true. In other words, the $\mathcal{M}'_\circlearrowleft, w'^0_\circlearrowleft$ is a neutral element for the product update.

The crucial invariant property (Inv) of an epistemic model is the existence of a chain of length $2k+1$ in any update of $\mathcal{M}, w^0$ by any sequence of $\mathcal{M}'_\circlearrowleft, w'^0_\circlearrowleft$ and $\mathcal{M}'_i, w'^0_i$.

The behavior of $\forall p_i$ in quantified Boolean logic consists in a universal choice of a truth value for p_i. It is translated by the update operator $[\mathcal{M}'_i, w'^0_i \cup \mathcal{M}'_\circlearrowleft, w'^0_\circlearrowleft]$ whose semantics is to choose *universally* the update of the epistemic model by $\mathcal{M}'_i, w'^0_i$, that will give a new updated epistemic model with a chain of length i, that is p_i is true, or by $\mathcal{M}'_\circlearrowleft, w'^0_\circlearrowleft$ that will let the new updated epistemic model without a chain of length i, that is p_i is false.

[1]The formula $(\langle B_a\rangle)^i\varphi$ is an abbreviation of $\underbrace{\langle B_a\rangle \ldots \langle B_a\rangle}_{i \text{ times}} \varphi$.

The behavior of $\exists p_i$ in quantified Boolean logic consists in an existential choice of a truth value for p_i. It is translated by the update operator $\langle \mathcal{M}'_i, w'^0_i \cup \mathcal{M}'_\circlearrowleft, w'^0_\circlearrowleft\rangle$ whose semantics is to choose *existentially* the update of the epistemic model by $\mathcal{M}'_i, w'^0_i$, that will give a new updated epistemic model with a chain of length i, that is p_i is true, or by $\mathcal{M}'_\circlearrowleft, w'^0_\circlearrowleft$, that will let the new updated epistemic model without a chain of length i, that is p_i is false. □

REMARK 1. *Note that the reduction used to prove that the model checking problem of $\mathcal{L}_{DEL}$ is PSPACE-hard uses only the precondition $\top$.*

4. SATISFIABILITY PROBLEM

The *satisfiability problem of* $\mathcal{L}_{DEL}$ is defined as follows:

Input: a formula $\varphi \in \mathcal{L}_{DEL}$;

Output: yes iff there exists a pointed epistemic model $(\mathcal{M}, w)$ such that $\mathcal{M}, w \models \varphi$.

The satisfiability problem is known to be decidable. Indeed, the standard reduction axioms of DEL [Baltag and Moss, 2004, p. 214] induce a translation $tr : \mathcal{L}_{DEL} \to \mathcal{L}_{EL}$ such that $\varphi \in \mathcal{L}_{DEL}$ is satisfiable iff $tr(\varphi) \in \mathcal{L}_{EL}$ is satisfiable. Since the size of $tr(\varphi)$ is at most exponential in the size of φ [Lutz, 2006] and the satisfiability problem of $\mathcal{L}_{EL}$ is PSPACE-complete, the satisfiability problem of $\mathcal{L}_{DEL}$ is in EXPSPACE. This upper bound is nevertheless not optimal: we are going to prove in this section that the satisfiability problem of $\mathcal{L}_{DEL}$ is NEXPTIME-complete.

4.1 Upper bound

In this subsection we present a tableau method that does not rely on reduction axioms and we prove that it provides a NEXPTIME procedure deciding the satisfiability problem.

4.1.1 Tableau method

Let $\mathfrak{Lab}$ be a countable set of labels designed to represent worlds of the epistemic model $(\mathcal{M}, w)$. Our tableau method manipulates terms that we call tableau terms and they are of the following kind:

- $(\sigma \quad \mathcal{M}'_1, w'_1; \ldots; \mathcal{M}'_i, w'_i \quad \varphi)$ where $\sigma \in \mathfrak{Lab}$ is a *node* (that represents a world in the initial model) and for all $j \in \{1, \ldots, i\}$, $\mathcal{M}'_j, w'_j$ is an event model. This term means that φ is true in the world denoted by σ after the execution of the sequence $\mathcal{M}'_1, w'_1, \ldots, \mathcal{M}'_i, w'_i$ and that the sequence is executable in the world denoted by σ;
- $(\sigma \quad \mathcal{M}'_1, w'_1; \ldots; \mathcal{M}'_i, w'_i \quad \checkmark)$ means that the sequence $\mathcal{M}'_1, w'_1, \ldots, \mathcal{M}'_i, w'_i$ is executable in the world denoted by σ;
- $(\sigma \quad \mathcal{M}'_1, w'_1; \ldots; \mathcal{M}'_i, w'_i \quad \otimes)$ means that the sequence $\mathcal{M}'_1, w'_1, \ldots, \mathcal{M}'_i, w'_i$ is not executable in the world denoted by σ;
- $(\sigma R_a \sigma_1)$ means that the world denoted by σ is linked by R_a to the world denoted by σ_1;
- $\bot$ denotes an inconsistency.

A *tableau rule* is represented by a *numerator* $\mathcal{N}$ above a line and a finite list of *denominators* $\mathcal{D}_1, \ldots, \mathcal{D}_k$ below this line, separated by vertical bars:

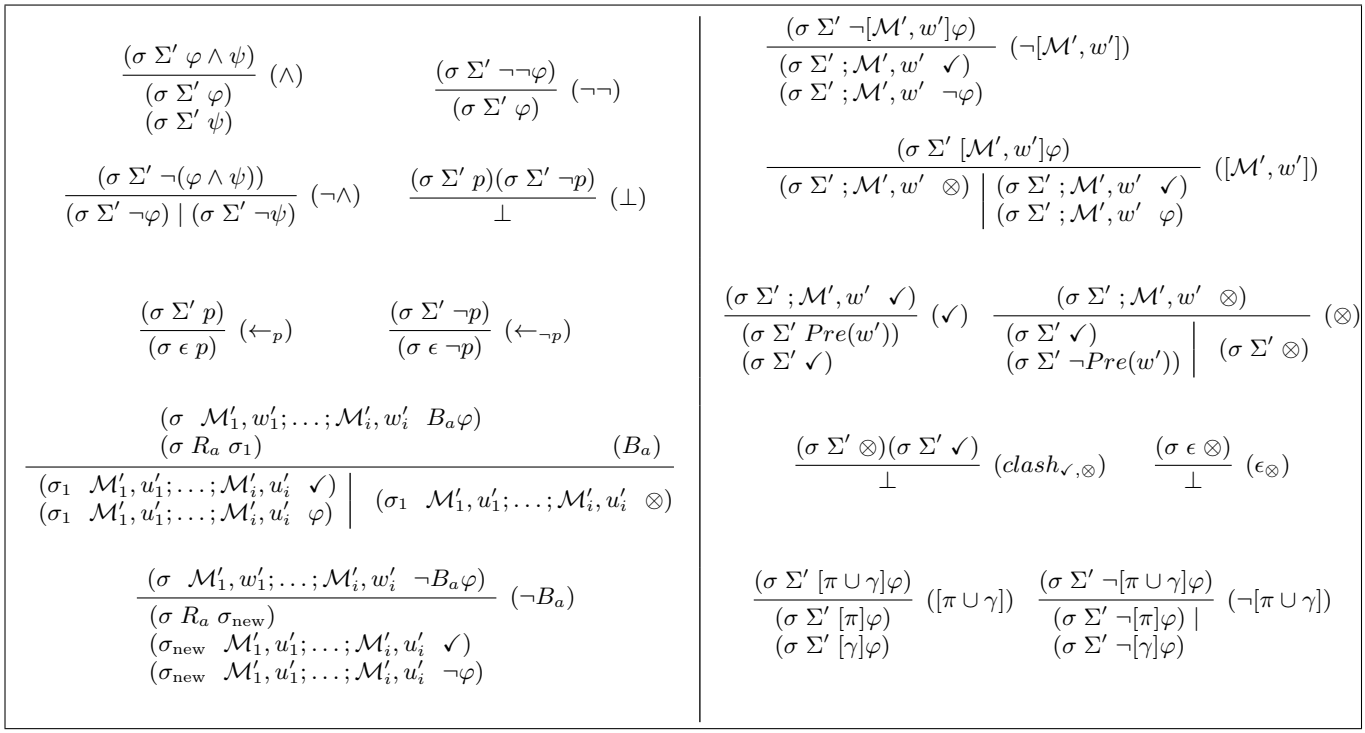


Figure 4: Tableau rules

$$\frac{\mathcal{N}}{\mathcal{D}_1 \mid \ldots \mid \mathcal{D}_k}$$

The numerator and the denominators are finite sets of tableau terms.

A *tableau tree* is a finite tree with a set of tableau terms at each node. A rule with numerator $\mathcal{N}$ and denominator $\mathcal{D}$ is *applicable* to a node carrying a set Γ if Γ contains an instance of $\mathcal{N}$ but not the instance of its denominator $\mathcal{D}$. If no rule is applicable, Γ is said to be *saturated*. We call a node σ an *end node* if the set of formulas Γ it carries is saturated, or if $\bot \in \Gamma$. The tableau tree is extended as follows:

1. Choose a leaf node n carrying Γ where n is not an end node, and choose a rule ρ applicable to n.
2. (a) If ρ has only one denominator, add the appropriate instantiation to Γ.
   (b) If ρ has multiple denominators, choose one of them and add to Γ the appropriate instantiation of this denominator.

A branch in a tableau tree is a path from the root to an end node. A branch is *closed* if its end node contains $\bot$, otherwise it is *open*. A tableau tree is *closed* if all its branches are closed, otherwise it is *open*. The *tableau tree for a formula* $\varphi \in \mathcal{L}_{DEL}$ is the tableau tree obtained from the root $\{(\sigma_0\ \epsilon\ \varphi)\}$ when all leafs are end nodes. We write $\vdash \varphi$ when the tableau for $\neg\varphi$ is closed.

The tableau rules of our tableau method are represented in Figure 4. In these rules, Σ' is a list of pointed event models $\mathcal{M}'_1, w'_1, \ldots, \mathcal{M}'_i, w'_i$ and ϵ is the empty list. The tableau method contains the classical Boolean rules $(\wedge), (\neg\neg), (\neg\wedge)$. The rules $(\leftarrow_p)$ and $(\leftarrow_{\neg p})$ handle atomic propositions. The rule $(\bot)$ makes the current execution fail. The rule for (B_a) is applied for all $j \in \{1, \ldots i\}$ and all u'_j such that $w'_j R'_a u'_j$. Similarly, the rule for $(\neg B_a)$ is applied by choosing non-deterministically for all $j \in \{1, \ldots i\}$ some u'_j such that $w'_j R'_a u'_j$ and creating a new fresh label σ_{new}. The rules $(\checkmark)$, $(\otimes)$, $(clash_{\checkmark,\otimes})$ and $(\epsilon_\otimes)$ handle the preconditions. The last two rules $([\pi \cup \gamma])$ and $(\neg[\pi \cup \gamma])$ handle the union operator.

THEOREM 3 (SOUNDNESS AND COMPLETENESS). *Let* $\varphi \in \mathcal{L}_{DEL}$. *It holds that* $\vdash \varphi$ *iff* $\models \varphi$.

EXAMPLE 4. *We prove with our tableau method that the formula* $\varphi = \neg[\mathcal{M}'_1, w'_1][\mathcal{M}'_2, w'_2]B_2B_1B_2p$ *from Example 3 is satisfiable, where* $\mathcal{M}'_1, w'_1$ *and* $\mathcal{M}'_2, w'_2$ *are defined in Example 2. In Figure 5, an open branch of the tableau tree for* φ *is represented. The set* Σ_{22} *is saturated: no more tableau rule is applicable. From this branch, we may extract a pointed epistemic model* $(\mathcal{M}, \sigma_0)$ *such that* $\mathcal{M}, \sigma_0 \models \varphi$.

4.1.2 NEXPTIME-membership

THEOREM 4. *The satisfiability problem of* $\mathcal{L}_{DEL}$ *is in NEXPTIME.*

PROOF SKETCH. Termination of our tableau method is proved by defining the size of a term $(\sigma\ \Sigma'\ \varphi)$ by $1 + \sum_{(\mathcal{M}', w') \in \Sigma'} (|\mathcal{M}'| + 1) + |\varphi|$. The depth of the tableau tree is linear in the size of the input formula, but the number of tableau terms at a node σ may be exponential, because of rule $(\neg B_a)$. As a consequence, the tableau tree has at most an exponential number of nodes and constructing non-deterministically such a tree can been done in an exponential amount of time. So, the procedure is in NEXPTIME. □

4.2 Lower bound

$$\Sigma_0 := \{(\sigma_0 \in \varphi)\}$$
$$\downarrow \quad (\neg[\mathcal{M}', w'])$$
$$\Sigma_1 := \Sigma_0 \cup \left\{ \begin{array}{l} (\sigma_0\ \mathcal{M}'_1, w'_1\ \checkmark) \\ (\sigma_0\ \mathcal{M}'_1, w'_1 \neg[\mathcal{M}'_2, w'_2] B_2 B_1 B_2 p) \end{array} \right\}$$
$$\downarrow \quad (\checkmark)$$
$$\Sigma_2 := \Sigma_1 \cup \{\ (\sigma_0 \in \checkmark), (\sigma_0 \in p)\ \}$$
$$\downarrow \quad (\neg[\mathcal{M}', w'])$$
$$\Sigma_3 := \Sigma_2 \cup \left\{ \begin{array}{l} (\sigma_0\ \mathcal{M}'_1, w'_1; \mathcal{M}'_2, w'_2\ \checkmark) \\ (\sigma_0\ \mathcal{M}'_1, w'_1; \mathcal{M}'_2, w'_2\ \neg B_2 B_1 B_2 p) \end{array} \right\}$$
$$\downarrow \quad (\checkmark)$$
$$\Sigma_4 := \Sigma_3 \cup \{\ (\sigma_0\ \mathcal{M}'_1, w'_1\ \checkmark), (\sigma_0\ \mathcal{M}'_1, w'_1\ B_2 p)\ \}$$
$$\downarrow \quad (\neg B_a)$$
$$\Sigma_5 := \Sigma_4 \cup \left\{ \begin{array}{l} (\sigma_0\ R_2\ \sigma_1) \\ (\sigma_1\ \mathcal{M}'_1, w'_1; \mathcal{M}'_2, u'_2\ \checkmark) \\ (\sigma_1\ \mathcal{M}'_1, w'_1; \mathcal{M}'_2, u'_2\ \neg B_1 B_2 p) \end{array} \right\}$$
$$\downarrow \quad (B_a)$$
$$\Sigma_6 := \Sigma_5 \cup \{\ (\sigma_1\ \mathcal{M}'_1, w'_1\ \checkmark), (\sigma_1\ \mathcal{M}'_1, w'_1\ p)\ \}$$
$$\downarrow \quad (\checkmark)$$
$$\Sigma_7 := \Sigma_6 \cup \{\ (\sigma_1\ \mathcal{M}'_1, w'_1\ \checkmark), (\sigma_1\ \mathcal{M}'_1, w'_1\ \neg(p \wedge \neg p))\ \}$$
$$\downarrow \quad (\neg\wedge, \neg\neg)$$
$$\Sigma_8 := \Sigma_7 \cup \{\ (\sigma_1\ \mathcal{M}'_1, w'_1\ p)\ \}$$
$$\downarrow \quad (\rightarrow_p)$$
$$\Sigma_9 := \Sigma_8 \cup \{\ (\sigma_1 \in p)\ \}$$
$$\downarrow \quad (\neg B_a)$$
$$\Sigma_{10} := \Sigma_9 \cup \left\{ \begin{array}{l} (\sigma_1\ R_1\ \sigma_2) \\ (\sigma_2\ \mathcal{M}'_1, u'_1; \mathcal{M}'_2, u'_2\ \checkmark) \\ (\sigma_2\ \mathcal{M}'_1, u'_1; \mathcal{M}'_2, u'_2\ \neg B_2 p) \end{array} \right\}$$
$$\downarrow \quad (\checkmark)$$
$$\Sigma_{11} := \Sigma_{10} \cup \{\ (\sigma_2\ \mathcal{M}'_1, u'_1\ \checkmark), (\sigma_2\ \mathcal{M}'_1, u'_1\ \neg(p \wedge \neg p))\ \}$$
$$\downarrow \quad (\neg\wedge, \neg\neg)$$
$$\Sigma_{12} := \Sigma_{11} \cup \{\ (\sigma_2\ \mathcal{M}'_1, u'_1\ p)\ \}$$
$$\downarrow \quad (\rightarrow_p)$$
$$\Sigma_{13} := \Sigma_{12} \cup \{\ (\sigma_2 \in p)\ \}$$
$$\downarrow \quad (\checkmark)$$
$$\Sigma_{14} := \Sigma_{13} \cup \{\ (\sigma_2 \in \checkmark), (\sigma_2 \in \top)\ \}$$
$$\downarrow \quad (\neg\wedge, \neg\neg)$$
$$\Sigma_{15} := \Sigma_{14} \cup \{\ (\sigma_2 \in p)\ \}$$
$$\downarrow \quad (\neg B_a)$$
$$\Sigma_{16} := \Sigma_{15} \cup \left\{ \begin{array}{l} (\sigma_2\ R_2\ \sigma_3) \\ (\sigma_3\ \mathcal{M}'_1, u'_1; \mathcal{M}'_2, u'_2\ \checkmark) \\ (\sigma_3\ \mathcal{M}'_1, u'_1; \mathcal{M}'_2, u'_2\ \neg p) \end{array} \right\}$$
$$\downarrow \quad (\rightarrow_{\neg p})$$
$$\Sigma_{17} := \Sigma_{16} \cup \{\ (\sigma_3 \in \neg p)\ \}$$
$$\downarrow \quad (\checkmark)$$
$$\Sigma_{18} := \Sigma_{17} \cup \left\{ \begin{array}{l} (\sigma_3\ \mathcal{M}'_1, u'_1\ \checkmark) \\ (\sigma_3\ \mathcal{M}'_1, u'_1; \mathcal{M}'_2, u'_2\ \neg(p \wedge \neg p)) \end{array} \right\}$$
$$\downarrow \quad (\checkmark)$$
$$\Sigma_{19} := \Sigma_{18} \cup \{\ (\sigma_3 \in \neg(p \wedge \neg p)\ \}$$
$$\downarrow \quad (\neg\wedge, \neg\neg)$$
$$\Sigma_{20} := \Sigma_{19} \cup \{\ (\sigma_3 \in \neg p)\ \}$$
$$\downarrow \quad (\neg\wedge, \neg\neg)$$
$$\Sigma_{21} := \Sigma_{20} \cup \{\ (\sigma_3\ \mathcal{M}'_1, u'_1; \mathcal{M}'_2, u'_2\ \neg p)\ \}$$
$$\downarrow \quad (\rightarrow_{\neg p})$$
$$\Sigma_{22} := \Sigma_{21} \cup \{\ (\sigma_3 \in \neg p)\ \}$$

Figure 5: An open branch of the tableau for φ

We prove that the algorithm based on our tableau method of the previous section is optimal in terms of computational complexity. To do so, we prove that the satisfiability problem of $\mathcal{L}_{DEL}$ is NEXPTIME-hard by reducing a NEXPTIME-complete tiling problem to it [Boas, 1997].

Let C be a countable and infinite set of colors. A *tile type* t is a 4-tuple of colors, denoted $t = (left(t), right(t), up(t), down(t)) \in C^4$. We consider the following *tiling problem*:

> **Input**: a finite set T of tile types, $t_0 \in T$ and a natural number k written in its binary form.
>
> **Output**: yes iff there exists a function τ from $\{0, \dots k\}^2$ to T satisfying the following constraints:
>
> $$\tau(0,0) = t_0; \tag{1}$$
>
> for all $x \in \{0, \dots, k\}$ and $y \in \{0, \dots, k-1\}$:
>
> $$up(\tau(x,y)) = down(\tau(x, y+1)); \tag{2}$$
>
> for all $x \in \{0, \dots, k-1\}$ and $y \in \{0, \dots, k\}$:
>
> $$right(\tau(x,y)) = left(\tau(x+1, y)). \tag{3}$$

In other words, the problem is to decide whether we can tile a $(k+1) \times (k+1)$ grid with the tile types of T, t_0 being placed onto $(0,0)$.

THEOREM 5. *The satisfiability problem of $\mathcal{L}_{DEL}$ is NEXPTIME -hard.*

PROOF. Without loss of generality, we assume that $k = 2^n$. Let us consider an instance of the NEXPTIME-hard tiling problem described above. Our goal is to provide a polynomial translation from this instance to an instance of the satisfiability problem of $\mathcal{L}_{DEL}$.

The idea is to embed *two identical* $k \times k$-tilings into a single tree. Each leaf of the tree represents both a position (x_1, y_1) in the first tiling and a position (x_2, y_2) in the second tiling. We need to encode *two identical* tilings because, in order to check constraints 2 and 3, we will need to refer to the tile located to the right or to the left of a given position in a tiling, and also to refer to the tile located above or below

it. This is hardly possible if we encode a single tiling at the leafs of a tree, because we would need to 'backtrack' in the tree to access these other positions.

We start by showing how to encode two identical tilings at the leafs of a tree. Then we will show how to express the three constraints 1, 2 and 3 in the definition of a tiling.

1. The coordinates (x_1, y_1) and (x_2, y_2) of the two tilings are represented by the valuations of atomic propositions $p_0, \ldots, p_{4n-1}$. More precisely, the set $X_1 = \{p_0, \ldots, p_{n-1}\}$ contains the atomic propositions encoding the binary representation of the integer x_1, $Y_1 = \{p_n, \ldots, p_{2n-1}\}$ contains the atomic propositions encoding the binary representation of the integer y_1, $X_2 = \{p_{2n}, \ldots, p_{3n-1}\}$ contains the atomic propositions encoding the binary representation of the integer x_2, and $Y_2 = \{p_{3n}, \ldots, p_{4n-1}\}$ contains the atomic propositions encoding the binary representation of the integer y_2. For instance, for $n = 4$, the coordinates $(x_1, y_1) = (4, 3)$ and $(x_2, y_2) = (11, 2)$ are represented at a leaf of the tree by the following valuation. We recall that in binary notation, 4 is represented by $\overline{100}$, 3 is represented by $\overline{11}$, 12 is represented by $\overline{1100}$ and 2 is represented by $\overline{10}$.

$$\underbrace{\neg p_0, p_1, \neg p_2, \neg p_3}_{4}\underbrace{\neg p_4, \neg p_5, p_6, p_7}_{3}$$

$$\underbrace{p_8, p_9, \neg p_{10}, \neg p_{11}}_{12}\underbrace{\neg p_{12}, \neg p_{13}, p_{14}, \neg p_{15}}_{2}$$

We then encode the existence of all valuations over $X_1 \cup Y_1 \cup X_2 \cup Y_2$ with the following formula:

$$\bigwedge_{l<4n} B_a^l\Big(\langle B_a\rangle p_l \wedge \langle B_a\rangle \neg p_l \wedge \bigwedge_{i<l}((p_i \to B_a p_i) \wedge (\neg p_i \to B_a \neg p_i))\Big). \quad (4)$$

Formula 4 is true at a pointed epistemic model iff this pointed epistemic model is bisimilar up to modal depth $4n$ to a binary tree of depth $4n$ whose leafs contain all the possible valuations associated to $p_0, \ldots, p_{4n-1}$.

In order to check Constraints 2 and 3 in the definition of a tiling, we will need to refer to the tile located to the right or to the left of a given position in a tiling, and also to refer to the tile located above or below it. The following formulas encode the fact that any pair of coordinates (x_1, x_2) and (y_1, y_2) of the two tilings satisfy the properties $x_1 = x_2$, $x_1 = x_2 + 1$, $y_1 = y_2$ and $y_1 = y_2 + 1$ respectively:

$$(x_1 = x_2) \triangleq \bigwedge_{i<n}(p_i \leftrightarrow p_{i+2n}) \quad (5)$$

$$(y_1 = y_2) \triangleq \bigwedge_{n\le i<2n}(p_i \leftrightarrow p_{i+2n}) \quad (6)$$

$$(x_1 = x_2 + 1) \triangleq \bigvee_{i<n}\Big(\bigwedge_{j<i}(p_{j+2n} \leftrightarrow p_j) \wedge \neg p_{i+2n} \wedge p_i \wedge \bigwedge_{i<j<n}(p_{j+2n} \wedge \neg p_j)\Big) \quad (7)$$

$$(y_1 = y_2 + 1) \triangleq \bigvee_{n\le i<2n}\Big(\bigwedge_{n\le j<i}(p_{j+2n} \leftrightarrow p_j) \wedge \neg p_{i+2n} \wedge p_i \wedge \bigwedge_{i<j<2n}(p_{j+2n} \wedge \neg p_j)\Big) \quad (8)$$

The tile types of the first tiling are represented by atomic propositions 1_t and the tile types of the second tiling are represented by atomic propositions $2_{t'}$, where t and t' range over T. They hold at a leaf of the tree whose coordinates correspond to (x_1, y_1) and (x_2, y_2) when the tile type of the first tiling at coordinate (x_1, y_1) is t and the tile type of the second tiling at coordinate (x_2, y_2) is t'.

Formulas 9 and 10 below encode the fact that, *at each leaf of the tree*, there is exactly one tile type for the first tiling and exactly one tile type for the second tiling. Formula 11 below encodes the fact that when these two pairs of coordinates coincide, that is when $x_1 = x_2$ and $y_1 = y_2$, then the tile type of the first tiling and the tile type of the second tiling are identical.

$$B_a^{4n}\Big(\bigvee_{t\in T} 1_t \wedge \bigvee_{t\in T} 2_t\Big) \quad (9)$$

$$B_a^{4n}\bigwedge\big\{(1_t \to \neg 1_{t'}) \wedge (2_t \to \neg 2_{t'}) \mid t, t' \in T, t \neq t'\big\} \quad (10)$$

$$B_a^{4n}\Big((x_1 = x_2) \wedge (y_1 = y_2) \to \bigwedge_{t\in T}(1_t \leftrightarrow 2_t)\Big) \quad (11)$$

However, it may be the case that in the tree, two different leafs with the *same* valuation have different tile types. Therefore, we also have to constrain the tree so that the leafs denoting the same position in the first tiling (resp. second tiling) contain the same tile type for the first tiling (resp. second tiling). This is expressed by the following two formulas:

$$[\mathcal{M}'_{p_0} \cup \mathcal{M}'_{\neg p_0}] \ldots [\mathcal{M}'_{p_{2n-1}} \cup \mathcal{M}'_{\neg p_{2n-1}}] \bigvee_{t\in T} B_a^{4n} 1_t \quad (12)$$

$$[\mathcal{M}'_{p_{2n}} \cup \mathcal{M}'_{\neg p_{2n}}] \ldots [\mathcal{M}'_{p_{4n-1}} \cup \mathcal{M}'_{\neg p_{4n-1}}] \bigvee_{t\in T} B_a^{4n} 2_t \quad (13)$$

where for a given a literal ℓ (p or $\neg p$), the pointed event model $\mathcal{M}'_\ell = (W', R', Pre, w'_0)$ is defined as follows: $W' = \{w'_i \mid i \in \{0, \ldots, 4n\}\}$; $R'_a = \{(w'_i, w'_{i+1}) \mid i \in \{0, \ldots, 4n-1\}\}$; and $Pre(w'_i) = \top$ for all $i < 4n$ and $Pre(w'_{4n}) = \ell$.

In formula 12, the sequence of pointed event models $[\mathcal{M}'_{p_0} \cup \mathcal{M}'_{\neg p_0}] \ldots [\mathcal{M}'_{p_{2n-1}} \cup \mathcal{M}'_{\neg p_{2n-1}}]$ non-deterministically picks a valuation v over $X_1 \cup Y_1$ and selects the branches of the tree whose leafs satisfy this valuation. Then, the formula $\bigvee_{t\in T} B_a^{4n} 1_t$ checks that these leafs, which denote the same position in the first tiling, are of the same tile type t. Likewise with formula 13 for the second tiling.

So, with formulas 9, 10, 11, 12 and 13, we have encoded in the tree two identical tilings in a single tree. Importantly, note that the tree is defined so that each leaf refers to two coordinates of the tiling, which can possibly be identical or consecutive. It is this feature which will allow us to express that constraints 2 and 3 of the definition of a tiling hold.

2. Constraints 1, 2 and 3 of the definition of a tiling are expressed respectively by the following formulas:

$$B_a^{4n}\Big(\Big(\bigwedge_{i<4n} \neg p_i\Big) \to t_0\Big) \quad (14)$$

$$B_a^{4n}\Big((x_1 = x_2) \wedge (y_1 = y_2 + 1) \to \bigwedge_{t\in T}\Big\{1_t \to \bigvee\big\{2_{t'} \mid t' \in T, down(t') = up(t)\big\}\Big\}\Big) \quad (15)$$

$$B_a^{4n}\Big((x_1 = x_2 + 1) \wedge (y_1 = y_2) \quad (16)$$
$$\rightarrow \bigwedge_{t \in T} \Big\{1_t \rightarrow \bigvee \{2_{t'} \mid t' \in T, left(t') = right(t)\}\Big\}\Big)$$

As we said at the beginning of the proof, these two constraints motivate the need to encode *two* tilings: for a given position in a tiling, we need to refer to the tile located to the right or to the left of it, and to refer to the tile located above or below it. This would not be possible with our epistemic language if the tiling was encoded by a single tree.

One can then check that there exists a tiling for the instance of the tiling problem iff the formula φ, which is the conjunction of fomulas 4, 9, 10, 11, 12, 13, 14, 15, and 16 is satisfiable in $\mathcal{L}_{DEL}$.

3. Finally, we show that the reduction is polynomial in the size of the instance of the tiling problem. The formula of Equation 4 is of size $O(n^2)$. The formulas of Equations 12, 13 are of size $O(n^2 + |T| \times n)$. The other formulas are clearly of size polynomial in the size of the input, so the result follows. Importantly, note that if we decided to rewrite the formulas 12 and 13 without using the union operator $\cup$, then the corresponding formula would be exponential in the size of the input. So, the use of the union operator is really crucial in order to have a polynomial reduction from the tiling problem to our satisfiability problem. □

5. RELATED WORK

5.1 Theory

There exists a terminating tableau method solving the satisfiability problem of $\mathcal{L}_{DEL}$ [Hansen, 2010]. This method writes subformulas by applying the reduction axioms [Baltag and Moss, 2004, p. 214]. It is therefore mainly a variant of the tableau method of classical multi-modal logic K_n. Even if we know that tr blows up exponentially the size of the input formula, the computational complexity of this tableau method is not studied. In this section, we review the existing results about computational complexity of DEL.

5.1.1 Public Announcement Logic (PAL)

Public Announcement Logic (PAL) [Plaza, 1989] is an extension of epistemic logic with a dynamic operator $[\psi!]\varphi$ whose truth conditions are defined as follows:

$$\mathcal{M}, w \models [\psi!]\varphi \quad \text{iff} \quad \mathcal{M}, w \models \psi \text{ implies } \mathcal{M}_\psi, w \models \varphi$$

where $\mathcal{M}_\psi$ is the restriction of $\mathcal{M}$ to the worlds which satisfy ψ. PAL is a fragment of DEL: the language of PAL is $\mathcal{L}_{DEL}$ restricted to event models consisting of a single possible event with reflexive arrows for all agents. There is a gap between PAL and DEL in terms of computational complexity, both for the model checking problem and the satisfiability problem. Indeed, the model checking of PAL is in P (also with common belief) [van Benthem and Kooi, 2004] and the satisfiability problem for PAL is PSPACE-complete [Lutz, 2006]. Despite the fact that there exist reduction axioms for PAL, it is difficult to implement a direct translation using reduction axioms. In fact, there are properties that can be expressed exponentially more succinctly in PAL than in epistemic logic [French et al., 2011]. Note that there exist PSPACE tableau methods for solving the satisfiability problem in PAL [de Boer, 2007, Balbiani et al., 2010].

5.1.2 DEL-sequents

DEL-sequents [Aucher, 2011] are triples of the form $\varphi, \varphi' \models \varphi''$ where $\varphi, \varphi'' \in \mathcal{L}_{EL}$ and φ' is a formula of a language for event models. A DEL-sequent $\varphi, \varphi' \models \varphi''$ holds when for all pointed epistemic model $(\mathcal{M}, w)$ such that $\mathcal{M}, w \models \varphi$, for all pointed event model $(\mathcal{M}', w')$ such that $\mathcal{M}', w' \models \varphi'$, if $(\mathcal{M}', w')$ is executable in $(\mathcal{M}, w)$, then $\mathcal{M} \otimes \mathcal{M}', (w, w') \models \varphi''$. The problem of determining whether a DEL-sequent holds is NEXPTIME-complete and there exists a tableau method for it. DEL-sequents have been generalized to sequences of the form $\varphi_0, \varphi_1', \varphi_1, \ldots, \varphi_n', \varphi_n \overset{\frac{1}{i}}{\models} \psi$ and $\varphi_0, \varphi_1', \varphi_1, \ldots, \varphi_n', \varphi_n \overset{\frac{2}{i}}{\models} \psi'$. The corresponding satisfiability problem is also NEXPTIME-complete [Aucher et al., 2012].

5.1.3 The sequence and 'star' iteration operators

The sequence and 'star' iteration operators are constructions enabling to build complex programs as in Propositional Dynamic Logic (PDL [Harel et al., 2000]). The truth conditions are defined as follows:

$\mathcal{M}, w \models [\pi; \gamma]\varphi$ iff $\mathcal{M}, w \models [\pi][\gamma]\varphi$
$\mathcal{M}, w \models [\pi^*]\varphi$ iff there is a *finite* sequence $\pi; \ldots; \pi$ such that $\mathcal{M}, w \models [\pi; \ldots; \pi]\varphi$

We do not know about the computational complexity of the model-checking problem when the operator $[\pi^*]\varphi$ is added to the language. In fact, we do not even know whether it is decidable. The computational complexity of the satisfiability problem remains the same when the sequential composition operator is added. However, adding a 'star' operator makes the satisfiability problem undecidable. This result is not really surprising, it is a direct corollary of the result of [Miller and Moss, 2005] stating that Public Announcement Logic with the 'star' operator is already undecidable.

5.1.4 The common belief operator

We may extend the language with the common belief operator $C_G\varphi$, where $G \subseteq AGT$. The truth conditions are defined as follows:

$$\mathcal{M}, w \models C_G\varphi \quad \text{iff} \quad \text{for all } v \in \left(\bigcup_{a \in G} R_a\right)^+ (w), \mathcal{M}, v \models \varphi$$

Intuitively, $C_G\varphi$ is an abbreviation of an infinite conjunction [Fagin et al., 1995]: $C_G\varphi = E_G^1\varphi \wedge E_G^2\varphi \wedge E_G^3\varphi \wedge \ldots$, where $E_G^k\varphi$ is defined inductively as follows: $E_G^1\varphi = \bigwedge_{a \in G} B_a\varphi$ and $E_G^{k+1}\varphi = E_G^1 E_G^k \varphi$.

We do not know about the computational complexity of the satisfiability problem when the common belief operator is added to the language $\mathcal{L}_{DEL}$. However, we know that it is decidable and that the language with common belief operator is more expressive than the *epistemic* language $\mathcal{L}_{EL}$ with common belief [Baltag et al., 1998, Baltag et al., 1999].

5.2 Implementation

There exist two implementations of our decision problems:

1. The model-checker DEMO [van Eijck, 2007], standing for Dynamic Epistemic MOdeling tool, can evaluate formulas of $\mathcal{L}_{DEL}$ in epistemic models, display graphically epistemic models, event models and updates of epistemic models by event models, translate formulas of $\mathcal{L}_{DEL}$ to formulas of PDL. DEMO is written in Haskel and has been applied in [van Ditmarsch et al., 2005] and [van Ditmarsch et al., 2006]. Also, it has been used to investigate the pros and cons of

modeling some well-known problems of computer security within the DEL framework [van Eijck and Orzan, 2007].

2. The program `Aximo` [Richards and Sadrzadeh, 2009], written in C++, implements an algorithm for proving properties of interactive multi-agent scenarios encoded in epistemic systems. Epistemic systems provide an algebraic semantics to DEL and were developed together with a sound and complete sequent calculus [Baltag et al., 2007].

6. CONCLUDING REMARKS

Our work contributes to the proof theory and the study of the computational complexity of DEL, which has been rather neglected so far. Although our results show that our decision problems are not tractable, it turns out that the DEMO implementation does not fare worse and often even better in terms of time of execution than other model-checkers modeling the same problems, without resorting to the DEL methodology [van Ditmarsch et al., 2006].

We still need to investigate whether or not the computational complexity remains the same when we consider other epistemic logics as the basis of DEL, such as S5. Moreover, our results rely on the fact that we use the union operator in the language, an open problem is to obtain similar results without this operator. Finally, we plan to implement our tableau method in LotrecScheme [Schwarzentruber, 2011].

Acknowledgment. We thank the reviewers for their comments and Sophie Pinchinat for discussions.

7. REFERENCES

[Aucher, 2011] Aucher, G. (2011). DEL-sequents for progression. *Journal of Applied Non-Classical Logics*, 21(3-4):289–321.

[Aucher et al., 2012] Aucher, G., Maubert, B., and Schwarzentruber, F. (2012). Generalized DEL-sequents. In del Cerro, L. F., Herzig, A., and Mengin, J., editors, *JELIA*, volume 7519 of *Lecture Notes in Computer Science*, pages 54–66. Springer.

[Balbiani et al., 2010] Balbiani, P., van Ditmarsch, H., Herzig, A., and de Lima, T. (2010). Tableaux for public announcement logic. *Journal of Logic and Computation*, 20(1):55–76.

[Baltag et al., 2007] Baltag, A., Coecke, B., and Sadrzadeh, M. (2007). Epistemic actions as resources. *Journal of Logic and Computation*, 17(3):555–585.

[Baltag and Moss, 2004] Baltag, A. and Moss, L. (2004). Logic for epistemic programs. *Synthese*, 139(2):165–224.

[Baltag et al., 1998] Baltag, A., Moss, L., and Solecki, S. (1998). The logic of common knowledge, public announcement, and private suspicions. In Gilboa, I., editor, *Proceedings of TARK98*, pages 43–56.

[Baltag et al., 1999] Baltag, A., Moss, L., and Solecki, S. (1999). The logic of public announcements, common knowledge and private suspicions. Technical report, Indiana University.

[Boas, 1997] Boas, P. (1997). The convenience of tilings. In *Complexity, Logic, and Recursion Theory*, pages 331–363. Marcel Dekker Inc.

[de Boer, 2007] de Boer, M. (2007). KE tableaux for public anouncement logic. In *Proceedings of FAMAS 07*, Durham UK.

[Fagin et al., 1995] Fagin, R., Halpern, J., Moses, Y., and Vardi, M. (1995). *Reasoning about knowledge*. MIT Press.

[Fitting, 1983] Fitting, M. (1983). *Proof methods for modal and intuitionistic logics*. D. Reidel, Dordrecht.

[French et al., 2011] French, T., van der Hoek, W., Iliev, P., and Kooi, B. P. (2011). Succinctness of epistemic languages. In Walsh, T., editor, *IJCAI*, pages 881–886. IJCAI/AAAI.

[Hansen, 2010] Hansen, J. (2010). Terminating tableaux for dynamic epistemic logics. *Electronic Notes in Theoretical Computer Science*, 262:141–156.

[Harel et al., 2000] Harel, D., Kozen, D., and Tiuryn, J. (2000). *Dynamic Logic*. MIT Press.

[Lutz, 2006] Lutz, C. (2006). Complexity and succinctness of public announcement logic. In *Proceedings of AAMAS 2006*, pages 137–143. ACM.

[Miller and Moss, 2005] Miller, J. and Moss, L. (2005). The undecidability of iterated modal relativization. *Studia Logica*, 79(3):373–407.

[Papadimitriou, 1995] Papadimitriou, C. H. (1995). *Computational complexity*. Addison Wesley.

[Plaza, 1989] Plaza, J. (1989). Logics of public communications. In Emrich, M. L., Pfeifer, M. Z., Hadzikadic, M., and Ras, Z. W., editors, *Proceedings of the 4th International Symposium on Methodologies for Intelligent Systems*, pages 201–216.

[Richards and Sadrzadeh, 2009] Richards, S. and Sadrzadeh, M. (2009). Aximo: Automated axiomatic reasoning for information update. *Electr. Notes Theor. Comput. Sci.*, 231:211–225.

[Schwarzentruber, 2011] Schwarzentruber, F. (2011). Lotrecscheme. *Electr. Notes Theor. Comput. Sci.*, 278:187–199.

[van Benthem and Kooi, 2004] van Benthem, J. and Kooi, B. (2004). Reduction axioms for epistemic actions. In Schmidt, R., Pratt-Hartmann, I., Reynolds, M., and Wansing, H., editors, *AiML-2004: Advances in Modal Logic*, number UMCS-04-9-1 in Technical Report Series, pages 197–211, University of Manchester.

[van Ditmarsch et al., 2007] van Ditmarsch, H., van der Hoek, W., and Kooi, B. (2007). *Dynamic Epistemic Logic*, volume 337 of *Synthese library*. Springer.

[van Ditmarsch et al., 2005] van Ditmarsch, H. P., Ruan, J., and Verbrugge, L. C. (2005). Model checking sum and product. In Zhang, S. and Jarvis, R., editors, *Australian Conference on Artificial Intelligence*, volume 3809 of *Lecture Notes in Computer Science*, pages 790–795. Springer.

[van Ditmarsch et al., 2006] van Ditmarsch, H. P., van der Hoek, W., van der Meyden, R., and Ruan, J. (2006). Model checking russian cards. *Electr. Notes Theor. Comput. Sci.*, 149(2):105–123.

[van Eijck, 2007] van Eijck, J. (2007). Demo — a demo of epistemic modelling. In van Benthem, J., Gabbay, D., and Löwe, B., editors, *Interactive Logic — Proceedings of the 7th Augustus de Morgan Workshop*, Texts in Logic and Games 1, pages 305–363.

[van Eijck and Orzan, 2007] van Eijck, J. and Orzan, S. (2007). Epistemic verification of anonymity. *Electronic Notes in Theoretical Computer Science*, 168(0):159 – 174.

The Shape of Reactive Coordination Tasks

Ido Ben-Zvi
Technion
Haifa, Israel
iddobz@gmail.com

Yoram Moses
Technion
Haifa, Israel
moses@ee.technion.ac.il

ABSTRACT

This paper studies the interaction between knowledge, time and coordination in systems in which timing information is available. Necessary conditions are given for the causal structure in coordination problems consisting of orchestrating a set of actions in a manner that satisfies a variety of temporal ordering assumptions. Results are obtained in two main steps: A specification of coordination is shown to require epistemic properties, and the causal structure required to obtain these properties is characterised via "*knowledge gain*" theorems. A new causal structure called a *centibroom* structure is presented, generalising previous causal structures for this model. It is shown to capture coordination tasks in which a sequence of clusters of events is performed in linear order, while within each cluster all actions must take place simultaneously. This form of coordination is shown to require the agents to gain a nested common knowledge of particular facts, which in turn requires a centibroom. Altogether, the results presented provide a broad view of the causal shape underlying partially ordered coordinated actions. This, in turn, provides insight into and can enable the design of efficient solutions to the coordination tasks in question.

Categories and Subject Descriptors

[**Artificial intelligence**]: Knowledge representation and reasoning — *Reasoning about belief and knowledge, Causal reasoning and diagnostics*; [**Artificial intelligence**]: Distributed artificial intelligence — *Cooperation and coordination, multi-agent systems*; [**Distributed computing methodologies**]

General Terms

Theory, Design, Algorithms, Verification

Keywords

Knowledge, Common knowledge, Epistemic logic, Temporal coordination, Causality and communication

1. INTRODUCTION

Coordinated action in distributed and multi-agent systems is closely related to knowledge and epistemic states. As a particular example, linearly ordered actions require nested knowledge. Namely, suppose that the occurrence of event e is guaranteed to trigger a response by each of the agents 1, 2, and 3, and, moreover, they must act in this order: first 1, then 2, and finally 3. Then, in a precise sense, $K_3K_2K_1\mathsf{occ}(e)$ (which we read as *"agent 3 knows that 2 knows that 1 knows that e has occurred"*) must hold when agent 3 acts [3]. This generalises from three agents to any finite number. In the theory of distributed systems, asynchronous systems, in which agents have no clock and no timing information is available, receive a great deal of attention [1, 17]. In such systems, Chandy and Misra's celebrated *Knowledge Gain* theorem [7] captures the necessary condition for attaining nested knowledge of this form. Roughly speaking, it implies the following. Suppose that a spontaneous event e takes place at agent 0's site in an asynchronous system. Then $K_3K_2K_1\mathsf{occ}(e)$ can hold only after a message chain is formed, that starts from agent 0 after e occurs, and passes through 1 and then through 2 to agent 3. (The message chain may pass through other sites as well; but it must visit these agents in the specified order.) As a result, the only way to coordinate a linearly ordered response to the event e in an asynchronous system is via such a message chain. This theorem captures the shape of the causal structure that underlies linear coordination.

The presence of clocks and timing information can greatly facilitate coordination tasks in distributed and multi-agent systems. In [3, 4] we initiated a study of coordination in a *synchronous* model, where agents have access to a global clock, and, for each particular channel, there is an upper bound on the time messages can spend in transit. In the presence of clocks the passage of time can be used to derive information about events at remote sites. As a result, message chains are not the only way to attain nested knowledge. A knowledge gain theorem capturing subtle interplay of communicated messages, the guaranteed bounds, and the passage of time is given in [3]. It shows that the causal "shape" underlying nested knowledge is captured by a structure called a *centipede* (see Figure 2). In a precise sense, this is the "synchronous" analogue of a message chain.

The connection between coordination and epistemics manifests itself beyond the connection between linearly ordered actions and nested knowledge. Halpern and Moses showed

that simultaneous actions are very closely related to common knowledge [14]: If a set of agents B can be guaranteed to all perform a particular action a *simultaneously* whenever any of them perform it, then when they perform a they have *common knowledge* that it is being performed. If the action a is performed only in response to a particular spontaneous event e, then they must also have common knowledge that e has occurred. Using more formal notation, if we denote common knowledge to B by C_{B}, then attaining $C_{\mathsf{B}}\mathsf{occ}(e)$ is a prerequisite for performing a. While common knowledge (as well as simultaneity) cannot be attained in asynchronous settings [7, 14], it can often be attained in the presence of clocks and time guarantees. A causal structure called a *broom* (Figure 3) was shown to be necessary for gaining common knowledge in systems with clocks [3]. Both centipedes and brooms are defined in terms of two relations (the different styled edges in Figures 2 and 3): *syncausality*, which captures message chains in the synchronous model, and *bound guarantee*, which provides the means to account for the information obtained by the passage of time. (We review the definitions of these relations and structures in Section 2.2.)

Characterizing the causal shape underlying coordination tasks provides insight into the structure of their solutions, and often enables the design of optimal solutions for such problems. In Section 3 we demonstrate this with a novel application by deriving an optimal solution to the *distributed snapshot* problem of [6] in the synchronous setting, based on the connection between brooms and simultaneous response.

In this paper, we extend the analysis of coordination in the synchronous model, to handle much more general forms of coordination. We follow the same scheme as above: Relate a class of coordination problems to a set of corresponding epistemic states, and then study the causal structure underlying these epistemic states via proving knowledge gain theorems.

Consider the following example, which constitutes a variation on one discussed by Chwe [8] and is related to numerous studies of information flow and agency in social networks [13, 22, 19].

EXAMPLE 1. *Under the yoke of the Roman conquerer, the repressed Judean people are bitter and rebellious. As a population, the agents are partitioned into several groups by their tendency to revolt: there is an instigator, and there are the hardline ideologists, the unsatisfied crowds, and the supporters of the old regime.*

- *The **instigator** is highly unpredictable. It may start a revolt at any time, independent of any other event.*
- *A **hard liner** will revolt if it knows that the instigator and all of the other hard liners are revolting together.*[1]
- *A member of the **unsatisfied masses** will revolt if it knows that the instigator, the hard liners and all of the other members of the unsatisfied masses are revolting.*
- *A supporter of the **old regime** will revolt only when it knows that all other members of the population are revolting.*

By means of sun clocks and camel-borne messages, the agents form a synchronous system with upper limits on message transmission times. If the agents are continually communicating with each other, then it is possible to arrange for a rebellion to start a finite number of days after the instigator revolts.[2]

The question we ask is — what pattern of communication would suffice to ensure that the whole population revolts, while keeping communication to a minimum, in the sense that unneeded messages are not sent (so as not to arouse the suspicion of the infamous Roman crucifixion police)?

The problem faced by the Judeans in Example 1 transcends both of the coordination tasks discussed earlier, because it involves both a linear ordering *and* simultaneity of events. More precisely, it involves a sequential ordering of clusters of responses, where the actions in every cluster are performed simultaneously. We will define a corresponding class of coordination problems, called *Ordered Joint Response* (OJR). In general, the sets of acting agents in every cluster will not be assumed to be disjoint. We will show that solving OJR requires attaining nested common knowledge of the form

$$C_{\mathsf{B}^k} C_{\mathsf{B}^{k-1}} \cdots C_{\mathsf{B}^1} \mathsf{occ}(e).$$

The main technical contribution of the paper is a nested common knowledge gain theorem (NCKG) for the synchronous model. It captures the causal structure underlying NCKG by a new form called a *centibroom* (see Figure 5), which is a hybrid structure, combining the centipede with brooms. Since the centibroom is necessary for getting the Judean groups from Example 1 to revolt in the proper order, it is also the minimal communication pattern.

Once we have established how linear sequences of joint responses can be ordered, we will take a bigger step and consider the general problem of ordering events according to any pre-specified ordering. Consider the following update on the situation in Judea.

EXAMPLE 2. *The situation is just as dire as in Example 1, but the social standing is more complex, as there are now three instigators, and the hardline ideologists are divided among themselves into two opposing groups: the Peoples Front of Judea (PFJ) and the Judean People's Front (JPF).*

- *The three instigators are: Jedediah, Jeremiah, and Brian - each of them operating on its own as before.*
- *A member of the PFJ will revolt if it knows that Jedediah, Jeremiah and the rest of the PFJ's members are revolting.*

[1] Assume for each of the groups in the population that common knowledge of "stalemate" is resolved by revolting. I.e., if it is common knowledge among the hardliners that the instigator is revolting, then each hardliner revolts too.

[2] In fact, the proposed solution to the distributed snapshot problem, discussed in Section 3, coud be used to achieve this in minimal time.

- *A member of the JPF will revolt if it knows that Brian and the rest of the JPF's members are revolting.*
- *The unsatisfied masses and the supporters of the old regime act as before.*

Figure 1 sums up the revolt dependencies as a directed acyclic graph. Once again we ask what pattern of communication would push the population into rebellion (provided that enough instigators revolt) while keeping communication to a minimum.

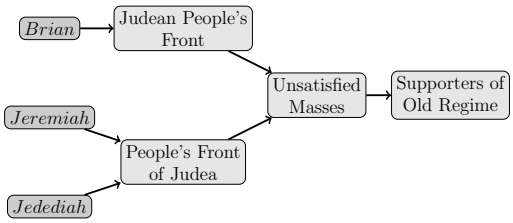


Figure 1: Judea, 71AD

In Example 2, if both Jedediah and Brian revolt (but not Jeremiah), then the country will not be swept by rebellion. The members of the JPF will also revolt, as they only look to Brian for guidance. But those of the PFJ will abstain — waiting for Jeremiah to revolt as well before joining in. The unsatisfied masses will see that the PFJ is not joining in, and will prefer to stay at home. In contrast, if all three instigators revolt (and there is sufficient communication to spread the word) then both hardliner factions will revolt too — eventually leading the unsatisfied masses to revolt, and even the supporters of the old regime to follow in their wake. Note that even though the members of the PFG all revolt simultaneously, as do all members of the JPF, these two simultaneous joint responses need not occur at the same time.

In order to derive the necessary epistemic state and communication pattern for solving such problems, we will consider *General Ordered Response* problems (GOR), in which a weak ordering among responses is specified by a general directed graph. An edge among two responses α and α' should imply that if α occurs at time t in a given execution and α' occurs at t', then $t \leq t'$. Note that we can encode both simultaneous sets of events, as well as an ordering on these sets, using the same partial order. We do this by making use of cycles on the graph, as all nodes on a cycle must be performed simultaneously. Thus, the GOR coordination problem can be used to specify a partial order on simultaneous clusters of responses. A we will show, while the OJR problem is solved by a new communication pattern called the centibroom, solutions to the more general GOR problem do not define a yet more complex structure that solves it. Rather, the communication pattern that characterizes it is best described as a set of unrelated centibrooms, thus capturing "the causal shape" of a very broad class of coordination problems in the synchronous model.

Our analysis is performed for *reactive* coordination tasks in which particular patterns of responses need to be performed in response to external triggering events. This is motivated by the fact that many distributed and multi-agent systems are embedded within a larger environment and need to coordinate their activities based on input that is supplied by it. This is true of an online bank, where customers may initiate transactions, a public safety application receiving the report of a smoke alarm activation, an online retailer (e.g. Amazon), a search engine (Google) that can accept requests, or a cloud computing application in which customers can submit computational tasks to be performed by the system. We focus on distributed settings in which activities in the system may be triggered by external events that are *spontaneous* as far as the system and its design are concerned. An external event of this sort may require a simple response by the system, but there are many cases in which it may trigger an extended transaction in which multiple events must take place, and these should be coordinated in various particular patterns.

This paper is organized as follows. The next section presents the model, and reviews the definitions of syncausality and bound guarantees, knowledge and common knowledge in distributed systems, and the centipede and broom structures from [3]. Section 3 illustrates the use of brooms by applying them to obtain an optimal distributed snapshot in the synchronous setting. Section 4 defines the ordered joint response problem, and relates the epistemic state of nested common knowledge to solutions to the problem. The notion of a centibroom is defined, and is used to capture nested common knowledge formulas, and sequential ordering of simultaneous responses. Finally, Section 5 defines the general ordered response problem, and states a theorem characterizing the shape of GOR solutions in terms of centibrooms. Section 6 closes the paper with discussion and further research.

2. BACKGROUND

2.1 Synchronous Networks

We focus on a simple synchronous setting in which agents are connected via a communication network and there are upper bounds on message transmission times. Agents share a global clock, and take steps at integer times. To analyze coordination in such a setting, we make use of the *interpreted systems* approach to modeling distributed systems (see [11]). Namely, we separate the definition of the environment for which protocols are designed, formally called the *context*, from the protocol being executed in that context. Formally, a context γ is a tuple $(\mathcal{G}_0, P_e, \tau)$, where $\mathcal{G}_0$ is a set of initial global states, P_e is a protocol for the environment, and τ is a *transition function*.[3] The environment is viewed as running a protocol (denoted by P_e) just like the agents; its protocol is used to capture nondeterministic aspects of the execution, such as the actual transmission times, external inputs into the system, etc. The transition function τ describes how the actions performed by the agents and by the environment change the global state.

A run is an infinite sequence of global states, Given a context γ and a protocol P designed to run in γ, there is a unique set $\mathcal{R} = \mathcal{R}(P, \gamma)$, of all possible runs of P in γ. This

[3]Depending on the application, a context can include additional components. See [11] for proper exposition.

set is called a *system*, and we study how knowledge evolves in systems. The reason why it does not suffice to consider just one system—say the system consisting of all possible runs in γ—is because the protocol being executed plays an important role in determining what is known. Typically, the information inherent in receiving a particular message (or in not receiving one) depends on the protocol being used.

The essential elements of the model are the following.

- We assume that agents can receive external inputs from the outside world. These are determined in a genuinely nondeterministic fashion, and are not correlated with anything that comes before in the execution or with external inputs of other agents.
- The set of agents is denoted by $\mathbb{P}$. The network consists of the weighted channels graph $\mathsf{Net} = (\mathbb{P}, \mathbb{C}, b)$ in which the weight of a channel $(i, j) \in \mathbb{C}$ consists of a discrete upper bound $b_{ij} \geq 1$. A copy of the network, as well as the current global time, are part of every agent's local state at all times.
- The scheduler, which we typically call the *environment*, is in charge of choosing these external inputs, and of determining message transmission times. The latter are also determined in a nondeterministic fashion, subject to the constraint that delivery satisfies the transmission bounds b_{ij}, and messages take at least one time step to be delivered.
- Time is identified with the natural numbers, and agents are assumed to take steps only at integer times. For simplicity, the agents follow deterministic protocols. Hence, a given protocol P for the agents and a given behavior of the environment completely determine the run.
- Events are sends, receives, arrivals of external inputs, and internal actions. All events in a run are distinct, and we denote a generic event by the letter e. For ease of exposition, we will assume that an agent's local state contains the set of response actions that the agent has performed. This assumption is needed only for the analysis of response problems, and can be obtained by adding an auxiliary variable keeping track of the history, to each of the agents.

We denote a context satisfying the above assumptions by γ^{max}, and use $\mathcal{R}^{\mathsf{max}}$ to denote a system $\mathcal{R}(P, \gamma^{\mathsf{max}})$ consisting of the set of all runs of some protocol P in synchronous context γ^{max}.[4]

Note that our model requires transmission times to obey the bounds specified in $\mathsf{Net} = (\mathbb{P}, \mathbb{C}, b)$, but it does not require the agents to have access to a global clock, or to any clocks at all. Nevertheless, the results will apply even in the case in which agents do share a precise global clock, and each agent is scheduled to move at every time step.

[4] We defer the rather tedious technical definition of γ^{max} for the full paper.

2.2 Syncausality and time bound guarantees

Messages and message chains are a primary tool in coordinating actions in a distributed system. In synchronous networks, in addition to messages, silence can also be used to transmit information. Indeed, as suggested by Lamport in [16], in the synchronous context γ^{max}, it is possible to consider the fact that a agent i does *not* send a message over the channel $(i, j) \in \mathbb{C}$ at time t as the sending of a *null* message over the channel. This null message is "received" by j at time $t + b_{ij}$. Motivated by this idea, we proposed the notion of *syncausality* (in [3]), generalizing Lamport's happened-before relation ([15]) to capture generalized message chains consisting of actual messages and null message. Since "not receiving at $t + b_{ij}$" is not an explicit event, it is convenient to define the syncausality relation between agent-time nodes rather than between events. An *agent-time node* (or simply *node*) is a pair $\theta = \langle i, t \rangle$, where i is a agent and t is a time. Such a node represents the instant at time t on i's timeline. Formally, syncausality is defined as follows:

DEFINITION 1 (SYNCAUSALITY). *The* ***Syncausality*** *relation in a given run r is the smallest relation $\rightsquigarrow_{\mathbf{r}}$ satisfying:*

Locality: *If $t \leq t'$ then $\langle i, t \rangle \rightsquigarrow_r \langle i, t' \rangle$;*

Send-rcv: *If a message sent at $\langle i, t \rangle$ is received at $\langle j, t' \rangle$ then $\langle i, t \rangle \rightsquigarrow_r \langle j, t' \rangle$;*

Null msg: *If no message is sent over $(i, j) \in \mathbb{C}$ at time t then $\langle i, t \rangle \rightsquigarrow_r \langle j, t + b_{ij} \rangle$; and*

Transitivity: *If $\theta \rightsquigarrow_r \theta'$ and $\theta' \rightsquigarrow_r \theta''$, then $\theta \rightsquigarrow_r \theta''$.*

Syncausality captures a notion of direct information flow via (generalized) message chains. If $\langle i, t \rangle \not\rightsquigarrow_r \langle j, t' \rangle$, then j at time t' does not have information regarding which nondeterministic (or spontaneous) events occur at $\langle i, t \rangle$. A straightforward but useful property of $\rightsquigarrow_r$ is:

FACT 1. *If $\langle i, t \rangle \neq \langle j, t' \rangle$ and $\langle i, t \rangle \rightsquigarrow_r \langle j, t' \rangle$, then $t < t'$.*

The second (Send-rcv) clause of the definition makes syncausality run-dependent, as actual delivery times depend on the adversary's actions. Hence the subscript r in the $\rightsquigarrow_r$ symbol. While syncausality captures direct information flow, the upper bounds on message transmission times allow agents to know about events at remote sites in a less direct fashion. Namely, if h knows about a message sent by i at time t to j, then after sufficient time has passed h can be guaranteed that j received i's message. Moreover, if the protocol specifies that j will perform particular actions after receiving this message, then h can know about actions of j without direct information flow from j. This can enable them to coordinate their actions without communicating directly. The interaction between communication and time is based on a combination of syncausality and the *bound guarantee* relation, a second causal relation between agent-time nodes that is based on time bounds. Denote by $\delta(i, j)$ the shortest distance between i and j in the weighted graph Net. Intuitively, if we think of a shortest path from i to j in Net as an "overlay channel" between i and j, then $\delta(i, j)$ would be

the upper bound for the transmission time over this channel. We define the *bound guarantee* relation as follows:

DEFINITION 2 (BOUND GUARANTEE [3]). *With respect to a network* $\mathsf{Net} = (\mathbb{P}, \mathbb{C}, b)$*, we write* $\langle i,t\rangle \dashrightarrow \langle j,t'\rangle$ *iff* $t + \delta(i,j) \leq t'$.

Intuitively, $\langle i,t\rangle \dashrightarrow \langle j,t'\rangle$ holds, then a message chain initiated at $\langle i,t\rangle$ can be guaranteed to reach j by $\langle j,t'\rangle$. No explicit acknowledgement from j is needed! Put another way, $\langle j,t'\rangle$ is sure to be within the cone of (causal) influence of events that occur at $\langle i,t\rangle$. While syncausality is sensitive to actually realized transmission times, the bound guarantee relation is not. It depends solely on the weighted network Net. This is one of the reasons why bound guarantees provides cross-site information of a type that is not available, for example, in asynchronous settings. In a precise sense, bound guarantees capture the run-invariant part of syncausality that is based solely on Net:

FACT 2. *If* $\langle i,t\rangle \dashrightarrow \langle j,t'\rangle$ *then* $\langle i,t'\rangle \rightsquigarrow_r \langle j,t'\rangle$*, for every run* r.

2.3 Definition of knowledge

We focus on a very simple logical language in which the set Φ of primitive propositions consists of propositions of the form $\mathsf{occ}(e)$ for events e of interest. To obtain the logical language $\mathcal{L}$, we close Φ under propositional connectives and knowledge formulas. Thus, $\Phi \subset \mathcal{L}$, and if $\varphi \in \mathcal{L}$, $i \in \mathbb{P}$, and $G \subseteq \mathbb{P}$, then $\{K_i\varphi, C_G\varphi\} \subset \mathcal{L}$.[5] The formula $K_i\varphi$ is read *agent* i *knows* φ, and $C_G\varphi$ is read φ *is common knowledge to* G. The truth of formulas is evaluated with respect to a triple (R,r,t) consisting of a system R, a run $r \in R$, and a time $t \in \mathbb{N}$, and we use $(R,r,t) \vDash \varphi$ to state that φ holds at time t in run r, with respect to system R. Denoting by $r_i(t)$ agent i's local state at time t in r, we inductively define

- $(R,r,t) \vDash \mathsf{occ}(e)$ if the event e occurs in r at a time $t' \leq t$;
- $(R,r,t) \vDash K_i\varphi$ if $(R,r',t) \vDash \varphi$ for every run r' satisfying $r_i(t) = r'_i(t)$;
- $(R,r,t) \vDash C_G\varphi$ if $(R,r,t) \vDash K_{i_h}K_{i_{h-1}}\cdots K_{i_1}\varphi$ holds for every $h > 0$ and every sequence $i_h, i_{h-1}, \ldots, i_1$ of agents in G.

Given the system R, the local state determines what facts are known. Intuitively, a fact φ is common knowledge to G if everyone in G knows φ, everyone knows that everyone knows φ, and so on *ad infinitum*. We remark that for singleton sets $G = \{i\}$, the operators $C_{\{i\}}$ and K_i coincide.

2.4 Centipedes and Brooms

In [3] we introduced *Ordered Response* (OR) and Simultaneous Response (SR), two coordination tasks that were simpler than the OJR and GOR problems studied here. We then uncovered the epistemic states communication structures that they necessitate. An instance $\mathsf{OR}\langle e_\mathsf{s}, \alpha_1, \ldots, \alpha_k\rangle$ of ordered response requires that, following occurrence of the triggering event e_s the set $\{\alpha_1, \ldots, \alpha_k\}$ of responses be performed in a linear temporal order. A central result of [3] is that every run of a protocol solving ordered response must contain a causal structure called a *centipede*:

DEFINITION 3 (CENTIPEDE). *Let* $r \in \mathcal{R}^{\mathsf{max}}$*, let* $\{i_0, \ldots, i_k\} \subseteq \mathbb{P}$*, and let* $t \leq t'$*. A* ***centipede*** *for* $\langle i_0, \ldots, i_k\rangle$ *in the interval* $(r, t..t')$ *is a sequence* $\theta_0 \rightsquigarrow_r \theta_1 \rightsquigarrow_r \cdots \rightsquigarrow_r \theta_k$ *of nodes such that (a)* $\theta_0 = \langle i_0, t\rangle$*, (b)* $\theta_k = \langle i_k, t'\rangle$*, and (c)* $\theta_h \dashrightarrow \langle i_h, t'\rangle$ *holds for* $h = 1, \ldots, k-1$.

A centipede is illustrated in Figure 2. The squiggly arrows depict syncausal (message) chains, while the dashed arrows stand for bound guarantees. In a precise sense, a centipede plays in the synchronous context a role analogous to that of message chains in asynchronous ones. In the asynchronous context, a response to the trigger in a protocol ensuring ordered response can occur only if a message chain from the trigger, passing through all previous responses, arrives at the acting agent. In our synchronous model, if e_s occurs at $\langle i_0, t\rangle$ and α_h is performed at time t_h in r, then there must be a centipede for $\langle i_0, \ldots, i_h\rangle$ in $(r, t..t_h)$.

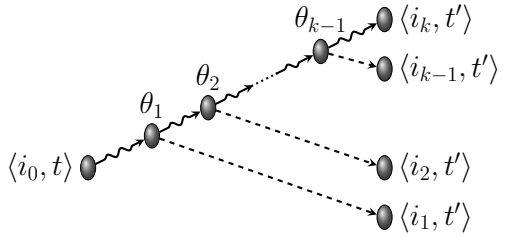


Figure 2: A centipede for $\langle i_0, \ldots, i_k\rangle$ in $(r, t..t')$.

A related causal structure, called a *broom* governs simultaneous coordination. The simultaneous response problem requires all responses to the trigger to occur simultaneously. The responders can act at time t' in response to a trigger at $\langle i_0, t\rangle$ only if a broom structure as in Figure 3 exists.

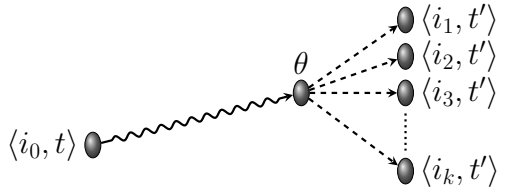


Figure 3: A broom for $\langle i_0, \{i_1 \ldots, i_k\}\rangle$ in $(r, t..t')$.

In a seminal result, Chandy and Misra showed that (Lamport) message chains are a prerequisite for attaining nested knowledge in asynchronous systems [7]. In our synchronous model, centipedes replace message chains in this role:

[5] This is a simplified logical language for ease of exposition.

THEOREM 1 (KNOWLEDGE GAIN, [3]). *Let P be a deterministic protocol, let $r \in \mathcal{R}^{\mathsf{max}} = \mathcal{R}(P, \gamma^{\mathsf{max}})$, and let e_{s} be an external input received in r at $\langle i_0, t\rangle$. If*

$$(\mathcal{R}^{\mathsf{max}}, r, t') \vDash K_{i_k} K_{i_{k-1}} \cdots K_{i_1} \mathsf{occ}(e_{\mathsf{s}})$$

then there is a centipede for $\langle i_0, \ldots, i_k\rangle$ in $(r, t..t')$.

Moreover, the synchronous model goes beyond the asynchronous one by also allowing for common knowledge gain. We have shown that common knowledge gain requires the existence of a broom.

THEOREM 2 (COMMON KNOWLEDGE GAIN, [3]). *Let P be a deterministic protocol, let $r \in \mathcal{R}^{\mathsf{max}} = \mathcal{R}(P, \gamma^{\mathsf{max}})$, let $G \subseteq \mathbb{P}$, and let e_{s} be an external input received in r at $\langle i_0, t\rangle$. If*

$$(\mathcal{R}^{\mathsf{max}}, r, t') \vDash C_G \mathsf{occ}(e_{\mathsf{s}})$$

then there is a broom for $\langle i_0, G\rangle$ in $(r, t..t')$.

3. BROOMS & THE DISTRIBUTED SNAPSHOT PROBLEM

Before embarking on the technical analysis of OJR and GOR problems, we now illustrate how a causal analysis (of the "shape" of solutions) can guide the development of efficient solutions to natural problems. We do this by describing the derivation of an optimal solution to the *Synchronous Global Snapshot* problem,[6] a variant of Chandy and Lamport's *Asynchronous Global Snapshot* problem [6]. Due to space constraints, the discussion will be somewhat informal. A formal version appears in [2] and is left for the full paper. A global snapshot of the system at a given time t in a particular run r, which we will denote by $\textsc{Snap}(r, t)$, consists of an instantaneous description of the local states of all agents in the system, as well as the contents of the communication channels, at that point in the run. Mechanisms for recording global states come in useful, for example, in association with recovery from system failure. In fact, many applications use such algorithms in order to retain "checkpoints": global states that can be "rolled back" into, when failure occurs (see [20]). Whereas in asynchronous systems the snapshot can only be approximated (see [6]), in systems with a global clock it is possible to compute $\textsc{Snap}(r, t)$ precisely. Indeed, since agents have access to a global clock, if they keep track of their full history, then such a snapshot can be recorded without the need of communication. But the cost of doing this is prohibitive. A natural solution is to record periodical snapshots every X rounds, say. More flexible would be a solution that allows snapshots to be initiated spontaneously, whenever there is good reason to do so. E.g, when some major transaction is completed, or when there is an external indication of an impending storm, requiring a snapshot to be taken.

Consider the problem of taking a spontaneously-generated snapshot. If each agent records its own local state in a global snapshot, then the recording actions are a simultaneous response to the snapshot trigger. By Theorem 2 this requires a causal broom structure with respect to the arrival of a spontaneous ext_Snap external triggering message. We now describe an optimal distributed snapshot protocol, for the model γ^{max} when agents share a global clock. The code for the protocol appears in Figure 4. For every agent $j \in \mathbb{P}$, define

$$\mathsf{Rad}(j) \quad = \quad \max\{\delta(j, h) : h \in \mathbb{P}\}.$$

Each agent i maintains a local variable named $\mathtt{Snap\_Time}_i$, which is initially set to ∞. A time-efficient solution would work as follows: Suppose that a spontaneous snapshot request appears at (i_0, t_0). Then if $t_0 + \mathsf{Rad}(i_0) < \mathtt{Snap\_Time}_{i_0}$, agent i_0 sets $\mathtt{Snap\_Time}_{i_0}$ to $t_0 + \mathsf{Rad}(i_0)$ and initiates a flooding of the network by sending a "Snap_msg" labelled with $\mathtt{Snap\_Time}_i$. When agent i receives a snap request labelled by a snap time T_j, it compares the current value of $\mathtt{Snap\_Time}_i$ with $t_j + \mathsf{Rad}(j)$ and with T_j. If T_j (or $t_i + \mathsf{Rad}(i)$) is smaller than $\mathtt{Snap\_Time}_i$, then agent i updates $\mathtt{Snap\_Time}_i$ to the lower value and initiates a flooding of the network with a $\mathsf{Snap\_msg}(\mathtt{Snap\_Time}_i)$ request. Finally, agents record their local states at the earliest time for which they received a snap message. (In order to account for the contents of the network's channels, they proceed to record messages received on incoming channels until the channels' bounds are met.)

```
OptimalDistributedSnapshot:          % code for agent i
01 Snap_Time_i ← ∞;
02 while True do
03   if time_i = Snap_Time_i then
04     STATE_i ← local state;
05     Snap_Time_i ← ∞;
06   else if ext_Snap or Snap_msg_j(T_j) msgs arrived then
07     candidate_i ← min{T_j : received Snap_msg_j(T_j)};
08     candidate_i ← min{candidate_i, time_i + Rad(i)};
09     if candidate_i < Snap_Time_i then
10       Snap_Time_i ← candidate_i;
11       broadcast Snap_msg_i(Snap_Time_i) to neighbors.
12 end while
```

Figure 4: An Optimal Distributed Snapshot Protocol

It is easy to see that every agent initiates at most one flooding in this algorithm, though in practice much fewer will be initiated. Moreover, the local states are recorded simultaneously, at the earliest time at which a broom exists for the arrival of external ext_Snap message. This ensures correctness. Finally, a broom is formed in a run of this protocol iff one would be formed in the corresponding run (in the sense that all transmission times and external ext_Snap messages are the same), of a full-information protocol and so this protocol is optimally fast in all cases. No protocol could beat this one, on any run when comparing corresponding runs. Formally, we obtain

THEOREM 3. *The* Optimal Distributed Snapshot *protocol of Figure 4 is* ***all-case optimal****: For every behavior of nature it records the state as soon as any protocol can.*

[6] We are thankful to Gadi Taubenfeld for suggesting this question.

4. RELATING KNOWLEDGE & ORDERED SIMULTANEOUS RESPONSES

We now define the Ordered Joint Response problem more formally.

DEFINITION 4 (ORDERED JOINT RESPONSE). *Let e_s be an external input and let $A^1, \ldots, A^k$ be disjoint sets of responses. A protocol P solves the instance $\mathsf{OJR}\langle e_s, A^1, \ldots, A^k\rangle$ of ordered joint response if P guarantees for every $h \leq k$ that in every run r in which some response $\alpha \in A^h$ takes place the following conditions are met:*

Triggering: *The trigerring event e_s and all responses in $A^1 \cup \cdots \cup A^h$ occur in r; and*

Simultaneity: *All responses in the same set A^g are performed simultaneously, for $1 \leq g \leq h$; and*

Linear Ordering: *$t_0 \leq t_1 \leq \cdots \leq t_h$, where t_0 is the time that e_s occurs in r and t_g is the time at which responses in A^g do, for $g = 1, \ldots, h$.*

The simultaneous response problem SR of [3] coincides with a particular subcase of OJR in which $k = 1$: Following the occurrence of the triggering event, all responses must be performed simultaneously. Similarly, the ordered response problem is also a sub-case of OJR, one in which $|A^h| = 1$ for all $h \leq k$.[7] Consider the shape of solutions satisfying $\mathsf{OJR}\langle e_s, A_1, A_2\rangle$, an instance with $k = 2$. Clearly, for every $\alpha_1 \in A_2$ and $\alpha_2 \in A_2$ occurring in a run r the protocol must solve ordered response and thus produce an appropriate centipede. Moreover, for each of A_1 and A_2 the protocol must solve simultaneous response, producing a broom. Does a solution need only to produce all of these induced centipedes and the two brooms? We shall show that more is required. Solutions satisfying OJR are associated with a particular shape that combines centipedes and brooms in a natural way. To show this, we apply the connection between simultaneity and common knowledge.

As has been well-established in the literature, simultaneously coordinated actions are intimately connected to common knowledge: When they are performed, the participants have common knowledge of this, and they also have common knowledge that all preconditions of the actions have been satisfied [10, 11, 14]. In the case of OJR, we can show that a particular nested common knowledge formula is a necessary condition for coordinated action. In what follows we denote the set of agents related to the response cluster A^h by $\mathsf{I}^h \subseteq \mathbb{P}$, for all $h \leq k$.

THEOREM 4. *Assume that P is a deterministic protocol solving the instance $\mathsf{OJR}\langle e_s, A^1, \ldots, A^k\rangle$, let $r \in \mathcal{R}^{\mathsf{max}} = \mathcal{R}(P, \gamma^{\mathsf{max}})$, and let $1 \leq h \leq k$. If the responses in A^h occur at time t_h in r, then*

$$(\mathcal{R}^{\mathsf{max}}, r, t_h) \vDash C_{\mathsf{I}^h} C_{\mathsf{I}^{h-1}} \cdots C_{\mathsf{I}^1} \mathsf{occ}(e_s).$$

[7] Actually OJR is defined along weaker constraints: if the responses occur they must do so simultaneously, whereas in SR and OR the responses must occur in every run where the trigger event e_s occurs.

PROOF. *(Sketch:)* Using the notations in the theorem statement, we prove by induction on $h \geq 1$ that for all $t'_h \geq t_h$: $(\mathcal{R}^{\mathsf{max}}, r, t'_h) \vDash C_{\mathsf{I}^h} C_{\mathsf{I}^{h-1}} \cdots C_{\mathsf{I}^1} \mathsf{occ}(e_s)$.
The results of [10] imply that when an action joint to I^h is performed, the members of I^h have common knowledge that it is being performed. The fact that each agent $j \in \mathsf{I}^h$ is assumed to recall the responses it performed (see Section 2) means that this common knowledge is maintained at all times $t'_h > t_h$. The claim follows inductively from the fact that A^1 is performed only if e_s occurs, while A^h for $h > 1$ is performed only at or after A^{h-1} has been performed, and so the corresponding subformula for $h - 1$ holds. □

Just as centipedes are closely related to ordered coordination and to nested knowledge formulas, and brooms correspond to common knowledge and simultaneous coordination, a natural composition of the two, which we call a *centibroom*, captures nested common knowledge, and, in turn, linearly ordered clusters of simultaneous actions. Formally,

DEFINITION 5 (CENTIBROOM). *Let $r \in \mathcal{R}^{\mathsf{max}}$, let $\mathsf{I}^h \subseteq \mathbb{P}$ for $1 \leq h \leq k$. A **centibroom** for $\langle i_0, \mathsf{I}^1, \ldots, \mathsf{I}^k\rangle$ in $(r, t..t')$ is a sequence of nodes $\theta_0 \rightsquigarrow_r \theta_1 \rightsquigarrow_r \cdots \rightsquigarrow_r \theta_k$ such that $\theta_0 = \langle i_0, t\rangle$, and $\theta_h \dashrightarrow \langle i^h_m, t'\rangle$ holds for all $h = 1, \ldots, k$ and $i^h_m \in \mathsf{I}^h$.*

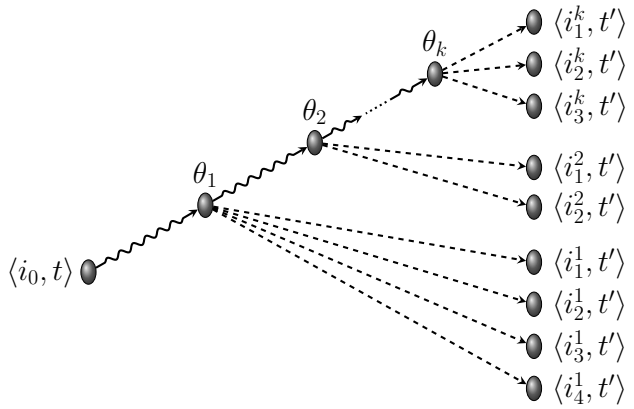


Figure 5: A centibroom for $\langle i_0, \mathsf{I}^1, \ldots, \mathsf{I}^k\rangle$ in $(r, t..t')$.

A centibroom for $\langle i_0, \mathsf{I}^1\rangle$ is one in which $k = 1$ and there is only one node θ_1. This is a *broom* (see Fig. 1(b)). As mentioned, brooms were shown in [3] to be closely related to common knowledge gain, and to coordinating a Simultaneous Response. A centibroom can be viewed as a generalized centipede, in which every "leg" is replaced by a broom structure.[8]

The Nested Common Knowledge Gain Theorem, our main technical result in this paper, now follows. The theorem shows that, in terms of communication, nested common knowledge requires, at the very least, the existence of a centibroom among the involved agents.

[8] The structure we now call a broom was originally called a *centibroom* in [3]. We have since changed the terminology because what is now called a centibroom consists of a centipede whose legs are replaced by brooms.

THEOREM 5 (NESTED COMMON KNOWLEDGE GAIN). *Let P be a deterministic protocol, $\mathsf{I}^h \subseteq \mathbb{P}$ for $h = 1 \ldots k$, and let $r \in \mathcal{R}^{\mathsf{max}} = \mathcal{R}(P, \gamma^{\mathsf{max}})$. Assume that e_{s} is the arrival of an external input at $\theta = \langle i_0, t\rangle$ in r. If*

$$(\mathcal{R}^{\mathsf{max}}, r, t') \vDash C_{\mathsf{I}^k} C_{\mathsf{I}^{k-1}} \cdots C_{\mathsf{I}^1} \mathsf{occ}(e_{\mathsf{s}}),$$

then there is a centibroom for $\langle i_0, \mathsf{I}^1, \ldots, \mathsf{I}^k\rangle$ in $(r, t..t')$.

PROOF. Assume the notations and conditions of the theorem. We will use $\mathsf{I}^h = \{i^h_1, \ldots, i^h_{s_h}\}$ to denote the set of agents $\{i | \langle i, a\rangle \in A^h\}$ participating in the responses A^h, for every $h < k$. Let $\bar{\mathcal{K}}_{\mathsf{I}^h} = K_{i^h_1} \cdots K_{i^h_{s_h}}$ denote the string of (nested) knowledge operators spanning the agents in I^h in sequence. We write $(\bar{\mathcal{K}}_{\mathsf{I}^h})^m$ to denote m consecutive copies of $\bar{\mathcal{K}}_{\mathsf{I}^h}$. Denote $d = t' - t + 1$. The fact that $(\mathcal{R}^{\mathsf{max}}, r, t') \vDash C_{\mathsf{I}^k} C_{\mathsf{I}^{k-1}} \cdots C_{\mathsf{I}^1} \mathsf{occ}(e_{\mathsf{s}})$ holds implies by definition of common knowledge that

$$(\mathcal{R}^{\mathsf{max}}, r, t') \vDash (\bar{\mathcal{K}}_{\mathsf{I}^k})^d \cdot (\bar{\mathcal{K}}_{\mathsf{I}^{k-1}})^d \cdots (\bar{\mathcal{K}}_{\mathsf{I}^1})^d \, \mathsf{occ}(e_{\mathsf{s}}).$$

By Theorem 1 (Knowledge Gain), there is a centipede

$$\begin{array}{rcl} \sigma & = & \langle i_0, t\rangle \rightsquigarrow_r \\ & & \theta^k_1 \rightsquigarrow_r \cdots \rightsquigarrow_r \theta^k_{d \cdot s_k} \rightsquigarrow_r \\ & & \theta^{k-1}_1 \rightsquigarrow_r \cdots \rightsquigarrow_r \theta^k_{d \cdot s_k} \rightsquigarrow_r \\ & & \quad \cdots \\ & & \theta^1_1 \rightsquigarrow_r \cdots \rightsquigarrow_r \theta^1_{d \cdot s_1} \end{array}$$

for $\langle i_0, (i^k_1, \ldots, i^k_{s_k})^d, \ldots, (i^1_1, \ldots, i^1_{s_1})^d\rangle$ in $(r, t..t')$.

We partition the centipede σ into the segments Θ_k to Θ_1 such that $\Theta_h = \langle \theta^h_1, .., \theta^h_{d \cdot s_h}\rangle$. Thus, each Θ^h corresponds to the $\bar{\mathcal{K}}_{\mathsf{I}^h}$ portion of the formula. Note that if $h < k$ then $\theta \rightsquigarrow_r \theta'$ for every $\theta \in \Theta_h$ and $\theta' \in \Theta_{h+1}$. Moreover, denoting $\theta = (i_\theta, t_\theta)$ and $\theta' = (i_{\theta'}, t_{\theta'})$, by Fact 1 we obtain that if $\theta \neq \theta'$ then $t_\theta < t_{\theta'}$. It follows that there can be at most $t' - t + 1 = d$ distinct nodes $\beta_1 \rightsquigarrow_r \beta_2 \rightsquigarrow_r \cdots \rightsquigarrow_r \beta_\ell$ in σ, and in particular at most d distinct nodes in every segment Θ_h of σ.

Given $h \leq k$, recall that $s_h = |\mathsf{I}^h|$ and that $\theta^h_\ell \dashrightarrow i^h_{(\ell \bmod s_h)+1}$ holds for each $\ell \leq d \cdot s_h$. As the segment Θ_h contains $d \cdot s_h$ nodes of which at most d are distinct, by the pigeonhole principle there must exist some node $\beta_h \in \Theta_h$ such that $\beta_h = \theta^h_x = \theta^h_{x+1} = \cdots = \theta^h_{x+s_h-1}$ for some $x \in [1..s_h \cdot d]$. By definition of centipede and the structure of our particular centipede σ, we get that $\beta_h \dashrightarrow \langle i^h_{x+\delta(\bmod s_h)}, t'\rangle$ for all $\delta \in [0..s_h - 1]$. It thus follows that $\beta_h \dashrightarrow \langle i, t'\rangle$ for all $i \in \mathsf{I}^h$. As noted above, we also have that $\theta_0 \rightsquigarrow_r \beta_1 \rightsquigarrow_r \beta_2 \rightsquigarrow_r \cdots \rightsquigarrow_r \beta_k$. We conclude that $\langle \theta_0, \beta_1, .., \beta_k\rangle$ is a centibroom for $\langle i_0, \mathsf{I}^1, \ldots, \mathsf{I}^k\rangle$ in $(r, t..t')$, as desired. □

Theorem 5 presents a strict and significant generalization of both the Knowledge Gain Theorem (Theorem 1 above) and of the Common Knowledge Gain Theorem of [3] to the case of nested *common knowledge*. It is the first nontrivial and useful nested CK gain theorem that we are aware of. Recall from Theorem 4 that nested common knowledge is a prerequisite for action in OJR problems. Combining the two theorems, we obtain a strict generalization of both the Centipede Theorem and the Broom Theorem of [3], matching centibrooms with ordered joint response.

COROLLARY 1 (CENTIBROOM THEOREM). *Assume that P satisfies the $\mathsf{OJR}\langle e_{\mathsf{s}}, A^1, \ldots, A^k\rangle$ property, that e_{s} occurs at $\langle i_0, t\rangle$ in $r \in \mathcal{R}(P, \gamma^{\mathsf{max}})$. Denote by I^m the set of agents responding in A^m, for $m = 1, \ldots, k$. For every $1 \leq h \leq k$, if the responses in A^h are performed at time t_h in r, then there is a centibroom for $\langle i_0, \mathsf{I}^1, \ldots, \mathsf{I}^h\rangle$ in $(r, t..t_h)$.*

5. CHARACTERIZING GENERAL ORDERED RESPONSE

We define a *response ordering* to be a finite directed graph $\mathrm{RO} = (V\langle T, A\rangle, \preceq)$ where the set of nodes $V = T \cup A$ is a disjoint union of the set of (externally initiated) triggering events T, and the a set of response actions A. Moreover, $\preceq$ is a preorder over $A \cup T$ (a reflexive and transitive binary relation), in which the nodes of T are all initial elements. Thus, $\beta \preceq \tau$ for $\tau \in T$ is possible only if $\beta = \tau$. Responses have the form $\alpha = (\mathsf{a}, i)$, where a is an action to be performed by agent i. We define base_α, the *trigger base* of a response $\alpha \in A$, by

$$\mathsf{base}_\alpha = \{e \in T : \ e \preceq \alpha\}$$

DEFINITION 6 (GENERAL ORDERED RESPONSE). *A response ordering $\mathrm{RO} = (V\langle T, A\rangle, \preceq)$ defines an instance $\mathsf{GR}\langle \mathrm{RO}\rangle$ of the **General Ordered Response** problem. A protocol P solves $\mathsf{GR}\langle \mathrm{RO}\rangle$ if it guarantees both*

Triggering: *A response $\alpha \in A$ occurs in a run iff all of the events in base_α occur; and*

Weak Ordering: *If $\alpha_1 \preceq \alpha_2$, and in a particular run α_1 occurs at time t_1 while α_2 occurs at t_2, then $t_1 \leq t_2$.*

Given the weak ordering clause, nodes on a cycle in the response ordering graph are responses that must be performed simultaneously in every solution to the problem. Characterizing the shape of GOR coordination is done by focusing on the ability of GOR to specify that a collection of disjoint sets of responses (we think of them as *clusters*) will be performed such that all responses in a cluster take place simultaneously. Moreover, any linearly ordered set of such clusters, together with an initial triggering event in their base, define an instance of OJR as a subproblem of the given GOR. By combining the above intuition with Theorems 4 and 5 that relate to the OJR problem, we can characterize the causal requirements for general response problems.

When the response ordering RO is a DAG, it specifies a *partial order* on the individual responses. Otherwise, it can be viewed as a directed graph, and every directed graph can be decomposed into its strongly connected components (SCCs) [9]. This decomposition naturally induces a graph on the SCCs, which is itself a DAG. Given an instance $\mathsf{GR}\langle A, T, \mathrm{RO}\rangle$, let $\mathrm{S} = \{\mathsf{scc}_1, \ldots, \mathsf{scc}_k\}$ be the set of strongly connected components of $A \in \mathrm{RO}$, and let I be a node labelling such that $\mathsf{I}(\mathsf{scc}_h) = \mathsf{I}^h$ is the set of agents performing responses in scc_h. (In particular, if scc_h is a single response, then $\mathsf{I}(\mathsf{scc}_h)$ is the agent performing it.) Then

CRO $= \langle \mathrm{S}, T, \preceq' \rangle$ where $\mathsf{scc}_i \preceq' \mathsf{scc}_j$ iff $\alpha_i \preceq \alpha_j$ for some $\alpha_i \in \mathsf{scc}_i$ and $\alpha_j \in \mathsf{scc}_j$, which we call the *DAG decomposition of* RO.

Going back to Example 2, a part of the detailed response ordering, containing cycles for agent groups that must react simultaneously, is shown in Figure 6. Figure 1, shown earlier, is the SCC decomposition of this full RO.

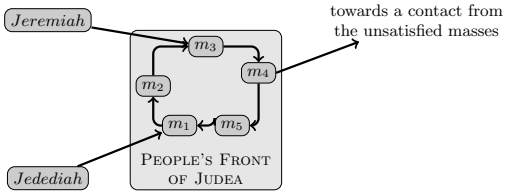


Figure 6: Part of the detailed RO for Judea, 71AD

Formally, we show:

THEOREM 6. *Fix a GOR instance* $\Gamma = \mathsf{GR}\langle A, T, \mathrm{RO} \rangle$ *and a response* $\alpha \in A$. *Assume that* P *solves* Γ, *and let* $r \in \mathcal{R}^{\mathsf{max}} = \mathcal{R}(P, \gamma^{\mathsf{max}})$ *be a run in which* α *takes place at the time* t'. *Then*

a) If RO *is a DAG, then* ***there exists a centipede*** *for*

$$\langle i_0, \mathsf{I}(\alpha_1), \ldots, \mathsf{I}(\alpha_k) \rangle$$

in $(r, t..t')$ ***for every path*** $e_0 \preceq \alpha_1 \preceq \alpha_2 \preceq \cdots \preceq \alpha_k = \alpha$ *in* RO *such that* e_0 *occurs at the* $\langle i_0, t \rangle$ *in* r. *More generally,*

b) In case RO *is not a DAG, let* CRO $= \langle \mathrm{S}, T, \preceq' \rangle$ *be the DAG decomposition of* RO. *Then* ***there exists a centibroom*** *for* $\langle i_0, \mathsf{I}(\mathsf{scc}_1), \ldots, \mathsf{I}(\mathsf{scc}_k) \rangle$ *in* $(r, t..t')$, ***for every path*** $e_0 \preceq' \mathsf{scc}_1 \preceq' \mathsf{scc}_2 \preceq' \cdots \preceq' \mathsf{scc}_k = \mathsf{scc}_\alpha$ *in* CRO *such that* $\alpha \in \mathsf{scc}_\alpha$ *and* e_0 *occurs at* $\langle i_0, t \rangle$ *in* r.

PROOF. *(Sketch:)* Part (a) is an instance of part (b) in which all SCCs are singletons, since a centipede is a centibroom in which every broom contains a single target node. It thus suffices to show part (b). Under the conditions and the notation of the theorem statement, the existence of the path $e_0 \preceq' \mathsf{scc}_1 \preceq' \mathsf{scc}_2 \preceq' \cdots \preceq' \mathsf{scc}_k = \mathsf{scc}_\alpha$ in CRO ensures that the protocol P must satisfy the $\mathsf{OJR}(e_0, \mathsf{scc}_1, \ldots, \mathsf{scc}_k)$ property. Denoting $\mathsf{I}^h = \mathsf{I}(\mathsf{scc}_h)$ for $1 \leq h \leq k$, Theorem 4 implies that $(\mathcal{R}^{\mathsf{max}}, r, t') \vDash C_{\mathsf{I}^k} C_{\mathsf{I}^{k-1}} \cdots C_{\mathsf{I}^1} \mathsf{occ}(e_0)$. The claim now follows immediately from Theorem 5 (nested common knowledge gain). □

Notice that a centipede contains a linear chain of syncausally-related agents that mimic the linear temporal ordering that is required of the responses in an Ordered Response. The shape of the OR problem and the "shape" of its solution are closely related. Theorem 6 shows that for more general specifications such as a partial-order GOR, the shapes are not as tightly connected. The partial order implies a set of linear orderings, and the centipedes for these must be constructed. In a precise sense, it is the required logical structure, specified in terms of a conjunction of nested knowledge and nested common knowledge formulas, that constrains the shape of the solution.

Theorem 6 states *necessary* conditions for any solution to GOR problems in a context in which the environment can deliver message subject to given upper bounds on transmission times. In a precise sense, this theorem cannot be strengthened: In the full paper we show (using the same technique as in [4]) that the condition in Theorem 6 is in a precise sense *sufficient*, as well as necessary: In a context in which the agents have access to a global clock there is a full-information protocol solving the GOR, in which each response α is performed at the first time t' at which all the required centibrooms for α by Theorem 6 exist. For every behavior of nature, the resulting protocols ensures that the agents respond in the fastest possible manner.

Using Theorem 6 to analyze Example 2, we obtain that in order for a rebellion to start in Judea we need that all three instigators revolt, and that there exist centibroom communication patterns for each of the following chains of population groups (using *masses* for the unsatisfied masses and *old regime* for supporters of the old regime):

- $Jeremiah \preceq' PFJ \preceq' masses \preceq' old\,regime$,
- $Jedediah \preceq' PFJ \preceq' masses \preceq' old\,regime$, and
- $Brian \preceq' JPF \preceq' masses \preceq' old\,regime$.

We remark that a GOR specifying a partial order on responses (with no simultaneous actions required), as in Theorem 6(a), is implementable in the asynchronous model as well. In the asynchronous model, centipedes reduce to message chains ([3]), and so the message chains must closely follow the paths in the graph of RO in this case (cf. Parikh and Krasucki [21]). This is no longer the case in the synchronous model, since there the centipedes impose a richer and more flexible structure in the shape of GOR implementations.

6. CONCLUSIONS

In summary, this paper uses an epistemic analysis to significantly extend our understanding of the interaction between knowledge, time and causality in multi-agent systems. This new understanding can be applied to a broad class of coordination problems, yielding insights and guidance regarding how to design efficient, even optimal, solutions to coordination tasks. Natural extensions currently being explored consider the analysis of coordination tasks stated in terms of explicit time bounds, rather than orderings. For example, if we specify that response α_2 must occur no later than 5 days after response α_1 has occurred, or even that responses α_1 and α_2 must occur exactly 3 days apart from each other. These issues are explored in [5] and [12].

The subject of network dynamics, the diffusion of ideas and actions through a social network, has been extensively studied since the seventies, and in particular in the last decade. We believe that our results pertaining to minimal networks will be of value in this ongoing effort. It is interesting to note that the minimal communication graph needed for achieving nested common knowledge in our system is significantly

sparser than that needed by Chwe in [8]. Currently our problem formulations are too far apart from Chwe's to allow for rigorous comparison, so further research is needed in order to get to the bottom of this.

The current paper extends and generalizes our previous work in [3], from the study of sequential or strictly simultaneous coordination to GOR problems. The latter allow coordination specified by an arbitrary partial order, or in terms of a partial order defined of clusters, where each cluster of actions is necessarily simultaneous. GOR problems also allow multiple triggering events, and responses must be performed if all of the spontaneous triggering events that they depend on occur. Thus, the dependence on triggers is conjunctive. A natural question that arises from the current investigation is, what would be the effect on required communication if the dependence on triggers could be more general, say defined by a general boolean function. Similarly, perhaps the interdependence among responses could also be specified more generally. Indeed, GOR problems can be specified by a suitably expressive temporal logic [18]. Is there a sensible way of relating general temporal-epistemic formulas to the causal structure required to attain them?

We illustrated the applicability of a causal analysis in terms of syncausality and bound guarantees in Section 3, showing how it can be used to derive an optimal solution to the synchronous distributed snapshot problem. Our new results can similarly allow the synthesis of efficient, even optimal, solutions to many other distributed tasks in synchronous settings. A promising direction for further study is to explore the epistemic underpinnings of particular tasks, and apply a causal analysis in this style, in order to improve the analysis and solutions for such tasks. There is much room for further investigation.

7. REFERENCES

[1] C. Attiya and J. Welch. *Distributed Computing: Fundamentals, Simulations, and Advanced Topics, 2nd Edition*. Wiley, 2004.

[2] I. Ben-Zvi. *Causality, Knowledge and Coordination in Distributed Systems*. PhD thesis, Technion, Israel, 2011.

[3] I. Ben-Zvi and Y. Moses. Beyond Lamport's happened-before: On the role of time bounds in synchronous systems. In *DISC 2010*, pages 421–436, 2010.

[4] I. Ben-Zvi and Y. Moses. On interactive knowledge with bounded communication. *Journal of Applied Non-Classical Logics*, 21(3-4):323–354, 2011.

[5] I. Ben-Zvi and Y. Moses. Agent-time epistemics and coordination. In *Proceedings of the 5th Indian conference on logic and its applications*, ICLA'13, 2013.

[6] K. M. Chandy and L. Lamport. Distributed snapshots: determining global states of distributed systems. *ACM Trans. on Computer Systems*, 3(1):63–75, 1985.

[7] K. M. Chandy and J. Misra. How processes learn. *Distributed Computing*, 1(1):40–52, 1986.

[8] M. S.-Y. Chwe. Communication and coordination in social networks. *Review of Economic Studies*, 67(1):1–16, 2000.

[9] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms, 3rd ed.* MIT Press, 2009.

[10] R. Fagin, J. Y. Halpern, Y. Moses, and M. Y. Vardi. Common knowledge revisited. *Annals of Pure and Applied Logic*, 96(1-3):89 – 105, 1999.

[11] R. Fagin, J. Y. Halpern, Y. Moses, and M. Y. Vardi. *Reasoning about Knowledge*. MIT Press, 2003.

[12] Y. Gonczarowski and Y. Moses. Timely common knowledge: Characterising asymmetric distributed coordination via vectorial fixed points. In *Proceedings of the 14th Conference on Theoretical Aspects of Rationality and Knowledge*, TARK XIV, 2013.

[13] M. Granovetter. The Strength of Weak Ties. *The American Journal of Sociology*, 78(6):1360–1380, 1973.

[14] J. Y. Halpern and Y. Moses. Knowledge and common knowledge in a distributed environment. *Journal of the ACM*, 37(3):549–587, 1990. A preliminary version appeared in *Proc. 3rd ACM Symposium on Principles of Distributed Computing*, 1984.

[15] L. Lamport. Time, clocks, and the ordering of events in a distributed system. *Communications of the ACM*, 21(7):558–565, 1978.

[16] L. Lamport. Using time instead of timeout for fault-tolerant distributed systems. *ACM Trans. Program. Lang. Syst.*, 6(2):254–280, 1984.

[17] N. A. Lynch. *Distributed Algorithms*. Morgan Kaufmann, 1996.

[18] Z. Manna and A. Pnueli. *The Temporal Logic of Reactive and Concurrent Systems*, volume 1. Springer-Verlag, Berlin/New York, 1992.

[19] S. Morris. Contagion. *Review of Economic Studies*, 67(1):57–78, January 2000.

[20] N. Neves and W. K. Fuchs. Using time to improve the performance of coordinated checkpointing. In *Proceedings of the 2nd International Computer Performance and Dependability Symposium (IPDS '96)*, 1996.

[21] R. Parikh and P. Krasucki. Levels of knowledge in distributed computing. *Sādhanā*, 17(1):167–191, 1992.

[22] T. Schelling. Dynamic models of segregation. *Journal of Mathematical Sociology*, 1(1):143–186, 1971.

Language-based Games

Adam Bjorndahl
Cornell University
Dept. Mathematics
Ithaca, NY 14853, USA
abjorndahl@math.cornell.edu

Joseph Y. Halpern
Cornell University
Dept. Computer Science
Ithaca, NY 14853, USA
halpern@cs.cornell.edu

Rafael Pass
Cornell University
Dept. Computer Science
Ithaca, NY 14853, USA
rafael@cs.cornell.edu

ABSTRACT

We introduce *language-based games*, a generalization of *psychological games* [6] that can also capture *reference-dependent preferences* [7]. The idea is to extend the domain of the utility function to *situations*, maximal consistent sets in some language. The role of the underlying language in this framework is thus particularly critical. Of special interest are languages that can express only *coarse* beliefs [9]. Despite the expressive power of the approach, we show that it can describe games in a simple, natural way. Nash equilibrium and rationalizability are generalized to this setting; Nash equilibrium is shown not to exist in general, while the existence of rationalizable strategies is proved under mild conditions.

Categories and Subject Descriptors

F.4.1 [**Mathematical Logic and Formal Languages**]: Mathematical Logic—*modal logic*; I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*multiagent systems*; J.4 [**Social and Behavioral Sciences**]: Economics

General Terms

Economics, Theory

Keywords

Psychological games, epistemic game theory, rationalizability

1. INTRODUCTION

In a classical, normal-form game, an *outcome* is a tuple of strategies, one for each player; intuitively, an outcome is just a record of which strategy each player chose to play. Players' preferences are formalized by utility functions defined on the set of all such outcomes. This framework thereby hard-codes the assumption that a player can prefer one state of the world to another only insofar as they differ in the outcome of the game.

Perhaps unsurprisingly, this model is too restrictive to account for a broad class of interactions that otherwise seem well-suited to a game-theoretic analysis. For example, one might wish to model players who feel guilt, wish to surprise their opponents, or are motivated by a desire to live up to what is expected of them. Work on *psychological game theory*, beginning with [6] and expanded in [3], is an enrichment of the classical setting meant to capture these kinds of preferences and motivations. In a similar vein, work on *reference-dependent preferences*, as developed in [7], formalizes phenomena such as loss-aversion by augmenting players' preferences with an additional sense of gain or loss derived by comparing the actual outcome to what was expected.

In both of these theories, the method of generalization takes the same basic form: the domain of the utility functions is enlarged to include not only the outcomes of the game, but also the beliefs of the players. The resulting structure may be fairly complex; for instance, in psychological game theory, since the goal is to model preferences that depend not only on beliefs about outcomes, but also beliefs about beliefs, beliefs about beliefs about beliefs, and so on, the domain of the utility functions is extended to include infinite hierarchies of beliefs.

The model we present in this paper, though motivated in part by a desire to capture belief-dependent preferences, is geared towards a much more general goal. Besides being expressive enough to subsume existing systems such as those described above, it establishes a general framework for modeling players with richer preferences. Moreover, it is equally capable of representing *impoverished* preferences, a canonical example of which are so-called "coarse beliefs" or "categorical thinking" [9]. More specifically, our formalism provides good practical and theoretical tools for handling beliefs as discrete rather than continuous objects, an advantage that is particularly relevant in the context of psychological effects in games.

Despite this expressive power, the system is easy to use: player preferences are represented in a simple and natural manner, narrowing the divide between intuition and formalism. As a preliminary illustration of some of these points, consider the following simple example.

Example 1. A surprise proposal. Alice and Bob have been dating for a while now, and Bob has decided that the time is right to pop the big question. Though he is not one for fancy proposals, he does want it to be a surprise. In fact, if Alice expects the proposal, Bob would prefer to postpone it entirely until such time as it might be a surprise. Otherwise, if Alice is not expecting it, Bob's preference is to take the opportunity.

We might summarize this scenario by the following table of payoffs for Bob:

TARK 2013, Chennai, India.

	p	$\neg p$
$B_A\, p$	0	1
$\neg B_A\, p$	1	0

Table 1: The surprise proposal.

In this table, we denote Bob's two strategies, proposing and not proposing, as p and $\neg p$, respectively, and use $B_A p$ (respectively, $\neg B_A p$) to denote that Alice is expecting (respectively, not expecting) the proposal.

Granted, whether or not Alice expects a proposal may be more than a binary affair: she may, for example, consider a proposal unlikely, somewhat likely, very likely, or certain. But there is good reason to think (see [9]) that an accurate model of her expectations stops here, with some small *finite* number k of distinct "levels" of belief, rather than a continuum. Table 1, for simplicity, assumes that $k = 2$, though this is easily generalized to larger values.

Note that although Alice does not have a choice to make (formally, her strategy set is a singleton), she does have beliefs about which strategy Bob will choose. To represent Bob's preference for a surprise proposal, we must incorporate Alice's beliefs about Bob's choice of strategy into Bob's utility function. In psychological game theory, this is accomplished by letting $\alpha \in [0, 1]$ be the probability that Alice assigns to Bob proposing, and defining Bob's utility function u_B in some simple way so that it is decreasing in α if Bob chooses to propose, and increasing in α otherwise:[1]

$$u_B(x, \alpha) = \begin{cases} 1 - \alpha & \text{if } x = p \\ \alpha & \text{if } x = \neg p. \end{cases}$$

The function u_B agrees with the table at its extreme points if we identify $B_A p$ with $\alpha = 1$ and $\neg B_A p$ with $\alpha = 0$. Otherwise, for the infinity of other values that α may take between 0 and 1, u_B yields a linear combination of the appropriate extreme points. Thus, in a sense, u_B is a continuous approximation to a scenario that is essentially discrete.

We view Table 1 as *defining* Bob's utility. To coax an actual utility function from this table, let the variable S denote a *situation*, which for the time being we can conceptualize as a collection of statements about the game; in this case, these include whether or not Bob is proposing, and whether or not Alice believes he is proposing. We then define

$$u_B(S) = \begin{cases} 0 & \text{if } p \in S \text{ and } B_A\, p \in S \\ 1 & \text{if } p \in S \text{ and } \neg B_A\, p \in S \\ 1 & \text{if } \neg p \in S \text{ and } B_A\, p \in S \\ 0 & \text{if } \neg p \in S \text{ and } \neg B_A\, p \in S. \end{cases}$$

In other words, Bob's utility is a function not merely of the outcome of the game (p or $\neg p$), but of a more general object we are calling a "situation", and his utility in a given situation S depends on his own actions combined with Alice's beliefs in exactly the manner prescribed by Table 1. As noted above, we may very well wish to refine our representation of Alice's state of surprise using more than two categories; indeed, we could allow a representation that permits continuous probabilities, as has been done in the literature. However, we will see that an "all-or-nothing" representation of belief is enough to capture some interesting and complex games. □

[1]Technically, in [6], Bob's utility can only be a function of his own beliefs; this is generalized in [3] in the context of extensive-form games, but the approach is applicable to normal-form games as well.

The central concept we develop in this paper is that of a *language-based game*, where utility is defined not on outcomes or the Cartesian product of outcomes with some other domain, but on *situations*. As noted, a situation can be conceptualized as a collection of statements about the game; intuitively, each statement is a description of something that might be relevant to player preferences, such as whether or not Alice believes that Bob will play a certain strategy. Of course, this notion crucially depends on just what counts as an admissible description. Indeed, the set of all admissible descriptions, which we refer to as the *underlying language* of the game, is a key component of our model. Since utility is defined on situations, and situations are sets of descriptions taken from the underlying language, a player's preferences can depend, in principle, on anything expressible in this language, and nothing more. Succintly: players can prefer one state of the world to another if and only if they can *describe* the difference between the two, where "describe" here means "express in the underlying language".

Language-based games are thus parametrized by the underlying language: changing the language changes the game. The power and versatility of our approach derives in large part from this dependence. Consider, for example, an underlying language that contains only terms refering to players' strategies. Players' preferences, then, can depend only on the outcome of the game, as is the case classically. Thus classical game theory is recovered as a special case of the present work (see Sections 2.1 and 2.2 for details).

Enriching the underlying language allows for an expansion and refinement of player preferences; in this manner we are able to subsume, for example, work on psychological game theory and reference-dependent preferences, in addition to providing some uniformity to the project of defining new and further expansions of the classical base. By contrast, restricting the underlying language coarsens the domain of player preference; this provides a framework for modeling phenomena like coarse beliefs. A combination of these two approaches yields a theory of belief-dependent preferences incorporating coarse beliefs.

For the purposes of this paper, we focus primarily on belief-dependent preferences and coarseness, although in Example 6, we examine a simple scenario where a type of procrastination is represented by a minor extension of the underlying language. We make three major contributions. First, as noted, our system is easy to use in the sense that players' preferences are represented with a simple and uncluttered formalism; complex psychological phenomena can thus be captured in a direct and intuitive manner. Second, we provide a formal game-theoretic representation of coarse beliefs, and in so doing, expose an important insight: a discrete representation of belief, often conceptually and technically easier to work with than its continuous counterpart, is sufficient to capture psychological effects that have heretofore been modeled only in a continuous framework. Section 3 provides several examples that illustrate these points. Third, we provide novel equilibrium analyses that do not depend on the continuity of the expected utility function as in [6]. (Note that such continuity assumptions are at odds with our use of coarse beliefs.)

The rest of the paper is organized as follows. In the next section, we develop the basic apparatus needed to describe

our approach. Section 3 presents a collection of examples intended to guide intuition and showcase the system. In Section 4, we show that there is a natural route by which solution concepts such as Nash equilibrium and rationalizability can be defined in our setting, and we address the question of existence. Proofs, more discussion, and further examples can be found in the full paper, which is available at `http://www.cs.cornell.edu/home/halpern/papers/lbg.pdf`.

2. FOUNDATIONS

2.1 Game forms and intuition

Much of the familiar apparatus of classical game theory is left untouched. A **game form** is a tuple $\Gamma = (N, (\Sigma_i)_{i \in N})$ where N is a finite set of *players*, which for convenience we take to be the set $\{1, \ldots, n\}$, and Σ_i is the set of *strategies available to player i*. Following standard notation, we set

$$\Sigma \coloneqq \prod_{i \in N} \Sigma_i \quad \text{and} \quad \Sigma_{-i} \coloneqq \prod_{j \neq i} \Sigma_j.$$

Elements of Σ are called *outcomes* or *strategy profiles*; given $\sigma \in \Sigma$, we denote by σ_i the ith component of the tuple σ, and by σ_{-i} the element of Σ_{-i} consisting of all but the ith component of σ.

Note that a game form does not come equipped with utility functions specifying the preferences of players over outcomes Σ. The utility functions we employ are defined on situations, which in turn are determined by the underlying language, so, before defining utility, we must first formalize these notions.

Informally, a *situation* is an exhaustive characterization of a given state of affairs using descriptions drawn from the underlying language. Assuming for the moment that we have access to a fixed "language", we might imagine a situation as being generated by simply listing all statements from that language that happen to be true of the world. Even at this intuitive level, it should be evident that the informational content of a situation is completely dependent on the expressiveness of the language. If, for example, the underlying language consists of exactly two descriptions, "It's raining" and "It's not raining", then there are only two situations:

$$\{\text{"It's raining"}\} \quad \text{and} \quad \{\text{"It's not raining"}\}.$$

Somewhat more formally, a situation S is a set of formulas drawn from a larger pool of well-formed formulas, the underlying language. We require that S include as many formulas as possible while still being consistent; we make this precise shortly.

The present formulation, informal though it is, is sufficient to allow us to capture a claim made in the introduction: any classical game can be recovered in our framework with the appropriate choice of underlying language. Specifically, let the underlying language be Σ, the set of all strategy profiles. Situations, in this case, are simply singleton subsets of Σ, as any larger set would contain distinct and thus intuitively contradictory descriptions of the outcome of the game. The set of situations can thus be identified with the set of outcomes, so a utility function defined on outcomes is readily identified with one defined on situations.

In this instance the underlying language, consisting solely of atomic, mutually incompatible formulas, is essentially structureless; one might wonder why call it a "language" at all, rather than merely a "set". Although, in principle, there are no restrictions on the kinds of objects we might consider as languages, it can be very useful to focus on those with some internal structure. This structure has two aspects: syntactic and semantic.

2.2 Syntax, semantics, and situations

The canonical form of syntactic structure in formal languages is *grammar*: a set of rules specifying how to compose well-formed formulas from atomic constituents. One of the best-known examples of a formal language generated by a grammar is the language of classical propositional logic.

Given a set Φ of *primitive propositions*, let $\mathcal{L}(\Phi)$ denote the propositional language based on Φ, namely, the set of formulas that can be obtained by starting with Φ and closing off under $\neg$ and $\wedge$. (We can define $\vee$ and $\rightarrow$ from $\neg$ and $\wedge$ as usual.) Propositional logic is easily specialized to a game-theoretic setting. Given a game form $\Gamma = (N, (\Sigma_i)_{i \in N})$, let

$$\Phi_\Gamma = \{play_i(\sigma_i) \,:\, i \in N,\, \sigma_i \in \Sigma_i\},$$

where we read $play_i(\sigma_i)$ as "player i is playing strategy σ_i". Then $\mathcal{L}(\Phi_\Gamma)$ is a language appropriate for reasoning about the strategies chosen by the players in Γ. We sometimes write $play(\sigma)$ as an abbreviation for $play_1(\sigma_1) \wedge \cdots \wedge play_n(\sigma_n)$.

Semantics provides a notion of truth. Recall that the semantics of classical propositional logic is given by *valuations* $v : \Phi \rightarrow \{\mathsf{true}, \mathsf{false}\}$. Valuations are extended to all formulas via the familiar truth tables for the logical connectives. Each valuation v thereby generates a *model*, determining the truth values of every formula in $\mathcal{L}(\Phi)$. In the case of the language $\mathcal{L}(\Phi_\Gamma)$, we restrict this class of models to those corresponding to an outcome $\sigma \in \Sigma$; that is, we consider only valuation functions v_σ defined by

$$v_\sigma(play_i(\sigma_i')) = \mathsf{true} \text{ if and only if } \sigma_i = \sigma_i'.$$

More generally, we consider only a set $\mathcal{M}$ of *admissible models*: the ones that satisfy some restrictions of interest.

A set of formulas F is said to be *satisfiable (with respect to a set $\mathcal{M}$ of admissible models)* if there is some model in $\mathcal{M}$ in which every formula of F is true. An $\mathcal{L}(\Phi)$*-situation* is then defined to be a *maximal* satisfiable set of formulas (with respect to the admissible models of $\mathcal{L}(\Phi)$): that is, a satisfiable set with no proper superset that is also satisfiable. Situations correspond to admissible models: a situation just consists of all the formulas true in some admissible model. Let $\mathcal{S}(\mathcal{L}(\Phi))$ denote the set of $\mathcal{L}(\Phi)$-situations. It is not difficult to see that $\mathcal{S}(\mathcal{L}(\Phi_\Gamma))$ can be identified with the set Σ of outcomes.

Having illustrated some of the principle concepts of our approach in the context of propositional logic, we now present the definitions in complete generality. Let $\mathcal{L}$ be a language with an associated semantics, that is, a set of admissible models providing a notion of truth. We often use the term "language" to refer to a set of well-formed formulas together with a set of admissible models (this is sometimes called a "logic"). An **$\mathcal{L}$-situation** is a maximal satisfiable set of formulas from $\mathcal{L}$. Denote by $\mathcal{S}(\mathcal{L})$ the set of $\mathcal{L}$-situations. A game form Γ is extended to an **$\mathcal{L}$-game** by adding utility functions $u_i : \mathcal{S}(\mathcal{L}) \rightarrow \mathbb{R}$, one for each player $i \in N$. $\mathcal{L}$ is called the **underlying language**; we omit it as a prefix when it is safe to do so.

If we view Γ as an $\mathcal{L}(\Phi_\Gamma)$-game, the players' utility functions are essentially defined on Σ, so an $\mathcal{L}(\Phi_\Gamma)$-game is really just a classical game based on Γ. As we saw in Section 2.1, this class of games can also be represented with the completely structureless language Σ. This may well be sufficient, especially in cases where all we care about are two or three formulas. However, having a structured underlying language makes it easier to analyze the much broader class of psychological games.

A psychological game is just like a classical game except that players' preferences can depend not only on what strategies are played, but also on what beliefs are held. While $\mathcal{L}(\Phi_\Gamma)$ is appropriate for reasoning about strategies, it cannot express anything about beliefs, so our first step is to define a richer language. Fortunately, we have at our disposal a host of candidates well-equipped for this task, namely those languages associated with epistemic logics.

Fix a game form $\Gamma = (N, (\Sigma_i)_{i\in N})$, and let $\mathcal{L}_B(\Phi_\Gamma)$ be the language obtained by starting with the primitive propositions in Φ_Γ and closing off under conjunction, negation, and the modal operators B_i, for $i \in N$. We read $B_i\varphi$ as "player i believes φ". Intuitively, this is a language for reasoning about the beliefs of the players and the strategy profiles being used.

We give semantics to $\mathcal{L}_B(\Phi_\Gamma)$ using Kripke structures, as usual. But for many examples of interest, understanding the (completely standard, although somewhat technical) details is not necessary. Example 1 was ultimately analyzed as an $\mathcal{L}_B(\Phi_\Gamma)$-game, despite the fact that we had not even defined the syntax of this language at the time, let alone its semantics. Section 3 provides more illustrations of this point.

A **Γ-structure** is a tuple $M = (\Omega, \vec{s}, Pr_1, \ldots, Pr_n)$ satisfying the following conditions:

(P1) Ω is a nonempty topological space;

(P2) each Pr_i assigns to each $\omega \in \Omega$ a probability measure $Pr_i(\omega)$ on Ω;

(P3) $\omega' \in Pr_i[\omega] \Rightarrow Pr_i(\omega') = Pr_i(\omega)$, where $Pr_i[\omega]$ abbreviates $supp(Pr_i(\omega))$, the support of the probability measure;

(P4) $\vec{s} : \Omega \to \Sigma$ satisfies $Pr_i[\omega] \subseteq \{\omega' : s_i(\omega') = s_i(\omega)\}$, where $s_i(\omega)$ denotes player i's strategy in the strategy profile $\vec{s}(\omega)$.

These conditions are standard for KD45 belief logics in a game-theoretic setting [1]. The set Ω is called the **state space**. Conditions (P1) and (P2) set the stage to represent player i's beliefs in state $\omega \in \Omega$ as the probability measure $Pr_i(\omega)$ over the state space itself. Condition (P3) says essentially that players are sure of their own beliefs. The function $\vec{s}$ is called the **strategy function**, assigning to each state a strategy profile that we think of as the strategies that the players are playing at that state. Condition (P4) thus asserts that each player is sure of his own strategy. The language $\mathcal{L}_B(\Phi_\Gamma)$ can be interpreted in any Γ-structure M via the strategy function, which induces a valuation $[\![\cdot]\!]_M : \mathcal{L}_B(\Phi_\Gamma) \to 2^\Omega$ defined recursively by:

$$\begin{array}{lcl} [\![play_i(\sigma_i)]\!]_M & := & \{\omega \in \Omega : s_i(\omega) = \sigma_i\} \\ [\![\varphi \wedge \psi]\!]_M & := & [\![\varphi]\!]_M \cap [\![\psi]\!]_M \\ [\![\neg\varphi]\!]_M & := & \Omega - [\![\varphi]\!]_M \\ [\![B_i\varphi]\!]_M & := & \{\omega \in \Omega : Pr_i[\omega] \subseteq [\![\varphi]\!]_M\}. \end{array}$$

Thus, the Boolean connectives are interpreted classically, and $B_i\varphi$ holds at state ω just in case all the states in the support of $Pr_i(\omega)$ are states where φ holds.

Pairs of the form (M, ω), where $M = (\Omega, \vec{s}, \vec{Pr})$ is a Γ-structure and $\omega \in \Omega$, play the role of admissible models for the language $\mathcal{L}_B(\Phi_\Gamma)$. Given $\varphi \in \mathcal{L}_B(\Phi_\Gamma)$, we sometimes write $(M, \omega) \models \varphi$ or just $\omega \models \varphi$ instead of $\omega \in [\![\varphi]\!]_M$, and say that ω **satisfies** φ or φ is **true at** ω; we write $M \models \varphi$ and say that φ is **valid in** M if $[\![\varphi]\!]_M = \Omega$. We say that φ is **satisfiable** if for some state ω in some Γ-structure M (i.e., for some admissible model), $\omega \models \varphi$. Given $F \subseteq \mathcal{L}_B(\Phi_\Gamma)$, we write $\omega \models F$ if for all $\varphi \in F$, $\omega \models \varphi$; we say that F is satisfiable if for some state ω in some M, $\omega \models F$.

With this notion of satisfiability, we gain access to the class of $\mathcal{L}_B(\Phi_\Gamma)$-games, where utility is defined on $\mathcal{L}_B(\Phi_\Gamma)$-situations, namely, maximal satisfiable subsets of $\mathcal{L}_B(\Phi_\Gamma)$. In particular, we can extend any game form Γ to an $\mathcal{L}_B(\Phi_\Gamma)$-game, a setting in which players' preferences can depend, in principle, on anything describable in the language $\mathcal{L}_B(\Phi_\Gamma)$.

It is not hard to show that when there is more than one player, $\mathcal{S}(\mathcal{L}_B(\Phi_\Gamma))$ is uncountable. A utility function $u_i : \mathcal{S}(\mathcal{L}_B(\Phi_\Gamma)) \to \mathbb{R}$ can therefore be quite complicated indeed. We will frequently be interested in representing preferences that are much simpler. For instance, though the surprise proposal scenario presented in Example 1 can be viewed as an $\mathcal{L}_B(\Phi_\Gamma)$-game, Bob's utility u_B does not depend on any situation as a whole, but rather is determined by a few select formulas. This motivates the following general definition, identifying a particularly easy to understand and well-behaved subclass of games.

Fix a language $\mathcal{L}$. A function $u : \mathcal{S}(\mathcal{L}) \to \mathbb{R}$ is called **finitely specified** if there is a finite set of formulas $F \subset \mathcal{L}$ and a function $f : F \to \mathbb{R}$ such that every situation $S \in \mathcal{S}(\mathcal{L})$ contains exactly one formula from F, and whenever $\varphi \in S \cap F$, $u(S) = f(\varphi)$. In other words, the value of u depends only on the formulas in F. Thus u is finitely specified if and only if it can be written in the form

$$u(S) = \begin{cases} a_1 & \text{if } \varphi_1 \in S \\ \vdots & \vdots \\ a_k & \text{if } \varphi_k \in S, \end{cases}$$

for some $a_1, \ldots, a_k \in \mathbb{R}$ and $\varphi_1, \ldots, \varphi_k \in \mathcal{L}$.

A language-based game is called finitely specified if each player's utility function is. Many games of interest are finitely specified. In a finitely specified game, we can think of a player's utility as being a function of the finite set F; indeed, we can think of the underlying language as being the structureless "language" F rather than $\mathcal{L}$.

3. EXAMPLES

We now give a few examples to exhibit both the simplicity and the expressive power of language-based games; more examples are given in the full paper. Since we focus on the language $\mathcal{L}_B(\Phi_\Gamma)$, we write $\mathcal{S}$ to abbreviate $\mathcal{S}(\mathcal{L}_B(\Phi_\Gamma))$.

Note that there is a unique strategy that player i uses in a situation $S \in \mathcal{S}$; it is the strategy σ_i such that $play_i(\sigma_i) \in S$. When describing the utility of a situation, it is often useful to extract this strategy; therefore, we define $\rho_i : \mathcal{S} \to \Sigma_i$ implicitly by the requirement $play_i(\rho_i(S)) \in S$. It is easy to check that ρ_i is well-defined.

Example 2. Indignant altruism. Alice and Bob sit down to play a classic game of prisoner's dilemma, with one twist:

neither wishes to live up to low expectations. Specifically, if Bob expects the worst of Alice (i.e. expects her to defect), then Alice, indignant at Bob's opinion of her, prefers to cooperate. Likewise for Bob. On the other hand, in the absense of such low expectations from their opponent, each will revert to their classical, self-serving behaviour.

The standard prisoner's dilemma is summarized in Table 2:

	c	d
c	(3,3)	(0,5)
d	(5,0)	(1,1)

Table 2: The classical prisoner's dilemma.

Let u_A, u_B denote the two players' utility functions according to this table, and let Γ denote the game form obtained by throwing away these functions: $\Gamma = (\{A, B\}, \Sigma_A, \Sigma_B)$ where $\Sigma_A = \Sigma_B = \{\mathsf{c}, \mathsf{d}\}$. We wish to define an $\mathcal{L}_B(\Phi_\Gamma)$-game that captures the given scenario; to do so we must define new utility functions on $\mathcal{S}$. Informally, if Bob is sure that Alice will defect, then Alice's utility for defecting is -1, regardless of what Bob does, and likewise reversing the roles of Alice and Bob; otherwise, utility is determined exactly as it is classically.

Formally, we simply define $u'_A : \mathcal{S} \to \mathbb{R}$ by

$$u'_A(S) = \begin{cases} -1 & \text{if } play_A(\mathsf{d}) \in S \text{ and} \\ & B_B\, play_A(\mathsf{d}) \in S \\ u_A(\rho_A(S), \rho_B(S)) & \text{otherwise,} \end{cases}$$

and similarly for u'_B.

Intuitively, cooperating is rational for Alice if she thinks that Bob is sure she will defect, since cooperating in this case would yield a minimum utility of 0, whereas defecting would result in a utility of -1. On the other hand, if Alice thinks that Bob is *not* sure she'll defect, then since her utility in this case would be determined classically, it is rational for her to defect, as usual.

This game has much in common with the surprise proposal of Example 1: in both games, the essential psychological element is the desire to surprise another player. Perhaps unsurprisingly, when players wish to surprise their opponents, *Nash equilibria* fail to exist—even mixed strategy equilibria. Although we have not yet defined Nash equilibrium in our setting, the classical intuition is wholly applicable: a Nash equilibrium is a state of play where players are happy with their choice of strategies *given accurate beliefs about what their opponents will choose*. But there is a fundamental tension between a state of play where everyone has accurate beliefs, and one where some player successfully surprises another.

We show formally in Section 4.2 that this game has no Nash equilibrium. On the other hand, players can certainly best-respond to their beliefs, and the corresponding iterative notion of *rationalizability* finds purchase here. In Section 4.3 we will import this solution concept into our framework and show that every strategy for the indignant altruist is rationalizable. □

Example 3. A deeply surprising proposal. Bob hopes to propose to Alice, but she wants it to be a surprise. He knows that she would be upset if it were not a surprise, so he would prefer not to propose if Alice so much as suspects it. Worse (for Bob), even if Alice does not suspect a proposal, if she suspects that Bob thinks she does, then she will also be upset, since in this case a proposal would indicate Bob's willingness to disappoint her. Of course, like the giant tortoise on whose back the world rests, this reasoning continues "all the way down"...

This example is adapted from a similar example given in [6]; in that example, the man is considering giving a gift of flowers, but rather than hoping to surprise the recipient, his goal is the exact opposite: to get her flowers just in case she *is* expecting them. Of course, the notion of "expectation" employed, both in their example and ours, is quite a bit more complicated than the usual sense of the word, involving arbitrarily deeply nested beliefs.

Nonetheless, it is relatively painless to represent Bob's preferences in the language $\mathcal{L}_B(\Phi_\Gamma)$, where $\Gamma = (\{A, B\}, \{\cdot\}, \{p, q\})$ and p and q stand for Bob's strategies of proposing and not proposing, respectively (Alice has no decision to make, so her strategy set is a singleton). For convenience, we use the symbol P_i to abbreviate $\neg B_i \neg$. Thus $P_i\varphi$ holds just in case player i is not sure that φ is false; this will be our gloss for Alice "so much as suspecting" a proposal. Define $u_B : \mathcal{S} \to \mathbb{R}$ by

$$u_B(S) = \begin{cases} 1 & \text{if } play_B(p) \in S \text{ and} \\ & (\forall k \in \mathbb{N})[P_A(P_BP_A)^k play_B(p) \notin S] \\ 1 & \text{if } play_B(q) \in S \text{ and} \\ & (\exists k \in \mathbb{N})[P_A(P_BP_A)^k play_B(p) \in S] \\ 0 & \text{otherwise,} \end{cases}$$

where $(P_BP_A)^k$ is an abbreviation for $P_BP_A \cdots P_BP_A$ (k times). In other words, proposing yields a higher utility for Bob in the situation S if and only if *none* of the formulas in the infinite family $\{P_A(P_BP_A)^k play_B(p) : k \in \mathbb{N}\}$ occur in S.

As in Examples 1 and 2, and in general when a player desires to surprise an opponent, it is not difficult to convince oneself informally that this game admits no Nash equilibrium. Moreover, in this case the infinitary nature of Bob's desire to "surprise" Alice has an even stronger effect: no strategy for Bob is even *rationalizable* (see Section 4.3). □

Example 4. Pay raise. Bob has been voted employee of the month at his summer job, an honour that comes with a slight increase (up to $1) in his per-hour salary, at the discretion of his boss, Alice. Bob's happiness is determined in part by the raw value of the bump he receieves in his wages, and in part by the sense of gain or loss he feels by comparing the increase Alice grants him with the minimum increase he expected to get. Alice, for her part, wants Bob to be happy, but this desire is balanced by a desire to save company money.

As usual, we first fix a game form that captures the players and their available strategies. Let $\Gamma = (\{A, B\}, \Sigma_A, \{\cdot\})$, where $\Sigma_A = \{s_0, s_1, \ldots, s_{100}\}$ and s_k represents an increase of k cents to Bob's per-hour salary (Bob has no choice to make, so his strategy set is a singleton). Notice that, in contrast to the other examples we have seen thus far, in this game Bob's preferences depend on his *own* beliefs rather than the beliefs of his opponent. Broadly speaking, this is an example of *reference-dependent preferences*: Bob's utility is determined in part by comparing the actual outcome of the game to some "reference level"—in this case, the minimum expected raise. This game also has much in common with a

scenario described in [3], in which a player Abi wishes to tip her taxi driver exactly as much as he expects to be tipped, but no more.

Define $u_B : \mathcal{S} \to \mathbb{R}$ by

$$u_B(S) = k + (k - r),$$

where k is the unique integer such that $play_A(s_k) \in S$, and

$$r \coloneqq \min\{r' \ : \ P_B\, play_A(s_{r'}) \in S\}.$$

Observe that r is completely determined by Bob's beliefs: it is the lowest raise he considers it possible that Alice will grant him. We think of the first summand k as representing Bob's happiness on account of receiving a raise of k cents per hour, while the second summand $k - r$ represents his sense of gain or loss depending on how reality compares to his lowest expectations.

Note that the value of r (and k) is encoded in S via a finite formula, so we could have written the definition of u_B in a fully expanded form where each utility value is specified by the presense of a formula in S. For instance, the combination $k = 5$, $r = 2$ corresponds to the formula

$$play_A(s_5) \wedge P_B\, play_A(s_2) \wedge \neg(P_B\, play_A(s_0) \vee P_B\, play_A(s_1)),$$

which therefore determines a utility of 8.

Of course, it is just as easy to replace the minimum with the maximum in the above definition (perhaps Bob feels entitled to the most he considers it possible he might get), or even to define the reference level r as some more complicated function of Bob's beliefs. The quantity $k - r$ representing Bob's sense of gain or loss is also easy to manipulate. For instance, given $\alpha, \beta \in \mathbb{R}$ we might define a function $f : \mathbb{R} \to \mathbb{R}$ by

$$f(x) = \begin{cases} \alpha x & \text{if } x \geq 0 \\ \beta x & \text{if } x < 0, \end{cases}$$

and set

$$u'_B(S) = k + f(k - r),$$

where k and r are determined as above. Choosing, say, $\alpha = 1$ and $\beta > 1$ results in Bob's utility u'_B incorporating *loss aversion*: Bob is more upset by a relative loss than he is elated by a same-sized relative gain. These kinds of issues are discussed in [7]; in the full paper we analyze a central example from this paper in detail.

Turning now to Alice's preferences, we are faced with a host of modeling choices. Perhaps Alice wishes to grant Bob the smallest salary increase he expects but nothing more. We can capture this by defining $u_A : \mathcal{S} \to \mathbb{R}$ by

$$u_A(S) = -|k - r|,$$

where k and r are as above. Or perhaps we wish to represent Alice as feeling some fixed sense of guilt if she undershoots, while her disutility for overshooting depends on whether she merely exceeded Bob's lowest expectations, or in fact exceeded even his highest expectations:

$$u'_A(S) = \begin{cases} -25 & \text{if } k < r \\ r - k & \text{if } r \leq k < R \\ r - R + 2(R - k) & \text{if } k \geq R, \end{cases}$$

where

$$R \coloneqq \max\{R' \ : \ P_B\, play_A(s_{R'}) \in S\}.$$

Or perhaps Alice's model of Bob's happiness is sophisticated enough to *include* his sensations of gain and loss, so that, for example,

$$u''_A(S) = u_B(S) - \delta k,$$

where δ is some scaling factor. Clearly the framework is rich enough to represent many possibilities. □

Example 5. Preparing for a roadtrip. Alice has two tasks to accomplish before embarking on a cross-country roadtrip: she needs to buy a suitcase, and she needs to buy a car.

Here we sketch a simple decision-theoretic scenario in a language-based framework. We choose the underlying language in such a way as to capture two well-known "irrationalities" of consumers. First, consumers often evaluate prices in a discontinuous way, behaving, for instance, as if the difference between \$299 and \$300 is more substantive than the difference between \$300 and \$301. Second, consumers who are willing to put themselves out (for example, drive an extra 5 kilometers) to save \$50 on a \$300 purchase are often not willing to drive that same extra distance to save the same amount of money on a \$20,000 purchase.

We do not claim a completely novel analysis; rather, we aim to show how naturally a language-based approach can account for these kinds of issues.

Both of the irrationalities described above can be captured by assuming a certain kind of coarseness, specifically, that the language over which Alice forms preferences does not describe prices with infinite precision. For example, we might assume that the language includes as primitive propositions terms of the form p_Q, where Q ranges over a given partition of the real line. We might further suppose that this partition has the form

$$\cdots \cup [280, 290) \cup [290, 300) \cup [300, 310) \cup \cdots,$$

at least around the \$300 mark. Any utility function defined over such a language cannot distinguish prices that fall into the same partition. Thus, in the example above, Alice would consider the prices \$300 and \$301 to be effectively the same as far as her preferences are concerned. At the borderline between cells of the partition, however, there is the potential for a "jump": we might reasonably model Alice as prefering a situation where $p_{[290,300)}$ holds to one where $p_{[300,310)}$ holds. A smart retailer, therefore, should set their price to be at the higher end of a cell of the consumers' partition.

To capture the second irrationality discussed above, it suffices to assume that the partition that determines the underlying language is not only coarse, but is coarser for higher prices. For example, around the \$20,000 mark, we might suppose that the partition has the form

$$\cdots \cup [19000, 19500) \cup [19500, 20000) \cup [20000, 20500) \cup \cdots.$$

In this case, while Alice may prefer a price of \$300 to a price of \$350, she cannot prefer a price of \$20,000 to a price of \$20,050, because that difference cannot be described in the underlying language. This has a certain intuitive appeal: the higher numbers get (or, more generally, the further removed something is, in space or time or abstraction), the more you "ballpark" it—the less precise your language is in describing it. Indeed, psychological experiments have demonstrated that Weber's law[2], traditionally applied to physical stimuli,

[2]Weber's law asserts that the minimum difference between two stimuli necessary for a subject to discriminate between them increases as the magnitude of the stimuli increases.

finds purchase in the realm of numerical perception: larger numbers are subjectively harder to discriminate from one another [8; 11]. Our choice of underlying language represents this phenomenon simply, while exhibiting its explanatory power. □

Example 6. Returning a library book. Alice has learned that a book she borrowed from the library is due back tomorrow. As long as she returns it by tomorrow, she'll avoid a late fee; returning it today, however, is mildly inconvenient.

Here we make use of an extremely simple example to illustrate how to model an ostensibly dynamic scenario in a normal-form framework by employing a suitable underlying language. The idea is straightforward: Alice has a choice to make *today*, but how she feels about it depends on what she might do tomorrow. Specifically, if she returns the library book tomorrow, then she has no reason to feel bad about not returning it today. Since the future has yet to be determined, we model Alice's preferences as depending on what action she takes in the present together with what she *expects* to do in the future.

Let $\Gamma = (A, \{\mathsf{return}, \mathsf{wait}\})$ be a game form representing Alice's two current options, and set $\Phi'_\Gamma := \Phi_\Gamma \cup \{\mathsf{tomorrow}\}$; thus Φ'_Γ is the usual set of primitive propositions (representing strategies) together with a single new addition, tomorrow, read "Alice will return the book tomorrow".

An $\mathcal{L}_B(\Phi'_\Gamma)$-game allows us to specify Alice's utility in a manner consistent with the intuition given above. In particular, we can define $u_A : \mathcal{S}(\mathcal{L}_B(\Phi'_\Gamma)) \to \mathbb{R}$ by

$$u_A(S) = \begin{cases} -1 & \text{if } play_A(\mathsf{return}) \in S \\ 1 & \text{if } play_A(\mathsf{wait}) \wedge B_A\mathsf{tomorrow} \in S \\ -5 & \text{otherwise,} \end{cases}$$

so Alice prefers to wait if she expects to return the book tomorrow, and to return the book today otherwise.

In this example, Alice's utility depends on her beliefs, as it does in psychological game theory. Unlike psychological game theory, however, her utility depends on her beliefs about features of the world aside from which strategies are being played. This is a natural extension of the psychological framework in a language-based setting.

This example also hints at another interesting application of language-based games. A careful look at the language $\mathcal{L}_B(\Phi'_\Gamma)$ reveals an oddity: as far as the semantics are concerned, $play_A(\mathsf{return})$ and tomorrow are independent primitive propositions, despite being intuitively contradictory. Of course, this can be rectified easily enough: we can simply insist in the semantics that whenever $play_A(\mathsf{return})$ holds at a state, tomorrow does not. But in so doing, we have introduced a further complexity: the strategy that Alice chooses now determines more about the situation than merely the fact of which strategy she has chosen.

This observation reveals the need for a good theory of counterfactuals. After all, it is not just the true state of the world that must satisfy the semantic contraints we impose, but also the counterfactual situations we consider when determining whether or not a player is behaving rationally. In Section 4.1, we give a formal treatment of rationality in $\mathcal{L}_B(\Phi_\Gamma)$-games that skirts this issue; however, we believe that a more substantive treatment of counterfactual reasoning in games is both important and interesting, and that the present framework is a promising setting in which to develop such a theory.

Returning to the example at hand, we might emphasize the new element of "control" Alice has by providing her with explicit mechanisms of influencing her own beliefs about tomorrow. For example, perhaps a third strategy is available to her, remind, describing a state of affairs where she keeps the book but places it on top of her keys, thus decreasing the likelihood that she will forget to take it when she leaves the next day.

More generally, this simple framework allows us to model *commitment devices* [5]: we can represent players who rationally choose to perform certain actions (like buying a year-long gym membership, or throwing away their "fat jeans") not because these actions benefit them immediately, but because they make it subjectively more likely that the player will perform certain other desirable actions in the future (like going to the gym regularly, or sticking with a diet) that might otherwise be neglected. In a similar manner, we can succinctly capture *procrastination*: if, for example, you believe that you will quit smoking tomorrow, then the health benefits of quitting today instead might seem negligible—so negligible, in fact, that quitting immediately may seem pointless, even foolish. Of course, believing you will do something tomorrow is not the same thing as actually doing it when tomorrow comes, thus certain tasks may be delayed repeatedly. □

4. SOLUTION CONCEPTS

A number of important concepts from classical game theory, such as *Nash equilibrium* and *rationalizability*, have been completely characterized epistemically, using Γ-structures. In $\mathcal{L}_B(\Phi_\Gamma)$-games (or, more generally, in language-based games where the language includes belief), we can use the epistemic characterizations as the *definitions* of these solution concepts. This yields natural definitions that generalize those of classical game theory. We begin by defining rationality in our setting.

4.1 Rationality

We call a player i *rational* if he is best-responding to his beliefs: the strategy σ_i he is using must yield an expected utility that is at least as good as any other strategy σ'_i he could play, given his beliefs. In classical game theory, the meaning of this statement is quite clear. Player i has beliefs about the strategy profiles σ_{-i} used by the other players. This makes it easy to compute what i's payoffs would be if he were to use some other strategy σ'_i: since i's utility just depends on the strategy profile being used, we simply replace σ_i by σ'_i in these strategy profiles, and compute the new expected utility. Thus, for example, in a two-player game, if player 1 places probability 1/2 on the two strategies σ_2 and σ'_2 for player 2, then his expected utility playing σ_1 is $(u_1(\sigma_1, \sigma_2) + u_1(\sigma_1, \sigma'_2))/2$, while his expected utility if he were to play σ'_1 is $(u_1(\sigma'_1, \sigma_2) + u_1(\sigma'_1, \sigma'_2))/2$.

We make use of essentially the same approach in language-based games. Let $(\Gamma, (u_i)_{i\in N})$ be an $\mathcal{L}_B(\Phi_\Gamma)$-game and fix a Γ-structure $M = (\Omega, \vec{s}, \vec{Pr})$. Observe that for each $\omega \in \Omega$ and each $i \in N$, there is a unique $\mathcal{L}_B(\Phi_\Gamma)$-situation S such that $\omega \models S$; we denote this situation by $S(M, \omega)$ or just $S(\omega)$ when the Γ-structure is clear from context.

If $play_i(\sigma_i) \in S(\omega)$, then given $\sigma'_i \in \Sigma_i$ we might naïvely let $S(\omega/\sigma'_i)$ denote the set $S(\omega)$ with the formula $play_i(\sigma_i)$ replaced by $play_i(\sigma'_i)$, and define $\hat{u}_i(\sigma'_i, \omega)$, the utility that i would get if he played σ'_i in state ω, as $u_i(S(\omega/\sigma'_i))$. Un-

fortunately, u_i is not necessarily defined on $S(\omega/\sigma_i')$, since it is not the case in general that this set is satisfiable; indeed, $S(\omega/\sigma_i')$ is satisfiable if and only if $\sigma_i' = \sigma_i$. This is because other formulas in $S(\omega)$, for example the formula $B_i\, play_i(\sigma_i)$, logically imply the formula $play_i(\sigma_i)$ that was removed from $S(\omega)$ (recall that our semantics insist that every player is sure of their own strategy). With a more careful construction of the "counterfactual" set $S(\omega/\sigma_i')$, however, we can obtain a definition of $\hat{u}_i$ that makes sense.

A formula $\varphi \in \mathcal{L}_B(\Phi_\Gamma)$ is called **i-independent** if for each $\sigma_i \in \Sigma_i$, every occurrence of $play_i(\sigma_i)$ in φ falls within the scope of some B_j, $j \neq i$. Intuitively, an i-independent formula describes a proposition that is independent of player i's choice of strategy, such as another player's strategy, another player's beliefs, or even player i's beliefs about the other players; on the other hand, player i's beliefs about his own choices are excluded from this list, as they are assumed to always be accurate, and thus dependent on those choices. Given $S \in \mathcal{S}$, set

$$\rho_{-i}(S) = \{\varphi \in S \,:\, \varphi \text{ is } i\text{-independent}\}.^3$$

Let $\mathcal{S}_{-i}$ denote the image of $\mathcal{S}$ under ρ_{-i}. Elements of $\mathcal{S}_{-i}$ are called **i-situations**; intuitively, they are complete descriptions of states of affairs that are out of player i's control. Informally, an i-situation $S_{-i} \in \mathcal{S}_{-i}$ determines everything about the world (expressible in the language) *except* what strategy player i is employing. This is made precise in Proposition 1. Recall that $\rho_i(S)$ denotes the (unique) strategy that i plays in S, so $play_i(\rho_i(S)) \in S$.

PROPOSITION 1. *For each $i \in N$, the map $\vec{\rho}_i : \mathcal{S} \to \Sigma_i \times \mathcal{S}_{-i}$ defined by $\vec{\rho}_i(S) = (\rho_i(S), \rho_{-i}(S))$ is a bijection.*

This identification of $\mathcal{S}$ with the set of pairs $\Sigma_i \times \mathcal{S}_{-i}$ provides a well-defined notion of what it means to alter player i's strategy in a situation S "without changing anything else". By an abuse of notation, we write $u_i(\sigma_i, S_{-i})$ to denote $u_i(S)$ where S is the unique situation corresponding to the pair (σ_i, S_{-i}), that is, $\vec{\rho}_i(S) = (\sigma_i, S_{-i})$. Observe that for each state $\omega \in \Omega$ and each $i \in N$ there is a unique set $S_{-i} \in \mathcal{S}_{-i}$ such that $\omega \models S_{-i}$. We denote this set by $S_{-i}(M, \omega)$, or just $S_{-i}(\omega)$ when the Γ-structure is clear from context. Then the utility functions u_i induce functions $\hat{u}_i : \Sigma_i \times \Omega \to \mathbb{R}$ defined by

$$\hat{u}_i(\sigma_i, \omega) = u_i(\sigma_i, S_{-i}(\omega)).$$

As in the classical case, we can view the quantity $\hat{u}_i(\sigma_i, \omega)$ as the utility that player i would have if he were to play σ_i at state ω. It is easy to see that this generalizes the classical approach in the sense that it agrees with the classical definition when the utility functions u_i depend only on the outcome.

For each $i \in N$, let $EU_i : \Sigma_i \times \Omega \to \mathbb{R}$ be the expected utility of playing σ_i according to player i's beliefs at ω. Formally:

$$EU_i(\sigma_i, \omega) = \int_\Omega \hat{u}_i(\sigma_i, \omega')\, dPr_i(\omega);$$

when Ω is finite, this reduces to

$$EU_i(\sigma_i, \omega) = \sum_{\omega' \in \Omega} \hat{u}_i(\sigma_i, \omega') \cdot Pr_i(\omega)(\omega').$$

Define $BR_i : \Omega \to 2^{\Sigma_i}$ by

$$BR_i(\omega) = \{\sigma_i \in \Sigma_i \,:\, (\forall \sigma_i' \in \Sigma_i)[EU_i(\sigma_i, \omega) \geq EU_i(\sigma_i', \omega)]\};$$

thus $BR_i(\omega)$ is the set of *best-reponses* of player i to his beliefs at ω, that is, the set of strategies that maximize his expected utility.

With this apparatus in place, we can expand the underlying language to incorporate *rationality* as a formal primitive. Let

$$\Phi_\Gamma^{rat} \coloneqq \Phi_\Gamma \cup \{RAT_i \,:\, i \in N\},$$

where we read RAT_i as "player i is rational". We also employ the syntactic abbreviation $RAT \equiv RAT_1 \wedge \cdots \wedge RAT_n$. Intuitively, $\mathcal{L}_B(\Phi_\Gamma^{rat})$ allows us to reason about whether or not players are being rational with respect to their beliefs and preferences.

We wish to interpret rationality as expected utility maximization. To this end, we extend the valuation function $[\![\cdot]\!]_M$ to $\mathcal{L}_B(\Phi_\Gamma^{rat})$ by

$$[\![RAT_i]\!]_M \quad \coloneqq \quad \{\omega \in \Omega \,:\, s_i(\omega) \in BR_i(\omega)\}.$$

Thus RAT_i holds at state ω just in case the strategy that player i is playing at that state, $s_i(\omega)$, is a best-response to his beliefs.

4.2 Nash equilibrium

Having formalized rationality, we are in a position to draw on work that characterizes solutions concepts in terms of RAT.

Let $\Gamma = (N, (\Sigma_i)_{i \in N})$ be a game form in which each set Σ_i is finite, and let $\Delta(\Sigma_i)$ denote the set of all probability measures on Σ_i. Elements of $\Delta(\Sigma_i)$ are the **mixed strategies** of player i. Given a *mixed strategy profile*

$$\mu = (\mu_1, \ldots, \mu_n) \in \Delta(\Sigma_1) \times \cdots \times \Delta(\Sigma_n),$$

we define a Γ-structure M_μ that, in a sense made precise below, captures "equilibrium play" of μ and can be used to determine whether or not μ constitutes a Nash equilibrium.

Set

$$\Omega_\mu = supp(\mu_1) \times \cdots \times supp(\mu_n) \subseteq \Sigma_1 \times \cdots \times \Sigma_n.$$

Define a probability measure π on Ω_μ by

$$\pi(\sigma_1, \ldots, \sigma_n) = \prod_{i=1}^{n} \mu_i(\sigma_i),$$

and for each $\sigma, \sigma' \in \Omega_\mu$, let

$$Pr_{\mu,i}(\sigma)(\sigma') = \begin{cases} \pi(\sigma')/\mu_i(\sigma_i) & \text{if } \sigma_i = \sigma_i' \\ 0 & \text{otherwise.} \end{cases}$$

Let $M_\mu = (\Omega_\mu, id_{\Omega_\mu}, \vec{Pr}_\mu)$. It is easy to check that M_μ is a Γ-structure; call it the **characteristic Γ-structure for μ**. At each state in M_μ, each player i is sure of his own strategy and has uncertainty about the strategies of his opponents; however, this uncertainty takes the form of a probability distribution weighted according to μ_{-i}, so in effect each player i correctly ascribes the mixed strategy μ_j to each of his opponents $j \neq i$. It is well known (and easy to show) that a

[3] As (quite correctly) pointed out by an anonymous reviewer, this notation is not standard, since ρ_{-i} is not a profile of functions of the type ρ_i. Nonetheless, we feel it is appropriate in the sense that, while ρ_i extracts from a given situation player i's strategy, ρ_{-i} extracts "all the rest" (cf. Proposition 1), the crucial difference here being that this includes far more than just the strategies of the other players.

mixed strategy profile μ is a Nash equilibrium in the classical sense if and only if each player is rational (i.e. maximizing expected utility) at every state in the characteristic Γ-structure for μ. Accordingly, we *define* a **Nash equilibrium** (in an $\mathcal{L}_B(\Phi_\Gamma)$-game) to be a mixed strategy profile μ such that $M_\mu \models RAT$. It is immediate that this definition generalizes the classical definition of Nash equilibrium.

We note that there are several other epistemic characterizations of Nash equilibrium besides the one presented here. While in the classical setting they all generate equivalent solution concepts, this need not be true in our more general model. We believe that investigating the solution concepts that arise by teasing apart these classically equivalent notions is an interesting and promising direction for future research.

In contrast to the classical setting, Nash equilibria are not guaranteed to exist in general; indeed, this is the case for the indignant altruism game of Example 2.

PROPOSITION 2. *There is no Nash equilibrium in the indignant altruism game.*

PROOF. We must show that for every mixed strategy profile

$$\mu = (\mu_A, \mu_B) \in \Delta(\{\mathsf{c}, \mathsf{d}\}) \times \Delta(\{\mathsf{c}, \mathsf{d}\}),$$

the corresponding characteristic Γ-structure $M_\mu \not\models RAT$.

Suppose first that $\mu_A(\mathsf{c}) > 0$. Then $M_\mu \models \neg B_B\, play_A(\mathsf{d})$, which implies that Alice's utility at every state in M_μ coincides with the classical prisoner's dilemma, so she is not rational at any state where she cooperates. Since, by definition, M_μ contains a state where Alice cooperates, we conclude that $M_\mu \not\models RAT_A$, so μ cannot be a Nash equilibrium.

Suppose instead that $\mu_A(\mathsf{c}) = 0$. Then $M_\mu \models B_B\, play_A(\mathsf{d})$, and so Alice, being sure of this, is not rational at any state where she defects, since by definition she is guaranteed a utility of -1 in that case. By definition, M_μ contains a state where Alice defects (in fact, Alice defects in every state), so we can conclude as above that $M_\mu \not\models RAT_A$, which means that μ cannot be a Nash equilibrium. □

What went wrong here? Roughly speaking, the utility functions in this game exhibit a kind of "discontinuity": the utility of defecting is -1 precisely when your opponent is 100% certain that you will defect. However, as soon as this probability dips below 100%, *no matter how small the drop*, the utility of defecting jumps up to at least 1.

Broadly, this issue arises in $\mathcal{L}$-games whenever $\mathcal{L}$ is limited to a coarse-grained notion of belief, such as the underlying language in this example, which only contains belief modalities representing 100% certainty. However, since coarseness is a central feature we wish to model, the lack of existence of Nash equilibria in general might be viewed as a problem with the notion of *Nash equilibrium* itself, rather than a defect of the underlying language. Indeed, the requirements that a mixed strategy profile must satisfy in order to qualify as a Nash equilibrium are quite stringent: essentially, each player must evaluate his choice of strategy *subject to the condition that his choice is common knowledge*! As we have seen, this condition is not compatible with rationality when a player's preference is to do something unexpected.

More generally, this tension arises with any solution concept that requires players to have common knowledge of the mixed strategies being played (the "conjectures", in the terminology of [2]). In fact, Proposition 2 relies only on second-order knowledge of the strategies: whenever Alice knows that Bob knows her play, she is unhappy. In particular, any alternative epistemic characterization of Nash equilibrium that requires such knowledge is subject to the same nonexistence result. Furthermore, we can use the same ideas to show that there is no *correlated equilibrium* [1] in the indignant altruism game either (once we extend correlated equilibrium to our setting).

4.3 Rationalizability

In this section, we define rationalizability in language-based games in the same spirit as we defined Nash equilibrium in Section 4.2. As shown by Tan and Werlang [12] and Brandenburger and Dekel [4], common belief of rationality characterizes rationalizable strategies. Thus, we define rationalizability that way here.

Let $\mathcal{L}_{CB}(\Phi_\Gamma^{rat})$ be the language obtained by starting with the primitive propositions in Φ_Γ^{rat} and closing off under conjunction, negation, the modal operators B_i, for $i \in N$, and the modal operator CB. We read $CB\varphi$ as "there is common belief of φ". Extend $[\![\cdot]\!]_M$ to $\mathcal{L}_{CB}(\Phi_\Gamma^{rat})$ by setting

$$[\![CB\varphi]\!]_M \;\coloneqq\; \bigcap_{k=1}^{\infty} [\![EB^k\varphi]\!]_M,$$

where

$$\begin{aligned} EB\varphi &\equiv B_1\varphi \wedge \cdots \wedge B_n\varphi, \text{ and} \\ EB^k\varphi &\equiv EB(EB^{k-1}\varphi). \end{aligned}$$

For convenience, we stipulate that $EB^0\varphi \equiv \varphi$. We read $EB\varphi$ as "everyone believes φ". Thus, intuitively, $CB\varphi$ holds precisely when everyone believes φ, everyone believes that everyone believes φ, and so on. We define a strategy $\sigma_i \in \Sigma_i$ to be **rationalizable** (in an $\mathcal{L}_B(\Phi_\Gamma)$-game) if the formula $play_i(\sigma_i) \wedge CB(RAT)$ is satisfiable in some Γ-structure.

Although there are no Nash equilibria in the indignant altruism game, as we now show, every strategy is rationalizable.

PROPOSITION 3. *Every strategy in the indignant altruism game is rationalizable.*

PROOF. Consider the Γ-structure in Figure 1.

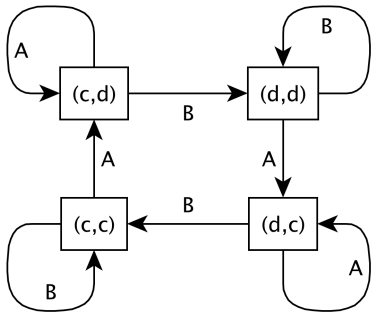


Figure 1: A Γ-structure for indignant altruism.

The valuations of the primitive propositions at each of the four states are labeled in the obvious way. Arrows labeled i based at state ω point to all and only those states in $Pr_i[\omega]$ (so every probability measure has exactly one state in its support).

As discussed in Example 2, it is rational to cooperate in this game if you believe that your opponent believes that you will defect, and it is rational to defect if you believe that your opponent believes you will cooperate. Given this, it is not difficult to check that RAT holds at each state of this Γ-structure, and therefore so does $CB(RAT)$. Thus, by definition, every strategy is rationalizable. □

Does every language-based game admit a rationalizable strategy? Every classical game does. This follows from the fact that every strategy in a Nash equilibrium is rationalizable, together with Nash's theorem that every (finite) game has a Nash equilibrium (cf. [10]). In the language-based setting, while it is immediate that every strategy in a Nash equilibrium is rationalizable, since Nash equilibria do not always exist, we cannot appeal to this argument. In fact, we have already seen an example of an $\mathcal{L}_B(\Phi_\Gamma)$-game that admits no rationalizable strategy.

PROPOSITION 4. *The deeply surprising proposal game has no rationalizable strategies.*

PROOF. Fix a Γ-structure $M = (\Omega, \vec{s}, \vec{Pr})$ and suppose for contradiction that $\omega \in \Omega$ is such that $\omega \models CB(RAT)$. Consider first the case where Alice does not *expect** a proposal at state ω, where "expect*" denotes the infinitary notion of expectation at play in this example: for all $k \geq 0$, $\omega \models \neg P_A(P_BP_A)^k play_B(p)$. Thus, for all $k \geq 0$, $\omega \models B_A(B_BB_A)^k \neg play_B(p)$; taking $k = 0$, it follows that for all $\omega' \in Pr_A[\omega]$, $\omega' \models \neg play_B(p)$. Moreover, since $CB(RAT)$ holds at ω, certainly $\omega' \models RAT_B$. But if Bob is rationally *not* proposing at ω', then he must at least consider it possible that Alice expects* a proposal: for some $k \in \mathbb{N}$, $\omega' \models P_BP_A(P_BP_A)^k play_B(p)$. But this implies that $\omega \models P_A(P_BP_A)^{k+1} play_B(p)$, contradicting our assumption. Thus, any state where $CB(RAT)$ holds is a state where Alice expects* a proposal.

So suppose that Alice expects* a proposal at ω. It follows that there is some state ω' satisfying $\omega' \models play_B(p) \wedge CB(RAT)$. But if Bob is rationally playing p at ω', there must be some state $\omega'' \in Pr_B[\omega']$ where Alice doesn't expect* it; however, we also know that $\omega'' \models CB(RAT)$, which we have seen is impossible.

This completes the argument: $CB(RAT)$ is not satisfiable. It is worth noting that this argument fails if we replace "expects*" with "expects$^{\leq K}$", where this latter term is interpreted to mean

$$(\forall k \leq K)[\neg P_A(P_BP_A)^k play_B(p)].$$

□

In the full paper, we provide a condition that guarantees the existence of rationalizable strategies in $\mathcal{L}_B(\Phi_\Gamma)$-games. The essential ingredient is a kind of compactness assumption on the language $\mathcal{L}_B(\Phi_\Gamma^{rat})$. Roughly speaking, we require that no player can fail to be rational for an "infinitary" reason. All finitely-specified $\mathcal{L}_B(\Phi_\Gamma)$-games turn out to satisfy this condition, so we obtain the following:

THEOREM 1. *Every finitely-specified $\mathcal{L}_B(\Phi_\Gamma)$-game has a rationalizable strategy.*

Since we expect to encounter finitely-specified games most often in practice, this suggests that the games we are likely to encounter will indeed have rationalizable strategies.

5. ACKNOWLEDGEMENTS

Bjorndahl is supported in part by NSF grants IIS-0812045, CCF-1214844, DMS-0852811, and DMS-1161175, and ARO grant W911NF-09-1-0281. Halpern is supported in part by NSF grants IIS-0812045, IIS-0911036, and CCF-1214844, by AFOSR grant FA9550-08-1-0266, and by ARO grant W911NF-09-1-0281. Pass is supported in part by an Alfred P. Sloan Fellowship, a Microsoft Research Faculty Fellowship, NSF Awards CNS-1217821 and CCF-1214844, NSF CAREER Award CCF-0746990, AFOSR YIP Award FA9550-10-1-0093, and DARPA and AFRL under contract FA8750-11-2-0211. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the US Government.

References

[1] R. J. Aumann. Correlated equilibrium as an expression of Bayesian rationality. *Econometrica*, 55:1–18, 1987.

[2] R. J. Aumann and A. Brandenburger. Epistemic conditions for Nash equilibrium. *Econometrica*, 63(5):1161–1180, 1995.

[3] P. Battigalli and M. Dufwenberg. Dynamic psychological games. *Journal of Economic Theory*, 144:1–35, 2009.

[4] A. Brandenburger and E. Dekel. Rationalizability and correlated equilibria. *Econometrica*, 55:1391–1402, 1987.

[5] I. Brocas, J. D. Carrillo, and M. Dwatripont. Commitment devices under self-control problems: an overview. In I. Brocas and J. D. Carrillo, editors, *The Psychology of Economic Decisions: Volume II: Reasons and Choices*, pages 49–67. Oxford University Press, Oxford, UK, 2004.

[6] J. Geanakoplos, D. Pearce, and E. Stacchetti. Psychological games and sequential rationality. *Games and Economic Behavior*, 1(1):60–80, 1989.

[7] B. Kőszegi and M. Rabin. A model of reference-dependent preferences. *The Quarterly Journal of Economics*, CXXI:1133–1165, 2006.

[8] R. S. Moyer and T. K. Landauer. Time required for judgements of numerical inequality. *Nature*, 215:1519–1520, 1967.

[9] S. Mullainathan. Thinking through categories. Unpublished manuscript, available at www.haas.berkeley.edu/groups/finance/cat3.pdf, 2002.

[10] M. J. Osborne and A. Rubinstein. *A Course in Game Theory*. MIT Press, Cambridge, Mass., 1994.

[11] F. Restle. Speed of adding and comparing numbers. *Journal of Experimental Psychology*, 83:274–278, 1978.

[12] T. Tan and S. Werlang. The Bayesian foundation of solution concepts of games. *Journal of Economic Theory*, 45(45):370–391, 1988.

Defeasible Modalities

Katarina Britz
Centre for Artificial Intelligence Research
CSIR Meraka Institute and UKZN, South Africa
arina.britz@meraka.org.za

Ivan Varzinczak
Centre for Artificial Intelligence Research
CSIR Meraka Institute and UKZN, South Africa
ivan.varzinczak@meraka.org.za

ABSTRACT

Nonmonotonic logics are usually characterized by the presence of some notion of 'conditional' that fails monotonicity. Research on nonmonotonic logics is therefore largely concerned with the defeasibility of argument forms and the associated normality (or abnormality) of its constituents. In contrast, defeasible *modes of inference* aim to formalize the defeasible aspects of modal notions such as actions, obligations and knowledge. In this work we enrich the standard possible worlds semantics with a preference ordering on worlds in Kripke models. The resulting family of modal logics allow for the elegant expression of defeasible modalities. We also propose a tableau calculus which is sound and complete with respect to our preferential semantics.

Keywords

Knowledge representation and reasoning; modal logic; preferential semantics; defeasible modes of inference

1. INTRODUCTION AND MOTIVATION

Defeasible reasoning, as traditionally studied in the literature on nonmonotonic reasoning, has focused mostly on one aspect of defeasibility, namely that of *argument forms*. Such is the case in the approach by Kraus et al. [33, 35], known as the KLM approach, and related frameworks [5, 6, 7, 8, 10, 15, 20, 21]. For instance, in the KLM approach (propositional) defeasible consequence relations $\mid\!\sim$ with a preferential semantics are studied. In this setting, the meaning of a defeasible statement (or a 'conditional', as it is sometimes referred to) of the form $\alpha \mid\!\sim \beta$ is that "all normal α-worlds are β-worlds", leaving it open for α-worlds that are, in a sense, exceptional not to satisfy β. With the theory that has been developed around this notion it becomes possible to cope with exceptionality when performing reasoning.

There are of course many other appealing and equally useful aspects of defeasibility besides that of arguments. These include notions such as typicality [4, 21], concerned with the most typical cases or situations (or even the most typical representatives of a class), and belief plausibility [2], which relates to the most plausible epistemic possibilities held by an agent, amongst others. It turns out that with KLM-style defeasible statements one cannot capture these aspects of defeasibility. This has to do partly with the syntactic restrictions imposed on $\mid\!\sim$, namely no nesting of conditionals, but, more fundamentally, it relates to where and how the notion of normality is used in such statements. Indeed, in a KLM defeasible statement $\alpha \mid\!\sim \beta$, the normality spotlight is somewhat put on α, as though normality was a property of the premise and not of the conclusion. Whether the situations in which β holds are normal or not plays no role in the reasoning that is carried out. In the original KLM framework, normality is also linked to the premise as a whole, rather than its constituents. Technically this meant one could not refer directly to normality of a sentence in the scope of logical operators. This limitation is overcome by taking a (modal) conditional approach *à la* Boutilier [5] — the resulting conditional logics are sufficiently general to allow for the expression of a number of different forms of defeasible reasoning. However, the emphasis remains on the defeasibility of arguments, or of conditionals.

In this paper we investigate a related, but incomparable, notion which we refer to as defeasible *modes of inference* [11].[1] These amount to defeasible versions of the traditional notions of actions, obligations, knowledge and beliefs, to name a few, as studied in modal logics. For instance, in an action context, one can say that normally the outcome of a given action a is α . However we may also want to state that the outcome of a is usually (or normally) α, which is different from the former statement. To see why, the first statement says that in the most normal worlds, the result of performing the action a is *always* α, whereas in the second one it is in the most normal situations resulting from a's execution that α holds — regardless of whether the situation in which the claim is uttered is normal or not.

For a concrete example, assume one arrives at a dark room and wants to toggle the light switch. Exceptionally, the light will not turn on. This can be either because the light bulb is blown (the current situation is abnormal) or because an overcharge was caused while switching the light (the action behaves abnormally). In the former case, the normality of the situation, or state, before the action is assessed, whereas in the latter the relative normality of the situation is assessed against all possible outcomes. Here we are interested in the formalization of the latter type of statement, where it becomes important to shift the notion of normality from the premise of an inference to the effect of an action, and, importantly, use it in the scope of other logical constructors.

Our next example concerns obligations and weaker versions thereof. There is a subtle difference between stating

[1]The present paper extends and refines the preliminary proposal which was presented at the 14th International Workshop on Non-Monotonic Reasoning (NMR).

TARK 2013, Chennai, India.

that, from the perspective of any normal situation, rhino poaching ought to carry a minimum prison sentence of 10 years, and stating that, from any perspective, the minimum sentence for rhino poaching normally ought to be 10 years. The shift in focus is again from normality of the present world, to relative normality amongst possible worlds. In the former statement, an abnormal present world would render the obligation unenforceable, whereas in the latter statement, the obligation is applicable in all relatively normal accessible worlds. We contend that the informal notion of 'normal, reasonable obligations' is more accurately modeled as defeasible modalities than as conditional statements.

Scenarios such as the ones depicted above require an ability to talk about the normality of effects of an action, relative normality of obligations, and so on. While existing modal treatments of preferential reasoning can express preferential semantics syntactically as modalities [5, 6, 20], they do not suffice to express defeasible modes of inference as described above. The ability to capture precisely these forms of defeasibility remains a fundamental challenge in the definition of a coherent theory of defeasible reasoning. At present we can formalize only the first type of statements above by, e.g. stating $\top \mathrel{|\!\sim} \Box\alpha$ in Britz et al.'s extension of preferential reasoning to modal languages [8, 10]. (As we shall see later in the paper, both Boutilier's [5] and Booth et al.'s [4] approaches also have to be enriched in order to capture the forms of defeasibility we are interested in here.)

In this paper we make the first steps towards filling this gap by introducing (non-standard) modal operators allowing us to talk about relative normality in accessible worlds. With our defeasible versions of modalities, we can make statements of the form "α holds in all of the relatively normal accessible worlds", thereby capturing defeasibility of what is 'expected' in target worlds. This notion of defeasibility in a modality meets a variety of applications in Artificial Intelligence, ranging from reasoning about actions to deontic and epistemic reasoning. For instance, a defeasible-action operator allows us to make statements of the form $\boxtimes_a\alpha$, which we read as "α is a normal necessary effect of a" (i.e., necessary in the most normal of a's outcomes), and with defeasible-obligation operators one can state formulae such as $\boxtimes_{\mathsf{A}}\alpha$, read as "$\alpha$ is a normal obligation of agent A".

These operators are defined within the context of a general preferential modal semantics obtained by enriching the standard possible worlds semantics with a preference order. The main difference between the approach we propose here and that of Boutilier [5] is in whether the underlying preference ordering alters the meaning of modalities or not. Boutilier's conditional is defined directly from a preference order in a bi-modal language, but the meanings of any additional, independently axiomatized, modalities are not influenced by the preference order. Our defeasible modalities correspond to a modification of the other modalities using the preference relation. Also, in contrast with the plausibility models of Baltag and Smets [2], the preference order we consider here does not define an agent's knowledge or beliefs. Rather, it is part of the semantics of the background ontology described by the theory or knowledge base at hand. As such, it informs the meaning of defeasible actions, which can fail in their outcome, or defeasible obligations, which may not hold in exceptional accessible worlds, in that it alters the classical semantics of these modalities. This allows for the definition of a family of modal logics in which defeasible modes of inference can be expressed, and which can be integrated with existing $\mathrel{|\!\sim}$-based nonmonotonic modal logics [8, 10].

The remainder of the present paper is structured as follows: After setting up the notation and terminology that we shall follow in this paper (Section 2), we revisit Britz et al.'s preferential semantics for modal logic [8, 10] (Section 3) by proposing a simplified version thereof. In Section 4 we present a logic enriched with defeasible modalities allowing for the formalization of defeasible versions of modes of inference. In Section 5 we present a detailed example illustrating the application of our constructions in an action context. Following that, we define a tableau system for the corresponding logic that we show to be sound and complete with respect to our preferential semantics (Section 6). In Section 7 we assess $\mathrel{|\!\sim}$-statements in our richer language. After a discussion of and comparison with related work (Section 8), we conclude with some comments and directions for further investigation. All the proofs of our results can be found in the Appendix.

2. MODAL LOGIC

We assume the reader is familiar with modal logic [14]. The purpose of this section is to make explicit the terminology and notation we shall use.

Here we work within a set of *atomic propositions* $\mathcal{P}$, using the logical connectives $\wedge$ (conjunction), $\neg$ (negation), and a set of modal operators $\Box_i$, $1 \leq i \leq n$. (In later sections we shall adopt a richer language.) We assume that the underlying multimodal logic is independently axiomatized (i.e., the logic is a fusion and there is no interaction between the modal operators [32]). Propositions are denoted by $p, q, \ldots$, and formulae by $\alpha, \beta, \ldots$, constructed in the usual way according to the rule: $\alpha ::= p \mid \neg\alpha \mid \alpha \wedge \alpha \mid \Box_i\alpha$. All the other truth functional connectives ($\vee$, $\rightarrow$, $\leftrightarrow$, ...) are defined in terms of $\neg$ and $\wedge$ in the usual way. Given $\Box_i$, $1 \leq i \leq n$, with $\Diamond_i$ we denote its *dual* modal operator, i.e., for any α, $\Diamond_i\alpha \equiv_{\text{def}} \neg\Box_i\neg\alpha$. We use $\top$ as an abbreviation for $p \vee \neg p$, and $\bot$ as an abbreviation for $p \wedge \neg p$, for some $p \in \mathcal{P}$.

With $\mathcal{L}$ we denote the set of all formulae of the modal language. The semantics is the standard possible-worlds one:

DEFINITION 1. *A Kripke model is a tuple $\mathscr{M} = \langle W, R, V\rangle$ where W is a (non-empty) set of possible worlds, $R = \langle R_1, \ldots, R_n\rangle$, where each $R_i \subseteq W \times W$ is an accessibility relation on W, $1 \leq i \leq n$, and $V : W \times \mathcal{P} \longrightarrow \{0, 1\}$ is a valuation function.*

Satisfaction of formulae with respect to possible worlds in a Kripke model is defined in the usual way:

DEFINITION 2. *Let $\mathscr{M} = \langle W, R, V\rangle$ and $w \in W$:*

- *$\mathscr{M}, w \Vdash p$ if and only if $V(w, p) = 1$;*
- *$\mathscr{M}, w \Vdash \neg\alpha$ if and only if $\mathscr{M}, w \nVdash \alpha$;*
- *$\mathscr{M}, w \Vdash \alpha \wedge \beta$ if and only if $\mathscr{M}, w \Vdash \alpha$ and $\mathscr{M}, w \Vdash \beta$;*
- *$\mathscr{M}, w \Vdash \Box_i\alpha$ if and only if $\mathscr{M}, w' \Vdash \alpha$ for all w' such that $(w, w') \in R_i$.*

Given $\alpha \in \mathcal{L}$ and $\mathscr{M} = \langle W, R, V\rangle$, we say that $\mathscr{M}$ *satisfies* α if there is at least one world $w \in W$ such that $\mathscr{M}, w \Vdash \alpha$. We say that $\mathscr{M}$ is a *model of* α (alias α is *true* in $\mathscr{M}$), denoted $\mathscr{M} \Vdash \alpha$, if $\mathscr{M}, w \Vdash \alpha$ for every world $w \in W$. Given a class

of models $\mathcal{M}$, we say that α is *valid* in $\mathcal{M}$ if every Kripke model $\mathscr{M} \in \mathcal{M}$ is a model of α.

Here we shall assume the system of normal modal logic K, of which all the other normal modal logics are extensions. Semantically, K is characterized by the class of all Kripke models (Definition 1). We say that α *locally entails* β in the system K (denoted $\alpha \models \beta$) if for every K-model $\mathscr{M}$ and every w in $\mathscr{M}$, $\mathscr{M}, w \Vdash \alpha$ implies $\mathscr{M}, w \Vdash \beta$.

Syntactically, K corresponds to the smallest set of sentences containing all propositional tautologies, all instances of the axiom schema $K : \Box_i(\alpha \rightarrow \beta) \rightarrow (\Box_i\alpha \rightarrow \Box_i\beta)$, $1 \leq i \leq n$, and closed under the *rule of necessitation* $RN : \alpha/\Box_i\alpha$, $1 \leq i \leq n$.

3. MODAL PREFERENTIAL SEMANTICS

In this section we modify the constructions for preferential reasoning in modal logic as studied by Britz et al. [8, 10]. We do so by enriching standard Kripke models with preference relations, instead of placing an ordering on states which are labeled with *pointed* Kripke models. Our starting point is therefore similar to the CT4O models of Boutilier [5] and the plausibility models of Baltag and Smets [2].

DEFINITION 3. *A preferential Kripke model is a tuple* $\mathscr{P} := \langle W, R, V, \prec \rangle$ *where* W *is a (non-empty) set of possible worlds,* $R = \langle R_1, \ldots, R_n \rangle$*, where each* $R_i \subseteq W \times W$ *is an accessibility relation on* W*,* $1 \leq i \leq n$*,* $V : W \times \mathcal{P} \longrightarrow \{0, 1\}$ *is a valuation function, and* $\prec \subseteq W \times W$ *is a co-Noetherian strict partial order on* W*, i.e.,* $\prec$ *is irreflexive, transitive and well-founded.*[2]

Given a preferential Kripke model $\mathscr{P} = \langle W, R, V, \prec \rangle$, we refer to $\mathscr{M} := \langle W, R, V \rangle$ as its associated standard Kripke model. If $\mathscr{P} = \langle W, R, V, \prec \rangle$ is a preferential Kripke model and $\alpha \in \mathcal{L}$, then with $[\![\alpha]\!] := \{w \in W \mid \mathscr{M}, w \Vdash \alpha$, where $\mathscr{M} = \langle W, R, V \rangle\}$ we denote the set of possible worlds satisfying α (α-*worlds* for short).

DEFINITION 4. *Let* $\mathscr{P} = \langle W, R, V, \prec \rangle$ *and let* $W' \subseteq W$*. With* $\min_\prec W'$ *we denote the minimal elements of* W' *with respect to* $\prec$*, i.e.,* $\min_\prec W' := \{w \in W' \mid$ *there is no* $w' \in W'$ *such that* $w' \prec w\}$*.*

The intuition behind the preference relation $\prec$ in a preferential Kripke model $\mathscr{P}$ is that worlds lower down in the order are *more preferred* (or *more normal* [4, 5]) than those higher up. Note that the preference relation in a preferential Kripke model, although a binary relation on W, is not to be seen as an accessibility relation. Indeed, the $\prec$-component in a preferential Kripke model has no counterpart in the syntax as each accessibility relation has.

As an example, Figure 1 below depicts the preferential Kripke model $\mathscr{P} = \langle W, R, V, \prec \rangle$, where $W = \{w_i \mid 1 \leq i \leq 5\}$, $R = \langle R_\Box \rangle$, with $R_\Box = \{(w_1, w_2), (w_1, w_4), (w_2, w_3), (w_2, w_5), (w_3, w_2), (w_4, w_5), (w_5, w_4)\}$, represented by the solid arrows in the picture, V is the obvious valuation function (in our pictorial representations of models we interpret absence of an atom as the atom being false in the respective world), and $\prec$ is the transitive closure of $\{(w_1, w_2), (w_1, w_3), (w_2, w_4), (w_3, w_4), (w_4, w_5)\}$, represented by the dashed arrows in the picture. (Note the direction of the dashed arrows, which point from less preferred to more preferred worlds.)

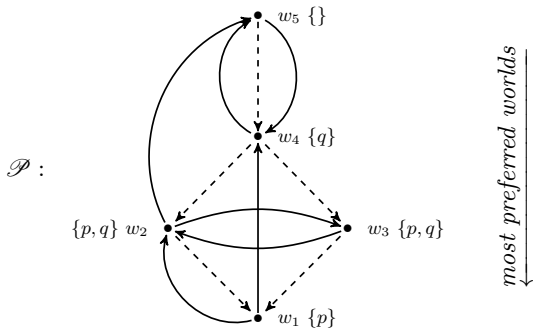


Figure 1: A preferential Kripke model for $\mathcal{P} = \{p, q\}$ and a single modality.

Given $\mathscr{P} = \langle W, R, V, \prec \rangle$ and $\alpha \in \mathcal{L}$, α is *satisfiable* in $\mathscr{P}$ if $[\![\alpha]\!] \neq \emptyset$, otherwise α is *unsatisfiable* in $\mathscr{P}$. We say that α is *true* in $\mathscr{P}$ (denoted $\mathscr{P} \Vdash \alpha$) if $[\![\alpha]\!] = W$. It is easy to see that the addition of the $\prec$-component preserves the truth of all (classical) modal formulae that are true in the remaining Kripke structure:

LEMMA 1. *Let* $\alpha \in \mathcal{L}$ *(i.e.,* α *is a classical modal formula). Let* $\mathscr{P} = \langle W, R, V, \prec \rangle$ *be a preferential Kripke model and* $\mathscr{M} = \langle W, R, V \rangle$ *its associated standard Kripke model. Then* $\mathscr{P} \Vdash \alpha$ *if and only if* $\mathscr{M} \Vdash \alpha$*.*

PROOF. See Appendix A.1. □

We can define *classes* of preferential Kripke models in the same way we do in the classical case. For instance, we can talk about the class of reflexive preferential Kripke models, in which the R-components are reflexive. We say that α is *valid* in the class $\mathcal{M}$ of preferential Kripke models if and only if α is true in every $\mathscr{P} \in \mathcal{M}$. Therefore, the following result is an immediate consequence of Lemma 1:

COROLLARY 1. *A modal formula* α *is valid in the class* $\mathcal{M}$ *of preferential Kripke models if and only if it is valid in the corresponding class of Kripke models.*

4. PREFERENCE-BASED MODALITIES

Recalling our discussion in the Introduction, we want to be able to state that a given sentence holds in *all* the relatively normal worlds that are accessible. This leads us to the definition of a 'weaker' version of the $\Box$ modalities. Through them we are then able to single out those normal situations that one cannot grasp via the classical $\Box$ modalities. Similarly, we want to be able to state that a given sentence holds in *at least one* relatively normal accessible world. This leads us to the definition of a stronger version of $\Diamond$, which may be read as *distinct* possibility.

We define a more expressive language than $\mathcal{L}$ by extending our modal language with a family of defeasible modal operators $\boxminus_i$ and $\diamondplus_i$, $1 \leq i \leq n$ (called, respectively, the 'flag' and the 'flame'), where n is the number of classical

[2]This implies the *smoothness condition* in Kraus et al.'s terminology [33], which basically says that $\prec$ has no infinitely descending chains. Even though well-foundedness is stronger than smoothness, here we prefer to stick to the term that is more broadly known outside the nonmonotonic reasoning circle.

modalities in the language. The formulae of the extended language are then recursively defined by:

$$\alpha \ ::= p \mid \neg\alpha \mid \alpha \wedge \alpha \mid \Box_i\alpha \mid \mathrel{|\!\sim}_i\alpha \mid \mathrel{|\!\sim}\!\!\Diamond_i\alpha$$

(As before, the other connectives are defined in terms of $\neg$ and $\wedge$ in the usual way, and $\top$ and $\bot$ are seen as abbreviations. It turns out that each $\mathrel{|\!\sim}\!\!\Diamond_i$ too is the dual of $\mathrel{|\!\sim}_i$, as we shall see below.) With $\widetilde{\mathcal{L}}$ we denote the set of all formulae of such a richer language.

The semantics of $\widetilde{\mathcal{L}}$ is in terms of our preferential Kripke models (see Definition 3). As before, given $\alpha \in \widetilde{\mathcal{L}}$ and a preferential Kripke model $\mathscr{P} = \langle W, R, V, \prec\rangle$, with $[\![\alpha]\!]$ we denote the set of elements of W satisfying α.

Definition 5. *Let $\mathscr{P} = \langle W, R, V, \prec\rangle$ be a preferential Kripke model. Then:*

- $[\![\mathrel{|\!\sim}_i\alpha]\!] := \{w \in W \mid \min_{\prec} R_i(w) \subseteq [\![\alpha]\!]\}$;
- $[\![\mathrel{|\!\sim}\!\!\Diamond_i\alpha]\!] := \{w \in W \mid \min_{\prec} R_i(w) \cap [\![\alpha]\!] \neq \emptyset\}$.

The intuition behind a sentence like $\mathrel{|\!\sim}_i\alpha$ is that α holds in the most 'normal' of R_i-accessible worlds. $\mathrel{|\!\sim}\!\!\Diamond_i\alpha$ intuitively says that α holds in at least one such relatively normal accessible world. To give a simple example (a more elaborated one is given in Section 5), if toggle denotes the action of toggling the light switch and light the proposition "the light is on", with the formula $\neg\mathsf{light} \rightarrow \mathrel{|\!\sim}_{\mathsf{toggle}}\mathsf{light}$ we formalize the example from the Introduction.

As mentioned before, in our enriched language the preference relation is not explicit in the syntax. The meaning of the new modalities is informed by the preference relation, which nevertheless remains tacit outside the realm of defeasible modalities. This stands in contrast to the approaches of Baltag and Smets [2], Boutilier [5], Britz et al. [6] and Giordano et al. [20], which cast the preference relation as an extra modality in the language. From a knowledge representation perspective, our approach has the advantage of hiding some complex aspects of the semantics from the user (e.g. a knowledge engineer who will write down sentences in an agent's knowledge base).

The notions of satisfaction in a preferential Kripke model, truth (in a model) and validity (in a class of preferential Kripke models) are extended to formulae with defeasible modalities in the obvious way.

We observe that, like in the classical (i.e., non-defeasible) case, the defeasible modal operators $\mathrel{|\!\sim}_i$ and $\mathrel{|\!\sim}\!\!\Diamond_i$ are the dual of each other:

$$\models \mathrel{|\!\sim}_i\alpha \leftrightarrow \neg\mathrel{|\!\sim}\!\!\Diamond_i\neg\alpha \tag{1}$$

The following validities are also easy to verify:

$$\models \mathrel{|\!\sim}_i\bot \leftrightarrow \Box_i\bot \quad \models \Diamond_i\top \leftrightarrow \mathrel{|\!\sim}\!\!\Diamond_i\top$$
$$\models \mathrel{|\!\sim}_i\top \leftrightarrow \top \quad \models \mathrel{|\!\sim}\!\!\Diamond_i\bot \leftrightarrow \bot$$

The following is the $\mathrel{|\!\sim}$-version of Axiom Schema K.

$$(\widetilde{K}) \ \models \mathrel{|\!\sim}_i(\alpha \rightarrow \beta) \rightarrow (\mathrel{|\!\sim}_i\alpha \rightarrow \mathrel{|\!\sim}_i\beta) \tag{2}$$

The validity below is easy to verify:

$$(\widetilde{R}) \ \models \mathrel{|\!\sim}_i(\alpha \wedge \beta) \leftrightarrow (\mathrel{|\!\sim}_i\alpha \wedge \mathrel{|\!\sim}_i\beta) \tag{3}$$

We also have $\models (\mathrel{|\!\sim}_i\alpha \vee \mathrel{|\!\sim}_i\beta) \rightarrow \mathrel{|\!\sim}_i(\alpha \vee \beta)$, but not the converse, as can easily be checked.

The following validity is an immediate consequence of our preferential semantics:

$$(N) \ \models \Box_i\alpha \rightarrow \mathrel{|\!\sim}_i\alpha \tag{4}$$

Intuitively, given $i = 1, \ldots, n$, where n is the number of modalities in the language, we want $\Box_i$ and $\mathrel{|\!\sim}_i$ to be 'tied' together in so far as one is the defeasible (or the 'hard') version of the other. Schema N is in line with the commonly accepted principle that whatever is classically the case is also defeasibly so.[3]

From duality of $\mathrel{|\!\sim}\!\!\Diamond$ and $\mathrel{|\!\sim}$ and contraposition of N we get:

$$\models \mathrel{|\!\sim}\!\!\Diamond_i\alpha \rightarrow \Diamond_i\alpha \tag{5}$$

It can easily be checked that in our preferential semantics, the standard rule of necessitation $RN : \alpha/\Box_i\alpha$ holds. The following rule of *normal necessitation* (RNN) follows from RN together with Schema N in (4) above:

$$(RNN) \ \frac{\alpha}{\mathrel{|\!\sim}_i\alpha} \tag{6}$$

From satisfaction of (1), (2) and (3), one can see that the logic of our defeasible modalities shares properties commonly characterizing the so-called *normal* modal logics [14]. In particular, we have that the following rule holds:

$$(NRK) \ \frac{(\alpha_1 \wedge \ldots \wedge \alpha_n) \rightarrow \beta}{(\mathrel{|\!\sim}_i\alpha_1 \wedge \ldots \wedge \mathrel{|\!\sim}_i\alpha_n) \rightarrow \mathrel{|\!\sim}_i\beta} \ (n \geq 0) \tag{7}$$

The observant reader would have noticed that we assume we have as many defeasible modalities as we have classical ones. That is, for each $\Box_i$, a corresponding $\mathrel{|\!\sim}_i$ (its defeasible version) is assumed. Moreover they are both linked together via Schema N in (4). In principle, from a technical point of view, nothing precludes us from having defeasible modalities with no corresponding classical version or the other way round. The latter is easily dealt with by simply not having $\mathrel{|\!\sim}_i$ for some i for which $\Box_i$ is present in the language. The former case, on the other hand, would require an elaboration of the semantics as satisfiability of $\mathrel{|\!\sim}$-formulae calls upon the accessibility relation R_i, associated with the $\Box_i$-modality.

The dependency between each (classical) modality and its defeasible counterpart is defined by a (fixed) preference order on worlds in the model. We do not have a Hilbert-style axiomatization of this dependency yet. What is certain is that such an axiomatization would require casting the preference order as a modality, in order to axiomatize the relationship between $\mathrel{|\!\sim}_i$, $\mathrel{|\!\sim}\!\!\Diamond_i$ and the preference order $\prec$, for each i. To this end, we may use, for example, the modal axiomatization of the preference order of Britz et al. [6], or one of Boutilier's modal systems [5]. Such an axiomatization is possible at the expense of moving to a more expressive language (see the remark below Definition 5 above and also the discussion in Section 8). Nevertheless, from a computational logic point of view, we shall suffice with the definition of a tableau-based decision procedure, which will be presented in Section 6.

We also observe that in order for us to capture the semantics of $\widetilde{\mathcal{L}}$ in standard conditional logics [14] we would require the addition of a preference relation on worlds, all standard modalities we want to work with and a suitably defined conditional for each modality in the language. Our contention here is that this route would hardly simplify matters.

[3] Similarly to what happens in KLM consequence relations ($\alpha \models \beta$ implies $\alpha \mathrel{|\!\sim} \beta$) [33] and in defeasible subsumption relations ($C \sqsubseteq D$ implies $C \mathrel{|\!\sim\!\!\sqsubseteq} D$) [9].

From the perspective of knowledge representation and reasoning, it becomes important to address the question of what it means for an $\widetilde{\mathcal{L}}$-sentence to be *entailed* from an $\widetilde{\mathcal{L}}$-knowledge base.

An $\widetilde{\mathcal{L}}$-*knowledge base* is a (possibly infinite) set $\mathcal{K} \subseteq \widetilde{\mathcal{L}}$. Given a preferential Kripke model $\mathscr{P}$, we extend the notion of satisfaction to knowledge bases in the obvious way: $\mathscr{P} \Vdash \mathcal{K}$ if and only if $\mathscr{P} \Vdash \alpha$ for every $\alpha \in \mathcal{K}$.

DEFINITION 6. *Let $\mathcal{K} \subseteq \widetilde{\mathcal{L}}$ and let $\alpha \in \widetilde{\mathcal{L}}$. We say that $\mathcal{K}$ (globally) entails α in the class $\mathcal{M}$ of preferential Kripke models (denoted $\mathcal{K} \models \alpha$) if and only if for every $\mathscr{P} \in \mathcal{M}$, if $\mathscr{P} \Vdash \mathcal{K}$, then $\mathscr{P} \Vdash \alpha$.*

Given this notion of entailment, its associated *consequence relation* is defined as follows:

$$Cn(\mathcal{K}) \equiv_{\text{def}} \{\alpha \mid \mathcal{K} \models \alpha\} \tag{8}$$

It can be checked that the consequence relation $Cn(\cdot)$ as defined in (8) above is a Tarskian consequence relation:

THEOREM 1. *Let $Cn(\cdot)$ be a consequence relation defined in terms of preferential entailment. Then $Cn(\cdot)$ satisfies the following properties:*

- $\mathcal{K} \subseteq Cn(\mathcal{K})$ *(Inclusion)*
- $Cn(\mathcal{K}) = Cn(Cn(\mathcal{K}))$ *(Idempotency)*
- *If $\mathcal{K}_1 \subseteq \mathcal{K}_2$, then $Cn(\mathcal{K}_1) \subseteq Cn(\mathcal{K}_2)$* *(Monotonicity)*

PROOF. See Appendix A.2. □

That is, in spite of the defeasibility features of $\sim$, we end up with a logic that is *monotonic* (at the entailment level).

5. AN APPLICATION EXAMPLE

Let us assume the following simple scenario depicting a nuclear power-plant [8]. In a particular power station there is an atomic pile and a cooling system, both of which can be either on or off. A surveillance agent is in charge of detecting hazardous situations so that the human controller can prevent the plant from malfunctioning (Figure 2).

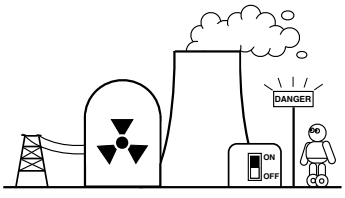


Figure 2: The power plant and its surveillance agent.

In what follows we shall illustrate our constructions from previous sections in reasoning about action using the aforementioned scenario.

We find in the AI literature a fair number of modal-based formalisms for reasoning about actions and change [12, 13, 16, 18, 29, 38, 41, 42, 43]. These are essentially variants of the modal logic K we presented in Section 2. Modal operators are determined by a (finite) set of *actions* $\mathcal{A} = \{a_1, \ldots, a_n\}$: For each $a \in \mathcal{A}$, there is associated a modal operator $\Box_a$. Given a Kripke model, $R_a \subseteq W \times W$ is therefore meant to represent possible executions of an (ontic) action a at specific worlds $w \in W$, i.e., R_a is the specification of a's behavior in a transition system. Hence, whenever $(w, w') \in R_a$, w' is a *possible outcome* of doing a in w. Formulae of the form $\Box_a\alpha$ are used to specify the *effects* of actions and they are read "after *every* execution of action a, the formula α holds". The operator $\Diamond_a$ is mostly used to specify the *executability* of actions: $\Diamond_a\top$ reads "there is a possible execution of action a".

In our nuclear power plant example, let $\mathcal{P} = \{\mathsf{p}, \mathsf{c}, \mathsf{h}\}$ be a set of propositions, where p stands for "the atomic pile is on", c for "the cooling system is on", and h for "hazardous situation". Moreover, let $\mathcal{A} = \{\mathsf{f}, \mathsf{m}\}$ be a set of atomic actions, where f stands for "flipping the pile switch", and m for (occurrence of) "a malfunction".

We first construct a preferential Kripke model (Definition 3) in which to check the satisfiability and truth of a few sentences. (The purpose is to illustrate the semantics of our notion of defeasibility in an action context rather than to present a comprehensive modeling of the nuclear power plant scenario.)

Let $\mathscr{P} = \langle W, R, V, \prec\rangle$ be the preferential Kripke model depicted in Figure 3, where $W = \{w_i \mid 1 \leq i \leq 4\}$, $R = \langle R_{\mathsf{f}}, R_{\mathsf{m}}\rangle$, with $R_{\mathsf{f}} = \{(w_1, w_2), (w_2, w_1), (w_3, w_1), (w_3, w_4), (w_4, w_2), (w_4, w_4)\}$ and $R_{\mathsf{m}} = \{(w_4, w_3), (w_4, w_4)\}$, V is the obvious valuation function, and $\prec$ is the transitive closure of $\{(w_1, w_2), (w_2, w_3), (w_3, w_4)\}$, i.e., of the relation represented by the dashed arrows in the picture. (Note again the direction of $\prec$ from less to more normal worlds.)

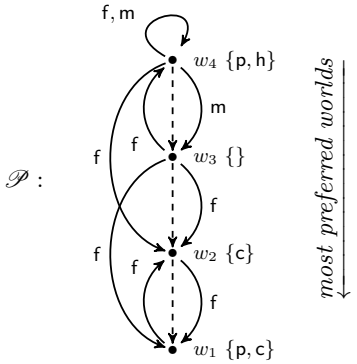


Figure 3: Preferential Kripke model for the power plant scenario.

The rationale of this partial order is as follows: The utility company selling the electricity generated by the power plant tries as far as possible to keep both the pile and the cooling system on, ensuring that the pile can easily be switched off (world w_1); sometimes the company has to switch the pile off for maintenance but then tries to keep the cooler running, because turning the pile on again would not cause a fault in the cooling system (world w_2); more rarely the company needs to switch off both the pile and the cooler, e.g. when the latter needs maintenance (world w_3); and, finally, only in very exceptional situations would the pile be on while the cooler is off, e.g. during a serious malfunction (world w_4).

In the preferential model $\mathscr{P}$ depicted above, one can check that $\mathscr{P} \Vdash (\mathsf{p} \wedge \neg\mathsf{c}) \leftrightarrow \mathsf{h}$, i.e., $[\![(\mathsf{p} \wedge \neg\mathsf{c}) \leftrightarrow \mathsf{h}]\!] = W$. Also, $w_4 \in [\![\mathsf{h} \wedge \Diamond_{\mathsf{f}}\neg\mathsf{h}]\!]$: at w_4 we have a hazardous situation, but

it is possible to switch the pile off having as a normal effect a safe condition. We have that w_1 satisfies $\boxtimes_{\mathsf{m}}\bot$: at w_1 a malfunction cannot occur (which is not true of w_4). In $\mathscr{P}$ we have $\mathscr{P} \Vdash \neg\mathsf{p} \rightarrow \boxtimes_{\mathsf{f}}\mathsf{p}$ (the normal outcome of switching the pile on is it being on), but $\mathscr{P} \nVdash \neg\mathsf{p} \rightarrow \Box_{\mathsf{f}}\mathsf{p}$ (see world w_3). We also have $\mathscr{P} \Vdash \mathsf{c} \rightarrow \boxtimes_{\mathsf{f}}\neg\mathsf{h}$ (if the cooler is on, the normal result of switching the pile is a safe situation). Finally we also have $\mathscr{P} \Vdash \mathsf{h} \rightarrow \diamondsuit_{\mathsf{m}}\top$: in any hazardous situation a meltdown is a distinct possibility — but fortunately $\mathscr{P} \Vdash \Diamond_{\mathsf{f}}\neg\mathsf{h}$: from every world it is possible to return to a non-hazardous world.

So far we have illustrated the preferential semantics of $\widetilde{\mathcal{L}}$-statements using a specific preferential Kripke model. In a knowledge representation context, though, we are interested in preferential entailment from an $\widetilde{\mathcal{L}}$-theory or knowledge base. The latter determines the preferential models that are permissible from the standpoint of the knowledge engineer. To illustrate this, consider the following $\widetilde{\mathcal{L}}$-knowledge base:

$$\mathcal{K} = \left\{ \begin{array}{c} (\mathsf{p} \wedge \neg\mathsf{c}) \leftrightarrow \mathsf{h},\ \mathsf{h} \rightarrow \diamondsuit_{\mathsf{m}}\top, \\ \mathsf{p} \rightarrow \boxtimes_{\mathsf{f}}\neg\mathsf{p},\ \mathsf{c} \rightarrow \boxtimes_{\mathsf{f}}\mathsf{c},\ \Diamond_{\mathsf{f}}\neg\mathsf{h} \end{array} \right\}$$

$\mathcal{K}$ basically says that "a hazardous situation is one in which the pile is on and the cooler off", "in a hazardous situation a malfunction is distinctly possible", "if the pile is on, then flipping its switch normally switches it off", "if the cooler is on, then switching the pile normally does not affect it" and "it is always possible to flip the pile switch". (Note that all the formulae in $\mathcal{K}$ are true in the preferential model $\mathscr{P}$ of Figure 3 above.) We can then conclude $\mathcal{K} \models \mathsf{p} \rightarrow \boxtimes_{\mathsf{f}}\neg\mathsf{h}$, $\mathcal{K} \models \boxtimes_{\mathsf{m}}\bot \rightarrow (\neg\mathsf{p} \vee \mathsf{c})$ and $\mathcal{K} \models (\mathsf{p} \vee \mathsf{c}) \rightarrow \boxtimes_{\mathsf{f}}\neg\mathsf{h}$, using the sound $\widetilde{\mathcal{L}}$-inference rules and validities presented in the previous section.

6. TABLEAU SYSTEM

In this section we present a simple tableau calculus for defeasible modalities based on labeled formulae and on explicit accessibility relations [22].[4] As we shall see, it also makes use of an auxiliary structure of which the intention is to build a preference relation on possible worlds. (For a discussion on the differences between our tableau method and the one by Giordano et al. [20], see end of Section 8.)

Definition 7. *If $n \in \mathbb{N}$ and $\alpha \in \widetilde{\mathcal{L}}$, then $n :: \alpha$ is a labeled formula.*

In a labeled formula $n :: \alpha$, n is the *label*. (As we shall see, informally, the idea is that the label stands for some possible world in a Kripke model.)

Let $mod(\widetilde{\mathcal{L}})$ denote the set of all *classical modalities* of $\widetilde{\mathcal{L}}$. (Remember our assumption that we have as many defeasible modalities as we have classical ones and that, for a given i, both $\Box_i$ and $\boxtimes_i$ depend on the same R_i.)

Definition 8. *A skeleton is a function $\Sigma : mod(\widetilde{\mathcal{L}}) \longrightarrow 2^{\mathbb{N}\times\mathbb{N}}$.*

Informally, a skeleton maps modalities in the language to accessibility relations on possible worlds.

Definition 9. *A preference relation $\prec$ is a binary relation on $\mathbb{N}$.*

[4]Our exposition here follows that given by Varzinczak [37] and Castilho et al. [12, 13].

As alluded to above, $\prec$ is meant to capture a preference relation on possible worlds. As we shall see below, like Σ, $\prec$ is built cumulatively through successive applications of the tableau rules we shall introduce.

Definition 10. *A branch is a tuple $\langle \mathcal{S}, \Sigma, \prec \rangle$, where $\mathcal{S}$ is a set of labeled formulae, Σ is a skeleton and $\prec$ is a preference relation.*

Definition 11. *A tableau rule is a rule of the form:*

$$\rho\ \ \frac{\mathcal{N}\ ;\ \Gamma}{\mathcal{D}_1\ ;\ \Gamma'_1\ \mid\ \ldots\ \mid\ \mathcal{D}_k\ ;\ \Gamma'_k}$$

where $\mathcal{N}; \Gamma$ is the numerator *and $\mathcal{D}_1\ ;\ \Gamma'_1\ \mid\ \ldots\ \mid\ \mathcal{D}_k\ ;\ \Gamma'_k$ is the* denominator.

Given a rule ρ, $\mathcal{N}$ represents one or more labeled formulae, called the *main formulae* of the rule, separated by ','. Γ stands for any additional *condition* (on Σ or $\prec$) that must be satisfied for the rule to be applicable. In the denominator, each $\mathcal{D}_i$, $1 \leq i \leq k$, has one or more labeled formulae, whereas each Γ'_i is a condition to be satisfied *after* the application of the rule (e.g. changes in the skeleton Σ or in the relation $\prec$). The symbol '|' indicates the occurrence of a *split* in the branch.

Figure 4 presents the set of tableau rules for $\widetilde{\mathcal{L}}$. In the rules we abbreviate $(n, n') \in \Sigma(i)$ as $n \xrightarrow{i} n'$, and $n' \in \Sigma(i)(n)$ as $n' \in \Sigma_i(n)$. Finally, with $n'^{\star}, n''^{\star}, \ldots$ we denote labels that have not been used before. We say that a rule ρ is *applicable* to a branch $\langle \mathcal{S}, \Sigma, \prec \rangle$ if and only if $\mathcal{S}$ contains an instance of the main formulae of ρ and the conditions Γ of ρ are satisfied by Σ and $\prec$.

$$(\bot)\ \frac{n :: \alpha,\ n :: \neg\alpha}{n :: \bot} \qquad (\neg)\ \frac{n :: \neg\neg\alpha}{n :: \alpha}$$

$$(\wedge)\ \frac{n :: \alpha \wedge \beta}{n :: \alpha,\ n :: \beta} \qquad (\vee)\ \frac{n :: \neg(\alpha \wedge \beta)}{n :: \neg\alpha \mid n :: \neg\beta}$$

$$(\Box_i)\ \frac{n :: \Box_i\alpha\ ;\ n \xrightarrow{i} n'}{n' :: \alpha} \qquad (\Diamond_i)\ \frac{n :: \neg\Box_i\alpha}{n'^{\star} :: \neg\alpha\ ;\ \Gamma'_1 \mid n'^{\star} :: \neg\alpha\ ;\ \Gamma'_2}$$

$$\text{where } \Gamma'_1 = \{n \xrightarrow{i} n'^{\star},\ n'^{\star} \in \min_{\prec} \Sigma_i(n)\} \text{ and}$$

$$\Gamma'_2 = \{n \xrightarrow{i} n'^{\star},\ n \xrightarrow{i} n''^{\star},\ n''^{\star} \prec n'^{\star},\ n''^{\star} \in \min_{\prec} \Sigma_i(n)\}$$

$$(\boxtimes_i)\ \frac{n :: \boxtimes_i\alpha\ ;\ n \xrightarrow{i} n',\ n' \in \min_{\prec} \Sigma_i(n)}{n' :: \alpha}$$

$$(\diamondsuit_i)\ \frac{n :: \neg\boxtimes_i\alpha}{n'^{\star} :: \neg\alpha\ ;\ n \xrightarrow{i} n'^{\star},\ n'^{\star} \in \min_{\prec} \Sigma_i(n)}$$

Figure 4: Tableau rules for $\widetilde{\mathcal{L}}$.

The Boolean rules together with $(\Box_i)$ are as usual and need no explanation. Rule $(\boxtimes_i)$ propagates formulae in the scope of a defeasible necessity operator to the most preferred (with respect to $\prec$) of all accessible nodes. Rule $(\diamondsuit_i)$ creates a preferred accessible node with the corresponding labeled formulae as content. Rule $(\Diamond_i)$ replaces the standard rule for $\Diamond$-formulae and requires a more thorough explanation. When creating a new accessible node, there are two possibilities: Either (*i*) it is minimal (with respect to $\prec$) amongst all the accessible nodes, in which case the result is the same as that of applying Rule $(\diamondsuit_i)$, or (*ii*) it is not minimal, in which case there must be a most preferred accessible node

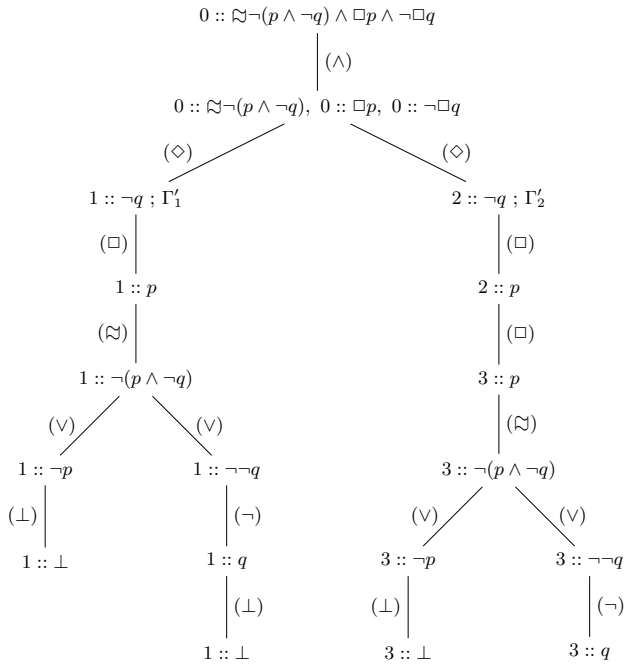
that is more preferred (with respect to $\prec$) than the newly created one. (This splitting is of the same nature as that in the $(\vee)$-rule, i.e., it fits the purpose of a proof by cases.)

DEFINITION 12. *A tableau $\mathcal{T}$ for $\alpha \in \widetilde{\mathcal{L}}$ is the limit of a sequence $\mathcal{T}^0, \ldots, \mathcal{T}^n, \ldots$ of sets of branches where the initial $\mathcal{T}^0 = \{\langle\{0 :: \alpha\}, \emptyset, \emptyset\rangle\}$ and every $\mathcal{T}^{i+1}$ is obtained from $\mathcal{T}^i$ by the application of one of the rules in Figure 4 to some branch $\langle \mathcal{S}, \Sigma, \prec\rangle \in \mathcal{T}^i$. Such a limit is denoted $\mathcal{T}^\infty$.*

We make the so-called *fairness assumption*: Any rule that *can* be applied *will* eventually be applied, i.e., the order of rule applications is not relevant. We say a tableau is *saturated* if no rule is applicable to any of its branches.

DEFINITION 13. *A branch $\langle \mathcal{S}, \Sigma, \prec\rangle$ is closed if and only if $n :: \bot \in \mathcal{S}$ for some n. A saturated tableau $\mathcal{T}$ for $\alpha \in \widetilde{\mathcal{L}}$ is closed if and only if all its branches are closed. (If $\mathcal{T}$ is not closed, then we say that it is an* open tableau*.)*

For an example of construction of a tableau, consider the sentence $\alpha = \approx(p \to q) \to (\Box p \to \Box q)$ (which is not valid). Figure 5 depicts the (open) tableau for $\neg\alpha = \approx\neg(p \wedge \neg q) \wedge \Box p \wedge \neg\Box q$.



Γ'_1 = add $(0,1)$ to Σ and 1 to $\min_\prec \Sigma(0)$
Γ'_2 = add $(0,2)$ and $(0,3)$ to Σ, $(3,2)$ to $\prec$ and 3 to $\min_\prec \Sigma(0)$

Figure 5: Visualization of an open tableau for the formula $\approx\neg(p \wedge \neg q) \wedge \Box p \wedge \neg\Box q$.

From the open tableau in Figure 5 we extract the preferential Kripke model $\mathscr{P}$ depicted in Figure 6. (In Figure 6 the understanding is that $3 \prec 2$ and that 0 is *incomparable* with respect to $\prec$ to the other possible worlds.)

We are now ready to state the main result of this section.

THEOREM 2. *The tableau calculus for $\widetilde{\mathcal{L}}$ is sound and complete with respect to the modal preferential semantics.*

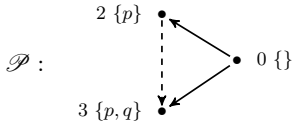


Figure 6: Preferential Kripke model $\mathscr{P}$ constructed from Figure 5.

PROOF. See Appendix A.3. □

It can easily be checked that in the construction of the tableau there is only a finite number of distinct states since every formula generated by the application of a rule is a sub-formula of the original one. Hence we have a decision procedure for $\widetilde{\mathcal{L}}$.

We end this section with a brief remark on complexity. It is well-known that satisfiability checking for modal logic K and K_n are both PSPACE-complete [23, 34]. The addition of $\approx$ and $\diamond$ to the language does not affect the space complexity of the resulting tableaux. If the formula at the root of the tableau is α, and $|\alpha| = m$, then the space requirement for each label is at most $O(m)$. Since there exists a saturated tableau with depth at most $O(m^2)$, the total space requirement is $O(m^3)$.

7. ADDING DEFEASIBLE ARGUMENTS

An obvious next step to the work presented here is the integration of $\widetilde{\mathcal{L}}$ with a KLM-style defeasible consequence relation $\mid\!\sim$, since this would allow for the expression of both defeasible modalities and defeasible argument forms.[5] First we need some definitions.

Given $\mathscr{P} = \langle W, R, V, \prec\rangle$ and $\alpha, \beta \in \mathcal{L}$, the defeasible statement $\alpha \mid\!\sim \beta$ holds in $\mathscr{P}$ (denoted $\mathscr{P} \Vdash \alpha \mid\!\sim \beta$) if and only if $\min_\prec[\![\alpha]\!] \subseteq [\![\beta]\!]$, i.e., every $\prec$-minimal α-world is a β-world. As an example, in the model $\mathscr{P}$ of Figure 1, we have $\mathscr{P} \Vdash p \mid\!\sim \Box q$ (but note that $\mathscr{P} \nVdash p \to \Box q$). We also have $\mathscr{P} \Vdash \neg p \mid\!\sim \diamond(\neg p \wedge \Box q)$ and $\mathscr{P} \nVdash \Box\neg q \mid\!\sim \neg q$ (from the latter follows $\mathscr{P} \nVdash \Box\neg q \to \neg q$).

It is worth noting that if only a classical modal language is assumed, then defeasible statements here still have the same intuition as mentioned in the Introduction. To witness, the statement $\diamond\alpha \mid\!\sim \Box\beta$ just says that "all normal worlds with an α-successor have only β-successors". That is, any $\mid\!\sim$-statement still refers only to normality in the premise, or, in this case, of the 'actual' world. In our enriched language we shall be able to make statements of the form $\alpha \mid\!\sim \approx\beta$.

We say that a preferential Kripke model $\mathscr{P}$ satisfies a set of defeasible statements if each such statement holds in $\mathscr{P}$. Given a set X of defeasible statements, we say that X (preferentially) entails the defeasible statement $\alpha \mid\!\sim \beta$ (denoted $X \models \alpha \mid\!\sim \beta$) if every preferential model satisfying all the statements in X also satisfies $\alpha \mid\!\sim \beta$. (It is easy to see that $\models$ here is exactly the same entailment relation from Definition 6, just restated in terms of $\mid\!\sim$-statements.)

We can now relate the truth of $\widetilde{\mathcal{L}}$-sentences in a preferential model with that of defeasible statements, as the following result shows.

[5]Here, $\mid\!\sim$ need not be a new connective in the language but can rather have the same status as, e.g., *subsumption* and defeasible versions thereof in description logics [1, 7, 9].

LEMMA 2. *Let $\alpha \in \widetilde{\mathcal{L}}$ and $\mathscr{P}$ be a preferential Kripke model. Then $\mathscr{P} \Vdash \alpha$ if and only if $\mathscr{P} \Vdash \neg\alpha \mathrel{|\!\sim} \bot$.*

PROOF. See Appendix A.4. □

This result raises the obvious question on whether and how entailment of $\widetilde{\mathcal{L}}$-sentences relates to that of $\mathrel{|\!\sim}$-statements.

DEFINITION 14. *Let $\mathcal{K} \subseteq \widetilde{\mathcal{L}}$. $\mathcal{K}^{\mathrel{|\!\sim}} := \{\neg\alpha \mathrel{|\!\sim} \bot \mid \alpha \in \mathcal{K}\}$.*

THEOREM 3. *$\mathcal{K} \models \alpha$ if and only if $\mathcal{K}^{\mathrel{|\!\sim}} \models \neg\alpha \mathrel{|\!\sim} \bot$.*

PROOF. See Appendix A.4. □

Hence, preferential entailment in $\widetilde{\mathcal{L}}$ reduces to preferential entailment of $\mathrel{|\!\sim}$-statements in the language of $\widetilde{\mathcal{L}}$. Note that soundness of KLM postulates for modal preferential reasoning [8, 10] is preserved when moving from $\mathcal{L}$ to $\widetilde{\mathcal{L}}$. An immediate consequence of this is that the existence of a sound and complete KLM-style $\mathrel{|\!\sim}$-based proof system [33] for $\widetilde{\mathcal{L}}$ would define a decision procedure for the extension of $\widetilde{\mathcal{L}}$ with $\mathrel{|\!\sim}$. At present we can only conjecture that a proof system along these lines exists, and is based on the integration of the tableau-based proof procedure for $\widetilde{\mathcal{L}}$ presented in Section 6 and the tableau calculi of Giordano et al [20].

8. DISCUSSION AND RELATED WORK

To the best of our knowledge, the first attempt to formalize a notion of relative normality in the context of defeasible reasoning was that of Delgrande [17] in which a conditional logic of normality is defined. Given the relationship between the general constructions on which we base our work and those by Kraus et al., most of the remarks in the comparison made by Lehmann and Magidor [35, Section 3.7] are applicable in comparing Delgrande's approach to ours and we do not repeat them here. We note though that, like Kraus et al. and Boutilier, Delgrande focuses on defeasibility of argument forms rather than modes of reasoning as we studied here. Contrary to them, Delgrande adopts the semantics of standard conditional logics [14, Chapter 10], which is based on a (general) selection function picking out the most normal worlds relative to the current one. In his setting, a conditional $\alpha \Rightarrow \beta$ holds at a world w if and only if the set of most normal α-worlds (relative to w) are also β-worlds. We can capture Delgrande's conditionals in our approach with $\bigtriangledown$-formulae of the form $\bigtriangledown(\alpha \rightarrow \beta)$ in the class of S5 preferential Kripke models.

Boutilier's expressive conditional logics of normality [5] act as unifying framework for a number of conditional logics, including those of Delgrande and Kraus et al. but do not suffice to define $\bigtriangledown$. This is because his modalities are defined directly from a preference order, and do not influence the meaning of any further modalities added to the language.

Baltag and Smets [2] also employ preference orders to refer to the normality of accessible worlds, but their aims and resulting semantics differ from ours in key aspects. They define multi-agent epistemic and doxastic *plausibility models* similar to our preferential Kripke models. Each accessibility relation is induced by a corresponding preference order and linked to an agent whose beliefs are determined by what the agent deems epistemically possible. Minimality, or *doxastic appearance*, is therefore determined relative to an epistemic context, which is induced as an equivalence relation on worlds. This results in modalities of knowledge, (conditional) belief and safe belief that are somewhat related to our defeasible modalities.

In contrast, our work offers a preferential semantic framework independent of a specific application area. We assume (for now) a single preference order across worlds in each Kripke model. The preference order informs the meaning of existing modalities by considering minimality in accessible worlds, where accessibility is determined independently from the preference order. The key difference between our proposal and plausibility models is therefore that our classical modalities are defined independently from any preference order. The special case of a single modality which does correspond to a (connected) preference order yields a logic in which $\bigtriangledown$ defines a belief operator. This follows from the conflation of accessibility and preference in plausibility models.

As we have seen, Britz et al. [8, 10] also propose a general semantic framework for preferential modal logics, but they focus on defeasible arguments rather than on defeasible modalities. As such, the semantics introduced there provides a foundation for the semantics of defeasible modalities, but the syntax of preferential modal logic also does not suffice to define preferential modalities such as ours.

Booth et al. [4] introduce an operator with which one can refer directly in the language to those *most typical* situations in which a given sentence is true. For instance, in their enriched language, a sentence of the form $\overline{\alpha}$ refers to the 'most typical' α-worlds in a semantics similar to ours. One of the advantages of such an extension is the possibility to make statements of the kind "all normal α-worlds are normal β-worlds", thereby shifting the focus of normality from the antecedent by also allowing us to talk about normality in the consequent. This additional expressivity can also be obtained by the addition of the modality $\Box$ of Modular Gödel-Löb logic to express normality syntactically [6, 20]:

$$\overline{\alpha} \equiv_{\text{def}} \Box\neg\alpha \wedge \alpha \tag{9}$$

Despite the gain in expressivity, both these proposals remain propositional in nature in that the only modality allowed is the one with semantics determined by the preference order. Britz et al. extended propositional preferential reasoning to the modal case [8, 10], but the modalities under consideration there remain classical — their meaning remains as in propositional modal logic, despite the underlying preferential semantics of the logic due to the extension of the language with conditional statements of the form $\alpha \mathrel{|\!\sim} \beta$.

If we internalize the preference relation as a modality and enrich our classical modal language with converse modalities and nominals [3], then $\bigtriangledown$ can be given an entirely classical treatment as follows:

$$\bigtriangledown\alpha \equiv_{\text{def}} \bigvee_{o \in \mathcal{O}} (o \wedge \Box(\neg\alpha \rightarrow \Diamond_{\prec}(\alpha \wedge \Diamond^{\smile} o))) \tag{10}$$

where $\Diamond_{\prec}$ is the dual of the modality characterizing the preference relation [6], $\Diamond^{\smile}$ is the converse of $\Diamond$ and $\mathcal{O}$ is a set of nominals. Then $\bigtriangledown\alpha$ is true at a world w in a (hybrid) Kripke model if and only if w is the denotation of some nominal $o \in \mathcal{O}$ and every $\neg\alpha$-world that is accessible from w is less normal than some α-world which is accessible from w. (Of course, besides ensuring that each nominal is interpreted as at most one possible world one also has to make sure that each possible world in a Kripke model is the denotation of

some nominal $o \in \mathcal{O}$. This is warranted in the class of *named* models [3, pp. 439–447].)

The definition in (10) above has the inconvenience of requiring infinitary disjunctions [30] in the language. We can replace (10) with an infinitely denumerable collection of axiom schemata given by:

$$\text{(F) } @_o \nsim \alpha \leftrightarrow @_o \Box(\Box_{\prec} \neg \Diamond\!\!\!\!\breve{} \, o \rightarrow \alpha) \tag{11}$$

As mentioned earlier, making use of such a machinery takes us to a much more expressive language. Note though that complexity-wise we remain in the same class — satisfiability in the basic hybrid logic like the one briefly sketched above is PSPACE-complete [3, Theorem 7.21].

Finally, despite the similarities between the tableau method we presented here and the one by Giordano et al. [20], they remain largely superficial. First, our preferential semantics counts as a proper generalization of the KLM approach to full modal logic, whereas theirs is an embedding of propositional KLM consequence relations in an enriched language. Second, again, in their approach the preference relation is explicit and cast as an additional modality, requiring a special tableau rule to deal with it. Here the preference relation is not present in the language and materializes only in the inner workings of our tableau method.

9. CONCLUSION AND FUTURE WORK

The main contribution of the present paper is the provision of a natural, simple and intuitive framework within which to represent defeasible modes of inference. The defeasible modalities we introduced here refer to the relative normality of *accessible worlds*, unlike syntactic characterizations of normality [4, 5, 20, 21], which refer to the relative normality of worlds in which a given sentence is true, or $\mid\!\sim$ [33, 35], which refers to the relative normality of the worlds in which the premise is true.

We have seen that the modal logics obtained through the addition of $\nsim_i$ are monotonic (Theorem 1). Although a logic based on $\widetilde{\mathcal{L}}$ can be extended to include a nonmonotonic conditional $\mid\!\sim$, such an extension does not make the addition of $\nsim_i$ a superfluous extension to the language, since $\nsim_i$ cannot be expressed in terms of $\mid\!\sim$. One avenue for future research is therefore integrating $\nsim_i$ with our approach to modal preferential reasoning [8, 10], since this would allow for the expression of both defeasible arguments and defeasible modalities. First steps towards this aim were presented in Section 7. Once this is in place, a deeper exploration of applications in various modal logics is warranted.

Here we have investigated the case where a single preference ordering among worlds is assumed. As we have seen, this fits the bill in capturing defeasibility of action effects or obligations, where an 'objective' or commonly agreed upon notion of normality can be quite easily justified. When moving to defeasible notions of knowledge or belief, though, a multi-preference based approach seems to be more appropriate, as agents may have different views on which worlds are more normal than others, i.e., preferences become *subjective* or at least relative to an agent [2].

Here we have investigated defeasible modalities in the system K. Our basic framework paves the way for exploring similar notions of defeasibility and additional properties in specific systems of modal logics. Once this is in place we will be able to investigate further applications of defeasible modalities in e.g. dynamic epistemic logic [36] as well as in other similarly structured logics, such as description logics [1]. We are currently investigating such extensions.

Finally, from a knowledge representation perspective, when one deals with knowledge bases, issues related to modularization [25, 26, 27, 28], knowledge base update and repair [24, 39, 40] as well as knowledge base maintenance and versioning [19] show up. These are tasks acknowledged as important by the community in the classical case [31] and that also make sense in a nonmonotonic setting. When moving to a defeasible approach, though, such tasks have to be reassessed and specific methods and techniques redesigned. This constitutes an avenue worthy of exploration.

10. REFERENCES

[1] F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation and Applications*. Cambridge University Press, 2 edition, 2007.

[2] A. Baltag and S. Smets. A qualitative theory of dynamic interactive belief revision. In G. Bonanno, W. van der Hoek, and M. Wooldridge, editors, *Logic and the Foundations of Game and Decision Theory (LOFT7)*, pages 13–60. Amsterdam Univ. Press, 2008.

[3] P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic*. Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, 2001.

[4] R. Booth, T. Meyer, and I. Varzinczak. PTL: A propositional typicality logic. In L. Fariñas del Cerro, A. Herzig, and J. Mengin, editors, *Proceedings of the 13th European Conference on Logics in Artificial Intelligence (JELIA)*, number 7519 in LNCS, pages 107–119. Springer, 2012.

[5] C. Boutilier. Conditional logics of normality: A modal approach. *Artificial Intelligence*, 68(1):87–154, 1994.

[6] K. Britz, J. Heidema, and W. Labuschagne. Semantics for dual preferential entailment. *Journal of Philosophical Logic*, 38:433–446, 2009.

[7] K. Britz, J. Heidema, and T. Meyer. Semantic preferential subsumption. In J. Lang and G. Brewka, editors, *Proc. International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pages 476–484. AAAI Press/MIT Press, 2008.

[8] K. Britz, T. Meyer, and I. Varzinczak. Preferential reasoning for modal logics. *Electronic Notes in Theoretical Computer Science*, 278:55–69, 2011.

[9] K. Britz, T. Meyer, and I. Varzinczak. Semantic foundation for preferential description logics. In D. Wang and M. Reynolds, editors, *Proc. Australasian Joint Conference on Artificial Intelligence*, number 7106 in LNAI, pages 491–500. Springer, 2011.

[10] K. Britz, T. Meyer, and I. Varzinczak. Normal modal preferential consequence. In M. Thielscher and D. Zhang, editors, *Proc. Australasian Joint Conference on Artificial Intelligence*, number 7691 in LNAI, pages 505–516. Springer, 2012.

[11] K. Britz and I. Varzinczak. Defeasible modes of inference: A preferential perspective. In *International Workshop on Nonmonotonic Reasoning (NMR)*, 2012.

[12] M. Castilho, O. Gasquet, and A. Herzig. Formalizing action and change in modal logic I: the frame problem. *Journal of Logic and Computation*, 9(5):701–735, 1999.

[13] M. Castilho, A. Herzig, and I. Varzinczak. It depends on the context! A decidable logic of actions and plans based on a ternary dependence relation. In *Intl. Workshop on Nonmonotonic Reasoning (NMR)*, 2002.

[14] B. Chellas. *Modal logic: An introduction*. Cambridge University Press, 1980.

[15] G. Crocco and P. Lamarre. On the connections between nonmonotonic inference systems and conditional logics. In R. Nebel, C. Rich, and W. Swartout, editors, *Proc. International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pages 565–571. Morgan Kaufmann Publishers, 1992.

[16] G. De Giacomo and M. Lenzerini. PDL-based framework for reasoning about actions. In M. Gori and G. Soda, editors, *Proceedings of the 4th Congress of the Italian Association for Artificial Intelligence (IA*AI)*, number 992 in LNAI, pages 103–114. Springer-Verlag, 1995.

[17] J. Delgrande. A first-order logic for prototypical properties. *Artificial Intelligence*, 33:105–130, 1987.

[18] R. Demolombe, A. Herzig, and I. Varzinczak. Regression in modal logic. *Journal of Applied Non-Classical Logics*, 13(2):165–185, 2003.

[19] E. Franconi, T. Meyer, and I. Varzinczak. Semantic diff as the basis for knowledge base versioning. In *International Workshop on Nonmonotonic Reasoning (NMR)*, 2010.

[20] L. Giordano, V. Gliozzi, N. Olivetti, and G. Pozzato. Analytic tableaux calculi for KLM logics of nonmonotonic reasoning. *ACM Transactions on Computational Logic*, 10(3):18:1–18:47, 2009.

[21] L. Giordano, N. Olivetti, V. Gliozzi, and G. Pozzato. $\mathcal{ALC} + T$: a preferential extension of description logics. *Fundamenta Informaticae*, 96(3):341–372, 2009.

[22] R. Goré. Tableau methods for modal and temporal logics. In M. D'Agostino, D. Gabbay, R. Hähnle, and J. Posegga, editors, *Handbook of Tableau Methods*, pages 297–396. Kluwer Academic Publishers, 1999.

[23] J. Halpern and Y. Moses. A guide to completeness and complexity for modal logics of knowledge and belief. *Artificial Intelligence*, 54:319–379, 1992.

[24] A. Herzig, L. Perrussel, and I. Varzinczak. Elaborating domain descriptions. In G. Brewka, S. Coradeschi, A. Perini, and P. Traverso, editors, *Proceedings of the 17th European Conference on Artificial Intelligence (ECAI)*, pages 397–401. IOS Press, 2006.

[25] A. Herzig and I. Varzinczak. Domain descriptions should be modular. In R. López de Mántaras and L. Saitta, editors, *Proceedings of the 16th European Conference on Artificial Intelligence (ECAI)*, pages 348–352. IOS Press, 2004.

[26] A. Herzig and I. Varzinczak. Cohesion, coupling and the meta-theory of actions. In L. Kaelbling and A. Saffiotti, editors, *Proc. International Joint Conference on Artificial Intelligence (IJCAI)*, pages 442–447. Morgan Kaufmann Publishers, 2005.

[27] A. Herzig and I. Varzinczak. On the modularity of theories. In R. Schmidt, I. Pratt-Hartmann, M. Reynolds, and H. Wansing, editors, *Advances in Modal Logic*, 5, pages 93–109. King's College Publications, 2005.

[28] A. Herzig and I. Varzinczak. A modularity approach for a fragment of $\mathcal{ALC}$. In M. Fisher, W. van der Hoek, B. Konev, and A. Lisitsa, editors, *Proceedings of the 10th European Conference on Logics in Artificial Intelligence (JELIA)*, number 4160 in LNAI, pages 216–228. Springer-Verlag, 2006.

[29] A. Herzig and I. Varzinczak. Metatheory of actions: beyond consistency. *Artificial Intelligence*, 171:951–984, 2007.

[30] C. Karp. *Languages with Expressions of Infinite Length*. North-Holland, 1964.

[31] B. Konev, D. Walther, and F. Wolter. The logical difference problem for description logic terminologies. In A. Armando, P. Baumgartner, and G. Dowek, editors, *Proc. International Joint Conference on Automated Reasoning (IJCAR)*, number 5195 in LNAI, pages 259–274. Springer-Verlag, 2008.

[32] M. Kracht and F. Wolter. Properties of independently axiomatizable bimodal logics. *Journal of Symbolic Logic*, 56(4):1469–1485, 1991.

[33] S. Kraus, D. Lehmann, and M. Magidor. Nonmonotonic reasoning, preferential models and cumulative logics. *Artificial Intelligence*, 44:167–207, 1990.

[34] R. Ladner. The computational complexity of provability in systems of modal propositional logic. *SIAM Journal on Computing*, 6(3):467–480, 1977.

[35] D. Lehmann and M. Magidor. What does a conditional knowledge base entail? *Artificial Intelligence*, 55:1–60, 1992.

[36] H. van Ditmarsch, W. van der Hoek, and B. Kooi. *Dynamic Epistemic Logic*. Springer, 2007.

[37] I. Varzinczak. Causalidade e dependência em raciocínio sobre ações ("Causality and dependency in reasoning about actions"). M.Sc. thesis, Universidade Federal do Paraná, Curitiba, Brazil, 2002.

[38] I. Varzinczak. *What is a good domain description? Evaluating and revising action theories in dynamic logic*. PhD thesis, Univ. Paul Sabatier, Toulouse, 2006.

[39] I. Varzinczak. Action theory contraction and minimal change. In J. Lang and G. Brewka, editors, *Proc. International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pages 651–661. AAAI Press/MIT Press, 2008.

[40] I. Varzinczak. On action theory change. *Journal of Artificial Intelligence Research*, 37:189–246, 2010.

[41] D. Zhang and N. Foo. EPDL: A logic for causal reasoning. In B. Nebel, editor, *Proc. International Joint Conference on Artificial Intelligence (IJCAI)*, pages 131–138. Morgan Kaufmann Publishers, 2001.

[42] D. Zhang and N. Foo. Interpolation properties of action logic: Lazy-formalization to the frame problem. In S. Flesca, S. Greco, N. Leone, and G. G. Ianni, editors, *Proceedings of the 8th European Conference on Logics in Artificial Intelligence (JELIA)*, number 2424 in LNCS, pages 357–368. Springer-Verlag, 2002.

[43] D. Zhang and N. Foo. Frame problem in dynamic logic. *Journal of Applied Non-Classical Logics*, 15(2):215–239, 2005.

APPENDIX

A. PROOFS OF MAIN RESULTS

A.1 Proof of Lemma 1

• Proving the 'only if' part: Let $\alpha \in \mathcal{L}$ be such that $\mathscr{M} \Vdash \alpha$, where $\mathscr{M} = \langle W, R, V\rangle$. Then $\mathscr{M}, w \Vdash \alpha$ for every $w \in W$. Let $\mathscr{P} = \langle W, R, V, \prec\rangle$ for some $\prec \subseteq W \times W$. Since $\alpha \in \mathcal{L}$, α's truth conditions do not depend on $\prec$. Then, given that α is true at every $w \in W$, it follows that $[\![\alpha]\!] = W$ and therefore $\mathscr{P} \Vdash \alpha$.

• Proving the 'if' part: Let $\alpha \in \mathcal{L}$ be such that $\mathscr{P} \Vdash \alpha$, where $\mathscr{P} = \langle W, R, V, \prec\rangle$. Then $[\![\alpha]\!] = W$. Since $\alpha \in \mathcal{L}$, it follows that $\mathscr{M}, w \Vdash \alpha$ for every $w \in W$ with $\mathscr{M} = \langle W, R, V\rangle$. Hence $\mathscr{M} \Vdash \alpha$. □

A.2 Proof of Theorem 1

• Showing Inclusion: Let $\alpha \in \mathcal{K}$. Since every preferential Kripke model of $\mathcal{K}$ is a model of α, it immediately follows that $\mathcal{K} \models \alpha$, from which follows $\alpha \in Cn(\mathcal{K})$.

• Showing Idempotency: Let $\alpha \in Cn(\mathcal{K})$. Then $Cn(\mathcal{K}) \models \alpha$ follows by the same argument given for Inclusion above. Hence $\alpha \in Cn(Cn(\mathcal{K}))$. For the other direction, let $\alpha \in Cn(Cn(\mathcal{K}))$. Then $Cn(\mathcal{K}) \models \alpha$. Assume that $\alpha \notin Cn(\mathcal{K})$. Then $\mathcal{K} \not\models \alpha$, and then there exists $\mathscr{P}$ such that $\mathscr{P} \Vdash \mathcal{K}$ but $\mathscr{P} \not\Vdash \alpha$. But from the definition of $Cn(\cdot)$ we have $\mathscr{P} \Vdash Cn(\mathcal{K})$, from which we derive a contradiction. Hence $\alpha \in Cn(\mathcal{K})$.

• Showing Monotonicity: Let $\alpha \in Cn(\mathcal{K}_1)$. Then $\mathcal{K}_1 \models \alpha$. Let $\mathscr{P}$ be such that $\mathscr{P} \Vdash \mathcal{K}_2$. Since $\mathcal{K}_1 \subseteq \mathcal{K}_2$, we have $\mathscr{P} \Vdash \mathcal{K}_1$ too. Hence $\mathscr{P} \Vdash \alpha$ and we have $\mathcal{K}_2 \models \alpha$, and therefore $\alpha \in Cn(\mathcal{K}_2)$. □

A.3 Proof of Theorem 2

We first show completeness of our tableau method, i.e., if $\alpha \in \widetilde{\mathcal{L}}$ is preferentially valid, then every tableau for $\neg\alpha$ is closed. Equivalently, if there is an open (saturated) tableau for α, then α is satisfiable, i.e., there exists a preferential Kripke model $\mathscr{P}$ in which $[\![\alpha]\!] \neq \emptyset$.

In the following, we show that from any open tableau $\mathcal{T}$ for $\alpha \in \widetilde{\mathcal{L}}$ one can construct a preferential Kripke model satisfying α, from which the result follows.

Let $\mathcal{T} = \mathcal{T}^{\infty}$ be an open saturated tableau for the formula $\alpha \in \widetilde{\mathcal{L}}$ (possibly infinite). Then there must be an open branch $\langle \mathcal{S}, \Sigma, \prec\rangle$ in $\mathcal{T}$ (cf. Definition 13). Let the tuple $\mathscr{P}_{\mathcal{T}} := \langle W_{\mathcal{T}}, R_{\mathcal{T}}, V_{\mathcal{T}}, \prec_{\mathcal{T}}\rangle$ be defined as follows:

- $W_{\mathcal{T}} := \{n \mid n :: \beta \in \mathcal{S}\}$;
- $R_{\mathcal{T}} := \langle R_1, \ldots, R_n\rangle$, where each $R_i := \Sigma(i)$, for $1 \leq i \leq n$;
- $V_{\mathcal{T}} := v$, where $v : W_{\mathcal{T}} \times \mathcal{P} \longrightarrow \{0, 1\}$ and $v(n, p) = 1$ if and only if $n :: p \in \mathcal{S}$, and
- $\prec_{\mathcal{T}} := \prec$.

Lemma 3. *$\mathscr{P}$ is a preferential Kripke model.*

Proof. That $\mathscr{M}_{\mathcal{T}} := \langle W_{\mathcal{T}}, R_{\mathcal{T}}, V_{\mathcal{T}}\rangle$ is a Kripke model follows immediately from the definition of $W_{\mathcal{T}}$, $R_{\mathcal{T}}$ and $V_{\mathcal{T}}$ above. It remains to show that $\prec_{\mathcal{T}}$ is a strict partial order satisfying the smoothness condition [33]. That is, one has to show that:

- $\prec_{\mathcal{T}}$ is irreflexive and transitive: This follows from the construction of $\prec$ in Rules $(\mathbin{\vert\!\sim}_i)$ and $(\Diamond_i)$, since (i) no pair (n, n) is ever added to $\prec$ and (ii) no chain of length greater than 2 is ever added to the preference structure.
- $\prec_{\mathcal{T}}$ has no infinitely descending chains: Clearly no pair (n, n') is added to $\prec$ beyond those added by Rules $(\mathbin{\vert\!\sim}_i)$ and $(\Diamond_i)$. Given this one can easily check that $\prec$ must have a minimum.

□

It remains to show that $\mathscr{P}$ above satisfies α.

Lemma 4. *Let $\mathscr{P} = \langle W_{\mathcal{T}}, R_{\mathcal{T}}, V_{\mathcal{T}}, \prec_{\mathcal{T}}\rangle$ and let β be a sub-formula of α. If $n :: \beta \in \mathcal{S}$, then $n \in [\![\beta]\!]$.*

Proof. The proof is by structural induction on the number of connectives in β.
Base case: β is a literal. We have two cases: (i) $\beta = p \in \mathcal{P}$. Then $n :: p \in \mathcal{S}$ if and only if $v(n, p) = 1$ if and only if $V_{\mathcal{T}}(n, p) = 1$ if and only if $n \in [\![p]\!] = [\![\beta]\!]$. (ii) $\beta = \neg p$ for some $p \in \mathcal{P}$. Then $n :: \neg p \in \mathcal{S}$, and therefore $n :: p \notin \mathcal{S}$, otherwise $n :: \bot \in \mathcal{S}$ (as $\mathcal{T}$ is saturated), contradicting the assumption that $\langle \mathcal{S}, \Sigma, \prec\rangle$ is open. Hence $v(n, p) = 0$, and then $n \notin [\![p]\!]$, from which follows $n \in W_{\mathcal{T}} \setminus [\![p]\!] = [\![\neg p]\!] = [\![\beta]\!]$.
Induction step: The Boolean cases are as usual. We analyze the modal cases (below $\mathscr{M}_{\mathcal{T}} = \langle W_{\mathcal{T}}, R_{\mathcal{T}}, V_{\mathcal{T}}\rangle$):

- $\beta = \Box_i\gamma$: If $n :: \Box_i\gamma \in \mathcal{S}$, then $n' :: \gamma \in \mathcal{S}$ by Rule $(\Box_i)$, for every n' such that $(n, n') \in R_i$. By the induction hypothesis, $n' \in [\![\gamma]\!]$ for every n' such that $(n, n') \in R_i$, i.e., $\mathscr{M}_{\mathcal{T}}, n' \Vdash \gamma$ for every n' such that $(n, n') \in R_i$. From this we conclude $\mathscr{M}_{\mathcal{T}}, n \Vdash \Box_i\gamma$ and therefore $n \in [\![\Box_i\gamma]\!]$.
- $\beta = \neg\Box_i\gamma$: If $n :: \neg\Box_i\gamma \in \mathcal{S}$, then by Rule $(\Diamond_i)$ there exists n' such that $(n, n') \in R_i$ and $n' :: \neg\gamma \in \mathcal{S}$. Then there exists n' such that $(n, n') \in R_i$ and $n' \in [\![\neg\gamma]\!]$, by the induction hypothesis. Hence $n \in [\![\neg\Box_i\gamma]\!]$.
- $\beta = \mathbin{\vert\!\sim}_i\gamma$: If $n :: \mathbin{\vert\!\sim}_i\gamma \in \mathcal{S}$, then $n' :: \gamma \in \mathcal{S}$ by Rule $(\mathbin{\vert\!\sim}_i)$, for every n' such that $n' \in \min_{\prec_{\mathcal{T}}} R_i(n)$. By the induction hypothesis, $n' \in [\![\gamma]\!]$ for every n' such that $n' \in \min_{\prec_{\mathcal{T}}} R_i(n)$, and therefore $n \in [\![\mathbin{\vert\!\sim}_i\gamma]\!]$.
- $\beta = \neg\mathbin{\vert\!\sim}_i\gamma$: If $n :: \neg\mathbin{\vert\!\sim}_i\gamma \in \mathcal{S}$, then by Rule $(\mathbin{\vert\!\sim}_i)$ there exists n' such that $n' \in \min_{\prec_{\mathcal{T}}} R_i(n)$ and $n' :: \neg\gamma \in \mathcal{S}$. Then there exists n' such that $n' \in \min_{\prec_{\mathcal{T}}} R_i(n)$ and $n' \in [\![\neg\gamma]\!]$, by the induction hypothesis. Hence $n \in [\![\neg\mathbin{\vert\!\sim}_i\gamma]\!]$.

□

Now, since $0 :: \alpha \in \mathcal{S}$, from Lemma 4 we conclude that $0 \in [\![\alpha]\!]$. Hence $[\![\alpha]\!] \neq \emptyset$ for the preferential Kripke model constructed as above, and therefore α is satisfiable, as we wanted to show. □

In the following we show soundness, i.e., if $\alpha \in \widetilde{\mathcal{L}}$ is (preferentially) satisfiable, then there is an open tableau for α. Equivalently, if all the tableaux for α are closed, then α is unsatisfiable, i.e., $\neg\alpha$ is valid.

Definition 15. *Let $\mathcal{S}$ be a set of labeled formulae. $\mathcal{S}(n) := \{\beta \mid n :: \beta \in \mathcal{S}\}$.*

Definition 16. $\widehat{\mathcal{S}(n)} := \bigwedge\{\beta \mid \beta \in \mathcal{S}(n)\}$.

LEMMA 5. *If, for every tableau rule that can be applied to* $\mathcal{T}^j = \{\ldots, \langle \mathcal{S}^j, \Sigma^j, \prec^j \rangle, \ldots\}$ *to produce* $\mathcal{T}^{j+1} = \{\ldots, \langle \mathcal{S}^{j+1}, \Sigma^{j+1}, \prec^{j+1} \rangle, \ldots\}$ *and for every branch* $\langle \mathcal{S}^j, \Sigma^j, \prec^j \rangle \in \mathcal{T}^j$ *there exists* n *such that* $\widehat{\mathcal{S}^{j+1}(n)}$ *is unsatisfiable, then* $\widehat{\mathcal{S}^j(n)}$ *is unsatisfiable.*

PROOF. We suffice with the cases of Rules ($\diamond_i$) and ($\Diamond_i$).

- Rule ($\diamond_i$): If $\mathcal{S}^j$ contains $n :: \neg\boxtimes_i\beta$, then an application of Rule ($\diamond_i$) creates a new label n', adds $n \xrightarrow{i} n'$ to $\Sigma^j(i)$ to obtain $\Sigma^{j+1}(i)$, adds $n' :: \neg\beta$ to $\mathcal{S}^j$ to obtain $\mathcal{S}^{j+1}$, and sets n' as a minimum in $\Sigma^{j+1}(i)$ with respect to $\prec^{j+1}$ (which extends $\prec^j$). Now, suppose $\widehat{\mathcal{S}^j(n)}$ is satisfiable, but $\widehat{\mathcal{S}^{j+1}(n')}$ is unsatisfiable. Since $\widehat{\mathcal{S}^{j+1}(n')} = \neg\beta$ (as $\mathcal{S}^{j+1}$ is the singleton $\{n' :: \neg\beta\}$ — n' the freshly added label), then $\neg\beta$ must be unsatisfiable, i.e., $\models \beta$. From this and normal necessitation — Rule (6) —, we have $\models \boxtimes_i\beta$. Hence $\widehat{\mathcal{S}^j(n)}$ is unsatisfiable too because $n :: \neg\boxtimes_i\beta \in \mathcal{S}^j$.
- Rule ($\Diamond_i$): If $\mathcal{S}^j$ contains $n :: \neg\Box_i\beta$, then an application of Rule ($\Diamond_i$) will create a new label n' and either (*i*) add $n \xrightarrow{i} n'$ to $\Sigma^j(i)$ to obtain $\Sigma^{j+1}(i)$, add $n' :: \neg\beta$ to $\mathcal{S}^j$ to obtain $\mathcal{S}^{j+1}$, and set n' as a minimum in $\Sigma^{j+1}(i)$ with respect to $\prec^{j+1}$ (thereby extending $\prec^j$) or (*ii*) add $n \xrightarrow{i} n'$ to $\Sigma^j(i)$ to obtain $\Sigma^{j+1}(i)$, add $n' :: \neg\beta$ to $\mathcal{S}^j$ to obtain $\mathcal{S}^{j+1}$, create a new label n'' and also add $n \xrightarrow{i} n''$ to $\Sigma^{j+1}(i)$, add (n'', n') to $\prec^j$ to obtain $\prec^{j+1}$ and set n'' as a minimum in $\Sigma^{j+1}(i)$ with respect to $\prec^{j+1}$. If (*i*) is the case, then we have the same argument as for Rule ($\diamond_i$) above. Let us assume (*ii*) is the case. Suppose $\widehat{\mathcal{S}^j(n)}$ is satisfiable, but either $\widehat{\mathcal{S}^{j+1}(n')}$ is unsatisfiable or $\widehat{\mathcal{S}^{j+1}(n'')}$ is unsatisfiable. If $\widehat{\mathcal{S}^{j+1}(n')}$ is unsatisfiable, since $\widehat{\mathcal{S}^{j+1}(n')} = \neg\beta$ we have the same argument as for Rule ($\diamond_i$) above. If $\widehat{\mathcal{S}^{j+1}(n'')}$ is unsatisfiable, then since $\widehat{\mathcal{S}^{j+1}(n'')} = \top$, we have $\models \bot$, which implies $\models \Box_i\bot$, and then $\models \Box_i\beta$. Hence $\widehat{\mathcal{S}^j(n)}$ is unsatisfiable too because $n :: \neg\Box_i\beta \in \mathcal{S}^j$.

□

From Lemma 5 we conclude that if all tableaux for α are closed, then every $\widehat{\mathcal{S}(n)}$ is unsatisfiable. In particular $\widehat{\mathcal{S}(0)} = \alpha$ is unsatisfiable. Hence all rules preserve satisfiability when transforming one set of branches into another. This warrants soundness of our tableau rules. □

A.4 Proofs of Lemma 2 and Theorem 3

Lemma 2: Let $\mathscr{P} = \langle W, R, V, \prec \rangle$. $\mathscr{P} \Vdash \alpha$ if and only if $[\![\alpha]\!] = W$ if and only if $[\![\neg\alpha]\!] = \emptyset$ if and only if $\min_\prec [\![\neg\alpha]\!] = \emptyset$ if and only if $\min_\prec [\![\neg\alpha]\!] \subseteq [\![\bot]\!]$ if and only if $\mathscr{P}\neg\alpha \mathrel{|\!\sim} \bot$. □

Theorem 3: Let $\mathcal{K}^{\mathrel{|\!\sim}}$ be obtained from $\mathcal{K}$ as in Definition 14. For the 'only if' part, let $\mathscr{P}$ be such that $\mathscr{P} \Vdash \mathcal{K}^{\mathrel{|\!\sim}}$, i.e., $\mathscr{P} \Vdash \neg\beta \mathrel{|\!\sim} \bot$ for every $\neg\beta \mathrel{|\!\sim} \bot$ in $\mathcal{K}^{\mathrel{|\!\sim}}$. From Lemma 2, this is the case if and only if $\mathscr{P} \Vdash \beta$ for every $\beta \in \mathcal{K}$. Hence $\mathscr{P} \Vdash \mathcal{K}$, and since $\mathcal{K} \models \alpha$, we have $\mathscr{P} \Vdash \alpha$ too. From Lemma 2 again we get $\mathscr{P} \Vdash \neg\alpha \mathrel{|\!\sim} \bot$. Now, for the 'if' part, let $\mathscr{P}$ be such that $\mathscr{P} \Vdash \mathcal{K}$, i.e., $\mathscr{P} \Vdash \beta$ for all $\beta \in \mathcal{K}$. From Lemma 2, it follows that $\mathscr{P} \Vdash \neg\beta \mathrel{|\!\sim} \bot$ for every $\beta \in \mathcal{K}$, and then $\mathscr{P} \Vdash \mathcal{K}^{\mathrel{|\!\sim}}$. From this and $\mathcal{K}^{\mathrel{|\!\sim}} \models \neg\alpha \mathrel{|\!\sim} \bot$ we have $\mathscr{P} \Vdash \neg\alpha \mathrel{|\!\sim} \bot$, and therefore by Lemma 2 again we get $\mathscr{P} \Vdash \alpha$. □

Acknowledgments

The authors are grateful to the anonymous referees for their constructive and useful remarks.

This work is based upon research supported by the National Research Foundation (NRF). Any opinion, findings and conclusions or recommendations expressed in this material are those of the authors and therefore the NRF do not accept any liability in regard thereto. This work was partially funded by Project # 247601, Net2: Network for Enabling Networked Knowledge, from the FP7-PEOPLE-2009-IRSES call.

Knowledge, awareness, and bisimulation

Hans van Ditmarsch
LORIA, CNRS - Univ. de Lorraine
hvd@us.es

Tim French
University of Western Australia
tim@csse.uwa.edu.au

Fernando R. Velázquez-Quesada
University of Seville
FRVelazquezQuesada@us.es

Yì N. Wáng
Bergen University College
yi.wang@hib.no

ABSTRACT

We compare different epistemic notions in the presence of awareness of propositional variables: the logics of implicit knowledge (in which explicit knowledge is definable), explicit knowledge, and speculative knowledge. Different notions of bisimulation are suitable for these logics. We provide correspondence between bisimulation and modal equivalence on image-finite models for these logics. The logic of speculative knowledge is equally expressive as the logic of explicit knowledge, and the logic of implicit knowledge is more expressive than both. We also provide axiomatizations for the three logics — only the one for speculative knowledge is novel. Then we move to the study of dynamics by recalling action models incorporating awareness. We show that any conceivable change of knowledge or awareness can be modelled in this setting, we give a complete axiomatization for the dynamic logic of implicit knowledge. The dynamic versions of all three logics are, surprising, equally expressive.

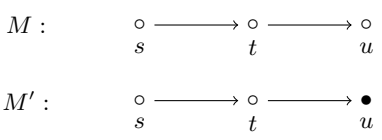
Keywords

modal logic, awareness, bisimulation, dynamics

1. INTRODUCTION

Motivating example. Explicit knowledge is often defined as implicit knowledge plus awareness, with implicit knowledge given by the standard modal box [4, 9]. Thus, to express that 'agent i knows φ explicitly', $K_i^E\varphi$, we use formulas of the form $\Box_i\varphi \wedge A_i\varphi$. In such frameworks, awareness is typically modelled as a function $\mathcal{A}$ that indicates the set of formulas each agent is aware of at each state; hence, $A_i\varphi$ is true at state s iff $\varphi \in \mathcal{A}_i(s)$. When the agents' awareness consists of all formulas built from a subset of atoms (the so-called propositional awareness), we can simply associate with a formula φ the set of atoms $Q \subseteq P$ occurring in φ, and we can then say that $A_i\varphi$ is true at state s iff $Q \subseteq \mathcal{A}_i(s)$.

This definition of explicit knowledge can lead to counter-intuitive situations. Consider the following models.

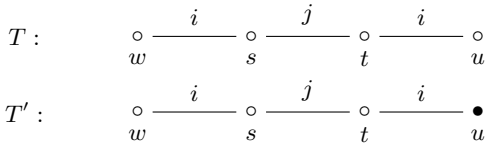


Model M has a domain $\{s, t, u\}$, a single agent i with accessibility relation $R = \{(s, t), (t, u)\}$, atom p true in all states, and the agent is aware of p only in state s. Awareness is not depicted. Model M' is like M, except that p is now false in u (the black dot).

As mentioned, the agent knows explicitly a given φ at a given state iff she is aware of the formula in that state and φ is true in all accessible states. Let us apply this to the depicted structures. In both, the agent is unaware of p at state t, and therefore of the value of p in u: she should see (M, t) and (M', t) as identical, and therefore (M, s) and (M', s) as well. We propose a notion of bisimilarity for which (M, s) and (M', s) are bisimilar.

Now here is the surprise: in the language with awareness and modal box, states (M, s) and (M', s) are not modally equivalent. Given explicit knowledge $K_i^E\varphi$ as $\Box_i\varphi \wedge A_i\varphi$, consider $K_i^E\Box_i p$. This is true in (M, s) but false in (M', s).

In logics of awareness [4] it is common only to consider models for knowledge (equivalence relations) and belief. However, as always in multi-agent logics, it is elementary to transform a single-agent model with directed (asymmetric) accessibility into a multi-agent model where intersecting equivalence classes for agents force such asymmetry. For example, consider the following.



Models T and T' have equivalence accessibility relations (a line represents a two-directions arrow, with reflexive and transitive arrows omitted) for agents i and j. Agent i is aware of p in the states w, and unaware of p in every other state; agent j is unaware of p in every state. The only difference between T and T' is that p is true at (T, u) and false in (T', u). Again, intuitively, these models are the same from agent i's perspective. But $K_i^E\Box_j\Box_i p$ is true above and false below.

TARK 2013, Chennai, India.

The problem here is the presence of the $\Box$. If the K^E operator is not defined by abbreviation but a primitive in the language, then the models cannot be distinguished, as we will prove. Explicit possibility L^E seems another desirable primitive, as it is not the dual of explicit knowledge (both require awareness). This led us to the comparison of logics where different epistemic notions are primitive. Instead of K^E and L^E as primitives, it turns out that we can equally well take K^E and A (awareness) as primitive, and this language then contrasts nicely with the initial one with $\Box$ and A as primitives. A third epistemic notion is also in our focus: speculative knowledge K^S [20, 21], and with that, the language with K^S and A. An agent i speculatively knows φ, $K_i^S\varphi$, if in any i-accessible state, in any state indistinguishable from that as far as awareness of i is concerned, φ is true. This is exactly the sense in which (M, s) and (M', s'), or (T, w) and (T', w), are similar for i.

Our results. This paper addresses the question of what a proper notion of knowledge should be in the presence of awareness, and what the proper notion of bisimulation should be in structures encoding knowledge and awareness; how these choices interact; and how adding dynamics of knowledge and awareness further affects this. We present two notions of bisimulation for the Fagin and Halpern structures of [4], *standard bisimulation* and *awareness bisimulation*; and we present three logics, all in the presence of operators $A_i\varphi$ for awareness of variables occurring in φ: *the logic of implicit knowledge* (with $\Box_i$, so that K_i^E is definable), *the logic of explicit knowledge* (with K_i^E), and *the logic of speculative knowledge* (with K_i^S), summarily introduced above as knowledge modulo speculation over unaware variables. We then show that, on image-finite models, standard bisimilarity corresponds to modal equivalence in the logic of implicit knowledge, but that awareness bisimilarity corresponds to modal equivalence in the logic of explicit knowledge, and also to modal equivalence in the logic of speculative knowledge. We continue by listing various expressivity results, mainly that the logic of implicit knowledge is (strictly) more expressive than the logic of explicit knowledge (reminiscent of [9]). After that we give axiomatizations for our three logics. The logic of implicit knowledge was already axiomatized in [4] and the logic of explicit knowledge in [9], but the axiomatization for the logic of speculative knowledge is novel. Then we investigate the dynamics of awareness and of knowledge, by way of *epistemic awareness action models*. The dynamic logic of speculative knowledge has already been reported in [22]. Here, we show that on the class of finite models every conceivable change of (implicit, explicit, or speculative) knowledge or awareness can be modelled in an epistemic awareness action model. Finally, we give a complete axiomatization for the dynamic logic of implicit knowledge. The dynamic versions of the logics are, surprising, equally expressive. This also gives us the axiomatization for the dynamic logic of explicit knowledge.

Overview of the literature. Our work is rooted in the tradition of epistemic logic [13] and in particular multi-agent epistemic logic [15, 5], in various works on the interaction between awareness and knowledge [4, 16, 9, 11, 12, 8, 10], and in modal logical research in propositional quantification, starting with [6] and followed up by work on bisimulation quantifiers [24, 14, 7].

Works treating awareness either follow a more *semantically* flavoured approach, where awareness is defined in terms of a set of propositional variables [17, 11], or a more *syntactically* flavoured approach, where awareness concerns all formulas of the language in a given set, in order to model 'limited rationality' of agents [4, 19]. Our proposal falls straight into the semantic corner: within the limits of their awareness, agents are fully rational.

2. LOGICS FOR AWARENESS

Throughout the contribution, given are a countable non-empty set of atomic propositions P and a (disjoint) finite non-empty set of agents N.

Definition 1 (Epistemic awareness model) *An* epistemic awareness model *is a tuple* $M = (S, R, \mathcal{A}, V)$ *where*

- S *(also denoted by* $\mathcal{D}(M)$*) is a non-empty set of* states*;*
- $R : N \to \mathcal{P}(S \times S)$ *is an* accessibility *function;*
- $\mathcal{A} : N \to S \to \mathcal{P}(P)$ *is an* awareness *function;*
- $V : P \to \mathcal{P}(S)$ *is a* valuation.

A pair (M, s) *with* $s \in S$ *is an* epistemic awareness state.

We write R_i for $R(i)$, $\mathcal{A}_i$ for $\mathcal{A}(i)$, and $R_i(s)$ for $\{t \in S \mid R_i(s, t)\}$. An epistemic awareness model is *image-finite* if all $R_i(s)$ are finite.

An epistemic awareness model is simply an epistemic model plus a propositional awareness function. We associate two notions of bisimulation [18, 3] with this. Standard bisimulation is the more obvious one, but awareness bisimulation is evidently the more suitable notion in view of our introductory examples. The motivation for awareness bisimulation was the lattice of state spaces in [11]; see [20, 21] for details.

Definition 2 (Standard bisimulation) *Let* $Q \subseteq P$*. A* Q standard bisimulation *between epistemic awareness models* $M = (S, R, \mathcal{A}, V)$ *and* $M' = (S', R', \mathcal{A}', V')$ *is a relation* $\mathfrak{R}[Q] \subseteq (S \times S')$ *such that, for every* $(s, s') \in \mathfrak{R}[Q]$*, for every agent* $i \in N$*, and for every* $p \in Q$*:*

- **atoms:** $s \in V(p)$ *iff* $s' \in V'(p)$*;*
- **aware:** $Q \cap \mathcal{A}_i(s) = Q \cap \mathcal{A}'_i(s')$*;*
- **forth:** *if* $t \in R_i(s)$ *then there is a* $t' \in R'_i(s')$ *such that* $(t, t') \in \mathfrak{R}[Q]$*;*
- **back:** *if* $t' \in R'_i(s')$ *then there is a* $t \in R_i(s)$ *such that* $(t, t') \in \mathfrak{R}[Q]$.

(M, s) *and* (M', s') *are* Q standard bisimilar*, notation* $(M, s) \simeq_Q (M', s')$*, if there is a* Q *standard bisimulation between* M *and* M' *that contains* (s, s')*.*

Definition 3 (Awareness bisimulation) *As Definition 2 but with the following clauses for* **forth** *and* **back** *instead.*

- **forth:** *if* $t \in R_i(s)$ *then there is a* $t' \in R'_i(s')$ *such that* $(t, t') \in \mathfrak{R}[Q \cap \mathcal{A}_i(s)]$*;*
- **back:** *if* $t' \in R'_i(s')$ *then there is a* $t \in R_i(s)$ *such that* $(t, t') \in \mathfrak{R}[Q \cap \mathcal{A}'_i(s')]$.

where $\mathfrak{R}[Q \cap \mathcal{A}_i(s)]$ *is a* $Q \cap \mathcal{A}_i(s)$ *awareness bisimulation and* $\mathfrak{R}[Q \cap \mathcal{A}'_i(s')]$ *is a* $Q \cap \mathcal{A}'_i(s')$ *awareness bisimulation. The notation for* Q *awareness bisimilarity is* $(M, s) \leftrightarrows_Q (M', s')$*.*

In an awareness bisimulation, the perspective of the agent is restricted to the variables that she is aware of, therefore in the **back** and **forth** steps bisimulation is only checked for the variables in $Q \cap \mathcal{A}_i(s)$ instead of the variables in Q. The following is therefore obvious.

Proposition 4 *Let (M,s) and (M',s') be epistemic awareness models, and $Q \subseteq P$. If $(M,s) \simeq_Q (M',s')$, then $(M,s) \leftrightarrows_Q (M',s')$.*

Example 1 *Awareness bisimularity does not imply standard bisimularity. The epistemic awareness states (M,s) and (M',s) of the introduction are $\{p\}$ awareness bisimilar. To see this, observe that (M,u) and (M',u') are $\emptyset$ awareness bisimilar; then, because of this, because $\{p\} \cap \mathcal{A}_i(t) = \emptyset$ and because t's states coincide in p's truth value and in i's awareness of p, epistemic awareness states (M,t) and (M',t') are $\{p\}$ awareness bisimilar. In turn, this, the fact that $\{p\} \cap \mathcal{A}_i(s) = \{p\}$ and the fact that s and s' coincide in p's truth value and in i's awareness of p, make epistemic awareness states (M,s) and (M',s') $\{p\}$ awareness bisimilar too.*

However, (M,s) and (M',s) are not $\{p\}$ standard bisimilar because, in turn, (M,t) and (M',t) are not $\{p\}$ standard bisimilar, and this is because (M,u) and (M',u) are not $\{p\}$ standard bisimilar: they differ in p's truth-value.

Definition 5 (Language) *The language $\mathcal{L}(\Box, K^E, L^E, K^S, A)$ is defined as follows, where $p \in P$ and $i \in N$.*

$$\varphi ::= \top \mid p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \Box_i\varphi \mid K_i^E\varphi \mid L_i^E\varphi \mid K_i^S\varphi \mid A_i\varphi$$

Given a language $\mathcal{L}$, $\mathcal{L}|Q$ is the language with the propositional variables restricted to $Q \subseteq P$.

We typically consider languages for subsets of these inductive rules. We write $\mathcal{L}^\Box$ for $\mathcal{L}(\Box, A)$, $\mathcal{L}^E$ for $\mathcal{L}(K^E, A)$, and $\mathcal{L}^S$ for $\mathcal{L}(K^S, A)$, as these three languages are the main focus of our investigations. We assume familiarity with the meaning of propositional constructs, the modal box, and awareness. Implication $\rightarrow$, disjunction $\vee$, equivalence $\leftrightarrow$, and the modal diamond $\Diamond_i$ are defined by abbreviation, as usual. Formula $\Box_i\varphi$ sometimes stands for 'the agent implicitly knows φ', but we also view it as a mere technical background notion. Formula $K_i^E\varphi$ stands for 'the agent *explicitly knows* that φ, $L_i^E\varphi$ stands for 'the agent *explicitly considers possible* that φ. (Explicit knowledge is not the dual of explicit possibility, as both require awareness.) Formula $K_i^S\varphi$ stands for 'the agent *speculatively knows* that φ. Speculative possibility $L_i^S\varphi$ is the dual of speculative knowledge and by abbreviation defined as $L_i^S\varphi$ iff $\neg K_i^S\neg\varphi$. More explanations will be given with the semantics.

Definition 6 (Free variables) *The* free variables *of a formula φ are defined by $v(p) := \{p\}$, $v(\neg\varphi) := v(\varphi)$, $v(\varphi \wedge \psi) := v(\varphi) \cup v(\psi)$ and $v(Y\varphi) := v(\varphi)$, where Y is one of $\Box_i, A_i, K_i^E, L_i^E, K_i^S$.*

Definition 7 (Semantics) *Let (M,s) be an epistemic awareness state, with $M = (S, R, \mathcal{A}, V)$. The non-propositional clauses are*

$$\begin{array}{ll}
(M,s) \models \Box_i\varphi & \text{iff } \forall t \in R_i(s), (M,t) \models \varphi \\
(M,s) \models A_i\varphi & \text{iff } v(\varphi) \subseteq \mathcal{A}_i(s) \\
(M,s) \models K_i^E\varphi & \text{iff } v(\varphi) \subseteq \mathcal{A}_i(s) \text{ and } \forall t \in R_i(s), (M,t) \models \varphi \\
(M,s) \models L_i^E\varphi & \text{iff } v(\varphi) \subseteq \mathcal{A}_i(s) \text{ and } \exists t \in R_i(s), (M,t) \models \varphi \\
(M,s) \models K_i^S\varphi & \text{iff } \forall t \in R_i(s), \forall (M',t') \leftrightarrows_{\mathcal{A}_i(s)} (M,t), \\
 & \qquad (M',t') \models \varphi
\end{array}$$

Model validity $M \models \varphi$ and validity $\models \varphi$ *are defined as usual. The logic (i.e., the set of validities) of language $\mathcal{L}^\Box$ is called $\mathsf{L}^\Box$, the logic of $\mathcal{L}^E$ is L^E, and the logic of $\mathcal{L}^S$ is L^S.*

We will refer to our standard logics as follows:

- $\mathsf{L}^\Box$: the logic of implicit knowledge
- L^E: the logic of explicit knowledge
- L^S: the logic of speculative knowledge

We pay attention to semantic relations between the non-propositional primitives in Section 5. E.g., it is the case that $A_i\varphi \leftrightarrow (K_i^E\varphi \vee L_i^E\neg\varphi)$.

Speculative knowledge is defined in terms of awareness bisimulation: agent i knows speculatively φ at (M,s) iff φ is the case in every epistemic awareness state that is $\mathcal{A}_i(s)$ awareness bisimilar to some state t accessible from s in M.

Speculative and explicit knowledge are different. For example, any agent knows $p \vee \neg p$ speculatively, even if she is unaware of p, because in every possible state $p \vee \neg p$ is true. Nevertheless, the agent only knows $p \vee \neg p$ explicitly when she is aware of p.

Speculative and implicit knowledge are also different. The agent may implicitly know p, but she cannot speculatively know that, because she can speculate about p being false. And if p were false, she cannot know that p.

More convincing examples of speculative knowledge involve dynamics. Suppose that the agent explicitly knows q but is unaware of p. She then speculatively knows: "If p is false then even if I were to become aware of p I cannot explicitly know that p and q are both true." (In the extended logic of Section 7 this is formally $\neg p \rightarrow [\mathsf{A}^{+p}]\neg K_i^E(p \wedge q)$, where $[\mathsf{A}^{+p}]$ is a dynamic modal operator.) But she does not explicitly know that, because she is unaware of p, and p occurs in the formula. For more intuitions, see [20, 21, 22].

Definition 8 (Modal equivalence) *Awareness epistemic states (M,s) and (M',s') are* modally equivalent in a language $\mathcal{L}$ up to $Q \subseteq P$, *notation $(M,s) \equiv_Q^{\mathcal{L}} (M',s')$, if for all $\varphi \in \mathcal{L}|Q$, $(M,s) \models \varphi$ iff $(M',s') \models \varphi$. For $\mathcal{L} = \mathcal{L}^\Box, \mathcal{L}^E, \mathcal{L}^S$ we write for that, respectively, $\equiv_Q^\Box$, $\equiv_Q^E$, and $\equiv_Q^S$.*

Example 2 *Consider the first introductory example. The formula $K_i^E\Box_i p$ is true in (M,s) and false in (M',s). The models are not modally equivalent in the logic $\mathcal{L}^\Box$. But they are modally equivalent in the logic $\mathcal{L}^E$ (without $\Box$), as we will show later.*

3. BISIMILARITY AND MODAL EQUIVALENCE

For the logic of implicit knowledge we have, as expected, that standard bisimilarity implies modal equivalence in $\mathcal{L}^\Box$. Moreover, in the class of image-finite models, modal equivalence in $\mathcal{L}^\Box$ implies standard bisimilarity. (Let (M,s) and (M',s') be epistemic awareness models, and $Q \subseteq P$...)

Proposition 9
$(M,s) \simeq_Q (M',s')$ implies $(M,s) \equiv_Q^\Box (M',s')$.

PROOF. The proof is standard, by induction on φ. The case for formulas of the form $A_i\varphi$ follows from the **aware** clause in Definition 2.

Proposition 10 *On image-finite models:*
$(M,s) \equiv_Q^\Box (M',s')$ implies $(M,s) \simeq_Q (M',s')$.

PROOF. Again, the proof is standard. For proving the **aware** clause, we use modal equivalence with respect to formulas of the form $A_i\varphi$.

Theorem 11 *On image-finite models:*
$(M,s) \simeq_Q (M',s')$ *iff* $(M,s) \equiv^{\Box}_Q (M',s')$.

For the logic of explicit knowledge the correspondence is between awareness bisimulation and modal equivalence in $\mathcal{L}^E$. We recall that awareness bisimulation is a weaker notion than standard bisimulation.

Proposition 12
$(M,s) \leftrightarrows_Q (M',s')$ *implies* $(M,s) \equiv^E_Q (M',s')$.

PROOF. We show the following:

> Let $\varphi \in \mathcal{L}^E$ and let $Q \subseteq P$ such that $v(\varphi) \subseteq Q$.
> Then for any (M,s) and (M',s'), $(M,s) \leftrightarrows_Q (M',s')$ implies that: $(M,s) \models \varphi$ iff $(M',s') \models \varphi$.

In this formulation it is important that φ is chosen before Q, and both φ and Q before the models, so that the inductive hypothesis may be used on a subformula of φ for *another* subset of P than the initial Q (and for any models). Again, the proof goes by induction on φ; all cases are trivial except $K^E_i\varphi$.

Case K^E_i Assume $(M,s) \leftrightarrows_Q (M',s')$, and suppose that $(M,s) \models K^E_i\varphi$ with $v(K^E_i\varphi) \subseteq Q$ (which implies $v(\varphi) \subseteq Q$). By semantic interpretation, $v(\varphi) \subseteq \mathcal{A}_i(s)$ and every state in $R_i(s)$ satisfies φ. First, take any $t' \in R'_i(s')$. By **back** there is a $t \in R_i(s)$ such that $(M,t) \leftrightarrows_{Q \cap \mathcal{A}_i(s)} (M',t')$. But $t \in R_i(s)$ so $(M,t) \models \varphi$. Moreover, $v(\varphi) \subseteq Q$ and $v(\varphi) \subseteq \mathcal{A}_i(s)$ so $v(\varphi) \subseteq Q \cap \mathcal{A}_i(s)$, and then we can use induction hypothesis to get $(M',t') \models \varphi$. Thus, every element of $R'_i(s')$ satisfies φ. Second, $(M,s) \leftrightarrows_Q (M',s')$ implies $Q \cap \mathcal{A}_i(s) = Q \cap \mathcal{A}'_i(s')$, so from $v(\varphi) \subseteq Q \cap \mathcal{A}_i(s)$ we get $v(\varphi) \subseteq Q \cap \mathcal{A}'_i(s')$ and thus $v(\varphi) \subseteq \mathcal{A}'_i(s')$. Hence, from the two parts we get $(M',s') \models K^E_i\varphi$, as needed. The other direction is similar.

Proposition 13 *On image-finite models:*
$(M,s) \equiv^E_Q (M',s')$ *implies* $(M,s) \leftrightarrows_Q (M',s')$.

PROOF. We will show that the relation of modal equivalence in $\mathcal{L}^E$ with formulas built from atoms in Q is a Q awareness bisimulation, i.e., that $\equiv^E_Q$ satisfies Definition 3.

Suppose that $(M,s) \equiv^E_Q (M',s')$.

- **Atoms**. Take any $p \in Q$ and suppose $s \in V(p)$; then $p \in \mathcal{L}^E|Q$ and $(M,s) \models p$ so $(M',s') \models p$, that is, $s' \in V'(p)$. The other direction is similar.
- **Aware**. Take any $i \in N$, and suppose $p \in Q \cap \mathcal{A}_i(s)$; then $p \in Q$ and $p \in \mathcal{A}_i(s)$. From the latter we get $(M,s) \models A_ip$ and therefore $(M',s') \models A_ip$, that is, $p \in \mathcal{A}'_i(s')$. We already had $p \in Q$, so $p \in Q \cap \mathcal{A}'_i(s')$. The other direction is similar.
- **Forth**. Take any $i \in N$, and suppose $t \in R_i(s)$; we want to find a $t' \in R'_i(s')$ such that $(M,t) \equiv^E_{Q \cap \mathcal{A}_i(s)} (M',t')$. We proceed by contradiction, so suppose no element of $R'_i(s')$ is modally equivalent to t with respect to formulas in $\mathcal{L}^E|(Q \cap \mathcal{A}_i(s))$.
  Observe how $R'_i(s')$ is a finite non-empty set: finite because of image-finiteness, and non-empty because $R_i(s) \neq \emptyset$ iff $(M,s) \models L_i\top$, and since $L_i\top \in \mathcal{L}^E|Q$, we should have $(M',s') \models L_i\top$ too. Now, since no element of $R'_i(s')$ is modally equivalent to t with respect to formulas with atoms in $Q \cap \mathcal{A}_i(s)$, then for each $t'_k \in R'_i(s')$ there should be a formula $\varphi_k \in \mathcal{L}^E|(Q \cap \mathcal{A}_i(s))$ that holds at t but fails at t'_k.
  Now define $\varphi := \varphi_1 \wedge \ldots \wedge \varphi_n$ (with n the cardinality of $R'_i(s')$). We have $(M,t) \models \varphi$ because every φ_k is true at t, but also $(M',t'_k) \not\models \varphi$ for every k because each φ_k fails in at least t'_k. Moreover, since $\varphi_k \in \mathcal{L}^E|(Q \cap \mathcal{A}_i(s))$ for every k, we have $\varphi \in \mathcal{L}^E|(Q \cap \mathcal{A}_i(s))$, and hence $v(\varphi) \subseteq Q \cap \mathcal{A}_i(s)$, that is, $v(\varphi) \subseteq \mathcal{A}_i(s)$. Now, from $t \in R_i(s)$, $(M,t) \models \varphi$ and $v(\varphi) \subseteq \mathcal{A}_i(s)$ we get $(M,s) \models L_i\varphi$. But $(M,s) \not\models L_i\varphi$ because no successor of s' satisfies φ. Then, $L_i\varphi$ distinguishes between s and s'. But since $\varphi \in \mathcal{L}^E|(Q \cap \mathcal{A}_i(s))$, we have $L_i\varphi \in \mathcal{L}^E|(Q \cap \mathcal{A}_i(s))$ and hence $L_i\varphi \in \mathcal{L}^E|Q$: this contradicts $(M,s) \equiv^E_Q (M',s')$. Hence, there should be a state $t' \in R'_i(s')$ such that $(M,t) \equiv^E_{Q \cap \mathcal{A}_i(s)} (M',t')$.
- **Back**. Similar to the **forth** clause.

Theorem 14 *On image-finite models:*
$(M,s) \leftrightarrows_Q (M',s')$ *iff* $(M,s) \equiv^E_Q (M',s')$.

Example 3 *The formula* $K^E_i\Box_i p$ *distinguishing the models in the introduction is in* $\mathcal{L}^{\Box}$ *(it is an abbreviation of* $\mathcal{A}_i\Box_i p \wedge \Box_i\Box_i p$*), but it is not in* $\mathcal{L}^E$*. It was unclear until now that it does not have an* $\mathcal{L}^E$ *equivalent. Now it is clear: the models* (M,s) *and* (M',s') *are p awareness bisimilar, and therefore modally equivalent in* $\mathcal{L}^E$*.*

That the language $\mathcal{L}^{\Box}$ of implicit knowledge is aligned with standard bisimulation rather than awareness bisimulation can be seen as a strong argument against the use of this language to specify interactions in epistemic awareness models: it is too rich from the point of view of an agent reasoning about its knowledge and awareness. The language of explicit knowledge $\mathcal{L}^E$ can be seen as its 'explicit' counterpart. Without the aspect of awareness, $\mathcal{L}^{\Box}$ is nothing but the standard multiagent epistemic language, built from the propositional connectives plus operators to talk about what the agent knows and considers possible. Similarly, language $\mathcal{L}^E$ can be seen as (relative to an expressivity result proved in Section 5) built from propositional connectives plus operators to talk about what the agent explicitly knows and explicitly considers possible.

Finally, speculative knowledge. Interestingly, modal equivalence in $\mathcal{L}^S$ for the logic of speculative knowledge is also characterized (on image-finite models) by awareness bisimulation.

Proposition 15
$(M,s) \leftrightarrows_Q (M',s')$ *implies* $(M,s) \equiv^S_Q (M',s')$.

PROOF. See [21, 22].

Proposition 16 *On image-finite models:*
$(M,s) \equiv^S_Q (M',s')$ *implies* $(M,s) \leftrightarrows_Q (M',s')$.

PROOF. Assume $(M,s) \equiv^S_Q (M',s')$; we will show that the relation $\equiv^S_Q$ defines a Q awareness bisimulation linking (M,s) and (M',s'). Clauses **atoms** and **aware** are straightforward; **back** is similar to **forth**.

- **Forth**. We proceed as in Proposition 13. Assume that $t \in R_i(s)$, that the i-successors of t are $t'_1, \ldots, t'_m$ (a finite number), and that none of those is $Q \cap \mathcal{A}_i(s)$ modally equivalent to t. Therefore there are difference formulas $\varphi_1, \ldots, \varphi_m \in \mathcal{L}^S|(Q \cap \mathcal{A}_i(s))$ that are false in $t'_1, \ldots, t'_m$, respectively, and true in t, so that their conjunction $\psi = \bigwedge_{1..m} \varphi_i$ is true in t. This conjunction ψ is also in $\mathcal{L}^S|(Q \cap \mathcal{A}_i(s))$. We now have that $(M,s) \models L^S_i\psi$, as there is an

i-accessible state from s, namely t, and a $Q \cap \mathcal{A}_i(s)$ awareness bisimilar state equivalent to (M,t), namely (M,t) itself, such that $(M,t) \models \psi$. (From $(M,t) \equiv_{Q\cap\mathcal{A}_i(s)} (M,t)$ follows by the definition of the bisimulation that $(t,t) \in \Re[Q \cap \mathcal{A}_i(s)]$.) On the other hand, $L_i^S\psi$ is false in s': clearly, ψ is false in any of the states $t'_1, \ldots, t'_m$ accessible from s', but any $Q \cap \mathcal{A}_i(s)$ modally equivalent state should also not satisfy ψ, as $\psi \in \mathcal{L}^S|(Q \cap \mathcal{A}_i(s))$.

Theorem 17 *On image-finite models:* $(M,s) \leftrightarrows_Q (M',s')$ *iff* $(M,s) \equiv^S_Q (M',s')$.

4. HAVING THE SAME KNOWLEDGE

We can now harvest the benefits from the previous section. We want to characterize when two epistemic awareness states are the same 'from the perspective of an agent', that is, when the agent's knowledge and ignorance is the same in both. This is weaker than being modally equivalent: two epistemic awareness states (M,s) and (M',s') that differ only in a propositional variable p look the same for an agent that is not aware of p in both, and they also look the same for an agent that is aware of p in both but such that the actual state is not accessible. The results for implicit, explicit and speculative knowledge are similar.

Definition 18 (Same knowledge) *Let $Q \subseteq P$, and $N' \subseteq N$. Assume epistemic awareness states (M,s) and (M',s').*

- *(M,s) and (M',s')* describe the same implicit knowledge *up to Q for the agents in N' iff, for every agent $i \in N'$ and every formula $\varphi \in \mathcal{L}^\Box|Q$, $(M,s) \models \Box_i\varphi$ iff $(M',s') \models \Box_i\varphi$.*
- *(M,s) and (M',s')* describe the same explicit knowledge *up to Q for the agents in N' iff, for every agent $i \in N'$ and every formula $\varphi \in \mathcal{L}^E|Q$, $(M,s) \models K_i^E\varphi$ iff $(M',s') \models K_i^E\varphi$, and $(M,s) \models L_i^E\varphi$ iff $(M',s') \models L_i^E\varphi$.*
- *(M,s) and (M',s')* describe the same speculative knowledge *up to Q for the agents in N' iff, for every agent $i \in N$ and every formula $\varphi \in \mathcal{L}^S$, $(M,s) \models K_i^S\varphi$ iff $(M',s') \models K_i^S\varphi$.*

If (M,s) and (M',s') describe the same explicit knowledge for agent i up to (at least) $\mathcal{A}_i(s)$, and $\mathcal{A}_i(s) = \mathcal{A}'_i(s')$, then we can simply say that they describe the same explicit knowledge for agent i; and similarly for speculative knowledge.

To define the same explicit knowledge, we need to refer to both K^E and L^E in the definition (both require awareness). For implicit knowledge and for speculative knowledge the part for the dual diamond version is simply the contraposition of the part for the box version. The 'at least' bit in the final part of the definition is there, because agent i does not explicitly know any formula with variables in $Q \setminus \mathcal{A}_i(s)$, both in s and s'.

Write $(M,s) \leftrightarrows^i (M',s')$ whenever $(M,s) \leftrightarrows_{\mathcal{A}_i(s)} (M',s')$ except for the valuation of atoms in s and s' (i.e., skip clause **atoms** in the root), and except for **back** and **forth** for all other agents than i, in the root. Then this $\leftrightarrows^i$ equivalence class encodes exactly 'what agent i knows in state s'. This works both for explicit knowledge and for speculative knowledge (for implicit knowledge we would require standard bisimulation, but we consider that case of lesser interest).

Proposition 19 *Let (M,s) and (M',s') be image-finite epistemic awareness models, and $i \in N$. Then $(M,s) \leftrightarrows^i (M',s')$ iff (M,s) and (M',s') describe the same explicit / speculative knowledge for agent i.*

PROOF. Directly from Theorem 14, resp., Theorem 17.

This structural characterization of explicit knowledge and speculative knowledge, for a given agent, was an important motivation for our investigation.

5. EXPRESSIVITY

Two models (M,s) and (M',s') can be distinguished in language $\mathcal{L}$ of logic L if there is formula $\varphi \in \mathcal{L}$ that is false in (M,s) and true in (M',s'); φ is called a *distinguishing formula*. A logic L with language $\mathcal{L}$ is at least as expressive as L' with language $\mathcal{L}'$ if all pairs of models distinguishable in $\mathcal{L}'$ are also distinguishable in $\mathcal{L}$. A standard way to prove this, is to show that any formula in $\mathcal{L}'$ is equivalent to a formula in $\mathcal{L}$ (and a trivial case is when $\mathcal{L}' \subseteq \mathcal{L}$), and a standard way to disprove it is to show that some pair of models distinguishable in $\mathcal{L}'$ is indistinguishable in $\mathcal{L}$. A logic L is (strictly) more expressive than a language L', given a class of models, if L is at least as expressive as L' but not vice versa. Instead of expressivity of logics one sometimes talks about the expressivity of languages. The latter is then, of course, relative to a semantics, i.e., it concerns after all a logic.

The expressivity hierarchy is a partial order $<$. We are interested in the relative expressivity of our main logics $\mathsf{L}^\Box$, L^E, and L^S. This is a total order: $\mathsf{L}^\Box > \mathsf{L}^E = \mathsf{L}^S$. Both terms in the equation are of interest. For example, L^E and L^S could just as well have been incomparable. Of further interest is that a number of other logics are equally expressive as $\mathsf{L}^\Box$. As we have a good naming device for languages but not for logics we will henceforth **in this section** talk about expressivity of languages, not logics, and we will write all languages in full, e.g., $\mathcal{L}(\Box, A)$ instead of $\mathcal{L}^\Box$, etc.

Proposition 20 (Equivalence class of $\mathsf{L}^\Box$) *The languages $\mathcal{L}(\Box, A)$, $\mathcal{L}(\Box, K^E)$, $\mathcal{L}(\Box, K^E, A)$ and $\mathcal{L}(\Box, K^E, L^E)$ are equally expressive.*

PROOF. This follows from the following equivalences:

$$\begin{array}{lcl} K_i^E\varphi & \Leftrightarrow & \Box_i\varphi \wedge A_i\varphi \\ L_i^E\varphi & \Leftrightarrow & \Diamond_i\varphi \wedge A_i\varphi \\ A_i\varphi & \Leftrightarrow & K_i^E\varphi \vee L_i^E\neg\varphi \quad \Leftrightarrow \quad K_i^E(\varphi \vee \neg\varphi) \end{array}$$

Proposition 21 (Equivalence class of L^E) *The languages $\mathcal{L}(K^E, L^E)$, $\mathcal{L}(K^E, A)$, $\mathcal{L}(K^E)$ and $\mathcal{L}(L^E, A)$ are equally expressive.*

PROOF. This follows from the following equivalences:

$$\begin{array}{lcl} K_i^E\varphi & \Leftrightarrow & \neg L_i^E\neg\varphi \wedge A_i\varphi \\ L_i^E\varphi & \Leftrightarrow & \neg K_i^E\neg\varphi \wedge A_i\varphi \\ A_i\varphi & \Leftrightarrow & K_i^E\varphi \vee L_i^E\neg\varphi \quad \Leftrightarrow \quad K_i^E(\varphi \vee \neg\varphi) \end{array}$$

Proposition 22 ($\mathsf{L}^\Box > \mathsf{L}^E$) *$\mathcal{L}(\Box, A)$ is more expressive than $\mathcal{L}(K^E, A)$.*

PROOF. Consider the models (M,s) and (M,s') of the first introductory example. We have seen that they are $\{p\}$ awareness bisimilar, and thus by Proposition 12 modally equivalent in $\mathcal{L}(K^E, A)$. On the other hand, $K_i^E\Box_i p \in \mathcal{L}(\Box, A)$ distinguishes between the two models. Hence, $\mathcal{L}(\Box, A)$ is more expressive than $\mathcal{L}(K^E, A)$.

Proposition 23 (Equivalence class of L^E, continued) *$\mathcal{L}(K^S, A)$ and $\mathcal{L}(K^E, A)$ are equally expressive.*

PROOF. To show that $\mathcal{L}(K^S, A)$ is at least as expressive $\mathcal{L}(K^E, A)$ it is enough to show that K^E and L^E are expressible in $\mathcal{L}(K^S, A)$. The following obvious (recursive) definitions are sufficient for this.

$$(K_i^E\varphi)' \stackrel{def}{\equiv} K_i^S\varphi' \wedge A_i\varphi \qquad (L_i^E\varphi)' \stackrel{def}{\equiv} L_i^S\varphi' \wedge A_i\varphi$$

For the converse, to show that $\mathcal{L}(K^E, A)$ is at least as expressive as $\mathcal{L}(K^S, A)$, we require the concept of a uniform interpolant [24]. It has been shown that the modal logic K has the uniform interpolation property, that is, if there is a formula φ whose variables are taken from the union of the disjoint sets of atoms Q and R, then there is a single formula φ^Q such that

1. $\varphi \to \varphi^Q$ is valid.
2. the validity of $\varphi \to \gamma$ implies the validity of $\varphi^Q \to \gamma$ for all formulae γ not containing any atoms from R.

This allows us to define a recursive translation (relative to the set Q of propositional atoms the agent is aware of):

$$(K_i^S\varphi)' \stackrel{def}{\equiv}_Q (K_i^E\varphi')^Q \qquad (L_i^S\varphi)' \stackrel{def}{\equiv}_Q (L_i^E\varphi')^Q$$

The proof of Prop. 23 required the presence of the awareness operator in $\mathcal{L}(K^S, A)$ (but not that of A in $\mathcal{L}(K^E, A)$, given Prop. 21). As speculative knowledge treats unaware atoms as their most general consistent interpretation, there is no semantic difference (with respect to just speculative knowledge) between an agent being unaware of an atom and an agent (speculatively) knowing nothing about it.

The lower end of this expressivity hierarchy is also of theoretical interest but maybe less of practical interest. We have various other results, that are given here without proof. Clearly the propositional language $\mathcal{L}(\emptyset)$ is less expressive than all of $\mathcal{L}(K^E)$, $\mathcal{L}(L^E)$, $\mathcal{L}(\Box)$, and $\mathcal{L}(K^S)$. More interesting is that, although we already established that $\mathcal{L}(K^E)$ is equally expressive as $\mathcal{L}(K^E, A)$, still, $\mathcal{L}(\Box)$, $\mathcal{L}(L^E)$ and $\mathcal{L}(K^S)$ are strictly less expressive than, respectively, $\mathcal{L}(\Box, A)$, $\mathcal{L}(L^E, A)$ and $\mathcal{L}(K^S, A)$. Interestingly, $\mathcal{L}(\Box)$ and $\mathcal{L}(K^S)$ are incomparable. And so on ...

6. AXIOMATIZATION

In this section we present complete axiomatizations for our logics.

Table 1 presents an axiomatization $\mathbf{L}^\Box$ characterizing the validities of the language $\mathcal{L}^\Box$ in epistemic awareness models (the logic $\mathsf{L}^\Box$). This axiomatization is provided in [4], modulo a minor variation (see Section 8).

All propositional tautologies	$A_i\top$
$\top$	$A_i\neg\varphi \leftrightarrow A_i\varphi$
$\Box_i(\varphi \to \psi) \to (\Box_i\varphi \to \Box_i\psi)$	$A_i(\varphi \wedge \psi) \leftrightarrow A_i\varphi \wedge A_i\psi$
From φ and $\varphi \to \psi$ infer ψ	$A_i\Box_j\varphi \leftrightarrow A_i\varphi$
From φ infer $\Box_i\varphi$	$A_iA_j\varphi \leftrightarrow A_i\varphi$

Table 1: Axiom system $\mathbf{L}^\Box$

Theorem 24 (Soundness and completeness) *Axiom system $\boldsymbol{L}^\Box$ is sound and complete for $\mathcal{L}^\Box$ with respect to epistemic awareness models.*

PROOF. Soundness is proved by showing that axioms in $\mathbf{L}^\Box$ are valid and that its rules preserve validity. Completeness is proved by using the canonical model technique in the standard way.

Table 2 presents an axiomatization $\mathbf{L}^E$ characterizing the validities of the language $\mathcal{L}^E$ in epistemic awareness models. A similar axiomatization, but with a different completeness proof, was provided in [9]. See again Section 8 for further discussion.

All propositional tautologies	$A_i\top$
$\top$	$A_i\neg\varphi \leftrightarrow A_i\varphi$
$K_i^E(\varphi \to \psi) \to (K_i^E\varphi \to K_i^E\psi)$	$A_i(\varphi \wedge \psi) \leftrightarrow A_i\varphi \wedge A_i\psi$
$K_i^E\varphi \to A_i\varphi$	$A_iK_j^E\varphi \leftrightarrow A_i\varphi$
From φ and $\varphi \to \psi$ infer ψ	$A_iA_j\varphi \leftrightarrow A_i\varphi$
From φ infer $A_i\varphi \to K_i^E\varphi$	

Table 2: Axiom system $\mathbf{L}^E$

Theorem 25 (Soundness and completeness) *Axiom system $\boldsymbol{L}^E$ is sound and complete for $\mathcal{L}^E$ with respect to epistemic awareness models.*

PROOF. Soundness is proved by showing that axioms in $\mathbf{L}^E$ are valid and that its rules preserve validity. Completeness is proved by using the canonical model technique in the standard way.

Table 3 presents an axiomatization $\mathbf{L}^S$ characterizing the validities of the language $\mathcal{L}^S$ in epistemic awareness models. In axiom * of Table 3, called **KS**, it is required that $p \notin v(\varphi)$.

All propositional tautologies	$A_i\top$
$\top$	$A_i\neg\varphi \leftrightarrow A_i\varphi$
$K_i^S(\varphi \to \psi) \to (K_i^S\varphi \to K_i^S\psi)$	$A_i(\varphi \wedge \psi) \leftrightarrow A_i\varphi \wedge A_i\psi$
$K_i^S\varphi \to (\neg A_ip \to K_i^S\varphi[p\backslash\psi])$ *	$A_iK_j^S\varphi \leftrightarrow A_i\varphi$
From φ and $\varphi \to \psi$ infer ψ	$A_iA_j\varphi \leftrightarrow A_i\varphi$
From φ infer $K_i^S\varphi$	

Table 3: Axiom system $\mathbf{L}^S$

Since the axiomatization for the logic of speculative knowledge is novel, we provide the results in detail.

Theorem 26 (Soundness) *Every theorem of L^S is valid.*

PROOF. This is a quite standard proof, and we only need to examine the axioms and rules involving speculative knowledge. Axiom $K_i^S(\varphi \to \psi) \to (K_i^S\varphi \to K_i^S\psi)$ and the rule of necessitation for K^S are straightforward, and are also found in [21].

Axiom **KS** is new. It says that if an agent speculatively knows a formula despite the formula using an atom of which the agent is unaware, then the agent would continue to know that formula if the atom were replaced with any other formula. This axiom captures the intuition of the speculative knowledge operator, where if an agent is unaware of an atom, the agent must assume the most general interpretation of that atom. In other words, this is according to the semantics for speculative knowledge.

To prove completeness, we use the canonical model technique.

Definition 27 (Canonical model) *The canonical model for $\mathcal{L}^S$ is a tuple $M^c = (S^c, R^c, \mathcal{A}^c, V^c)$ where*

- *S^c is the set of all theories (maximal consistent sets) of $\mathcal{L}^S$;*
- *For any $i \in N$, R_i^c is a binary relation on S^c such that $R_i^c(\Phi, \Psi)$ iff for all φ, $K_i^S\varphi \in \Phi$ implies $\varphi \in \Psi$;*
- *$\mathcal{A}^c : N \to S^c \to P$ is such that $p \in \mathcal{A}_i^c(\Phi)$ iff $A_i p \in \Phi$;*
- *$V^c : P \to S^c$ is such that $\Phi \in V^c(p)$ iff $p \in \Phi$.*

Lemma 28 *For all formulas $\varphi \in \mathcal{L}^S$ and all maximal consistent sets $\Phi \in S^c$, $v(\varphi) \subseteq \mathcal{A}_i^c(\Phi)$ iff $A_i\varphi \in \Phi$.*

PROOF. This follows directly from the definition of a canonical model (Definition 27) and the axioms involving awareness (right-hand side of Table 3).

Lemma 29 *Suppose that (M, s) is image-finite. Let $Th(M, s)$ be $\{\varphi \in \mathcal{L}^S \mid (M, s) \models \varphi\}$. Then $(M, s) \leftrightarrows (M^c, Th(M, s))$.*

PROOF. This follows from the definitions of the canonical model (Def. 27) and awareness bisimulation (Def. 3). Define the relation $B = \{(s, Th(M, s)) \mid s \in \mathcal{D}(M)\}$. It can be easily seen that B satisfies clauses **atoms** and **aware**. For **forth**, if $t \in R_i(s)$, then we have $(t, Th(M, t)) \in B$, and we note that $Th(M, t) \in R_i^c(Th(M, s))$ since if $(M, s) \models K_i^S(\varphi)$, then $(M, t) \models \varphi$. For **back**, if $\Phi \in R_i^c(Th(M, s))$, then for every formula $\varphi \in \Phi$ we must have $(M, s) \models L_i^S\varphi$. Since M is image-finite there must be some $t \in R_i(s)$ such that for all $\varphi \in \Phi$, $(M, t) \models \varphi$. Therefore $(t, \Phi) \in B$ and we are done.

Lemma 30 (Truth) *For all formulas $\varphi \in \mathcal{L}^S$ and all maximal consistent sets $\Phi \in S^c$, $(M^c, \Phi) \models \varphi$ iff $\varphi \in \Phi$.*

PROOF SKETCH. This is shown by induction over the complexity of formulas; we only show the non-trivial case $K_i^S\varphi$.

Suppose $(M^c, \Phi) \models K_i^S\varphi$. Then for all $\Psi \in R_i^c(\Phi)$ and for all $(N, t) \leftrightarrows_{\mathcal{A}_i^c(\Phi)} (M^c, \Psi)$ we have $(N, t) \models \varphi$. Suppose for contradiction that $K_i^S\varphi \notin \Phi$; then there must be some $\Psi \in R_i^c(\Phi)$ such that $\neg\varphi \in \Psi$ (this follows from the maximality of Φ, from propositional reasoning, and the axioms for distribution of K^S over $\to$ and necessitation for K^S). By induction hypothesis, $(M^c, \Psi) \models \neg\varphi$ and, since awareness bisimulation is reflexive, we have the required contradiction, so we must have $K_i^S\varphi \in \Phi$.

Now suppose that $K_i^S\varphi \in \Phi$ and define $Q := v(\varphi) \backslash \mathcal{A}_i^c(\Phi)$. From axiom **KS** we also have that

$$\big(K_i^S\varphi \wedge \bigwedge_{q \in Q} \neg A_i q\big) \to K_i^S\varphi[Q\backslash\overline{\psi}],$$

where $\overline{\psi}$ is any vector of formulas in one-to-one correspondence with Q (so that $[Q\backslash\overline{\psi}]$ stands for simultaneous substitution). So $K_i^S\varphi[Q\backslash\overline{\psi}] \in \Phi$. Now suppose that $\Psi \in R_i^c(\Phi)$, and that (N, t) is any model such that $(N, t) \leftrightarrows_{\mathcal{A}_i(\Phi)} (M^c, \Psi)$.

By Theorem 26, $Th(N, t)$ must be a maximally consistent set. For every atom $p \in Q$, we define a characteristic formula, $\chi(p)$,that is true exactly when p is in the sets reachable from $Th(N, t)$, up to the modal depth of φ. This can be done by taking the intersection of these sets with the closure set of φ (all subformulas of φ and their negations) and the set $\mathcal{A}_i^c(\Phi)$. Applying axiom **KS**, substituting $\chi(p)$ for p we can show that $L_i^S\psi \in \Phi$ for all $\psi \subseteq \varphi$ where $(N, t) \models \psi$. Since $K_i^S\varphi \in \Phi$ for every subtitution of Q, it follows that $(N, t) \models \varphi$ as required. Therefore, $(M^c, \Phi) \models K_i^S\varphi$ as required.

Theorem 31 (Completeness)
Let $\Phi \subseteq \mathcal{L}^S$ and $\varphi \in \mathcal{L}^S$. Then $\Phi \models \varphi$ implies $\Phi \vdash \varphi$.

7. DYNAMICS

7.1 Epistemic awareness action models

Epistemic awareness models represent the information of agents who may be uncertain about the truth of some propositional variables and unaware of others. The information of such agents can change via informational acts. Epistemic awareness action models represent awareness change and knowledge change. They were introduced in [22] for the logic of speculative knowledge. The definition adds a component for awareness to the action models of [1] (and a component for postconditions, as in [23]).

Definition 32 (Epistemic awareness action model) *An* epistemic awareness **action** model *is a tuple* $\mathsf{M} = (\mathsf{S}, \mathsf{R}, \mathsf{A}, \mathsf{pre}, \mathsf{post})$ *where*

- S *is a non-empty set of* actions*;*
- $\mathsf{R} : N \to \mathcal{P}(\mathsf{S} \times \mathsf{S})$ *is an* accessibility function*;*
- $\mathsf{A} : \{+, -\} \to N \to \mathsf{S} \to \mathcal{P}(P)$ *is an* awareness change function*, indicating the disjoint sets of atoms each agent $i \in N$ will become aware (+) and unaware of (-) after the execution of $\mathsf{s} \in \mathsf{S}$;*
- $\mathsf{pre} : \mathsf{S} \to \mathcal{L}$ *is a* precondition function *specifying, for each action $\mathsf{s} \in \mathsf{S}$, the requirement for its execution;*
- $\mathsf{post} : \mathsf{S} \to P \to \mathcal{L}$ *is a* postcondition function *specifying, for each action in $\mathsf{s} \in \mathsf{S}$, how the truth value of each atomic proposition $p \in P$ will change.*

A pair (M, s) with $\mathsf{s} \in \mathsf{S}$ is an epistemic awareness action.

The language $\mathcal{L}$ of the preconditions and postconditions is a fixed parameter of this definition. We write A_i^+ for $\mathsf{A}(+)(i)$ and A_i^- for $\mathsf{A}(-)(i)$.

Example 4 *Particular kinds of epistemic awareness action models can be considered. Some examples:*

- *If A^+ and A^- are both empty, the standard action models for knowledge change reappear.*
- *The singleton epistemic awareness action model with action s accessible to all agents, with trivial precondition and postcondition, and such that $\mathsf{A}_i^+(\mathsf{s}) = \{p\}$ for all agents i, represents 'all agents become aware of p' (without any knowledge change). For this action we write A^{+p}. (Similarly, A^{-p}, for becoming unaware of a variable.)*
- *The singleton epistemic awareness action model that is similar to the previous, but with precondition φ and $\mathsf{A}_i^+(\mathsf{s}) = v(\varphi)$, represents a 'public announcement of a novel issue φ' — all agents become aware of the variables in φ as part of the announcement. For this action we write $!_A\varphi$.*

We can now indicate how an epistemic awareness action model modifies an epistemic awareness model.

Definition 33 (Action model execution) *Let $M = (S, R, \mathcal{A}, V)$ be an epistemic awareness model, and let $\mathsf{M} = (\mathsf{S}, \mathsf{R}, \mathsf{A}, \mathsf{pre}, \mathsf{post})$ be an epistemic awareness action model. The epistemic awareness model $M \otimes \mathsf{M} = (S', R', \mathcal{A}', V')$ – the result of executing M in M – is defined as follows:*

$$S' := \{(s,\mathsf{s}) \mid (M,s) \models \mathsf{pre}(\mathsf{s})\}$$
$$R_i' := \{((s,\mathsf{s}),(s',\mathsf{s}')) \mid s' \in R_i(s) \text{ and } (\mathsf{s},\mathsf{s}') \in \mathsf{R}_i\}$$
$$\mathcal{A}_i'(s,\mathsf{s}) := (\mathcal{A}_i(s) \cup \mathsf{A}_i^+(\mathsf{s})) \backslash \mathsf{A}_i^-(\mathsf{s})$$
$$V'(p) := \{(s,\mathsf{s}) \mid (M,s) \models \mathsf{post}(\mathsf{s},p)\}$$

The new set of states is the restricted cartesian product of S and S: a pair (s,s) is a state in the new model iff s satisfies s's precondition in M. Since the precondition is a formula of a language $\mathcal{L}$, we assume a satisfiability relation $\models$ that evaluates it. For the accessibility relation of the new model, we combine the accessibility relation of the 'static' model and the 'action' model: a state (s',s') is R_i'-accessible from state (s,s) iff s' is R_i-accessible from s, and s' is R_i-accessible from s. For the awareness function of each agent i in each state (s,s), we add to $\mathcal{A}_i(s)$ the atoms in $\mathsf{A}_i^+(\mathsf{s})$ and remove those in $\mathsf{A}_i^-(\mathsf{s})$ (in whatever order, as these sets are disjoint). For the valuation: an atomic proposition p is true at state (s,s) iff s satisfies $\mathsf{post}(\mathsf{s},p)$ in M.

7.2 Language and semantics

Instead of interpreting action models relative to a given logical language, we can also consider the set of action model frames as an additional parameter in an inductively defined language with a clause $[\mathsf{M},\mathsf{s}]\varphi$ (where the precondition of actions should be lower in the inductive hierarchy); this stands for 'after execution of (M,s), φ (is true)'.

Definition 34 (Language) *The language $\mathcal{L}(\otimes)$ extends any $\mathcal{L}$ with an additional inductive clause $[\mathsf{M},\mathsf{s}]\varphi$, where (M,s) is an epistemic awareness action satisfying that: its domain is finite, the postcondition function changes the valuation of only a finite number of atomic propositions, and the awareness function returns two finite sets of atomic propositions. For $\mathcal{L}(\Box,A,\otimes)$ we write $\mathcal{L}^{\Box\otimes}$, for $\mathcal{L}(K^E,A,\otimes)$ we write $\mathcal{L}^{E\otimes}$, and $\mathcal{L}(K^S,A,\otimes)$ we write $\mathcal{L}^{S\otimes}$.*

Definition 35 (Free variables)
An additional inductive clause $v([\mathsf{M},\mathsf{s}]\varphi)$ is defined as

$$v(\varphi) \cup \bigcup_{\mathsf{t}\in\mathcal{D}(\mathsf{M})} v(\mathsf{pre}(\mathsf{t})) \cup \bigcup_{\mathsf{t}\in\mathcal{D}(\mathsf{M}),p\in\mathcal{A}_i^+(\mathsf{t})\cup\mathcal{A}_i^-(\mathsf{t})} (p \cup v(\mathsf{post}(\mathsf{t})(p)))$$

This definition of free variables formalizes that an agent is aware of an action $[\mathsf{M},\mathsf{s}]$ if she is aware of all variables that occur in a precondition or postcondition of an action in the model M. This can be called a conservative stance. For example, the agent can only be aware of $\Box_i p \to [\mathsf{A}^{+p}]K_i^E p$ (if the agent implicitly knows φ, then after becoming aware of p, the agent explicitly knows that p) if the agent is already aware of p before the action. There is much wiggle room here that may also depend on philosophical considerations. For example, alternatively one could call a variable p that occurs in a construct $[\mathsf{A}^{+p}]K_i^E p$ a *closed* variable. The variables that an agent is aware of now would then exclude those that she may become aware of later. We think this stance is conceptually problematic.

Definition 36 (Semantics)
Let $M = (S,R,\mathcal{A},V)$ and $s \in S$.

$$(M,s) \models [\mathsf{M},\mathsf{s}]\varphi \text{ iff } (M,s) \models \mathsf{pre}(\mathsf{s}) \Rightarrow (M \otimes \mathsf{M},(s,\mathsf{s})) \models \varphi$$

The set of validities of language $\mathcal{L}^{X\otimes}$ is called the logic $\mathsf{L}^{X\otimes}$ (for $X = \Box, E, S$).

Example 5 *The dynamic operator $[\mathsf{M},\mathsf{s}]$ is not awareness bisimulation preserving. Consider this: The models (M,s) and (M',s) of the introduction are p awareness bisimilar. And modally equivalent in $\mathcal{L}^E$. But after we make the agent aware of p, they are no longer p awareness bisimilar. The formula $K_i^E K_i^E p$ is now a distinguishing formula. And therefore, $[\mathsf{A}^{+p}]K_i^E K_i^E p$ is true in (M,s) and false in (M',s). So (M,s) and (M',s) are not modally equivalent in $\mathcal{L}^{E\otimes}$.*

The dynamic operator $[\mathsf{M},\mathsf{s}]$ is not awareness bisimulation preserving, but it is standard bisimulation preserving. The proof is similar for all three dynamic logics. In the proof we use modal equivalence in $\mathcal{L}^{X\otimes}$ (for $X = E,S,\Box$ of epistemic awareness states up to Q, denoted by $\equiv_Q^{X\otimes}$, defined analogously to $\equiv_Q^X$.

Proposition 37 *Let $\varphi \in \mathcal{L}^{X\otimes}$, $Q \subseteq P$, and (M,s), (M',s') given. If $(M,s) \simeq_Q (M',s')$, then $(M,s) \equiv_Q^{X\otimes} (M',s')$.*

Proof. The proof is very similar to that in [22] for speculative knowledge. (Theorem 8 in [22] contains an error. It is here corrected.) The difference between implicit, speculative and explicit knowledge plays no role in the inductive case for action models. We only show that case.

Inductive case $[\mathsf{M},\mathsf{s}]\varphi$: Suppose $(M,s) \models [\mathsf{M},\mathsf{s}]\varphi$. Then $(M,s) \models \mathsf{pre}(\mathsf{s})$ implies $(M \otimes \mathsf{M},(s,\mathsf{s})) \models \varphi$. By induction, $(M,s) \models \mathsf{pre}(\mathsf{s})$ iff $(M',s') \models \mathsf{pre}(\mathsf{s})$. The modal product construction in $(M \otimes \mathsf{M})$ is (standard) bisimulation preserving [1]; an easily observable fact when one realizes that pairs in the new accessibility relation require the first argument to be in the accessibility relation in the original model (given $(t,t') \in \mathfrak{R}[Q]$, the induced bisimulation $\mathfrak{R}'[Q]$ on the product is defined as $((t,\mathsf{t}),(t',\mathsf{t})) \in \mathfrak{R}'[Q]$). We now also have to satisfy the **aware** requirement. In the model $M \otimes \mathsf{M}$ the level of awareness $\mathcal{A}_i(t,\mathsf{t})$ is a function of the prior level of awareness $\mathcal{A}_i(t)$ in t and the added or deleted propositional variables $\mathsf{A}_i^+(\mathsf{t})$ and $\mathsf{A}_i^-(\mathsf{t})$. As the prior awareness $\mathcal{A}_i(t)$ is the same in any Q awareness bisimilar state t', and the added or deleted atoms are also the same, the posterior awareness must therefore also be the same for any pairs (t,t) and (t',t) in the Q awareness bisimulation. Therefore, $(M \otimes \mathsf{M},(s,\mathsf{s})) \leftrightarrows_Q (M' \otimes \mathsf{M},(s',\mathsf{s}))$. Now using induction again, we conclude $(M' \otimes \mathsf{M},(s',\mathsf{s})) \models \varphi$, and from that and $(M',s') \models \mathsf{pre}(\mathsf{s})$ we conclude $(M',s') \models [\mathsf{M},\mathsf{s}]\varphi$.

Given the variety of knowledge and awareness changes that can be modelled by epistemic awareness action models, as shown in Example 4, the following is an important theorem. It demonstrates the adequacy of the framework.

Theorem 38 *Let (M,s) and (M',s') be finite epistemic awareness states. Then there is an epistemic awareness action (M,s) such that $(M,s) \otimes (\mathsf{M},\mathsf{s})$ is standardly bisimilar to (M',s').*

Proof. The proof is an extension of the one in [23]. We sketch the proof. *First*, delete the structure of (M,s) by a public announcement of its characteristic formula (as M is finite, this characteristic formula exists [2]). The result is a singleton epistemic awareness state consisting of s only. It does not matter what the valuation is or the level of awareness because, *next*, we execute an epistemic awareness action with precondition true and with the exact structure of the target model (M',s'), using postconditions in actions instead of valuations in states (setting then the value of

propositional variables to the value of the valuation in the corresponding state), and awareness change function in actions instead of awareness functions in states (setting then the level of awareness of propositional variables to that in the corresponding state). This last part on awareness is the extension with respect to [23].

An alternative construction is the straightforward execution in (M, s) of an epistemic awareness action with the structure of the target model (M', s'), and then the result is an epistemic awareness state bisimilar to (M', s') (but typically larger than in the previous construction, it now has size $|M \otimes M'|$ instead of size $|M'|$).

7.3 Axiomatization

We now give the axiomatization of the logic $\mathsf{L}^{\Box\otimes}$. In Table 4 we only give the axioms involving action models. The ones for awareness after actions were presented in [22] and the one for implicit knowledge after action is novel, but has the standard shape of [1]. These are rewrite rules, that allow us to eliminate epistemic awareness action from formulas (given an innermost action model, one pushes it deeper and deeper into a formula until one of the first two axioms can be applied at which moment it has disappeared on the right-hand side). This proves the completeness of the axiomatization and the logic $\mathsf{L}^{\Box\otimes}$ is therefore also equally expressive as the logic of implicit knowledge $\mathsf{L}^{\Box}$.

$[\mathsf{M},\mathsf{s}]\top \leftrightarrow \top$
$[\mathsf{M},\mathsf{s}]p \leftrightarrow \big(\mathsf{pre}(\mathsf{s}) \rightarrow \mathsf{post}(\mathsf{s},p)\big)$
$[\mathsf{M},\mathsf{s}]\neg\varphi \leftrightarrow \big(\mathsf{pre}(\mathsf{s}) \rightarrow \neg[\mathsf{M},\mathsf{s}]\varphi\big)$
$[\mathsf{M},\mathsf{s}](\varphi\wedge\psi) \leftrightarrow \big([\mathsf{M},\mathsf{s}]\varphi \wedge [\mathsf{M},\mathsf{s}]\psi\big)$
$[\mathsf{M},\mathsf{s}]A_i\varphi \leftrightarrow \neg\mathsf{pre}(\mathsf{s})$ if $v(A_i\varphi)\cap \mathcal{A}_i^-(\mathsf{s}) \neq \emptyset$
$[\mathsf{M},\mathsf{s}]A_i\varphi \leftrightarrow (\mathsf{pre}(\mathsf{s}) \rightarrow A_i\varphi[\mathcal{A}_i^+(\mathsf{s})\backslash\top])$ otherwise
$[\mathsf{M},\mathsf{s}]\Box_i\varphi \leftrightarrow (\mathsf{pre}(\mathsf{s}) \rightarrow \bigwedge_{\mathsf{t}\in\mathsf{R}_i(\mathsf{s})} \Box_i[\mathsf{M},\mathsf{t}]\varphi)$
From φ infer $[\mathsf{M},\mathsf{s}]\varphi$

Table 4: Axioms for action models in $\mathbf{L}^{\Box\otimes}$

Proposition 39 *$\boldsymbol{L}^{\Box\otimes}$ is sound and complete.*

Example 6 *To get an idea for the axioms involving awareness after actions, consider $[\mathsf{A}^{+p}]A_ip$. Surely we want the agents to be aware of p after becoming aware of p. The righthand side of this axiom computes to $A_ip[p\backslash\top]$ which is $A_i\top$, a theorem.*

The other axiom applies when the agent becomes unaware. For example, consider $[\mathsf{A}^{-p}]$, which stands for 'the agents become unaware of φ' (not to be seen as gradual fading out, but as conscious abstraction). After that, the agents are no longer aware of p, so $[\mathsf{A}^{-p}]A_ip$ should be false. The righthand side of the axiom is $\neg\mathsf{pre}([\mathsf{A}^{-p}])$. The action $[\mathsf{A}^{-p}]$ is always executable: precondition $\top$. Its negation is therefore the contradiction $\bot$, as desired.

7.4 Expressivity

In this short section we show that the logics $\mathsf{L}^{\Box\otimes}$, $\mathsf{L}^{E\otimes}$, $\mathsf{L}^{S\otimes}$ are all equally expressive (and therefore, as $\mathsf{L}^{\Box\otimes} = \mathsf{L}^{\Box}$, all equally expressive as $\mathsf{L}^{\Box}$).

Proposition 40 *$\boldsymbol{L}^{\Box\otimes}$ and $\boldsymbol{L}^{E\otimes}$ are equally expressive.*

Proof. This follows from the following equivalences (embeddings). The first demonstrates that $\mathsf{L}^{\Box\otimes} < \mathsf{L}^{E\otimes}$ and the second (wherein we use a familiar equivalence, but now within the language $\mathcal{L}^{\Box\otimes}$ instead of $\mathcal{L}^{\Box}$) that $\mathsf{L}^{\Box\otimes} > \mathsf{L}^{E\otimes}$.

$$\begin{array}{lcl} \Box_i\varphi & \Leftrightarrow & [\mathsf{A}^{+v(\varphi)}]K_i^E\varphi \\ K_i^E\varphi & \Leftrightarrow & A_i\varphi \wedge \Box_i\varphi \end{array}$$

Proposition 41 *$\boldsymbol{L}^{E\otimes}$ and $\boldsymbol{L}^{S\otimes}$ are equally expressive.*

Proof. The same argument as in Prop. 23 applies here.

This is an unmistakable though somewhat (we think) surprising result. Even though the logic of implicit knowledge is *more* expressive than the logic of explicit knowledge, the dynamic logic of implicit knowledge is *equally* expressive as the dynamic logic of explicit knowledge. And similarly for speculative knowledge. Example 5 clearly demonstrates the increase of expressive power when dynamics are added: all of a sudden we can distinguish the models (M, s) and (M', s)!

To conclude the picture — and this paper — the axiomatization for the dynamic logic of explicit knowledge is therefore simply the one wherein you write $K_i^E\varphi$ as $\Box_i\varphi \wedge A_i\varphi$ and then derive that in $\mathbf{L}^{\Box\otimes}$. This does not get us the axiomatization for the dynamic logic of speculative knowledge yet, a missing piece in this puzzle, but as the expressivity of this logic is now known, this seems of decidedly minor interest.

8. RELATED WORKS

Our epistemic awareness models are those of [4]. The language used there is $\mathcal{L}(\Box, K^E, A)$, but it has the same expressivity as $\mathcal{L}(\Box, A)$, since $K_i^E\varphi$ is definable as $\Box_i\varphi \wedge A_i\varphi$ (see Proposition 20). The setting of [4] is otherwise different. They assume the accessibility relations to be serial, transitive and euclidean ($KD45$). For the axiomatization one can simply add the characterizing axioms. The complete axiomatization provided there defines awareness A_ip by abbreviation as $K_i^E(p \vee \neg p)$.

Another pertinent investigation is [9]. It focusses on axiomatizations, not on expressivity issues. In [9], Halpern presents axiomatizations for the logics with languages $\mathcal{L}(\Box, A)$, $\mathcal{L}(K^E, A)$ and $\mathcal{L}(K^E)$, for the model class where the ($KD45$) agents also know their own awareness: $t \in R_i(s)$ implies $\mathcal{A}_i(s) = \mathcal{A}_i(t)$. In the axiomatization for $\mathcal{L}(\Box, A)$ we find this as $A_i\varphi \rightarrow \Box_iA_i\varphi$ and $\neg A_i\varphi \rightarrow \Box_i\neg A_i\varphi$. In the axiomatization for $\mathcal{L}(K^E, A)$ this property is, instead, described by an axiom $A_i\varphi \rightarrow K_i^EA_i\varphi$ and a rule Irr.: "If no propositional variables in φ appear in ψ, then from $\neg A_i\varphi \rightarrow \psi$ infer ψ" (with the suggestion that the rule might be derivable from the axiomatization). The rule Irr. is also discussed in [10]. These additional features seem to explain that the completeness proof for the logic of explicit knowledge in [9] is more involved than ours.

The language $\mathcal{L}(K^E)$ is shown in [9] to have the same expressivity as $\mathcal{L}(K^E, A)$ but with the crucial difference that this is on models with euclidean accessibility relations and knowledge of awareness. In such models awareness can be defined in terms of explicit knowledge (as also done in [17]): $A\varphi \leftrightarrow K^E\varphi \vee K^E\neg K^E\varphi$. We recall that in our approach $A\varphi \leftrightarrow K^E(\varphi\vee\neg\varphi)$ (similar to [4], see above), but this equivalence does not hold on the more restricted model class.

Some recent studies on dynamics, such as [12, 8, 19] take a somewhat different approach to awareness, namely syntactic awareness, but employ similar ideas for the dynamics: updates of structures.

9. CONCLUSIONS AND FURTHER WORK

We have the described the logics of implicit, explicit, and speculative knowledge, related modal equivalence in these logics to different forms of bisimulation, compared their expressivity, and provided sound and complete axiomatizations. Then we investigated the dynamics of these logics, where we have shown that any conceivable change of knowledge or awareness can be modelled, we axiomatized the dynamic logic of implicit knowledge, and showed that all three dynamic logics are equally expressive.

Concerning further work, we wish to close some (we think) little gaps. The axiomatization of the logic of speculative knowledge with respect to $S5$ structures is not necessarily an extension of the current axiomatization. This is because the speculative knowledge operator has a built-in quantification over awareness bisimilar structures. Quantifying over structures in a more restricted model class therefore changes the semantics of speculative knowledge; and therefore, also its axiomatic properties. Another little gap is that, even though we know the expressivity of the dynamic logic of speculative knowledge, we do not have (as mentioned above) its axiomatization (with or without the $S5$ restriction).

Further ahead, there are alternative notions of knowledge beyond implicit / explicit / speculative that employ propositional awareness, for example: an agent knows a formula φ in state s iff in all accessible states t, φ is true and the agent is aware of φ (a version explored in [19]). Or consider knowledge employing a recursive version of awareness: agent i is aware of $K_i^E\varphi$ in s iff it is aware of φ in s and aware of φ in all t i-accessible from s. Alternative notions of knowledge would correspond to yet other notions of bisimulation.

The result of Theorem 38 that awareness action models can encode any form of knowledge and awareness change, is very strong. But from another perspective, it is also very weak, because typically only certain protocols or a given and commonly known set of actions are allowed. Investigating the dynamic logics of explicit and speculative knowledge for those settings may be relevant for game theory.

10. ACKNOWLEDGMENTS

We are in debt to the TARK reviewers whose very detailed and enthusiastic comments we have tried to do justice in the final version. We thank Joe Halpern for clarifying a technical detail concerning the relation between our axiomatization and that in his publication [9]. This work was done while Hans van Ditmarsch was employed by the University of Seville, Spain. Hans van Ditmarsch is also affiliated to IMSc, Chennai, as a research associate. Yì Wáng gratefully acknowledges funding support from the Major Project of National Social Science Foundation of China (No. 11&ZD088).

11. REFERENCES

[1] A. Baltag, L. Moss, and S. Solecki. The logic of public announcements, common knowledge, and private suspicions. In *Proc. of 7th TARK*, pages 43–56, 1998.

[2] J. Barwise and L. Moss. *Vicious Circles*. CSLI Publications, Stanford, 1996.

[3] P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic*. Cambridge University Press, Cambridge, 2001. Cambridge Tracts in Theoretical Computer Science 53.

[4] R. Fagin and J. Halpern. Belief, awareness, and limited reasoning. *Artificial Intelligence*, 34(1):39–76, 1988.

[5] R. Fagin, J. Halpern, Y. Moses, and M. Vardi. *Reasoning about Knowledge*. MIT Press, Cambridge MA, 1995.

[6] K. Fine. Propositional quantifiers in modal logic. *Theoria*, 36(3):336–346, 1970.

[7] T. French. *Bisimulation quantifiers for modal logic*. PhD thesis, University of Western Australia, 2006.

[8] D. Grossi and F. Velázquez-Quesada. Twelve angry men: A study on the fine-grain of announcements. In *Proc. of 1st LORI*, pages 147–160. Springer, 2009. LNCS 5834.

[9] J. Halpern. Alternative semantics for unawareness. *Games and Economic Behavior*, 37(2):321–339, 2001.

[10] J. Halpern and L. Rego. Reasoning about knowledge of unawareness. *Games and Economic Behavior*, 67(2):503–525, 2009.

[11] A. Heifetz, M. Meier, and B. Schipper. Interactive unawareness. *Journal of Economic Theory*, 130:78–94, 2006.

[12] B. Hill. Awareness dynamics. *Journal of Philosophical Logic*, 39:113–137, 2010.

[13] J. Hintikka. *Knowledge and Belief*. Cornell University Press, Ithaca, NY, 1962.

[14] M. Hollenberg. *Logic and bisimulation*. PhD thesis, University of Utrecht, 1998.

[15] J.-J. Meyer and W. van der Hoek. *Epistemic Logic for AI and Computer Science*. Cambridge University Press, 1995. Cambridge Tracts in Theoretical Computer Science 41.

[16] S. Modica and A. Rustichini. Awareness and partitional information structures. *Theory and Decision*, 37:107–124, 1994.

[17] S. Modica and A. Rustichini. Unawareness and partitional information structures. *Games and Economic Behavior*, 27:265–298, 1999.

[18] C. Stirling. The joys of bisimulation. In *MFCS '98: Proc. of the 23rd International Symposium on Mathematical Foundations of Computer Science*, pages 142–151. Springer, 1998. LNCS 1450.

[19] J. van Benthem and F. Velázquez-Quesada. The dynamics of awareness. *Synthese*, 177 (Supp-1):5–27, 2010.

[20] H. van Ditmarsch and T. French. Awareness and forgetting of facts and agents. In *Proc. of WI-IAT Workshops 2009*, pages 478–483. IEEE Press, 2009.

[21] H. van Ditmarsch and T. French. Becoming aware of propositional variables. In *Proc. of 4th ICLA*, pages 204–218. Springer, 2011. LNCS 6521.

[22] H. van Ditmarsch, T. French, and F. Velazquez-Quesada. Action models for knowledge and awareness. Proc. of AAMAS Valencia, 2012.

[23] H. van Ditmarsch and B. Kooi. Semantic results for ontic and epistemic change. In *Proc. of 7th LOFT*, Texts in Logic and Games 3, pages 87–117. Amsterdam University Press, 2008.

[24] A. Visser. Bisimulations, model descriptions and propositional quantifiers, 1996. Logic Group Preprint Series 161, Department of Philosophy, Utrecht University.

Bounded rationality in a dynamic alternate game

Eduardo Espinosa-Avila
eespinosa@uxmcc2.iimas.unam.mx
National University of Mexico (UNAM)

Francisco Hernández-Quiroz
fhq@ciencias.unam.mx
National University of Mexico (UNAM)

ABSTRACT

From the standpoint of game theory, dominoes is a game that has not received much attention (specially the variety known as *draw*). It is usually thought that this game is already solved, given general results in game theory. However, the determination of equilibria is not feasible for the general case because of the well known problem of node explosion in the tree expressing the game. We propose a new model based in limited forecast as a kind of bounded rationality for dynamic alternate games.

1. INTRODUCTION

There are a lot of possible games to play with dominoes tiles. The variety we analize here is known as *Draw*, which is part of a bigger group called *blocking games*. In this kind of games, tiles are initially randomly distributed among all players and the one with the biggest *double* (a tile with the same number in both sides) draws it on the table, starting the *game train*. Afterwards, players draw tiles alternately matching a *free-end*. The game ends when one player draws all of her tiles or the game is *blocked*, which happens when none of the players can draw a matching tile. At the end of the game the sum of points in the tiles of the other players are the winner's profit.

Even though there are several theoretical results proposing a solution to similar games, we think this is an interesting problem because in practice it is not computationally possible to apply these results to the general (i.e. with an arbitrary number of points). Also the node explosion problem makes necessary to find alternative techniques to compute the game equilibria and to determine the best strategy in order to get the best profit in the match.

We define a model of the game considering limitations on forecast as referred in [1, 4, 10], but in our approach limitations are not fixed anf instead they are a function of some bounded optimization parameter [5].

2. PREVIOUS MODEL

Philippe Jehiel [1] presents a model of limited horizon forecast applied to repeated alternate games. The key features of this class of games are that there are two players moving sequentially in discrete time steps. In each period t, the current payoff for player i depends on her own action chosen in that time and the action made by the opponent in the last period. The action spaces are finite and remain the same throughout the match.

It is assumed that each player has a *limited ability to forecast* the future. Player i is characterized by the lenght of her forecast n_i (a constant). At period t, player i formulates predictions for the forthcoming n_i moves after her own move. Therefore, she must make her choice of the current action on the basis of her limited forecast *only*. This is because:

1. Player i cannot build her criterion on what will come after n_i periods, since she cannot make predictions about (she has no idea of) it, and,
2. Given the stationarity of the game, the average payoff over the lenght of foresight may be perceived as a good approximation of the true objective function.

Jehiel also defines a solution concept called $(n_1, n_2)-solution$ which requires two preliminary notions:

1. A strategy for player i is *justified* by a sequence of forecasts if the strategy provides actions that maximize the average payoff obtained over the lenght of her forecast, and,
2. A sequence of forecasts for player i is *consistent* with a strategy profile if the forecast coincide with the truncation to the first n_i actions of the respective actions of the respective continuation paths induced by the strategy profile.

Hence a $(n_1, n_2) - solution$ is defined as a strategy profile that can be justified by consistent sequences for players 1 and 2. In other words, in a $(n_1, n_2) - solution$:

1. Current actions are chosen to maximize the average payoff over the lenght of her foresight, and,
2. At any period t where player i must move, her forecasts for the forthcoming n_i actions as a function of her current action are correct.

It should be mentioned that the predictions for the forthcoming n_i actions include her own actions and that the equilibrium forecasts about all these actions are assumed to be correct whatever her current action and not only on the equilibrum path.

Another important issue mentioned by Jehiel is the fact that there is no improvement by incrementing the forecast for player, since the game is cyclic.

3. OUR MODEL

In the case of dynamic alternate games, like dominoes, there are substantial differences:

♢ While the movements are alternated, the game is dynamic, meaning that the action space is updated after each move and the search space is reduced.

♢ According to the classification presented in [2], dominoes is a convergent, imperfect information and sudden death game. Given this, the length of the forecast is not constant and forecasting can be better based upon a function of the computing capability of the agent[1].

♢ Since in games with similar nature to dominoes subgame perfect equilibrium can be applied, a reasonable[2] approach to the solution concept could be *subgame perfect equilibrium with limited forecast*, wich would have to compute at each period a new equilibrium according to some desired benefit (payoff function).

♢ A problem in applying the bounded forecast to the game of chess for instance, is the difficulty of determining a *reasonable function* to estimate the payoff at the end of the horizon bounded by the forecast. In the case of dominoes we can use heuristics or *guidelines* known in the folklore of the game to determine this payoff function [7].

In order to develop the model, it is important to have a clear notion of awareness. Therefore, the first step is to answer the questions raised in [3] for dominoes:

♢ Awareness of what? The player must be aware of the actions made up to the current period of the game, as well as the tiles she holds. In addition, she must be aware of the actions she might take in her turn.

♢ What is the environment? The environment consists of the current state of the game: how many tiles each player holds and the game train. A query to the environment consists of trying to reconstruct the history of the match using the current turn and following the match train.

♢ What is the enumeration process? The acquisition of the set of possible actions is made by touring the decision tree of the match as the match evolves. This path can return a set or a particular state. However, building the entire tree requires exponential space.

♢ What is the decision making process? Once the enumeration returns a state or a set, she can select the best possible action from among its outgoing edges by following the subgame perfect equilibrium.

Now we define a model of limited horizon forecast as a function of the computing capability of the agent applied to *dynamic alternate games*. The key feature of this class of games is that there are two players moving sequentially in discrete time steps. In each period t, the current expected payoff for player i depends on her own action chosen in that time and the action made by the opponent in the previous period. The action spaces are finite and the search space is reduced as the match evolves.

[1] *Computing capability* in this context indicates the ability to generate and visit a certain number of nodes in the future.
[2] *Reasonable* in this context is used as a synonym for common sense.

The latter ensures that in the final steps of the match, the number of nodes is very small and can be determined in reasonable time. In other words, the number of leaves is very small compared to the number of branches at the beginning of the match.

We assume that each player has a *limited ability to forecast* the future. Player i is characterized by her *ability to generate and visit states* in the future c_i. At period t, player i formulates predictions for the forthcoming $n_i = f(c_i)$ moves after her own move. Therefore, she must make her choice of the current action on the basis of her limited forecast *only*. This is because:

1. Player i cannot build her criterion on what will come after n_i periods, since she cannot make predictions about (she is not aware of) it, and,

2. The subgame perfect equilibrium payoff over the lenght of foresight may be perceived as a good approximation of the true objective function.

3.1 Dynamic alternate games

We consider two players indexed by $i = 1, 2$; player i takes actions a_i from a finite action space A_i. Players take actions in discrete time and the horizon is finite. Time periods are indexed by $t = 1, 2, 3, ...$ At time t player i's period payoff is a function of the current actions a_i^t of the two players $i = 1, 2$.

Players take actions sequentially and player 1 moves first. At each odd period ($t = 1, 3, 5, \ldots$), player 1 choses an action from her set. Similarly, player 2 choses her actions at each even period ($t = 2, 4, 6, \ldots$). In both cases, the action taken modifies the immediate next action of the opponent and reduces the search spaces of both players. We call games like this *dynamic alternate games*.

A stream of action profiles $\left\{q_i^t\right\}_{t=1}^{n_{max}} = \left\{q_1^{2k-1}, q_2^{2k}\right\}_{t=1}^{n_{max}}$, where $q_1^{2k-1} \in A_1$ and $q_2^{2k} \in A_2$ is known as a path and is denoted by Q. Since players move each two periods, an action taken at period t is combined with the action taken by the opponent in the last period $t-1$ to modify the structure of the game tree (they prune it) and therefore, the payoff of player i induced by path Q.

Notation

1. Let R_n be an arbitrary n-length stream of actions of alternate actions; $\phi_i(R_n)$ denotes a function that, given the current state for player i, returns the expected payoff to player i induced by R_n. This function considers both the rules of the game and/or *guidelines* known from the game in question.

2. $[Q]_n$ denotes the truncation of path $Q = \left\{q_i^t\right\}_{t=1}^{n_{max}}$, where $n \leq n_{max}$, to the first n actions.

3. $[q]^N$ denotes the truncation of path q to the last N actions.

4. (q, q') denotes the concatenation of $q = \left\{q_i^t\right\}_{t=v}^{s}$ with $q' = \left\{q_i^t\right\}_{t=s+1}^{w}$: $(q, q') = \left\{q_i^t\right\}_{t=v}^{w}$.

3.2 The solution concept

Similar to Jehiel [1], we assume that players have a *limited ability to forecast* the future and *bounded recall*; however, unlike his proposal, forecasting ability in our model is not

fixed, but a function of the ability of the agent to generate and visit future states in the game tree. The idea of having units of brain power to study the future and partly to the analysis of the past is maintained, but in the case of the units dedicated to the future they are intended to dynamically compute the next possible branches. Therefore, Player i has a two-dimensional ability, on the one hand N_i represents her memory capacity and, on the other side $n_i = f(c_i)$ is the number of steps that player i is able to foresight, as a function of her ability to generate and visit. At each period where player i must take an action, she determines new forecasts about the future. Her forecasts are limited to the next n_i steps. Additionally, as she has bounded memory, her forecasts about the future may only depend on the last N_i periods and her current action.

Notation and auxiliary definitions

Let $\mathcal{H}(N_i)$ be the set of histories of alternate actions of length N_i, in which last action is an element of A_j $(j \neq i)$ and h an arbitrary element of $\mathcal{H}(N_i)$.

1. An n_i-length prediction, where $n_i = f(c_i)$, for player i is a stream of alternate actions of length n_i, starting with an action in A_j $(j \neq i)$. The set of n_i-length predictions (shorter in the last steps of the match) is denoted by P_{n_i} (a subtree).

2. An n_i-length forecast for player i at a period t where she must move is denoted by f_i^t. It maps, for every history of length N_i, $h \in \mathcal{H}(N_i)$, the set of actions A_i available for the set of predictions P_{n_i}. Formally, $f_i^t = \left\{f_i^t(\cdot|h)\right\}_h$, where, $\forall h \in \mathcal{H}(N_i), f_i^t(\cdot|h) : A_i \to P_{n_i} : f_i^t(a_i|h)$ is the prediction about the forthcoming n_i actions made by player i at period t if she currently choses a_i given the last N_i actions $h \in \mathcal{H}(N_i)$.

3. $f_i = \left\{f_i^t\right\}_t$denotes an arbitrary sequence of forecasts f_i^t for every period t where player i must move. The set of f_i is denoted by $\mathcal{F}_i$. A pair $(f_1, f_2) \in \mathcal{F}_1 \times \mathcal{F}_2$ is denoted by$f$ and $\mathcal{F}$ denotes the set of f.

4. A pure strategy for player i is denoted by σ_i. It is a sequence of functions σ_i^t, one for each period t where player i must take an action. The function at period t, σ_i^t, is the behavior strategy for player i at that period. It determines player i's action at period t as a function of the last N_i actions. A strategy profile (σ_1, σ_2) is denoted by σ and the set of strategy profiles $\Sigma_1 \times \Sigma_2$ is denoted by Σ.

Any strategy profile $\sigma \in \Sigma$ generates a path $Q(\sigma) = \left\{q_i^t(\sigma)\right\}_t$, $i = 1$ if t is odd and $i = 2$ if t is even. Let $\mathcal{H}^t$ be the set of histories of alternate actions of length t and let h^* be an arbitrary history of length $t-1$, i. e., $h^* \in \mathcal{H}^{t-1}$. The strategy profile and the induced path by σ on the subgame h^* are denoted by $\sigma|_h$ and $Q(\sigma|_h)$ respectively. Given $h^* \in \mathcal{H}^{t-1}$ and an action $a_i \in A_i$ at period t, the continuation path induced by σ after (h^*, a_i) is thus $Q(\sigma|_{h^* a_i})$. The set of continuation paths at period t, referred as the continuation set, is denoted by $Q^t(\sigma) = \{(a_i, Q(\sigma|_{h^* a_i}))\}_{h^* a_i}$. The sequence of continuation sets $Q^t(\sigma), t = 1, 2, \ldots$ is denoted by $\hat{Q}(\sigma) = \left\{Q^t(\sigma)\right\}_t$.

The key idea in this construction is that the strategies of player i, (i. e. her choices of actions), are based on her forecast, limited by her computing capability. Hence, to define the solution concept it is necessary to (1) specify a criterion based on forecast as a function of computing capability and (2) show how equilibrium forecasts are related to equilibrium strategies.

The criterion for calculating the payoff is to determine the largest profit among all the *branches* in the subtree she can see by applying subgame perfect equilibrium. Such a criterion is natural, given the features of this class of games. In the early stages (where the player cannot see the horizon of the match) this criterion will indicate what action will give her the best profit even if this profit may not be the best in the whole game tree. In the final stages (where player can see all subtrees from the current node) the agent can determine the *actual equilibrium* and take the *best actual action* according to the subgame perfect equilibrium. In other words, player assumes she can see the whole tree and based on this assumption, she can calculate a subgame perfect equilibrium at every turn. Formally:

Definition 1. A strategy $\sigma_i \in \Sigma_i$ is *justified* by a sequence of forecasts $f_i = \left\{f_i^t\right\} \in \mathcal{F}_i$ if at each stage player i chooses the action that delivers the largest profit by applying subgame perfect equilibrium to the game tree produced by her forecast limited by her computing capability.

We next assume that player i's equilibrium forecasts are related to equilibrium strategies by a *consistency relationship*, defined as follows. Given a history h^* of length $t-1$, and any action a_i at the current period t, $Q(\sigma|_{h^* a_i})$ is the continuation path induced by σ after $(h^* a_i)$. At period t, the N_i last actions are $h = [h^*]^{N_i}$. Consistency requires for every (h^*, a_i), the forecast $f_i^t(a_i|h)$ coincides with the truncation to the first $n_i = f(c_i)$ actions of the continuation path induced by $\sigma, [Q(\sigma|_{h^* a_i})]_{n_i}$. In other words, consistency means that forecasts are correct on and off the equilibrium path. Formaly:

Definition 2. $f_i = \left\{f_i^t\right\} \in \mathcal{F}_i$ is *consistent* with $\sigma \in \Sigma$ if for every period t where player i must move: $\forall a_i \in A_i$, $\forall h^* \in \mathcal{H}^{t-1}$, $f_i^t(a_i|h) = [Q(\sigma|_{h^* a_i})]_{n_i}$ with $h = [h^*]^{N_i}$.

Now define the solution concept. A $(c_1, c_2) - solution$ is a strategy profile that can be *justified* by *consistent* forecasts for players 1 and 2, i. e., a strategy profile that is associated with sequences of forecasts such that (1) players choose their actions in order to maximize the payoff received by applying the subgame perfect equilibrium over the length of her current forecast and (2) player i's forecasts for the forthcoming $n_i = f(c_i)$ actions after her own move are correct on and off the equilibrium path. Formally:

Definition 3 The solution concept. A strategy profile $\sigma = (\sigma_1, \sigma_2) \in \Sigma$ is a *subgame perfect equilibrium with limited forecast* $((c_1, c_2) - solution)$ if and only if there exist sequences of forecasts $f = (f_1, f_2) \in \mathcal{F}$ such that for $i = 1, 2$.

1. σ_i is *justified* for f_i and

2. f_i is *consistent* with σ.

Similarly to Jehiel, we do not provide justification for why forecasts should be correct in equilibrium, but in [8] Jehiel discusses a learning process based on limited predictions such that players eventually learn to have correct forecasts.

Hence, players eventually behave as in $(c_1, c_2) - solution$.

3.3 Construction

Given a forecast f_i^T of player i at period T, if f_i^T is associated with a $(c_1, c_2) - solution$, is it possible to derive f_i^{T-1} backward on the sole basis of f_i^T? It is not possible, because player i generates subtrees at each period T where she must move, hence the opponent cannot know in advance which tree she will generate (since it depends on her computing capability). There are two ways in wich the opponent may foresee all the moves down to the leaves: if she has exponential capability and when she is *sufficiently near* to the horizon of the match; if this is the case, she will be able to determine a subgame perfect equilibrium.

Hence in dynamic alternate games the construction is performed forward, but in each period T where player i moves, she must apply backward induction to calculate the subgame perfect equilibrium over the generated subtree. This construction is similar to the *forward looking procedure* introduced in [9].

As stated above, on each step player i generates a set of predictions P_{n_i} of length (depth) $n_i = f(c_i)$, this set is the subtree computed at the current period. Given P_{n_i} we apply a function that returns a payoff for each outgoing edge from the current node to the *leafs* of each prediction. When player i cannot see the *real* horizon of the game from the current node this function estimates the gains from the rules and/or *guidelines* known from the game in question. Once the player can see the whole subtree starting from the current node, the function returns the payoff for each of the leafs. Therefore the player can obtain a series of *subgame perfect equilibria* wich corresponds with a $(c_1, c_2) - solution$.

Moreover, this function is useful to make the best choice at each time step and to generate a sequence of subtrees $\mathcal{P}_{n_i}$. Each $P_{n_i} \in \mathcal{P}_{n_i}$ meets that $length(P_{n_i}) \leq length(P_{n_{i+1}})$ where n_i is such that the current player cannot see the leafs. Once the player is able to observe the entire subtree from any node, the relationship is reversed $length(P_{n_i}) \geq length(P_{n_{i+1}})$, since the number of nodes in the last steps of the game tree is much lower than in the early stages due to the considerable reduction of the search space as the game progresses.

3.4 Properties

A dynamic alternated game always has at least one subgame perfect equilibrium with limited forecast. By applying a Kuhn's Corollary of the Zermelo-von Neumann's Theorem [10], we guarantee the existence of a subgame perfect equilibrium for each subtree that player i may generate, hence we may construct a subgame perfect equilibrium with limited forecast by concatenating the equilibria calculated in each period.

Equilibrium forecasts associated with $(c_1, c_2) - solution$ are history independent, as a decision made in the current period is taken on the sole basis of the last action (taken by the opponent).

3.5 When a player is better off with a larger foresight

Given the nature of dynamic games, the search space is reduced at each period and since the game is finite (of sudden death), contrary to the model developed by Jehiel [1], in this class of games player i gets advantage by having larger computing capability and, therefore a larger foresight.

Compared to a completely random player, this model behaves better. At any stage, while a purely random player chooses her actions completely at random within the range of possible options, a player implementing the model presented here can take into account both knowledge and preferences of the player and other players (the guidelines).

On the other hand, this model requires less computational power than that of "rational man". Endowed with polynomial capability, the player's behaviour approaches that of the rational man as the game advances.

3.6 Example

To illustrate the usefulness of the model, we now present how to apply it to a small instance of dominoes. As stated above, the problem with games like chess is the difficulty to define a payoff function. However, in games like dominoes we may use basic guides like those shown in [7] to define the expected payoff function ϕ. Some of them can easily be adapted to the case of individual games.

We consider an instance of players $\{1, 2\}$ with 6 tiles (each player gets 3 at the beginning). To show a concrete example, player 1 gets tiles $\{(0,0), (0,1), (2,2)\}$ and player 2 $\{(0,2), (1,1), (1,2)\}$; additionally, the computing capability for each player is $2n$ with n the number of tiles each player gets at the beginning. To simplify the example we will consider the guide of drawing the tile with higher value. The main goal of the match is to win by drawing all the tiles she possesses first and if she cannot win in the current period she will apply the guide mentioned above. It should be noticed that when nodes generated by an agent do not cover a full level, we consider the she does not possesses any information about the truncated level, as we cannot know what set of nodes she will generate.

The game described above generates the game tree shown in figure 1. We observe that if player 1 draws the tile $(0, 0)$ in her first turn, there exists a path that gives her a gain of 3. However, as she has limited forecast she is not aware of this possibility. At the beginning the number of levels player 1 may visit is $n_1 = 1$; therefore she evaluates her profit with the guide of "getting rid of heavier tokens" and she draws tile $(2, 2)$.

Figures show in boxes the levels that player i can generate and visit at each turn and in light gray the branches cannot be played. Figure 3 shows that player 2 can foresee $n_2 = 2$ levels down.

Now player 2 has two options: draw tiles $(0, 2)$ and $(1, 2)$. If she draws tile $(0, 2)$, she might win the match; but as she is not aware of it she chooses to draw tile $(1, 2)$, the highest.

At this point, figure 3 shows that player 1 can only draw tile $(0, 1)$. She can see the complete rest of the game tree $n_1 = 4$ and she is aware she will win, getting a gain of 2.

Figure 4 shows the final stages of the match. Player 2 must draw tile $(0, 2)$ but she can leave free-end $[0, 0]$ or $[2, 2]$ in both cases she loses. She chooses to leave $[2, 2]$ in order to *lock* the tile $(0, 0)$ of player 1.

This example shows that the length of the forecast grows as the game evolves because the number of future states is reduced at each period. This feature together with the intrinsic finiteness of the game makes feasible to build a subgame equilibrium (and strategies) that *gets closer* to the perfect at each step.

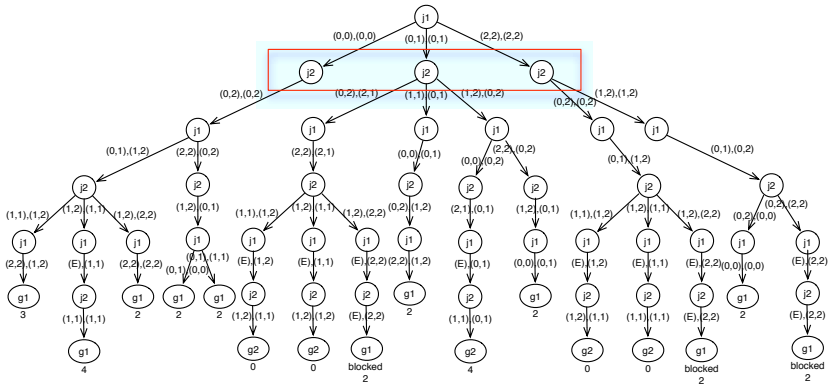


Figure 1: Full game tree

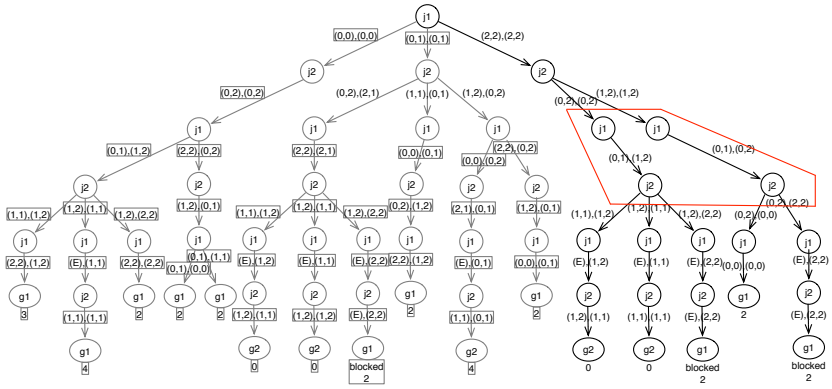


Figure 2: Game tree after the first turn of player 1

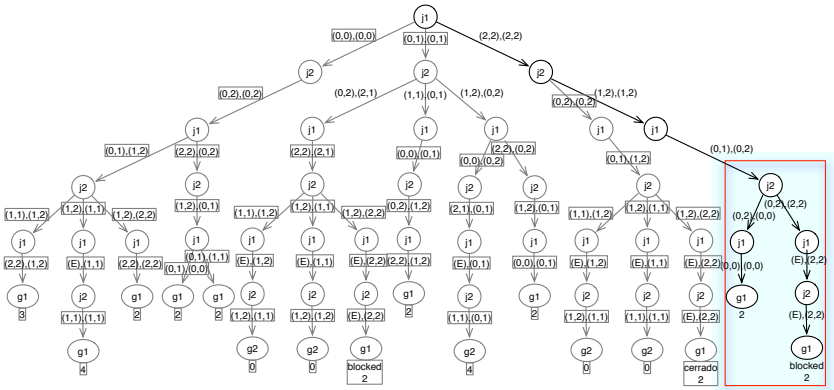


Figure 3: Game tree after the first turn of player 2

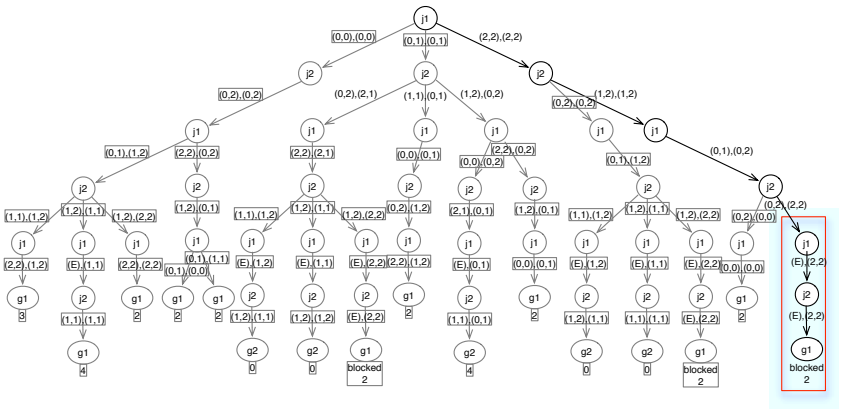


Figure 4: Game tree in the final stages of the match

4. CONCLUSIONS AND FURTHER WORK

In this paper we have proposed a model of limited foresight as a function of the computing capability of an agent applied to the class of dynamic alternate games. We also define a solution concept called subgame perfect equilibrium with limited forecast which uses the well known subgame perfect equilibrium for each subtree that player builds at each period of the game.

Additionally, we provide a couple of properties for the class of games as well as the solution concept. Finally, we show a concrete example of the model applied to an instance of the game dominoes in order to show its applicability.

We intend to extend the results in this paper in the following directions:

- ♢ Currently we seek to prove that our model performs better than a purely random one.
- ♢ Develop a generic mathematical model applicable to a class of games, not only dominoes.
- ♢ Derive other interesting properties about the solution concept.
- ♢ Present concrete examples on how to use the model, initially by applying it to dominoes adding several guides and then with other dynamic games.
- ♢ Obtain models that combine other characterizations of bounded rationality.

5. REFERENCES

[1] Jehiel, P. "Limited Horizon Forecast in Repeated Alternate Games." Journal of Economic Theory 67, 497–519, 1995.

[2] "Searching for Solutions in Games and Articial Intelligence", Allis, L. Victor, Phd Thesis, 1994, University of Limburg, http://fragrieu.free.fr/SearchingForSolutions.pdf.

[3] A Computational Theory of Awareness and Decision Making, Nikhil R Devanur and Lance Fortnow, Proceedings of TARK XII (2009), Stanford, California.

[4] Herbert A. Simon, A Behavioral Model of Rational Choice, The Quarterly Journal of Economics, Vol. 69, No. 1. Feb., 1955, pp. 99-118.

[5] Rubinstein, A. Modeling Bounded Rationality. MIT Press, 1998.

[6] Russell, S., and Subramanian, D. Provably bounded-optimal agents. Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence, 1993.

[7] Tejeiro-Arias, G. A. "How to play latin partnership dominoes" Hats Off Books, 2001.

[8] Jehiel, P. "Learning to play limited forecast equilibria." mimeo C. E. R. A. S., 1995.

[9] Mirrokni, V., Thain, N. and Vetta, A., On the Implications of Lookahead Search in Game Playing on ArXiv e-prints. Feb., 2012. http://adsabs.harvard.edu/abs/2012arXiv1202.4134M.

[10] Kuhn, H. W. "Extensive Games and the Problem of Information", pp 46-68 in Classics in Game Theory, Princeton University Press, 1997.

Appendix

Kuhn's Corollary of the Theorem of Zermelo-von Neumann

A general n-person game Γ with perfect information always has an equilibrium point in pure strategies [10].

Basic strategies for dominoes

Here are descriptions of basic moves and strategies [7].

1. **Commanding/Leading a Strong Number.** Generally a player should lead a strong number with the objective of playing it later in the game. A strong number is a number that occurs often in a player's hand.
2. **Indicating/Showing the Number of the Double Tile.** A player should command a number that includes any respective double tile in her hand, so that her partner knows her most difficult tiles to play.
3. **Hitting/Blocking a Number Commanded by the Opposition.** When a player blocks a number led by the opposition.
4. **Leaving a Number Open.** When a player avoids drawing a number that he has been trying to play.
5. **Repeating a Number.** A player should repeat a strong number.
6. **Taking Care of the Hand.** When a player avoids being void at a given number.
7. **Avoid Leading an Orphan Number.** Generally, leading an orphan number should be avoided because the player who does so is providing inaccurate information to her partner. An orphan number is a number that occurs only once in a hand.
8. **Protecting Your Partner/Avoiding a Possible Pass.** If a player does not have the relative control of her couple, she should avoid forcing her partner to pass on her next turn.
9. **Indicating/Showing Your Type of Hand.** Each player should try to show the value of her hand (low or high) so that her partner knows the tiles she should try to play.
10. **Stealing the Game.** When a player does not have the relative control of her couple, she should hit a number commanded by her partner with her own strong number.
11. **Playing Aggressively (for Low Hands).** A player should try to play in a manner that high tiles cannot be played if she feels that her couple can win the game or she does not have a high hand with at least one high double hand.
    From this strategy comes the first objective of dominoes: try to win by getting the **highest** amount of points.
12. **Playing Conservatively (for High Hands).** A player should try to play in a manner that high tiles are played if she feels that her couple cannot win the game or she has a high hand with at least one high double hand.
    From this strategy comes the second objective of dominoes: try to lose by giving the **lowest** amount of points.
13. **Playing to Accumulate Points.** The couple has the option to play aggressively (for low hands) if the score is not close to the upper limit of points.
14. **Playing not to Accumulate Points.** The couple should play conservatively (for high hands) if the score is close to the upper limit of points. The definition of a "close" score is subjective and depends on the player's appraisal.

Universal Interactive Preferences

[Abstract]

Jayant V. Ganguli
Department of Economics
University of Essex
jayantvivek@gmail.com

Aviad Heifetz
Department of Economics and Management
Open University of Israel
aviadhe@openu.ac.il

1. ABSTRACT

We prove that a universal preference type space exists under much more general conditions than those postulated by [1] for a large class of preferences beyond [4]. To wit, it is enough that preferences can be encoded by a countable collection of continuous functionals, while the preferences themselves need not necessarily be continuous or regular, like, e.g., in the case of lexicographic preferences. The proof relies on a far-reaching generalization of a method developed by [3]. The full statements and proofs are provided in [2].

2. ACKNOWLEDGEMENTS

We thank Simone Cerria-Vioglio, Yi Chun-Chen, Sander Heinsalu, Fabio Maccheroni, Miklos Pinter, Amanda Friedenberg, and the TARK XIV program committee for helpful comments and discussions. The usual disclaimer applies.

3. REFERENCES

[1] L. Epstein and T. Wang. 'Beliefs about beliefs' without probabilities. *Econometrica*, 64:1343–1373, 1996.

[2] J. V. Ganguli and A. Heifetz. Universal interactive preferences. mimeo, http://ssrn.com/abstract=2174371, 2012.

[3] A. Heifetz and D. Samet. Topology-free typology of beliefs. *Journal of Economic Theory*, 82:324–341, 1998.

[4] L. J. Savage. *The foundations of statistics*. John Wiley and Sons, 1954.

TARK 2013, Chennai, India.

Timely Common Knowledge

Characterising Asymmetric Distributed Coordination via Vectorial Fixed Points

Yannai A. Gonczarowski
Einstein Institute of Mathematics and
Center for the Study of Rationality
Hebrew University of Jerusalem
Jerusalem 91904, Israel
yannai@gonch.name

Yoram Moses
Department of Electrical Engineering
Technion—Israel Institute of Technology
Haifa 32000, Israel
moses@ee.technion.ac.il

ABSTRACT

Coordinating activities at different sites of a multi-agent system typically imposes epistemic constraints on the participants. Specifying explicit bounds on the relative times at which actions are performed induces combined temporal and epistemic constraints on when agents can perform their actions. This paper characterises the interactive epistemic state that arises when actions must meet particular temporal constraints. The new state, called *timely common knowledge*, generalizes common knowledge, as well as other variants of common knowledge. While known variants of common knowledge are defined in terms of a fixed point of an epistemic formula, timely common knowledge is defined in terms of a *vectorial* fixed point of temporal-epistemic formulae. A general class of coordination tasks with timing constraints is defined, and timely common knowledge is used to characterise both solvability and optimal solutions of such tasks. Moreover, it is shown that under natural conditions, timely common knowledge is equivalent to an infinite conjunction of temporal-epistemic formulae, in analogy to the popular definition of common knowledge.

Categories and Subject Descriptors

[**Artificial intelligence**]: Knowledge representation and reasoning — *Reasoning about belief and knowledge, Temporal reasoning, Causal reasoning and diagnostics*; [**Artificial intelligence**]: Distributed artificial intelligence — *Cooperation and coordination, multi-agent systems*; [**Real-time systems**]: Real-time system specification; [**Distributed computing methodologies**]

General Terms

Theory, Algorithms, Verification

Keywords

Common Knowledge, Epistemic Logic, Temporal coordination, Real-time constraints

1. INTRODUCTION

The fact that knowledge is closely related to coordinated action in distributed and multi-agent systems is well established by now. Ensuring that actions are performed in linear temporal order requires the agents to obtain appropriate nested knowledge (knowledge about knowledge) [5], while coordinating simultaneous actions requires attaining common knowledge of particular facts [17]. The latter connection has found uses in the analysis of distributed protocols (see, e.g. [17, 11, 28]). One of the contributions of [17] was in relating approximations of simultaneous coordination to weaker variants of common knowledge, called *epsilon*-common knowledge and *eventual* common knowledge. While common knowledge is typically defined and thought of as an infinite conjunction of nested knowledge formulae, it may also be defined as a fixed point [3, 8]. The variants of common knowledge defined by Halpern and Moses in [17] are most naturally obtained by appropriately modifying the fixed-point definition of common knowledge. All of the forms of coordination analyzed in [17] are symmetric in nature, in the sense that they are invariant under renaming of agents. For example, ε-common knowledge arises when the agents are guaranteed to act at most ε time units apart. In many natural situations, however, asymmetric forms of coordination arise. Let us consider an example.

Example 1.1 (Robotic Car Wash).
In an automated robotic car-wash enterprise, there are two washing robots L and R, (with L fitted to soap & rinse the left sides of cars, and R fitted to soap & rinse the right sides), and one drying robot, denoted D. At some point after a car enters, it must be soaped & rinsed from both sides, and then dried. The robot L is a new model, which takes only 4 minutes to perform its duty, while R is an older model, requiring 6 minutes. The drying is applied to the whole car, and it must commence only after washing of both sides is complete. Moreover, drying should not begin more than 5 minutes after the first of the washing robots finishes rinsing the car, as water stains might otherwise incur. It follows that, in particular, no more than 5 minutes may elapse between the time at which the rinsing of the car's left side ends and the time at which the rinsing of its right side ends. This, in turn, implies that L must start washing the car no later than 7 minutes after — and no more than 3 minutes before — R starts washing it. Finally, it is obviously desirable to minimize the time that the car spends in the Car Wash.

The temporal constraints in the car wash example make the design of the robots' control (the protocol that they follow) a delicate matter. With respect to a given car, each of the robots has only one decision to make: when to start treating the car — we shall refer to this as *the robot's action*. The times at which the robots act must satisfy the

TARK 2013, Chennai, India.

interactive constraints implied by the example. Clearly, the decision to act depends on when each of the other robots can (and will) commence treatment of this car. Before it can act, a robot must know (i.e., be sure) that the others will act in time, which requires, in particular, that the others will in turn know that *they* can act. More concretely, in our example, when L starts washing a car, it must know that between 7 minutes earlier and 3 minutes later, R will have started washing it, and that between 4 and $4+5=9$ minutes afterward, the drying robot D will have started drying it. Conversely yet asymmetrically, when R starts washing a car, it must know that between 3 minutes earlier and 7 minutes later, L will have started washing it and that between 6 and $6+5=11$ minutes afterward, D will have started drying it. We can similarly calculate D's required knowledge about L and R. Notice that this dependence is asymmetric — each robot calculates different bounds between its action and those of the two others.

The above discussion suggests that the robots in our example must reach some form of "temporal-epistemic equilibrium" in order to act. More generally, analogous situations seem to arise whenever a set of agents must coordinate their actions in a manner satisfying possibly asymmetric timing constraints. Our purpose in this paper is to concisely and usefully capture this form of interdependence in coordinated action. We shall do this by defining a new epistemic condition called *timely* common knowledge, which is, in a precise sense, necessary and sufficient for coordination as in the above example. Timely common knowledge generalizes and significantly extends common knowledge and its popular variants. Mathematically, the new notion is formally captured by way of a *vectorial* fixed point. Whereas common knowledge of an event ψ can be defined as the greatest fixed point of the function $x \mapsto E(\psi \wedge x)$, mapping events to events, where E is the operator denoting "everyone knows that...", a vectorial fixed point is the fixed point of a function mapping tuples of events to tuples of events. To our knowledge, such a technique has never before been utilized with regard to epistemic analysis. Roughly speaking, in the case of the car wash example, let $\bar{\xi} = (\xi_l, \xi_r, \xi_d)$ be the greatest fixed point of the function

$$\begin{pmatrix} x_l \\ x_r \\ x_d \end{pmatrix} \mapsto \begin{pmatrix} K_l(\, \psi_c \ \wedge\ \circledcirc^{\leq 3} x_r \ \ \wedge\ \circledcirc^{\leq 9} x_d \ ) \\ K_r(\, \psi_c \ \wedge\ \circledcirc^{\leq 7} x_l \ \ \wedge\ \circledcirc^{\leq 11} x_d \ ) \\ K_d(\, \psi_c \ \wedge\ \circledcirc^{\leq -4} x_l \ \wedge\ \circledcirc^{\leq -6} x_r \ ) \end{pmatrix},$$

where ψ_c is the event "the car is here", where K_i denotes "i knows that..." and where $\circledcirc^{\leq \varepsilon} x$ means that "x holds at some (past, present or future) point in time, no later than ε minutes after the current time". In the fixed point $\bar{\xi}$, robot L's coordinate ξ_l holds iff $K_l(\psi_c \wedge \circledcirc^{\leq 3} \xi_r \wedge \circledcirc^{\leq 9} \xi_d)$ does, and similarly for the other coordinates. Our results imply that the car-wash problem may be solved by having each robot i perform its task as soon as ξ_i holds, and that this solution is, in a precise sense, time-optimal. Roughly speaking, the tuple of events $\bar{\xi}$ will constitute *timely common knowledge* of ψ_c (with respect to the timing constraints of Example 1.1).[1] Notice that $\bar{\xi}$ does not correspond to a single fact (or event) that may be true or false at a single point in time. Rather, it represents a tuple of facts, one for each agent of interest. Each of the facts should hold at its own individual time, and the different times jointly satisfy the conditions in the fixed point definition.

In Section 4, we relate timely common knowledge to coordination. We define a class of *timely coordination* specifications, in which actions by various agents must satisfy timing conditions as in the Car Wash example. Timely coordination allows both symmetric and asymmetric forms of communication, and it strictly generalizes many symmetric forms of coordination previously studied in the literature. We also show, in a precise sense, that timely common knowledge strictly generalizes standard common knowledge and some of its variants. In Section 6, we show another close connection between timely common knowledge and standard common knowledge. Recall that common knowledge is often described as an infinite conjunction of nested knowledge formulae. A temporal-epistemic variant applies in the case of timely common knowledge. Roughly speaking, consider the point p at which L acts in the above car wash example. Recall that ψ_c denotes the fact that the car c has arrived, then clearly $K_l\psi_c$ must hold at p. It is not hard to check that $K_l \circledcirc^{\leq 3} K_r\psi_c$ should also hold at p, as should $K_l \circledcirc^{\leq 3} K_r \circledcirc^{\leq 7} K_l\psi_c$. Indeed, it is possible to generate arbitrarily deeply nested formulae that must hold at p. A different set of formulae must hold when R acts, and yet another set when D does. Thus, timely common knowledge implies an infinite set of nested formulae at each point of action. We show that it is in fact equivalent to a tuple of such sets under natural assumptions.

As an example of a natural application of our analysis, in Section 5 we present and mathematically analyze *timely-coordinated response* — a novel class of multi-agent coordination tasks. Roughly speaking, a timely-coordinated response task involves a prespecified *triggering* event ψ, such as the activation of a smoke detector or the arrival of a car to the car-wash facility. Should the trigger ψ occur, then each agent i in a set I of agents should perform an action (called its *response* to ψ) specified by the task, and the timing of the actions must satisfy a constraint of the following form: for all $i, j \in I$, if i acts at time t_i and j at t_j, then $-\delta(j,i) \leq t_j - t_i \leq \delta(i,j)$. The trigger ψ, the response actions, and the bounds δ are parameters specified in a given task. E.g. in the car wash example, the trigger is a car's arrival ψ_c, responses are robots' initiating their respective services, while $\delta(L,R) = 3$, $\delta(L,D) = 9$, $\delta(R,L) = 7$, $\delta(R,D) = 11$, $\delta(D,L) = -4$ and $\delta(D,R) = -6$. Timely-coordinated response is inspired by, and strictly generalizes, the response problems presented and studied by Ben-Zvi and Moses [5, 4, 6, 7].

We show that timely common knowledge is, in a precise sense, the epistemic counterpart of timely coordination. We use timely common knowledge to phrase a necessary and sufficient condition characterising protocols solving timely-coordinated response. Moreover, we show how timely common knowledge can be used to give a general technique for deriving a time-optimal solution (i.e. an optimal protocol) for any instance of timely-coordinated response.

The main contributions of this paper are:

- The theory of coordination in multi-agent systems is extended to treat timely coordination, in which general interdependent constraints are allowed;
- Timely common knowledge is defined as a vectorial fixed point and the mathematical soundness and key

[1] The definition of timely common knowledge is made in Section 4 with respect to general timing constraints, and is, naturally, more subtle.

properties of its definition are established;

- The solvability of, and optimal solutions to, a general class of timely coordination tasks are characterised using timely common knowledge;
- Timely common knowledge is shown to generalize common knowledge and many of its variants; and
- Timely common knowledge is shown to be equivalent to an infinite conjunction under natural assumptions.

2. RELATED WORK

The notion of common knowledge was defined by the philosopher David Lewis in [24]. Its relevance to game theory was shown by Aumann [2] and to AI by McCarthy [26]. Halpern and Moses [17] introduced it to distributed computing, showed its connection to simultaneity, and defined weaker variants of common knowledge corresponding to "approximations" of simultaneity. Common knowledge and its variants have had various applications in distributed computing [9, 11, 28, 19, 22, 12]. More recently, Ben-Zvi and Moses studied how time bounds on message transmission impact coordination in message-passing systems [5, 6, 4]. Most of their work studied coordination problems that are specified by partial orders. In [7], Ben-Zvi and Moses consider a notion of *tightly-timed* coordination in which agents act at precise time differences from each other. This gives rise to a generalization of common knowledge in which agents are considered at different prespecified times. All fixed-point epistemic notions (common knowledge and its variants) in the above works are based on a standard (scalar) fixed-point definition. The analysis in this paper significantly extends the connection between coordination and epistemic notions.

3. MODEL AND NOTATION

For ease of exposition, we adopt the multi-agent systems model, based on contexts, runs and systems, of Fagin *et al.* [12]. The model captures the possible histories, called runs, of a finite set of agents $\mathbb{I}$. We model time as being discrete, ranging over the set $\mathbb{T} = \mathbb{N} \cup \{0\}$ of nonnegative integers.[2] Each agent $i \in \mathbb{I}$ may be thought of as an automaton, existing at any specific time $t \in \mathbb{T}$ in one of a set of possible states $\mathbb{L}_i$. The set of possible global states of the model, describing a snapshot of the system at some given time, is thus $\mathbb{L}_e \times \bigtimes_{i\in\mathbb{I}} \mathbb{L}_i$, where $\mathbb{L}_e$ is a set of possible states for the environment. We denote by $\mathcal{R}$ the set of possible runs, or possible histories, of the model, where a run $r \in \mathcal{R}$ is a function $r : \mathbb{T} \to \mathbb{L}_e \times \bigtimes_{i\in\mathbb{I}} \mathbb{L}_i$, from times to global states. A *point* is a run-time pair $p = (r, t) \in \mathcal{R} \times \mathbb{T}$, denoting time t in the run r. The local state of an agent $i \in \mathbb{I}$ at the point (r, t) is denoted by $r_i(t)$. We denote by $\mathbb{P}$ the set of protocols, where a protocol $P \in \mathbb{P}$ is a tuple $P = (P_i)_{i\in\mathbb{I}}$, in which each P_i is a function from the set $\mathbb{L}_i$ of i's local states to sets of possible actions (or to a single option, if P is deterministic) for the actions to be performed by i when at that state. Finally, a context γ is a specification of a protocol for the environment, possible initial global states, any relevant constraints on runs (e.g. an agent may not perform two certain given actions at the same time), and a transition function from the global state and all actions performed at any time t, to the global state at $t+1$. For a context γ and a protocol $P \in \mathbb{P}$, we denote by $R(P, \gamma)$ the set of runs of P in γ.

[2] All results in this paper hold verbatim if we consider infinite sets $\mathbb{I}$ of agents, and with only trivial changes if time is continuous, so that $\mathbb{T} \triangleq \mathbb{R}_{\geq 0}$. We avoid modifying the model to handle continuous time for ease of exposition.

3.1 Events, Knowledge Operators and Temporal Operators

There are two equivalent ways of defining knowledge in systems, one in terms of propositions and modal operators in modal logic [12], and the other, proposed by Aumann [2], in terms of events and of functions on events. We follow the latter, since it facilitates the formulation of fixed points, which play a role in our analysis. Informally, however, we use the terms *fact* and *event* interchangeably. As in probability theory, we represent events using the set of points at which they hold. A set of runs R gives rise to a *(R-)universe* $\Omega_R \triangleq R \times \mathbb{T}$, and a corresponding σ-algebra of events $\mathcal{F}_R \triangleq 2^{\Omega_R}$. Thus, for example, the event "agent i is performing action α", is formally associated with all points $(r, t) \in \Omega_R$ at which i performs α.

We make use of several temporal operators applied to events. These are very much in the spirit of standard linear-time operators (see Manna and Pnueli [25]), except that in our case two of the operators may refer to the past as well as the future. We thus use slight variations on the standard symbols. A few basic properties of these operators are explored in Appendix C. We define three temporal operators as functions $\mathcal{F}_R \to \mathcal{F}_R$ as follows;[3] fix $R \subseteq \mathcal{R}$ and let $\psi \in \mathcal{F}_R$.

- $\diamondsuit\psi \triangleq \big\{(r, t) \in \Omega_R \mid \exists\, t' \in \mathbb{T} : (r, t') \in \psi\big\}$; the event "$\psi$ holds at some past, present or future time (during the current run)",
- $\circledcirc^{\varepsilon}\psi \triangleq \big\{(r, t) \in \Omega_R \mid (r, t + \varepsilon) \in \psi\big\}$, for $\varepsilon \in \mathbb{Z}$; the event "$\psi$ holds at *exactly* ε time units from now", and
- $\circledcirc^{\leq\varepsilon}\psi \triangleq \left\{(r, t) \in \Omega_R \,\middle|\, \exists\, t' \in \mathbb{T} : \begin{array}{l} t' \leq t + \varepsilon \;\;\&\; \\ (r, t') \in \psi \end{array}\right\}$, for $\varepsilon \in \mathbb{Z} \cup \{\infty\}$; the event "$\psi$ holds at some (past, present, or future) time, no later than ε time units from now".

The standard definition of knowledge in this setting is also a function on events. Intuitively, an agent's information is captured by its local state $r_i(t)$. Accordingly, two points (r, t) and (r', t') are considered *indistinguishable* in the eyes of i if i's local state at both points is the same. We use K_i to denote i's knowledge, and define the event "i knows ψ" by

- $K_i\psi \triangleq \big\{(r, t) \in \Omega_R \mid (r', t') \in \psi \text{ whenever } r'_i(t) = r_i(t)\big\}$.

Since $K_i\psi$ is itself an event, nested knowledge facts such as $K_jK_i\psi$ are immediately well defined. This gives rise to a standard S5 notion of knowledge, equivalent to the standard definition in terms of partitions. See Appendix A for a discussion, and for a definition of common knowledge.

[3] While the following definitions depend on R, we omit R from these notations for readability, as the set of runs will be clear from the discussion. We follow this convention when presenting some other definitions below as well.

3.2 Event Ensembles

Roughly speaking, it is possible for an agent i to act precisely whenever an event $\psi \in \Omega_R$ occurs, only if at every point at which ψ holds, i knows that it does, i.e. if $\psi = K_i\psi$. Such an event is said to be *i-local*. Equivalently, ψ is i-local if its truth is determined by i's local state, i.e. if there exists $S \subseteq \mathbb{L}_i$ s.t. for every $(r,t) \in \Omega_R$, we have $(r,t) \in \psi$ iff $r_i(t) \in S$. In the study of coordination, we are usually interested in the interaction between the actions of several agents. Consider, for example, a scenario in which two agents, Alice and Bob, must perform two respective actions α and β in some coordinated manner. Then the set $\boldsymbol{e}_A$ of points at which Alice performs α is a local event for Alice, and likewise for the corresponding set $\boldsymbol{e}_B$ for Bob and β. The pair $\bar{\boldsymbol{e}} \triangleq (\boldsymbol{e}_A, \boldsymbol{e}_B)$ is called an *ensemble* for Alice and Bob. More generally, following Fagin *et al.*, given a set of agents $I \subseteq \mathbb{I}$, we define an *I-ensemble* to be an I-tuple of events $\bar{\boldsymbol{e}} = (\boldsymbol{e}_i)_{i\in I} \in {\mathcal{F}_R}^I$, in which $\boldsymbol{e}_i$ is i-local, for each $i \in I$. Returning to Alice and Bob, consider a deterministic protocol in which whenever Alice performs action α, Bob is guaranteed to simultaneously perform action β and vice versa. Since α and β are guaranteed to be simultaneous actions, we have $\boldsymbol{e}_A = \boldsymbol{e}_B$. An ensemble $\bar{\boldsymbol{e}}$ with this property is thus said to be *perfectly coordinated*. Fagin *et al.* [13] have studied the properties of such ensembles, as well as of ensembles satisfying weaker forms of coordination (eventual coordination and ε-coordination) defined in [17]. See Appendix B.1 for more details.

4. TIMELY COORDINATION & TIMELY COMMON KNOWLEDGE

Given a set of agents I, we denote by the set of distinct pairs of agents in I by $I^{\bar{2}} \triangleq \{(i,j) \in I^2 \mid i \neq j\}$. We define a *timely-coordination spec* to be a pair (I,δ), where $I \subseteq \mathbb{I}$ is a set of agents and $\delta : I^{\bar{2}} \to \mathbb{Z} \cup \{\infty\}$. Intuitively, $\delta(i,j)$ denotes an upper bound on the time from when i performs her action, until when j performs his.[4] We can now formally define timely coordination.

Definition 4.1 (Timely-Coordination). *Given a timely-coordination spec (I,δ) and a system $R \subseteq \mathcal{R}$, we say that an I-ensemble $\bar{\boldsymbol{e}} \in {\mathcal{F}_R}^I$ is **δ-coordinated** (in R) if for every $(i,j) \in I^{\bar{2}}$ and for every $(r,t) \in \boldsymbol{e}_i$, there exists $t' \leq t + \delta(i,j)$ s.t. $(r,t') \in \boldsymbol{e}_j$.*

While, as discussed in Appendix A, the popular definition of common knowledge is in terms of an infinite conjunction of nested knowledge formulae, Barwise [3], following Harman [8], has defined common knowledge as a fixed point. Indeed, if we denote $E_I\psi = \bigcap_{i\in I} K_i\psi$ ("everybody in I knows"), then the following is an equivalent way of formulating common knowledge as a fixed point.[5]

Theorem 4.2 ([17]). *Let $R \subseteq \mathcal{R}$ and $I \subseteq \mathbb{I}$. Then $C_I\psi$ is the greatest fixed point of the function $f_\psi : \mathcal{F}_R \to \mathcal{F}_R$ given by $x \mapsto E_I(\psi \cap x)$, for every event $\psi \in \mathcal{F}_R$.*

[4] If time were continuous (i.e. $\mathbb{T} = \mathbb{R}_{\geq 0}$), then the range of δ would be $(\mathbb{T} - \mathbb{T}) \cup \{\infty\} = \mathbb{R} \cup \{\infty\} = (-\infty, \infty]$.

[5] The equivalence is in the standard models; see Barwise [3] for a discussion of various accepted definitions for common knowledge and of models in which they do not coincide.

As mentioned in the introduction, a classic result [17], which stems from Theorem 4.2, is that common knowledge tightly relates to perfect coordination. One manifestation of this is in the fact that if an action α is guaranteed to be performed simultaneously by a set of agents whenever any of them performs it, then these agents must have common knowledge of the occurrence of α when it is performed. (Intuitively, the guaranteed simultaneity of α causes its joint occurrence to be inferred at once by all participants who perform it.) Conversely, whenever common knowledge of a fact arises among a set of agents, it does so simultaneously for all agents. See Appendix B.2 for further details, as well as a review of the analogous analysis for the weaker variants of common knowledge defined in [17]. Our purpose is to similarly relate timely coordination to an epistemic notion. Consider the points at which the robots act in the Car Wash example. In general, the robots may act at different times. Moreover, while the local events that the various robots must respectively know in order for them to act are interdependent, they differ from one another. Therefore, instead of seeking a fixed point of a function on (single events in) $\mathcal{F}_R$ as done for common knowledge and previous variants, we define a function on ${\mathcal{F}_R}^I$ — the set of I-tuples of events. Given an event $\psi \in \mathcal{F}_R$ and a timely-coordination spec (I,δ), we define a function f^δ_ψ on ${\mathcal{F}_R}^I$ in which each coordinate i captures the respective constraints of the agent i, based on ψ and δ. The greatest fixed point of f^δ_ψ, denoted by by $C^\delta_I\psi$ (this is an I-tuple of events), is shown to capture timely coordination, and is thus the desired ensemble. Since f^δ_ψ is a function of several variables, it is a *vectorial* function, and its fixed point is a vectorial fixed point [1].[6]

4.1 Timely Common Knowledge as a Vectorial Fixed Point

We start by defining a lattice structure on ${\mathcal{F}_R}^I$. A *greatest* fixed point of a function f on ${\mathcal{F}_R}^I$ is a fixed point of f that is greater than any other fixed point thereof, according to the partial order $\leq$ of the lattice. Recall that a member of ${\mathcal{F}_R}^I$ is a tuple of events of the form $\bar{\varphi} \triangleq (\varphi_i)_{i\in I}$.

Definition 4.3 (Lattice Structure on ${\mathcal{F}_R}^I$). *Let $R \subseteq \mathcal{R}$ and let $I \subseteq \mathbb{I}$. The following partial order relation and binary operations define a lattice structure on ${\mathcal{F}_R}^I$.*

- *Order:* $\bar{\varphi} \leq \bar{\xi}$ *iff* $\forall i \in I : \varphi_i \subseteq \xi_i$.
- *Join:* $\bar{\varphi} \vee \bar{\xi} \triangleq (\varphi_i \cup \xi_i)_{i\in I}$.
- *Meet:* $\bar{\varphi} \wedge \bar{\xi} \triangleq (\varphi_i \cap \xi_i)_{i\in I}$.

We are now ready to define timely common knowledge.

Definition 4.4 (Timely Common Knowledge). *Let $R \subseteq \mathcal{R}$ and let (I,δ) be a timely-coordination spec. For each $\psi \in \mathcal{F}_R$, we define **δ-common knowledge** of ψ by I, denoted by $C^\delta_I\psi$, to be the greatest fixed point of the function $f^\delta_\psi : {\mathcal{F}_R}^I \to {\mathcal{F}_R}^I$ given by*

$$f^\delta_\psi : \quad (x_i)_{i\in I} \quad \mapsto \quad \left(K_i\Big(\psi \cap \bigcap_{j\in I\setminus\{i\}} \circledcirc^{\leq\delta(i,j)} x_j\Big)\right)_{i\in I}.$$

[6] While vectorial fixed points may alternatively be captured by nested fixed points [1, Chapter 1], in our case we argue that the vectorial representation better parallels the underlying intuition. We are not aware of either vectorial, or nested fixed points being used in an epistemic setting before.

We justify Definition 4.4 in three steps. First, we show that $C_I^\delta\psi$ is well-defined and satisfies a natural induction rule and a monotonicity property. (For proofs of all propositions given in this paper, see Appendix C.)

LEMMA 4.5. *Let (I,δ) be a timely-coordination spec, let $R \subseteq \mathcal{R}$ and let $\psi \in \mathcal{F}_R$.*

1. *$C_I^\delta\psi$ is well-defined, i.e. f_ψ^δ has a greatest fixed point.*
2. *Induction Rule: Every $\bar{\xi} \in {\mathcal{F}_R}^I$ satisfying $\bar{\xi} \leq f_\psi^\delta(\bar{\xi})$ also satisfies $\bar{\xi} \leq C_I^\delta\psi$.*
3. *C_I^δ is monotone: $\psi \subseteq \phi \;\Rightarrow\; C_I^\delta\psi \leq C_I^\delta\phi$, for every $\psi, \phi \in \mathcal{F}_R$.*

The induction rule is a powerful tool for analyzing situations giving rise to timely common knowledge. It states that if ξ_i implies the K_i statement in Definition 4.4, with x_j substituted by ξ_j everywhere, then each agent i knows its respective coordinate of $C_I^\delta\psi$ whenever ξ_i holds.

A timely-coordination spec is a fairly general tool for defining relative timing constraints. Particular simple instances can capture previously studied forms of coordination. Namely, if $\delta \equiv \infty$, timely coordination coincides with eventual coordination, and for any $\varepsilon < \infty$, the form of coordination obtained by setting $\delta \equiv \varepsilon$ closely relates to ε-coordination (and hence to perfect coordination when $\delta \equiv 0$). Indeed, for coordinate-wise stable ensembles (see Appendix C.4) and for ensembles with at most a single point per agent per run (see Section 5 for an example), $\delta \equiv \varepsilon$ precisely captures ε-coordination and $\delta \equiv 0$ specifies perfect coordination. Furthermore, timely common knowledge is closely related to the corresponding variants of common knowledge, for each of these special cases of constant δ. (See Appendix D.2 for the precise details.) Our second step is to show that timely common knowledge closely corresponds to timely coordination, in the same sense in which common knowledge corresponds to perfect coordination, and variants of common knowledge to their respective forms of coordination. (See, once again, Appendix B.2.) The following theorem establishes this correspondence. (While phrasing this theorem, and henceforth, we use the shorthand notation $\cup\bar{\xi} \triangleq \bigcup_{i\in I}\xi_i$, for every $\bar{\xi} = (\xi_i)_{i\in I} \in {\mathcal{F}_R}^I$.)

THEOREM 4.6. *Let $R \subseteq \mathcal{R}$ and let (I,δ) be a timely-coordination spec.*

1. *$C_I^\delta\psi$ constitutes a δ-coordinated I-ensemble, for every $\psi \in \mathcal{F}_R$.*
2. *$\cup C_I^\delta\psi \subseteq \psi$, for every $\psi \in \mathcal{F}_R$.*
3. *If $\bar{\boldsymbol{e}} \in {\mathcal{F}_R}^I$ is a δ-coordinated I-ensemble satisfying $\cup\bar{\boldsymbol{e}} \subseteq \psi$ for some $\psi \in \mathcal{F}_R$, then $\bar{\boldsymbol{e}} \leq C_I^\delta\psi$.*
4. *If $\bar{\boldsymbol{e}} \in {\mathcal{F}_R}^I$ is a δ-coordinated I-ensemble, then $\bar{\boldsymbol{e}} \leq C_I^\delta(\cup\bar{\boldsymbol{e}})$.*
5. *If $\bar{\boldsymbol{e}} \in {\mathcal{F}_R}^I$ is a δ-coordinated I-ensemble, then $\cup\bar{\boldsymbol{e}} = \cup C_I^\delta(\cup\bar{\boldsymbol{e}})$.*

Theorem 4.6 highlights some key properties of the fundamental connection between δ-coordination and δ-common knowledge: (Parts 1, 4 and 5 are analogues of Theorems B.4, B.5 and B.6, the latter part being stronger in a sense than its counterparts from Theorems B.5 and B.6 regarding eventual- and ε-common knowledge, respectively.) Parts 1–3 characterise δ-common knowledge of ψ as the greatest δ-coordinated event ensemble that implies ψ.[7] Moreover, Part 3 provides convenient means to prove that timely common knowledge holds. Part 4 says that regardless of the way a δ-coordinated ensemble is formed (be it using δ-common knowledge of some event ψ, or otherwise), the fact that its i'th coordinate holds implies that the i'th coordinate of δ-common knowledge of (the disjunction of) this ensemble holds as well. Finally, part 5 captures the fact that the union of any δ-coordinated ensemble is a fixed point of $\cup C_I^\delta$, and, together with Part 1, implies the idempotence of $\cup C_I^\delta$. Our third step is demonstrating the usefulness of timely common knowledge, which we do in the next section.

5. TIMELY-COORDINATED RESPONSE

We now harness the machinery developed in the previous sections to study a class of coordination problems. In these problems, the occurrence of a particular event ϕ must trigger responses by a set $I \subseteq \mathbb{I}$ of agents, and the responses must be timely coordinated according to a given spec δ.[8] The triggering event ϕ may be the arrival of a car at the Car Wash, the ringing of a smoke alarm, or some other event that requires a response. A run r during which ϕ occurs (i.e. $(r,t) \in \phi$ for some $t \in \mathbb{T}$) is called *ϕ-triggered*. Following in the spirit of [5] and generalizing their definitions (see Appendix D.1), we define this class of coordination problems as follows.

DEFINITION 5.1 (TIMELY-COORDINATED RESPONSE). *A **timely-coordinated response** problem, or TCR, is a quintuplet $\tau = (\gamma, \phi, I, \delta, \bar{\alpha})$, where γ is a context, $\phi \in \mathcal{F}_\mathcal{R}$ is an event, (I,δ) is a timely-coordination spec and $\bar{\alpha} = (\alpha_i)_{i\in I}$ is a tuple of actions, one for each $i \in I$. A protocol P is said to **solve** a TCR $\tau = (\gamma, \phi, I, \delta, \bar{\alpha})$ if for every $r \in R(P,\gamma)$,*

- *If r is ϕ-triggered and ϕ first occurs in r at $t_\phi \in \mathbb{T}$, then each agent $i \in I$ responds (i.e. performs α_i) in r exactly once, at a time $t_i \geq t_\phi$ s.t. for every $(i,j) \in I^2$, it holds that $t_j \leq t_i + \delta(i,j)$.*
- *If r is not ϕ-triggered, then none of the agents in I respond in r.*

We say that τ is *solvable* if there exists a protocol $P \in \mathbb{P}$ that solves it. We now show that attaining timely common knowledge is a necessary condition for action in a protocol solving timely-coordinated response, in the sense that an agent cannot respond unless is has attained its respective coordinate of timely common knowledge.[9] Indeed, Theo-

[7] Neither eventual- nor ε-common knowledge give way for a clean analogous characterisation. (See Appendix D.2 for more details.)

[8] For ease of exposition, we assume that each agent is associated with exactly one action. Essentially the same analysis applies if we allow each agent to be associated with more than one response action.

[9] In the following propositions, we work in the universe $\Omega_{R(P,\gamma)}$ defined by the system of runs of the given protocol in question. All knowledge and temporal operators are therefore relative to this universe. Furthermore, we slightly abuse notation by writing ϕ to refer to $\phi \cap \Omega_{R(P,\gamma)}$, i.e. the restriction of ϕ to this universe.

rem 4.6(3) implies:[10]

COROLLARY 5.2. *Let $P \in \mathbb{P}$ be a protocol solving a TCR $\tau = (\gamma, \phi, I, \delta, \bar{\alpha})$, and let $r \in R(P, \gamma)$ be a ϕ-triggered run. If $i \in I$ responds at time t_i in r, then $(r, t_i) \in \left(C_I^\delta(\circledcirc^{\leq 0}\phi)\right)_i$.*

In fact, timely common knowledge is not only necessary for solving a TCR, but also *sufficient* for doing so. (See below.) Indeed, we now argue that timely common knowledge can be used to design time-optimal solutions for arbitrary TCRs. For the notion of time-optimality to be well defined, we define it with regard to each family of protocols that are the same in all aspects, except for possibly the time at which (and whether) agents respond. To this end, we restrict ourselves to protocols that may be represented as a pair $P = (P_{-\bar{\alpha}}, P_{\bar{\alpha}})$, s.t. the output of P is a Cartesian product of the outputs of its two parts, where $P_{\bar{\alpha}}$ specifies whether to respond, while $P_{-\bar{\alpha}}$ specifies everything else. (Natural examples for such protocols are those in which the choice of whether to respond is deterministic.) Furthermore, we restrict ourselves to contexts in which none of $\bar{\alpha}$ affect the agents' transitions in any way (and hence do not affect any future states or actions). Under these conditions, given two protocols $P = (P_{-\bar{\alpha}}, P_{\bar{\alpha}})$ and $P' = (P_{-\bar{\alpha}}, P'_{\bar{\alpha}})$ that share same non-response component $P_{-\bar{\alpha}}$, there exists a natural isomorphism $\sigma : R(P, \gamma) \xrightarrow{\sim} R(P', \gamma)$, in which corresponding runs agree in all aspects except for possibly the times at which (and whether) responses are performed; we thus say that two such protocols are *run-equivalent*. Furthermore, we slightly abuse notation by writing $R(P_{-\bar{\alpha}}, \gamma)$ to refer to both $R(P, \gamma)$ and $R(P', \gamma)$, which coincide using σ. We say that a protocol $P = (P_{-\bar{\alpha}}, P_{\bar{\alpha}})$ is a *time-optimal* solution for a TCR τ if P solves τ and, moreover, responses are never performed in P later than in any solution P' of τ that is run-equivalent to P. More formally, we demand that for every ϕ-triggered $r \in R(P, \gamma)$ and for every $i \in I$, if i responds at time t_i in r and at time t'_i in $\sigma(r) \in R(P', \gamma)$ (with σ as above), then necessarily $t_i \leq t'_i$. It should be noted that it is not *a priori* clear that TCRs admit time-optimal solutions. We now show not only that all solvable TCRs do, but moreover, that for every solution there exists a run-equivalent time-optimal solution and that all time-optimal solutions have each agent responding at the first instant at which it attains its respective coordinate of timely common knowledge.[11]

COROLLARY 5.3. *Let $\tau = (\gamma, \phi, I, \delta, \bar{\alpha})$ be a solvable TCR and let $P = (P_{-\bar{\alpha}}, P_{\bar{\alpha}})$ be a protocol solving it. The run-equivalent protocol $P' = (P_{-\bar{\alpha}}, P'_{\bar{\alpha}})$ in which every $i \in I$ responds at the first instant at which $\left(C_I^\delta(\circledcirc^{\leq 0}\phi)\right)_i$ holds (in $\Omega_{R(P_{-\bar{\alpha}}, \gamma)}$), is a time-optimal solution for τ.*

Indeed, we may now formalize our previous statement regarding timely common knowledge being necessary and sufficient for solving a TCR $\tau = (\gamma, \phi, I, \delta, \bar{\alpha})$: a protocol P is run-equivalent to a solution of τ iff $C_I^\delta(\circledcirc^{\leq 0}\phi)$ is attained in each of its ϕ-triggered runs. (See Corollary C.13.)

Attaining true (*not* timely) common knowledge of a fact of interest is often an effective and intuitive way of synchronization, which may also be used to solve TCRs. However, in addition to such a solution being suboptimal in many cases, timely common knowledge is often attainable even when common knowledge is not. In the Car Wash setting, for example, if the arrival of a car is guaranteed to be observed by each robot (privately) within at most 2 time units, then the TCR can be readily solved (and timely common knowledge attained) even though techniques of [17] may be used to show that the arrival of the car might never become common knowledge.

We conclude this section by noting that in contexts supporting full-information protocols (see, e.g. [12]), the above tools may be applied to obtain both a globally time-optimal solution to, as well as a solvability criterion for, arbitrary TCRs. We defer the details to the full paper.

6. A CONSTRUCTIVE DEFINITION FOR TIMELY COMMON KNOWLEDGE

The analysis of Section 5 provides us with time-optimal solutions for timely-coordinated response. The fly in the ointment, though, is how to implement these solutions, i.e. how to check whether a certain coordinate of timely common knowledge holds, given the state of the corresponding agent. We now take a step in this direction, which also sheds some more light on the fixed-point analysis of the previous section, and makes the notion of timely common knowledge more concrete. Under natural assumptions (see Theorem C.20 for details), we obtain, for every $i \in I$:

$$(C_I^\delta\psi)_i = \bigcap_{(i,i_2,\ldots,i_n)\in I^{\bar{*}}} K_i \circledcirc^{\delta(i,i_2)} K_{i_2} \circledcirc^{\delta(i_2,i_3)} K_{i_3} \cdots \circledcirc^{\delta(i_{n-1},i_n)} K_{i_n}\psi, \tag{1}$$

where $I^{\bar{*}} \triangleq \left\{(i_1, \ldots, i_n) \in I^* \mid \forall m : i_m \neq i_{m+1}\right\}$ denotes the set of all finite non-stuttering sequences of elements of I. Note that for $\delta \equiv 0$ (perfect coordination), (1) yields in each coordinate a familiar definition (see Observation A.4) of common knowledge as an infinite conjunction: (cf. the more popular Definition A.3, which is generalized by eventual- and ε-common knowledge, but is symmetric in nature, and therefore less natural for generalization in our setting.)

$$C_I\psi \quad = \bigcap_{(i_1,\ldots,i_n)\in I^{\bar{*}}} K_{i_1}K_{i_2}K_{i_3}\cdots K_{i_n}\psi.$$

The formulation of timely common knowledge in terms of an infinite conjunction provides a constructive interpretation of the time-optimal solution from Corollary 5.3. Roughly speaking, each agent $i \in I$ should respond at the first instant at which all nested-knowledge formulae of the form $K_i \circledcirc^{\delta(i,i_2)} K_{i_2} \circledcirc^{\delta(i_2,i_3)} K_{i_3} \cdots \circledcirc^{\delta(i_{n-1},i_n)} K_{i_n} \circledcirc^{\leq 0} \phi$ hold for all $(i, i_2, \ldots, i_n) \in I^{\bar{*}}$. (See Corollary C.22 for the precise phrasing.) While this may appear to take us a step closer to implementing time-optimal solutions, a naïve implementation may still require potentially infinitely many tests. In fact, as in the case of common knowledge, in practice timely common knowledge may be established using the induction rule of Theorem 4.6(3). We also refer the reader to [16, Chapters 6 and 9] for a study of the causal structure of these tests, which uses a different set of tools and which is,

[10] Observe that $\circledcirc^{\leq 0}$ stands for the temporal operator "previously".

[11] As noted in Appendix C, in some runs of certain systems $R(P_{-\bar{\alpha}}, \gamma)$ in a continuous-time model, the set of times at which $\left(C_I^\delta(\circledcirc^{\leq 0}\phi)\right)_i$ holds does not attain its infimum value. It is possible to similarly show that in such pathological cases, no time-optimal protocol that is run-equivalent to P exists.

therefore, out of the scope of this paper.

7. CONCLUDING REMARKS

This paper suggests a broader connection between epistemic analysis and distributed coordination than was previously realized. The novel concept of timely common knowledge provides a formal connection between distributed protocols and a new form of equilibria, thus bringing distributed and multi-agent protocols closer to the realm of games, even in the absence of utilities and preferences. It should be noted, however, that the equilibrium in our analysis is not merely among strategies; in the Car Wash scenario, for example, the particular time instants at which the various robots act are at a *temporal-epistemic equilibrium*.

While this paper introduces vectorial fixed-point epistemic analysis as a tool for defining timely common knowledge, we believe that it will prove to be applicable well beyond the scope of problems considered here. We are currently pursuing generalizations and variations on the techniques presented in this paper for varying purposes, from generalizations of timely common knowledge to analyses of significantly different tasks, such as distributed agreement problems, which do not involve any form of timely coordination.

Fixed points, be they scalar or vectorial, be they temporal-epistemic or of any other kind, provide formal, yet intuitive, means of capturing equilibria in multi-agent systems. Many systems around us, from subatomic physical systems to astrophysical ones, and from animal societies to stock markets, exist in some form of equilibrium, possibly reached as a result of a long-forgotten spontaneous symmetry breaking. It is thus only natural to conjecture that fixed-point analyses of distributed algorithms and multi-agent systems hold the potential to provide significant further insights that are yet to be discovered.

8. ACKNOWLEDGMENTS

This work was supported in part by the Israel Science Foundation (ISF) under Grant 1520/11, and by the European Research Council under the European Community's Seventh Framework Programme (FP7/2007-2013) / ERC grant agreement no. [249159]. We would like to thank the reviewers for their useful comments. The first author would like to thank Gil Kalai, the co-advisor of his M.Sc. thesis [16]; this paper is based upon Chapters 7, 8 and 10 thereof.

9. REFERENCES

[1] A. Arnold and D. Niwiński. *Rudiments of μ-Calculus*, volume 146 of *Studies in Logic and the Foundations of Mathematics*. North-Holland, Amsterdam, Netherlands, 2001.

[2] R. J. Aumann. Interactive epistemology I: Knowledge. *International Journal of Game Theory*, 28(3):263–300, 1999.

[3] J. Barwise. Three views of common knowledge. In *Proceedings of the 2nd Conference on Theoretical Aspects of Reasoning about Knowledge (TARK)*, pages 365–379, 1988.

[4] I. Ben-Zvi. *Causality, Knowledge and Coordination in Distributed Systems*. PhD thesis, Technion, Israel Institute of Technology, Haifa, Israel, 2011.

[5] I. Ben-Zvi and Y. Moses. Beyond Lamport's happened-before: On the role of time bounds in synchronous systems. In *Proceedings of the 24th International Symposium on Distributed Computing (DISC)*, pages 421–436, 2010.

[6] I. Ben-Zvi and Y. Moses. On interactive knowledge with bounded communication. *Journal of Applied Non-Classical Logics*, 21(3-4):323–354, 2011.

[7] I. Ben-Zvi and Y. Moses. Agent-time epistemics and coordination. In *Proceedings of the 5th Indian Conference on Logic and its Applications (ICLA)*, 2013. To appear.

[8] J. Bennett. Review of linguistic behaviour by Jonathan Bennet. *Language*, 53(2):417–424, 1977.

[9] K. M. Chandy and J. Misra. How processes learn. *Distributed Computing*, 1(1):40–52, 1986.

[10] B. A. Coan, D. Dolev, C. Dwork, and L. Stockmeyer. The distributed firing squad problem. In *Proceedings of the 17th Annual ACM Symposium on Theory of Computing (STOC)*, pages 335–345, 1985.

[11] C. Dwork and Y. Moses. Knowledge and common knowledge in a Byzantine environment: crash failures. *Information and Computation*, 88(2):156–186, 1990.

[12] R. Fagin, J. Y. Halpern, Y. Moses, and M. Y. Vardi. *Reasoning About Knowledge*. The MIT Press, Cambridge, MA, USA, 1995.

[13] R. Fagin, J. Y. Halpern, Y. Moses, and M. Y. Vardi. Common knowledge revisited. *Annals of Pure and Applied Logic*, 96(1–3):89–105, 1999.

[14] M. F. Friedell. On the structure of shared awareness. *Behavioral Science*, 14(1):28–39, 1969.

[15] Y. A. Gonczarowski. Satisfiability and canonisation of timely constraints. Manuscript submitted for publication, 2012.

[16] Y. A. Gonczarowski. Timely coordination in a multi-agent system. Master's thesis, Hebrew University of Jerusalem, Jerusalem, Israel, 2012.

[17] J. Halpern and Y. Moses. Knowledge and common knowledge in a distributed environment. *Journal of the ACM*, 37(3):549–587, 1990.

[18] J. Y. Halpern, Y. Moses, and O. Waarts. A characterization of eventual byzantine agreement. *SIAM Journal on Computing*, 31(3):838–865, 2001.

[19] J. Y. Halpern and L. D. Zuck. A little knowledge goes a long way: knowledge-based derivations and correctness proofs for a family of protocols. *Journal of the ACM*, 39(3):449–478, 1992.

[20] S. C. Kleene. *Introduction to Metamathematics*. North-Holland Publishing Company, Amsterdam, Netherlands, 1952.

[21] I. I. Kolodner. On completeness of partially ordered sets and fixpoint theorems for isotone mappings. *American Mathematical Monthly*, 75(1):48–49, 1968.

[22] F. Kuhn, Y. Moses, and R. Oshman. Coordinated consensus in dynamic networks. In *Proceedings of the 30th Annual ACM Symposium on Principles of Distributed Computing (PODC)*, pages 1–10, 2011.

[23] J. L. Lassez, V. L. Nguyen, and E. A. Sonenberg. Fixed point theorems and semantics: A folk tale. *Information Processing Letters*, 14(3):112–116, 1982.

[24] D. Lewis. *Convention, A Philosophical Study*. Harvard University Press, Cambridge, MA, USA, 1969.

[25] Z. Manna and A. Pnueli. *The Temporal Logic of*

Reactive and Concurrent Systems, volume 1. Springer-Verlag, Berlin, Germany / New York, NY, USA, 1992.

[26] J. McCarthy. Formalization of two puzzles involving knowledge. Manuscript, Computer Science Department, Stanford University, 1978.

[27] E. F. Moore. The firing squad synchronization problem. In E. F. Moore, editor, *Sequential Machines: Selected Papers*, pages 213–214. Addison-Wesley, Reading, MA, USA, 1964.

[28] Y. Moses and M. R. Tuttle. Programming simultaneous actions using common knowledge. *Algorithmica*, 3:121–169, 1988.

[29] A. Tarski. A lattice-theoretical fixpoint theorem and its applications. *Pacific Journal of Mathematics*, 5(2):285–309, 1955.

APPENDIX

A. KNOWLEDGE AND COMMON KNOWLEDGE

We first survey a few immediate (and well-known) properties of the knowledge operator, which is defined in Section 3.

OBSERVATION A.1. *Let $R \subseteq \mathcal{R}$ and let $i \in \mathbb{I}$. By definition of K_i, we have:*

- *Knowledge Axiom: $K_i\psi \subseteq \psi$, for every $\psi \in \mathcal{F}_R$.*
- *Positive Introspection Axiom: $K_iK_i\psi = K_i\psi$, for every $\psi \in \mathcal{F}_R$.*
- *Monotonicity: $\psi \subseteq \phi \;\Rightarrow\; K_i\psi \subseteq K_i\phi$, for every $\psi, \phi \in \mathcal{F}_R$.*
- *K_i commutes with intersection: $K_i(\cap\Psi) = \bigcap\{K_i\psi \mid \psi \in \Psi\}$, for every set of events $\Psi \subseteq \mathcal{F}_R$.*

We now build upon the definition of knowledge and define the notions of "everyone knows" and of "common knowledge".

DEFINITION A.2 (EVERYONE KNOWS). *Let $R \subseteq \mathcal{R}$ and let $I \subseteq \mathbb{I}$. For every $\psi \in \mathcal{F}_R$, denote $E_I\psi \triangleq \bigcap_{i\in I} K_i\psi$.*

One popular, constructive definition of common knowledge [14] is the following, defining that an event is common knowledge to a set of agents when all know it, all know that all know it, etc.

DEFINITION A.3 (COMMON KNOWLEDGE). *Let $R \subseteq \mathcal{R}$ and let $I \subseteq \mathbb{I}$. For every $\psi \in \mathcal{F}_R$, denote $C_I\psi \triangleq \bigcap_{n=1}^{\infty} {E_I}^n\psi$, where ${E_I}^0\psi \triangleq \psi$ and ${E_I}^n\psi \triangleq E_I{E_I}^{n-1}\psi$ for every $n \in \mathbb{N}$.*

OBSERVATION A.4. *Equivalently, by Definition A.2,*

$$C_I\psi \quad = \bigcap_{(i_1,\ldots,i_n)\in I^*} K_{i_1}\cdots K_{i_n}\psi \quad = \bigcap_{(i_1,\ldots,i_n)\in I^{\bar{*}}} K_{i_1}\cdots K_{i_n}\psi,$$

where $I^{\bar{*}} \triangleq \{(i_1,\ldots,i_n) \in I^* \mid \forall m \in [n-1] : i_m \neq i_{m+1}\}$ denotes the set of all finite non-stuttering sequences of elements of I.

B. BACKGROUND: SYMMETRIC FORMS OF COORDINATION

In this section, we survey a few forms of coordination previously defined and analyzed by Halpern and Moses [17], as formulated for ensembles in [12, Section 11.6]. We reformulate these using events and adapt them to our notation.

B.1 Definitions

DEFINITION B.1 (PERFECT COORDINATION). *Let $R \subseteq \mathcal{R}$ and let $I \subseteq \mathbb{I}$. An I-ensemble $\bar{\boldsymbol{e}} \in {\mathcal{F}_R}^I$ is said to be **perfectly coordinated** (in R) if $\boldsymbol{e}_i = \boldsymbol{e}_j$ for every $i, j \in I$.*

DEFINITION B.2 (EVENTUAL COORDINATION [17, 12]). *Let $R \subseteq \mathcal{R}$ and let $I \subseteq \mathbb{I}$. An I-ensemble $\bar{\boldsymbol{e}} \in {\mathcal{F}_R}^I$ is said to be **eventually coordinated** (in R) if for every $i, j \in I$ and for every $(r,t) \in \boldsymbol{e}_i$, there exists $t' \in \mathbb{T}$ s.t. $(r,t') \in \boldsymbol{e}_j$.*

DEFINITION B.3 (ε-COORDINATION [17, 12]). *Let $R \subseteq \mathcal{R}$, let $I \subseteq \mathbb{I}$ and let $\varepsilon \geq 0$. An I-ensemble $\bar{\boldsymbol{e}} \in {\mathcal{F}_R}^I$ is said to be **$\boldsymbol{\varepsilon}$-coordinated** (in R) if for every $i \in I$ and for every $(r,t) \in \boldsymbol{e}_i$, there exists an interval $T \subseteq \mathbb{T}$ of length at most ε, s.t. $t \in T$ and s.t. for every $j \in I$ there exists $t' \in T$ s.t. $(r,t') \in \boldsymbol{e}_j$.*

B.2 Fixed-Point Analysis

While phrasing the propositions in this section, and henceforth, we use the shorthand notation $\cup\bar{\xi} \triangleq \bigcup_{i\in I} \xi_i$, for every I-ensemble $\bar{\xi} = (\xi_i)_{i\in I} \in {\mathcal{F}_R}^I$.

THEOREM B.4 ([17, 12]). *Let $R \subseteq \mathcal{R}$ and let $I \subseteq \mathbb{I}$.*

1. *$(C_I\psi)_{i\in I}$ constitutes a perfectly coordinated I-ensemble, for every $\psi \in \mathcal{F}_R$.*
2. *If $\bar{\boldsymbol{e}} \in {\mathcal{F}_R}^I$ is a perfectly-coordinated I-ensemble, then $\boldsymbol{e}_i \subseteq C_I(\cup\bar{\boldsymbol{e}})$ for every $i \in I$.*
3. *If $\bar{\boldsymbol{e}} \in {\mathcal{F}_R}^I$ is a perfectly-coordinated I-ensemble, then $\cup\bar{\boldsymbol{e}} = C_I(\cup\bar{\boldsymbol{e}})$.*

THEOREM B.5 ([17, 12]). *Let $R \subseteq \mathcal{R}$ and let $I \subseteq \mathbb{I}$.*

1. *For every $\psi \in \mathcal{F}_R$, the function $f^{\diamondsuit}_{\psi} : \mathcal{F}_R \to \mathcal{F}_R$ given by $x \mapsto \cap_{i\in I} \diamondsuit K_i(\psi \cap x)$ has a greatest fixed point, denoted by $C^{\diamondsuit}_I\psi$ — for **eventual common knowledge** of ψ by I.*
2. *$(K_iC^{\diamondsuit}_I\psi)_{i\in I}$ constitutes an eventually-coordinated I-ensemble, for every $\psi \in \mathcal{F}_R$.*
3. *If $\bar{\boldsymbol{e}} \in {\mathcal{F}_R}^I$ is an eventually-coordinated I-ensemble, then $\boldsymbol{e}_i \subseteq K_iC^{\diamondsuit}_I(\cup\bar{\boldsymbol{e}})$ for every $i \in I$.*
4. *If $\bar{\boldsymbol{e}} \in {\mathcal{F}_R}^I$ is an eventually-coordinated I-ensemble, then $\cup\bar{\boldsymbol{e}} \subseteq C^{\diamondsuit}_I(\cup\bar{\boldsymbol{e}})$.*

We note that for $\varepsilon \equiv 0$, ε-coordination is the same as perfect coordination, and thus the following theorem also implies Theorem B.4 as a special case thereof.

THEOREM B.6 ([12]). *Let $R \subseteq \mathcal{R}$, let $I \subseteq \mathbb{I}$ and let $\varepsilon \geq 0$. For every $\psi \in \mathcal{F}_R$, denote*

$$E^{\varepsilon}_I(\psi) \triangleq \left\{ (r,t) \in \Omega_R \;\middle|\; \begin{array}{c} \exists T \subseteq \mathbb{T} : \\ t \in T \;\;\&\;\; \sup\{T - T\} \leq \varepsilon \;\;\& \\ \forall i \in I \; \exists t' \in T : (r,t') \in K_i\psi \end{array} \right\}.$$

1. *For every $\psi \in \mathcal{F}_R$, the function $f^{\varepsilon}_{\psi} : \mathcal{F}_R \to \mathcal{F}_R$ given by $x \mapsto E^{\varepsilon}_I(\psi \cap x))$ has a greatest fixed point, denoted by $C^{\varepsilon}_I\psi$ — for **$\boldsymbol{\varepsilon}$-common knowledge** of ψ by I.*
2. *$(K_iC^{\varepsilon}_I\psi)_{i\in I}$ constitutes an ε-coordinated I-ensemble, for every $\psi \in \mathcal{F}_R$.*
3. *If $\bar{\boldsymbol{e}} \in {\mathcal{F}_R}^I$ is an ε-coordinated I-ensemble, then $\boldsymbol{e}_i \subseteq K_iC^{\varepsilon}_I(\cup\bar{\boldsymbol{e}})$ for every $i \in I$.*
4. *If $\bar{\boldsymbol{e}} \in {\mathcal{F}_R}^I$ is an ε-coordinated I-ensemble, then $\cup\bar{\boldsymbol{e}} \subseteq C^{\varepsilon}_I(\cup\bar{\boldsymbol{e}})$.*

C. PROOFS

C.1 Preliminaries

OBSERVATION C.1. *Let $R \subseteq \mathcal{R}$ and $i \in I$. By the positive introspection axiom, the event $K_i\psi$ is i-local for every $\psi \in \mathcal{F}_R$.*

DEFINITION C.2. *To aid the readability of the proofs below, we define $\Delta = \mathbb{Z} \cup \{\infty\}$ — the set of suprema of sets of time differences. (For every timely-coordination spec (I, δ), this is the range of δ. See the definition of a timely-coordination spec in Section 4 for more details.)*[12]

OBSERVATION C.3. *By definition of $\circledcirc^{\leq\varepsilon}$,*

- $\circledcirc^{\leq\infty} = \diamondsuit$.
- $\circledcirc^{\leq 0}\psi$ *means "ψ has occurred, either now or in the past".*
- *Additivity:* $\circledcirc^{\leq\varepsilon_1}\circledcirc^{\leq\varepsilon_2}\psi = \circledcirc^{\leq\varepsilon_1+\varepsilon_2}\psi$ *for every $\varepsilon_1, \varepsilon_2 \in \Delta$ and for every $\psi \in \mathcal{F}_R$.*
- *Monotonicity:* $(\varepsilon_1 \leq \varepsilon_2 \ \&\ \psi \subseteq \phi) \Rightarrow \circledcirc^{\leq\varepsilon_1}\psi \subseteq \circledcirc^{\leq\varepsilon_2}\phi$, *for every $\varepsilon_1, \varepsilon_2 \in \Delta$ and for every $\psi, \phi \in \mathcal{F}_R$.*
- $\circledcirc^{\leq\varepsilon}(\cap\Psi) \subseteq \bigcap\{\circledcirc^{\leq\varepsilon}\psi \mid \psi \in \Psi\}$, *for every $\varepsilon \in \Delta$ and for every set of events $\Psi \subseteq \mathcal{F}_R$.*

OBSERVATION C.4. *By definition of $\circledcirc^{\varepsilon}$, for every event $\psi \in \mathcal{F}_R$ we have:*

- $\circledcirc^{\varepsilon_1}\ \circledcirc^{\leq\varepsilon_2}\ \psi = \circledcirc^{\leq\varepsilon_1}\ \circledcirc^{\varepsilon_2}\ \psi = \circledcirc^{\leq\varepsilon_1+\varepsilon_2}\psi$, *for every $\varepsilon_1, \varepsilon_2 \in \Delta \setminus \{\infty\}$.*
- $\circledcirc^{\varepsilon}\psi \subseteq \circledcirc^{\leq\varepsilon}\psi$, *for every $\varepsilon \in \Delta \setminus \{\infty\}$.*
- $\circledcirc^{\varepsilon}$ *commutes with intersection for every $\varepsilon \in \Delta \setminus \{\infty\}$:* $\circledcirc^{\varepsilon}(\cap\Psi) = \bigcap\{\circledcirc^{\varepsilon}\psi \mid \psi \in \Psi\}$ *for every set of events $\Psi \subseteq \mathcal{F}_R$.*

C.2 Proofs of Propositions from Section 4

The soundness of our definition of timely common knowledge is based on the following part of Tarksi's celebrated theorem.

DEFINITION C.5 (COMPLETE LATTICE). *A lattice L is called* ***complete*** *if each subset $S \subseteq L$ has both a supremum (i.e. least upper bound, denoted $\bigvee S$) and an infimum (i.e. greatest lower bound, denoted $\bigwedge S$).*

THEOREM C.6 (TARSKI [29]). *Let L be a complete lattice. Every monotone function $f : L \to L$ has a greatest fixed point. Furthermore, this greatest fixed point is given by $\bigvee\{l \in L \mid l \leq f(l)\}$.*

OBSERVATION C.7. *${\mathcal{F}_R}^I$, equipped with the lattice structure from Definition 4.3, constitutes a complete lattice; the supremum of every subset of ${\mathcal{F}_R}^I$ is given by coordinate-wise union, and its infimum — by coordinate-wise intersection.*

PROOF OF LEMMA 4.5. By monotonicity of K_i for every $i \in \mathbb{I}$ and of $\circledcirc^{\leq\varepsilon}$ for every $\varepsilon \in \Delta$, we obtain that f^δ_ψ is monotone. By Observation C.7, and by Tarski's Theorem C.6, the set of fixed points of f^δ_ψ has a greatest element, which equals $\bigvee\{\bar{\xi} \in {\mathcal{F}_R}^I \mid \bar{\xi} \leq f^\delta_\psi(\bar{\xi})\}$. This proves both that $C^\delta_I\psi$ is well-defined (part 1 of the lemma) and the induction rule for timely common knowledge (part 2).

To prove monotonicity of C^δ_I (part 3), let $\psi, \phi \in \mathcal{F}_R$ s.t. $\psi \subseteq \phi$. Once again, by monotonicity of K_i for every $i \in \mathbb{I}$, we obtain that $f^\delta_\psi(\bar{\varphi}) \leq f^\delta_\phi(\bar{\varphi})$ for every $\bar{\varphi} \in {\mathcal{F}_R}^I$. By substituting $\bar{\varphi} \triangleq C^\delta_I\psi$, and by definition of $C^\delta_I\psi$, we obtain $C^\delta_I\psi = f^\delta_\psi(C^\delta_I\psi) \leq f^\delta_\phi(C^\delta_I\psi)$. By directly applying the induction rule for timely common knowledge with $\bar{\xi} \triangleq C^\delta_I\psi$, we obtain that $C^\delta_I\psi \leq C^\delta_I\phi$. □

PROOF OF THEOREM 4.6. We begin the proof of part 1 by noting that for every $i \in I$, by definition $C^\delta_I\psi = f^\delta_\psi(C^\delta_I\psi)$, and therefore $(C^\delta_I\psi)_i$ is of the form $K_i(\cdots)$. Hence, by Observation C.1, $C^\delta_I\psi$ is an I-ensemble. Let $(i,j) \in I^{\bar{2}}$ and $(r,t) \in (C^\delta_I\psi)_i$. By definition of C^δ_I and by the knowledge axiom,

$$(C^\delta_I\psi)_i \ = \ K_i\Big(\psi \cap \bigcap_{k\in I\setminus\{i\}} \circledcirc^{\leq\delta(i,k)}(C^\delta_I\psi)_k\Big) \ \subseteq$$
$$\subseteq \ \psi \cap \bigcap_{k\in I\setminus\{i\}} \circledcirc^{\leq\delta(i,k)}(C^\delta_I\psi)_k \ \subseteq \ \circledcirc^{\leq\delta(i,j)}(C^\delta_I\psi)_j.$$

Thus, we obtain that $(r,t) \in \circledcirc^{\leq\delta(i,j)}(C^\delta_I\psi)_j$. By definition of $\circledcirc^{\leq\delta(i,j)}$, there exists $t' \in \mathbb{T}$ such that $t' \leq t + \delta(i,j)$ and $(r,t') \in (C^\delta_I\psi)_j$, and the proof of part 1 is complete. Similarly, we have

$$(C^\delta_I\psi)_i \ = \ K_i\Big(\psi \cap \bigcap_{k\in I\setminus\{i\}} \circledcirc^{\leq\delta(i,k)}(C^\delta_I\psi)_k\Big) \ \subseteq$$
$$\subseteq \ \psi \cap \bigcap_{k\in I\setminus\{i\}} \circledcirc^{\leq\delta(i,k)}(C^\delta_I\psi)_k \ \subseteq \ \psi$$

for every $i \in I$, thus proving part 2 as well.

We move on to proving part 3. Let $\bar{\boldsymbol{e}}$ be a δ-coordinated I-ensemble s.t. $\cup\bar{\boldsymbol{e}} \subseteq \psi$. First, we show that $\bar{\boldsymbol{e}} \leq f^\delta_\psi(\bar{\boldsymbol{e}})$. Let $i \in I$. Let $(r,t) \in \boldsymbol{e}_i$ and let $j \in I \setminus \{i\}$. Since $\bar{\boldsymbol{e}}$ is δ-coordinated, there exists $t' \in \mathbb{T}$ s.t. $t' \leq t + \delta(i,j)$ and $(r,t') \in \boldsymbol{e}_j$. By definition of $\circledcirc^{\leq\delta(i,j)}$, we therefore obtain $(r,t) \in \circledcirc^{\leq\delta(i,j)}\boldsymbol{e}_j$. Thus, and since $\cup\bar{\boldsymbol{e}} \subseteq \psi$, we have

$$\boldsymbol{e}_i \ \subseteq \ \psi \cap \bigcap_{j\in I\setminus\{i\}} \circledcirc^{\leq\delta(i,j)}\boldsymbol{e}_j.$$

By definition of an ensemble, $\boldsymbol{e}_i$ is i-local, and thus $\boldsymbol{e}_i = K_i\boldsymbol{e}_i$. Hence, by monotonicity of K_i,

$$\boldsymbol{e}_i \ = \ K_i\boldsymbol{e}_i \ \subseteq \ K_i\Big(\psi \cap \bigcap_{j\in I\setminus\{i\}} \circledcirc^{\leq\delta(i,j)}\boldsymbol{e}_j\Big) \ = \ \big(f^\delta_\psi(\bar{\boldsymbol{e}})\big)_i.$$

By the induction rule for timely common knowledge, we thus have $\bar{\boldsymbol{e}} \leq C^\delta_I\psi$, completing the proof of part 3. Part 4 follows from part 3 by setting $\psi \triangleq \cup\bar{\boldsymbol{e}}$. Finally, one direction of part 5 follows from part 4 by taking the union of both sides, while the other follows by setting $\psi \triangleq \cup\bar{\boldsymbol{e}}$ in part 2. □

C.3 Proofs of Propositions from Section 5

C.3.1 Preliminaries

In order to harness the tools of Section 4 to analyzing timely-coordinated response, we introduce some machinery relating agent responses in a protocol $P \in \mathbb{P}$ to an ensemble in the space $\Omega_{R(P,\gamma)}$ defined by the set of runs of P. Recall that as mentioned above, we slightly abuse notation at times

[12] As noted above, we more generally define the set of time differences as $\Delta = (\mathbb{T} - \mathbb{T}) \cup \{\infty\}$. E.g. if $\mathbb{T} = \mathbb{R}_{\geq 0}$, then $\Delta = (-\infty, \infty]$.

when working in $\Omega_{R(P,\gamma)}$ for some protocol P, by writing ϕ to refer to $\phi \cap \Omega_{R(P,\gamma)}$.

DEFINITION C.8. *Let $\tau = (\gamma, \phi, I, \delta, \bar{\alpha})$ be a TCR and let $P \in \mathbb{P}$. We denote by $\boldsymbol{e}^{P_{\bar{\alpha}}} \in {\mathcal{F}_{R(P,\gamma)}}^I$ the I-ensemble $\boldsymbol{e}_i^{P_{\bar{\alpha}}} \triangleq \{(r,t) \in \Omega_{R(P,\gamma)} \mid i \text{ performs } \alpha_i \text{ at } (r,t) \text{ according to } P\}$, for every $i \in I$.*

OBSERVATION C.9. *Let $\tau = (\gamma, \phi, I, \delta, \bar{\alpha})$ be a TCR and let $P \in \mathbb{P}$. Since the actions of each agent $i \in I$ at each point are defined by its state at that point, it follows that $\boldsymbol{e}_i^{P_{\bar{\alpha}}}$ is i-local, and thus $\bar{\boldsymbol{e}}^{P_{\bar{\alpha}}}$ is indeed an I-ensemble.*

OBSERVATION C.10. *Let $\tau = (\gamma, \phi, I, \delta, \bar{\alpha})$ be a TCR. A protocol $P \in \mathbb{P}$ solves τ iff all the following hold in $\Omega_{R(P,\gamma)}$:*

- *$\boldsymbol{e}_i^{P_{\bar{\alpha}}}$ occurs* at most *once during each run $r \in R(P,\gamma)$, for every agent $i \in I$.*
- *$\bar{\boldsymbol{e}}^{P_{\bar{\alpha}}}$ is δ-coordinated.*
- *$\cup\bar{\boldsymbol{e}}^{P_{\bar{\alpha}}} \subseteq \circledcirc^{\leq 0}\phi$. (I.e. ϕ must occur before or when any response does.)*
- *$\phi \subseteq \diamondsuit\boldsymbol{e}_i^{P_{\bar{\alpha}}}$, for every $i \in I$. (I.e. all responses must occur at some point along any ϕ-triggered run.)*

OBSERVATION C.11. *Let $\tau = (\gamma, \phi, I, \delta, \bar{\alpha})$ be a TCR. A protocol $P \in \mathbb{P}$ is a time-optimal solution to τ iff it both solves τ and for every protocol P' solving τ that is run-equivalent to P, we have $\boldsymbol{e}_i^{P'_{\bar{\alpha}}} \subseteq \circledcirc^{\leq 0}\boldsymbol{e}_i^{P_{\bar{\alpha}}}$ in $\Omega_{R(P_{-\bar{\alpha}},\gamma)}$ for every $i \in I$.*

C.3.2 Proofs

PROOF OF COROLLARY 5.2. We must show that under the conditions of the corollary, $\bar{\boldsymbol{e}}^{P_{\bar{\alpha}}} \leq C_I^\delta(\circledcirc^{\leq 0}\phi)$ holds in ${\Omega_{R(P,\gamma)}}^I$. Since P solves τ, by Observation C.10 we have both that $\bar{\boldsymbol{e}}^{P_{\bar{\alpha}}}$ is δ-coordinated and that $\cup\bar{\boldsymbol{e}}^{P_{\bar{\alpha}}} \subseteq \circledcirc^{\leq 0}\phi$. Thus, by Theorem 4.6(3), we obtain $\bar{\boldsymbol{e}}^{P_{\bar{\alpha}}} \leq C_I^\delta(\circledcirc^{\leq 0}\phi)$, as required. □

The following somewhat technically-phrased lemma lies at the heart of Corollaries C.13 and 5.3, whose proofs follow below.

LEMMA C.12. *Let $\tau = (\gamma, \phi, I, \delta, \bar{\alpha})$ be a TCR and let $P_{-\bar{\alpha}}$ be a non-response component of a protocol such that $\phi \subseteq \diamondsuit\big(C_I^\delta(\circledcirc^{\leq 0}\phi)\big)_i$ holds in $\Omega_{R(P_{-\bar{\alpha}},\gamma)}$ for some $i \in I$. The protocol $P = (P_{-\bar{\alpha}}, P_{\bar{\alpha}})$ s.t. in $P_{\bar{\alpha}}$ each $i \in I$ responds at the first instant at which $\big(C_I^\delta(\circledcirc^{\leq 0}\phi)\big)_i$ holds (in $\Omega_{R(P_{-\bar{\alpha}},\gamma)}$), is a time-optimal solution for τ.*

PROOF. We note that by Theorem 4.6(1), $\big(C_I^\delta(\circledcirc^{\leq 0}\phi)\big)_j$ is j-local for every $j \in I$, and thus $P_{\bar{\alpha}}$ is well-defined.[13] We now show that P solves τ by showing that it satisfies all four conditions of Observation C.10.

By definition of $P_{\bar{\alpha}}$, for each $j \in I$ the event $\boldsymbol{e}_j^{P_{\bar{\alpha}}}$ occurs at most once during each $r \in R(P,\gamma)$. Let $(j,k) \in I^2$ and let $(r,t) \in \boldsymbol{e}_j^{P_{\bar{\alpha}}}$. By definition of $P_{\bar{\alpha}}$, we have that $(r,t) \in \big(C_I^\delta(\circledcirc^{\leq 0}\phi)\big)_j$. By Theorem 4.6(1), $C_I^\delta(\circledcirc^{\leq 0}\phi)$ is a δ-coordinated ensemble, and thus there exists $t' \leq t + \delta(j,k)$ s.t. $(r,t') \in \big(C_I^\delta(\circledcirc^{\leq 0}\phi)\big)_k$. By definition of $P_{\bar{\alpha}}$, there exists $t'' \leq t'$ s.t. $(r,t'') \in \boldsymbol{e}_k^{P_{\bar{\alpha}}}$. As $t'' \leq t' \leq t + \delta(j,k)$, we obtain that $\bar{\boldsymbol{e}}^{P_{\bar{\alpha}}}$ is δ-coordinated.

Let $j \in I$. By Observation C.3 (monotonicity), we conclude that $\boldsymbol{e}_i^{P_{\bar{\alpha}}} \subseteq \circledcirc^{\leq \delta(i,j)}\boldsymbol{e}_j^{P_{\bar{\alpha}}} \subseteq \diamondsuit\boldsymbol{e}_j^{P_{\bar{\alpha}}}$. By definition of $P_{\bar{\alpha}}$, we have $\big(C_I^\delta(\circledcirc^{\leq 0}\phi)\big)_i \subseteq \circledcirc^{\leq 0}\boldsymbol{e}_i^{P_{\bar{\alpha}}}$. By both of these, by the conditions of the lemma, and once again by Observation C.3 (monotonicity), we obtain $\phi \subseteq \diamondsuit\big(C_I^\delta(\circledcirc^{\leq 0}\phi)\big)_i \subseteq \diamondsuit\circledcirc^{\leq 0}\boldsymbol{e}_i^{P_{\bar{\alpha}}} \subseteq \diamondsuit\circledcirc^{\leq 0}\diamondsuit\boldsymbol{e}_j^{P_{\bar{\alpha}}} = \diamondsuit\boldsymbol{e}_j^{P_{\bar{\alpha}}}$. Finally, by definition of $P_{\bar{\alpha}}$ and by Theorem 4.6(2), we have $\cup\bar{\boldsymbol{e}}^{P_{\bar{\alpha}}} \subseteq \cup C_I^\delta(\circledcirc^{\leq 0}\phi) \subseteq \circledcirc^{\leq 0}\phi$, thus completing the proof of P solving τ.

We move on to show that P constitutes a time-optimal solution to τ. Let $P' = (P_{-\bar{\alpha}}, P'_{\bar{\alpha}})$ be a protocol solving τ that is run-equivalent to P. Let $j \in I$. By Corollary 5.2, we have $\boldsymbol{e}_j^{P'_{\bar{\alpha}}} \subseteq \big(C_I^\delta(\circledcirc^{\leq 0}\phi)\big)_j$. By definition of $P_{\bar{\alpha}}$, we have $\big(C_I^\delta(\circledcirc^{\leq 0}\phi)\big)_j \subseteq \circledcirc^{\leq 0}\boldsymbol{e}_j^{P_{\bar{\alpha}}}$. We combine these to obtain $\boldsymbol{e}_j^{P'_{\bar{\alpha}}} \subseteq \circledcirc^{\leq 0}\boldsymbol{e}_j^{P_{\bar{\alpha}}}$, and thus, by Observation C.11, the proof is complete. □

COROLLARY C.13. *Let $\tau = (\gamma, \phi, I, \delta, \bar{\alpha})$ be a TCR and let $P \in \mathbb{P}$. The following are equivalent:*

1. *P is run-equivalent to a protocol that solves τ.*
2. *$\phi \subseteq \diamondsuit\big(C_I^\delta(\circledcirc^{\leq 0}\phi)\big)_i$ in $\Omega_{R(P,\gamma)}$, for* every *$i \in I$.*
3. *$\phi \subseteq \diamondsuit\big(C_I^\delta(\circledcirc^{\leq 0}\phi)\big)_i$ in $\Omega_{R(P,\gamma)}$, for* some *$i \in I$.*

PROOF.

1 ⇒ 2: Let $i \in I$. Let P' be a protocol solving τ that is run-equivalent to P. Recall that $\Omega_{R(P',\gamma)} \simeq \Omega_{R(P,\gamma)}$. By Observation C.10, we have $\phi \subseteq \diamondsuit\boldsymbol{e}_i^{P'_{\bar{\alpha}}}$. By Corollary 5.2, we have $\boldsymbol{e}_i^{P'_{\bar{\alpha}}} \subseteq \big(C_I^\delta(\circledcirc^{\leq 0}\phi)\big)_i$. We combine these two with Observation C.3 (monotonicity) to obtain $\phi \subseteq \diamondsuit\big(C_I^\delta(\circledcirc^{\leq 0}\phi)\big)_i$.

2 ⇒ 3: Immediate.

3 ⇒ 1: Follows immediately from Lemma C.12, since $\Omega_{R(P,\gamma)} \simeq \Omega_{R(P_{-\bar{\alpha}},\gamma)}$. □

PROOF OF COROLLARY 5.3. By Corollary C.13(1 ⇒ 2), we have $\phi \subseteq \diamondsuit\big(C_I^\delta(\circledcirc^{\leq 0}\phi)\big)_i$ holding in $\Omega_{R(P,\gamma)} \simeq \Omega_{R(P_{-\bar{\alpha}},\gamma)}$. By Lemma C.12, the proof is complete. □

C.4 From Fixed-Point Definition to Nested-Knowledge Definition

C.4.1 Definitions and Propositions

In order to precisely phrase our nested-knowledge characterisation of timely common knowledge, we first introduce an additional definition.[14]

[13] In some runs of certain contexts under a continuous-time model, the set of times at which $\big(C_I^\delta(\circledcirc^{\leq 0}\phi)\big)_j$ holds does not attain its infimum value, and thus "$\big(C_I^\delta(\circledcirc^{\leq 0}\phi)\big)_j$ holds for the first time" is not necessarily a j-local event. To accommodate such cases, we may adapt the response component $P_{\bar{\alpha}}$ s.t. each $j \in I$ responds exactly 1 time unit after the infimum of times at which $\big(C_I^\delta(\circledcirc^{\leq 0}\phi)\big)_j$ holds. (It is straightforward to show that this is indeed a j-local event). The proof is easily adaptable to both show that this definition yields a solution for τ and to prove that in such pathological cases, no time-optimal solution for τ exists.

[14] As our notation $\mathcal{P}(G_\delta)$ may suggest, this is in fact the set of paths in a directed graph G_δ having I as vertices and with edges wherever $\delta < \infty$. For an in-depth graph-theoretic study of G_δ and of its elaborate relation to tuples of δ-coordinated timestamps, we refer the reader to [15] or to [16, Chapter 5]. For a study of the connection between the graph-theoretic properties of G_δ and the required delivery guarantees required to solve a TCR, we refer the reader to [16, Chapter 9].

DEFINITION C.14. *Let I be a set and let $\delta : I^{\bar{2}} \to \Delta$. We define*

$$\mathcal{P}(G_\delta) \triangleq \big\{(i_1,\ldots,i_n)\in I^{\bar{*}} \mid \forall m\in[n-1] : \delta(i_m,i_{m+1}) < \infty\big\}.$$

EXAMPLE C.15. *By the above definition, if $I = \{i,j\}$, then every element of $\mathcal{P}(G_\delta)$ is either $(\underbrace{i,j,i,j,i,j,\ldots}_{n})$ or $(\underbrace{j,i,j,i,j,i,\ldots}_{n})$, for some $n \in \mathbb{N}$. (If $|I| > 2$, then $\mathcal{P}(G_\delta)$ is much richer.)*

Second, we present a variation of a definition from [12, Chapter 4], which we utilize in this section.

DEFINITION C.16 (PERFECT RECALL). *A system $R \subseteq \mathcal{R}$ is said to exhibit **perfect recall** if for every $r \in R$, for every $i \in \mathbb{I}$ and for every $t \in \mathbb{T}$, the state of i at t in r uniquely determines the set $\big\{r_i(t') \mid t' \in \mathbb{T} \setminus [t,\infty)\big\}$ of states of i in r prior to t.*

OBSERVATION C.17. *If P^{fip}_γ is a full-information protocol in a context γ, then $R(P^{\text{fip}}_\gamma,\gamma)$ exhibits perfect recall.*

Third, we present a definition based upon [12, Chapter 4] and some basic properties thereof.

DEFINITION C.18 (STABILITY). *Let $R \subseteq \mathcal{R}$. An event $\psi \in \mathcal{F}_R$ is said to be **stable** if once ψ holds at some time during a run $r \in \mathcal{R}$, it continues to hold for the duration of r. Formally, using our notation, ψ is stable iff $\psi = \circledcirc^{\leq 0}\psi$.*

OBSERVATION C.19. *By Definition C.18,*

- *By Observation C.3 (additivity), $\circledcirc^{\leq 0}$ is idempotent. Thus, $\circledcirc^{\leq 0}\phi$ is a stable event for every $\phi \in \mathcal{F}_R$.*
- *$\psi \cap \phi$ is a stable event for any two stable events $\psi, \phi \in \mathcal{F}_R$.*

Indeed, since $\circledcirc^{\leq 0}\phi$ is stable for every ϕ, we do not lose much in the perspective of Section 5 if we restrict our study to timely common knowledge of stable events. We can now precisely phrase our constructive characterisation of timely common knowledge. See the following sections for a proof and a discussion of the various requirements of the following theorem.

THEOREM C.20. *Let (I,δ) be a timely-coordination spec, let $R \subseteq \mathcal{R}$ be a system exhibiting perfect recall and let $\psi \in \mathcal{F}_R$ be a stable event. Assume, furthermore, that either of the following holds:*

1. *$\delta < \infty$.*
2. *$R = R(P,\gamma)$, for some protocol P and context γ s.t. P either solves $(\gamma,\psi,I,\delta,\bar{\alpha})$ for some $\bar{\alpha}$, or is run-equivalent to a protocol that does.*

For every $i \in I$,

$$(C^\delta_I\psi)_i = \bigcap_{(i,i_2,\ldots,i_n)\in\mathcal{P}(G_\delta)} K_i \circledcirc^{\delta(i,i_2)} K_{i_2} \circledcirc^{\delta(i_2,i_3)} K_{i_3} \cdots \circledcirc^{\delta(i_{n-1},i_n)} K_{i_n}\psi \tag{2}$$

holds in Ω_R.

OBSERVATION C.21. *By Observation C.17, and since it is straitforward to show that a TCR is solvable iff it is solvable by a full-information protocol, condition 2 of Theorem C.20 is met if $R = R(P^{\text{fip}}_\gamma,\gamma)$, for a context γ admitting a full-information protocol P^{fip}_γ s.t. $(\gamma,\psi,I,\delta,\bar{\alpha})$ is solvable (by some protocol) for some $\bar{\alpha}$.*

COROLLARY C.22. *The time-optimal solution from Corollary 5.3, under (any of) the conditions of Theorem C.20 (with regard to $R \triangleq R(P_{-\bar{\alpha}},\gamma)$ and $\psi \triangleq \circledcirc^{\leq 0}\phi$), is for each agent $i \in I$ to respond at the first instant at which all nested-knowledge formulae of the form*

$$K_i \circledcirc^{\delta(i,i_2)} K_{i_2} \circledcirc^{\delta(i_2,i_3)} \cdots K_{i_{n-1}} \circledcirc^{\delta(i_{n-1},i_n)} K_{i_n} \circledcirc^{\leq 0} \phi$$

hold (in $\Omega_{R(P_{-\bar{\alpha}},\gamma)}$) for all $(i,i_2,\ldots,i_n) \in \mathcal{P}(G_\delta)$.

C.4.2 Background

In order to prove Theorem C.20, we perform an analysis of timely common knowledge of stable events. For reasons that will soon be apparent, we conduct this analysis under the assumption of perfect recall. To make our analysis somewhat cleaner and more generic, we first aim to distill the property of sets of runs exhibiting perfect recall that is of interest to us, namely that in such sets of runs, knowledge of a stable event is itself stable. The following is given in [12, Exercise 4.18(b)], and its proof follows directly from the definitions of stability and of knowledge.

CLAIM C.23. *Let $R \subseteq \mathcal{R}$ be a system exhibiting perfect recall and let $\psi \in \mathcal{F}_R$. If ψ is stable, then $K_i\psi$ is stable as well, for every $i \in \mathbb{I}$.*

C.4.3 Proof

Returning to our results and working toward proving Theorem C.20, we first derive a stability property for timely common knowledge (given in Claim C.25.)

CLAIM C.24. *Let $R \subseteq \mathcal{R}$ be a system exhibiting perfect recall. For every event $\psi \in \mathcal{F}_R$ and for every agent $i \in I$, it holds that $\circledcirc^{\leq 0}K_i\psi \subseteq K_i \circledcirc^{\leq 0}\psi$.*

PROOF. By Observation C.3, we have $\psi \subseteq \circledcirc^{\leq 0}\psi$. Thus, by monotonicity of $\circledcirc^{\leq 0}$ and of K_i, we have $\circledcirc^{\leq 0}K_i\psi \subseteq \circledcirc^{\leq 0}K_i \circledcirc^{\leq 0}\psi$. By Observation C.19, $\circledcirc^{\leq 0}\psi$ is stable, and therefore, by Claim C.23, $K_i \circledcirc^{\leq 0}\psi$ is stable as well, and thus equals $\circledcirc^{\leq 0}K_i \circledcirc^{\leq 0}\psi$, by applying Observation C.19 once more. We combine all these to obtain $\circledcirc^{\leq 0}K_i\psi \subseteq \circledcirc^{\leq 0}K_i \circledcirc^{\leq 0}\psi = K_i \circledcirc^{\leq 0}\psi$, as required. □

CLAIM C.25. *Let (I,δ) be a timely-coordination spec and let $R \subseteq \mathcal{R}$ be a set of runs exhibiting perfect recall. For every stable $\psi \in \mathcal{F}_R$, all coordinates of $C^\delta_I\psi$ are stable.*

PROOF. Let $i \in I$. By Definition C.18 and by Observation C.3, it is enough to show that $\circledcirc^{\leq 0}(C^\delta_I\psi)_i \subseteq (C^\delta_I\psi)_i$. Indeed, we have

$$\begin{aligned}
&\circledcirc^{\leq 0}(C^\delta_I\psi)_i = && \text{by definition of } C^\delta_I\\
&= \circledcirc^{\leq 0} K_i\Big(\psi \cap \bigcap_{j\in I\setminus\{i\}} \circledcirc^{\leq\delta(i,j)}(C^\delta_I\psi)_j\Big) \subseteq && \text{by Claim C.24}\\
&\subseteq K_i \circledcirc^{\leq 0}\Big(\psi \cap \bigcap_{j\in I\setminus\{i\}} \circledcirc^{\leq\delta(i,j)}(C^\delta_I\psi)_j\Big) \subseteq && \text{by Observation C.3}\\
&\subseteq K_i\Big(\circledcirc^{\leq 0}\psi \cap \bigcap_{j\in I\setminus\{i\}} \circledcirc^{\leq 0}\circledcirc^{\leq\delta(i,j)}(C^\delta_I\psi)_j\Big) \subseteq && \text{by Observation C.3 (additivity)}
\end{aligned}$$

$$\subseteq K_i\Big(\circledcirc^{\leq 0}\psi \cap \bigcap_{j\in I\setminus\{i\}} \circledcirc^{\leq\delta(i,j)}(C_I^\delta\psi)_j\Big) \subseteq$$

by stability of ψ

$$\subseteq K_i\Big(\psi \cap \bigcap_{j\in I\setminus\{i\}} \circledcirc^{\leq\delta(i,j)}(C_I^\delta\psi)_j\Big) =$$

by definition of C_I^δ

$$= (C_I^\delta\psi)_i.$$

□

Claims C.23 and C.25 lead us to consider, for stable ψ and given perfect recall, a slightly different definition for f_ψ^δ than the one given in Definition 4.4. This modified version of f_ψ^δ, which we denote by g_ψ^δ, differs by the use of $\circledcirc^{\delta(i,j)}$ in lieu of $\circledcirc^{\leq\delta(i,j)}$, and by not intersecting over eventual knowledge requirements.

Definition C.26. *Let (I,δ) be a timely-coordination spec and let $R\subseteq\mathcal{R}$. For each $\psi\in\mathcal{F}_R$, we define a function $g_\psi^\delta:\mathcal{F}_R^I\to\mathcal{F}_R^I$ by*

$$g_\psi^\delta:\quad (x_i)_{i\in I} \quad\mapsto\quad \left(K_i\Big(\psi\cap\bigcap_{\substack{j\in I\setminus\{i\}\\ \delta(i,j)<\infty}}\circledcirc^{\delta(i,j)}x_j\Big)\right)_{i\in I},$$

and denote its greatest fixed point by $\text{Ç}_I^\delta\psi$.

Using an argument completely analogous to the proof of Lemma 4.5, it may be shown that $\text{Ç}_I^\delta\psi$ is well-defined. Furthermore, the same argument shows that Ç_I^δ also satisfies the obvious analogues of the induction rule (with regard to g_ψ^δ) and of the monotonicity property from Lemma 4.5.

We now present a key observation, which stands at the heart of our proof of Theorem C.20. While, even in the presence of perfect recall and when ψ is stable, $g_\psi^\delta\neq f_\psi^\delta$ (e.g. when applied to certain unstable events), it so happens that under certain conditions, the greatest fixed points of both of these functions coincide.

Lemma C.27. *Let (I,δ) be a timely-coordination spec, let $R\subseteq\mathcal{R}$ be a set of runs exhibiting perfect recall and let $\psi\in\mathcal{F}_R$. Furthermore, assume that either $\psi\subseteq\circledast(C_I^\delta\psi)_i$ for every $i\in I$, or $\delta<\infty$. If ψ is stable, then $\text{Ç}_I^\delta\psi=C_I^\delta\psi$.*

Proof.

≥: For every $i\in I$, we have

$$(C_I^\delta\psi)_i = \quad\text{by definition of } C_I^\delta$$

$$= K_i\Big(\psi\cap\bigcap_{j\in I\setminus\{i\}}\circledcirc^{\leq\delta(i,j)}(C_I^\delta\psi)_j\Big)\subseteq$$

intersecting over fewer events

$$\subseteq K_i\Big(\psi\cap\bigcap_{\substack{j\in I\setminus\{i\}\\ \delta(i,j)<\infty}}\circledcirc^{\leq\delta(i,j)}(C_I^\delta\psi)_j\Big) =$$

by Observation C.4

$$= K_i\Big(\psi\cap\bigcap_{\substack{j\in I\setminus\{i\}\\ \delta(i,j)<\infty}}\circledcirc^{\delta(i,j)}\circledcirc^{\leq 0}(C_I^\delta\psi)_j\Big) =$$

by Claim C.25

$$= K_i\Big(\psi\cap\bigcap_{\substack{j\in I\setminus\{i\}\\ \delta(i,j)<\infty}}\circledcirc^{\delta(i,j)}(C_I^\delta\psi)_j\Big) =$$

by definition of g_ψ^δ

$$= \big(g_\psi^\delta(C_I^\delta\psi)\big)_i.$$

Thus, by the induction rule for Ç_I^δ and for g_ψ^δ, we obtain $C_I^\delta\psi\leq\text{Ç}_I^\delta\psi$, as required.

≤: For every $i\in I$, by monotonicity of K_i we have

$$(\text{Ç}_I^\delta\psi)_i = \quad\text{by definition of } \text{Ç}_I^\delta$$

$$= K_i\Big(\psi\cap\bigcap_{\substack{j\in I\setminus\{i\}\\ \delta(i,j)<\infty}}\circledcirc^{\delta(i,j)}(\text{Ç}_I^\delta\psi)_j\Big)\subseteq$$

by Observation C.4

$$\subseteq K_i\Big(\psi\cap\bigcap_{\substack{j\in I\setminus\{i\}\\ \delta(i,j)<\infty}}\circledcirc^{\leq\delta(i,j)}(\text{Ç}_I^\delta\psi)_j\Big)\subseteq$$

as $\psi\subseteq\circledast(C_I^\delta\psi)_j$ for every $j\in I$ (expression unchanged if $\delta<\infty$)

$$\subseteq K_i\Big(\psi\cap\bigcap_{\substack{j\in I\setminus\{i\}\\ \delta(i,j)=\infty}}\circledast(C_I^\delta\psi)_j\cap\bigcap_{\substack{j\in I\setminus\{i\}\\ \delta(i,j)<\infty}}\circledcirc^{\leq\delta(i,j)}(\text{Ç}_I^\delta\psi)_j\Big)\subseteq$$

by the other direction (≥) of this proof, and by monotonicity of $\circledast$

$$\subseteq K_i\Big(\psi\cap\bigcap_{\substack{j\in I\setminus\{i\}\\ \delta(i,j)=\infty}}\circledast(\text{Ç}_I^\delta\psi)_j\cap\bigcap_{\substack{j\in I\setminus\{i\}\\ \delta(i,j)<\infty}}\circledcirc^{\leq\delta(i,j)}(\text{Ç}_I^\delta\psi)_j\Big) =$$

as $\circledast=\circledcirc^{\leq\infty}$

$$= K_i\Big(\psi\cap\bigcap_{j\in I\setminus\{i\}}\circledcirc^{\leq\delta(i,j)}(\text{Ç}_I^\delta\psi)_j\Big) =$$

by definition of f_ψ^δ

$$= \big(f_\psi^\delta(\text{Ç}_I^\delta\psi)\big)_i.$$

Thus, by the induction rule for timely common knowledge, we have $\text{Ç}_I^\delta\psi\leq C_I^\delta\psi$.[15] □

One may wonder why we have worked so hard, and added the additional assumption of perfect recall (among others), to obtain $C_I^\delta\psi$, under the above assumptions, as a fixed point of g_ψ^δ rather than of f_ψ^δ. The answer is simple: g_ψ^δ commutes with the meet operation, while f_ψ^δ does not. (Moreover, as a result, g_ψ^δ is downward-continuous while f_ψ^δ, even in a

[15] It should be noted that we could have saved ourselves some hardship in this direction of the proof, by not intersecting over eventual knowledge requirements when defining f_ψ^δ. While this would still have allowed us to obtain some of our main results, such as Corollary 5.3, in this case many of our other results regarding timely common knowledge would have required the additional assumption that $\psi\subseteq\circledast(C_I^\delta\psi)_i$, reducing from their generality and usefulness. The added strength of the approach we have chosen presents itself not only while discussing eventual common knowledge in Appendix D.2, but in other settings [16, Section 9.3] as well.

discrete-time model, is generally not.) This fact paves our way toward proving Theorem C.20.

PROOF OF THEOREM C.20. The following proof applies to prove both parts of the theorem. As ψ is stable and R exhibits perfect recall, by Lemma C.27,[16] we obtain $C_I^\delta\psi = \mathcal{C}_I^\delta\psi$.

It is easy to verify that g_ψ^δ commutes with both finite and infinite meet. Thus, it is downward-continuous and by a well-known theorem popularly referred to as Kleene's fixed-point theorem[17], we obtain

$$\mathcal{C}_I^\delta\psi \;=\; \bigwedge_{n\in\mathbb{N}} (g_\psi^\delta)^n\,(\Omega_R{}^I). \tag{3}$$

Since $\circledcirc^\varepsilon$ commutes with intersection for every $\varepsilon \in \Delta$, and K_i commutes with intersection for every $i \in I$, we thus obtain, for every $i \in I$, that

$$\begin{aligned}
&(C_I^\delta\psi)_i = \\
&= \bigcap_{n\in\mathbb{N}} \Big((g_\psi^\delta)^n\,(\Omega_R{}^I)\Big)_i \;= \\
&= K_i\psi \cap K_i\Big(\psi \cap \bigcap_{\substack{i_2\in I\setminus\{i\}\\ \delta(i,i_2)<\infty}} \circledcirc^{\delta(i,i_2)} K_{i_2}\psi\Big) \cap \\
&\quad \cap K_i\Big(\psi \cap \bigcap_{\substack{i_2\in I\setminus\{i\}\\ \delta(i,i_2)<\infty}} \circledcirc^{\delta(i,i_2)} K_{i_2}\Big(\psi \cap \bigcap_{\substack{i_3\in I\setminus\{i_2\}\\ \delta(i_2,i_3)<\infty}} \circledcirc^{\delta(i_2,i_3)} K_{i_3}\psi\Big)\Big) \cap \\
&\quad \cap \cdots = \\
&= K_i\psi \cap K_i\Big(\bigcap_{\substack{i_2\in I\setminus\{i\}\\ \delta(i,i_2)<\infty}} \circledcirc^{\delta(i,i_2)} K_{i_2}\psi\Big) \cap \\
&\quad \cap K_i\Big(\bigcap_{\substack{i_2\in I\setminus\{i\}\\ \delta(i,i_2)<\infty}} \circledcirc^{\delta(i,i_2)} K_{i_2}\Big(\bigcap_{\substack{i_3\in I\setminus\{i_2\}\\ \delta(i_2,i_3)<\infty}} \circledcirc^{\delta(i_2,i_3)} K_{i_3}\psi\Big)\Big) \cap \\
&\quad \cap \cdots = \\
&= \bigcap_{(i,i_2,\ldots,i_n)\in\mathcal{P}(G_\delta)} K_i \circledcirc^{\delta(i,i_2)} K_{i_2} \circledcirc^{\delta(i_2,i_3)} K_{i_3} \cdots \circledcirc^{\delta(i_{n-1},i_n)} K_{i_n}\psi.
\end{aligned}$$

□

C.4.4 Discussion

Theorem C.20, which we have just proved, hinges on quite a few conditions, especially when $\delta \not< \infty$. The two conditions that are required even when $\delta < \infty$, namely perfect recall and stability of ψ, allow us to define g_ψ^δ using $\circledcirc^{\delta(i,j)}$ instead of $\circledcirc^{\leq\delta(i,j)}$. Without this modification, g_ψ^δ would not commute with intersection, resulting, instead of (2), in

$$\bigcap_{(i,i_2,\ldots,i_n)\in\mathcal{P}(G_\delta)} K_i\Big(\psi\cap\circledcirc^{\delta(i,i_2)} K_{i_2}\big(\cdots\psi\cap\circledcirc^{\delta(i_{n-1},i_n)} K_{i_n}(\psi)\cdots\big)\Big).$$

When $\delta \not< \infty$, it is condition 2 of Theorem C.20 that allows us to define g_ψ^δ without intersecting over eventual knowledge requirements. Without this modification, g_ψ^δ would not be downward-continuous. (This would also have been the case, had g_ψ^δ been defined using $\circledcirc^{\leq\delta(i,j)}$ instead of $\circledcirc^{\delta(i,j)}$ when under a continuous-time model.) Without downward continuity of g_ψ^δ, Kleene's fixed-point theorem could not have been utilized, forcing us to go beyond the "ω'th power" of g_ψ^δ in the r.h.s. of (3), to a greater ordinal power thereof [1, Theorem 1.2.11]. Incidentally, this may be viewed as a concrete example, of sorts, of Barwise's statement in [3] regarding various definitions of common knowledge, according to which in some models, taking only the intersection of finite approximations (i.e. only the results of finitely-many iterations of the relevant function f, starting from the top of the lattice) yields a weaker state of knowledge than the fixed-point of f, which is equivalent to taking the intersection of all (i.e. including transfinite) approximations.

We conclude this section with an observation. For certain δ functions, $\mathcal{P}(G_\delta)$ is finite,[18] and thus the intersection in (2) is finite. (See the discussion of ordered response in Appendix D.1 for an example.) This observation may seem, at first glance, to clash with the infinitary nature of fixed points in general, and of greatest fixed points in particular. It is worthwhile to note that what reconciles these is that in this case, $(g_\psi^\delta)^{|I|}$ is constant and therefore its value, which is a finite intersection of nested-knowledge events, is its only fixed point, and thus its greatest fixed point, and hence the greatest fixed point of g_ψ^δ as well. Furthermore, by Corollary C.13, solvability of τ implies that $\psi \subseteq \diamondsuit(C_I^\delta\psi)_i$ for every $i \in I$ and thus, as noted above, Corollaries 5.2 and 5.3 would have still held had we defined f_ψ^δ without intersecting over eventual-knowledge requirements (i.e. similarly to g_ψ^δ, but using $\circledcirc^{\leq\delta(i,j)}$ in lieu of $\circledcirc^{\delta(i,j)}$). In this case, the function $(f_\psi^\delta)^{|I|}$ would have also been constant, and thus, similarly, its value would have been its greatest fixed point, and thus the greatest fixed point of f_ψ^δ as well.

D. COMPARISON TO, AND DERIVATION OF PREVIOUS RESULTS

In this section, we show how some previously-known results may be derived from the novel results we have introduced in this paper.

D.1 Response Problems

In this section, we survey the response problems defined and studied by Ben-Zvi and Moses [5, 4, 6, 7], and their knowledge-theoretic results for these problems. We reformulate these problems, their results, and the associated definitions to match our notation, and show how our definitions and results from Section 5 extend each one of these, even though the tools used to derive our results are vastly different than their tools. This provides us with a "sanity check" of sorts, verifying that we have not committed the sin of generalizing our tools to the extent of weakening the results they yield for simple cases.

The first, most-basic response problem defined in [5] is that of *ordered response*. In this problem, finitely many agents $I = \{i_m\}_{m=1}^n$ must respond to an event in a pre-

[16] For the proof given condition 2, at this point we also use the fact that by Corollary C.13, we have that $\psi \subseteq \diamondsuit\big(C_I^\delta(\circledcirc^{\leq 0}\psi)\big)_i = \diamondsuit(C_I^\delta\psi)_i$ for every $i \in I$.

[17] See [23] for an investigation of the origins of this theorem, see [20, p. 348] for Kleene's first recursion theorem and for its proof that implies this theorem, and see [21] or [1, Theorem 1.2.14] for a statement of this theorem in terms of lattices, continuity and greatest fixed points.

[18] This happens iff both $|I| < \infty$ and G_δ has only trivial (i.e. singleton) strongly connected components.

defined order: i_{m+1} may not respond before i_m does. Using our notation, this is a special case of timely-coordinated response, for

$$\delta(i_k, i_l) \triangleq \begin{cases} 0 & k = l+1 \\ \infty & \text{otherwise.} \end{cases} \tag{4}$$

For this problem, they have shown that whenever an agent i_m responds in a solving protocol exhibiting perfect recall, it holds that

$$K_m K_{m-1} \cdots K_1 \circledcirc^{\leq 0} \phi, \tag{5}$$

and that in a full-information protocol, for each i_m to respond as soon as (5) holds constitutes what we have defined as a time-optimal solution.[19]

By Theorem C.20, we have, for δ as defined in (4), that when τ is solvable and in the presence of perfect recall (e.g. in a full-information protocol),

$$\left(C_I^\delta(\circledcirc^{\leq 0}\phi)\right)_m = \bigcap_{\substack{k\in\mathbb{N} \\ 1\leq i_1<\cdots<i_k\leq m}} K_{i_k}\cdots K_{i_1} \circledcirc^{\leq 0} \phi = K_m K_{m-1}\cdots K_1 \circledcirc^{\leq 0} \phi.$$

Thus, for ordered response, Corollaries 5.2 and 5.3 reduce to the above results.

The second problem presented in [5, 4] is a variant of the firing squad problem [27, 10] called *simultaneous response*. In this problem, all agents I must respond to an event simultaneously. Using our notation, this is a special case of timely-coordinated response, for $\delta \equiv 0$. For this problem, they have shown that $C_I \circledcirc^{\leq 0} \phi$ is the associated state of knowledge, in the same sense as above, i.e. when the agents respond (in a solving protocol exhibiting perfect recall), they share common knowledge of the fact that ϕ has occurred, and for each agent to respond as soon as she knows that common knowledge of $\circledcirc^{\leq 0}\phi$ has been attained constitutes a time-optimal solution (for a full-information protocol, when the problem is solvable). Once again, Corollaries 5.2 and 5.3 reduce to the above results under the above assumptions, since by Theorem C.20 and by Observation A.4, we have $\left(C_I^\delta(\circledcirc^{\leq 0}\phi)\right)_i = K_i C_I \circledcirc^{\leq 0} \phi = C_I \circledcirc^{\leq 0} \phi$, for $\delta \equiv 0$.

The third and last problem presented in [4] is a generalization of both ordered response and simultaneous response, called *ordered joint response*. In this problem, the agents are partitioned into pairwise-disjoint sets $I = \bigcup_{m=1}^{n} I_m$, and the agents in each such set must respond simultaneously, s.t. the agents in a set I_{m+1} may not respond before the agents in I_m do. Under our notation, this is a special case of timely-coordinated response, for

$$\delta(i,j) \triangleq \begin{cases} 0 & \exists k \in [n] : \{i,j\} \subseteq I_k \quad \text{or} \\ & \exists k \in [n-1] : i \in I_{k+1} \ \ \& \ \ j \in I_k \\ \infty & \text{otherwise.} \end{cases} \tag{6}$$

For this problem, they have shown that the associated state of knowledge, in the above sense, for an agent $i \in I_m$, is $C_{I_m} C_{I_{m-1}} \cdots C_{I_1} \circledcirc^{\leq 0} \phi$.

By Theorem C.20, by Observation A.4 and by K_j commuting with intersection for every $j \in I$, we have, in this case, for δ as defined in (6) and for $i \in I_m$,

$$\begin{aligned} \left(C_I^\delta(\circledcirc^{\leq 0}\phi)\right)_i &= \\ &= \bigcap_{(i,i_2,\ldots,i_n)\in\mathcal{P}(G_\delta)} K_i K_{i_2}\cdots K_{i_n} \circledcirc^{\leq 0} \phi = \\ &= K_i \circ \Big(\bigcap_{(i_1,\ldots,i_n)\in {I_m}^{\circledast}} K_{i_1}K_{i_2}\cdots K_{i_n} \Big) \circ \Big(\bigcap_{(i_1,\ldots,i_n)\in (I_{m-1})^{\circledast}} K_{i_1}K_{i_2}\cdots K_{i_n} \Big) \circ \\ &\qquad \circ \cdots \circ \Big(\bigcap_{(i_1,\ldots,i_n)\in {I_1}^{\circledast}} K_{i_1}K_{i_2}\cdots K_{i_n} \Big) \circledcirc^{\leq 0} \phi = \\ &= K_i C_{I_m} C_{I_{m-1}} \cdots C_{I_1} \circledcirc^{\leq 0} \phi = \\ &= C_{I_m} C_{I_{m-1}} \cdots C_{I_1} \circledcirc^{\leq 0} \phi. \end{aligned}$$

Thus, once more, Corollaries 5.2 and 5.3 reduce to the above results in this case as well.

The analogous results of Ben-Zvi and Moses for the rest of the response problems that they define (general ordered response [4], weakly-timed response [7] and tightly-timed response [7]) may be readily derived from our results in a similar manner — the details are left for the reader.

Having surveyed all the above response problems, one property, which is common to all of them (as well as to the rest of the response problems defined by Ben-Zvi and Moses) should be spelled out explicitly: they are all representable as special cases of timely-coordinated response, using δ s.t. for each $(i,j) \in I^2$, either $\delta(i,j) = \infty$, or $\delta(j,i) = \infty$, or $\delta(i,j) = -\delta(j,i)$, i.e. the difference between the response times of i and j is bounded either from one side at most, or tightly (i.e. specified exactly). We note that the absence of this property in timely-coordinated response introduced a significant amount of complexity into our analysis, both technically and conceptually, and that without it, the machinery with which we analyzed timely-coordinated response could have been significantly simplified. Incidentally, for an analysis of timely-coordinated response that follows and extends the synchronous causality ("syncausality") approach of Ben-Zvi and Moses for analyzing response problems (and which makes this statement about the complexity introduced by an arbitrary δ function more concrete), the reader is referred to [16, Chapter 6].

D.2 Common Knowledge and Variants

For the duration of this section, fix a system $R \subseteq \mathcal{R}$, an event $\psi \in \mathcal{F}_R$ and a set of agents $I \subseteq \mathbb{I}$. As noted above, while all previously-studied variants of common knowledge that are surveyed in Appendix B (and other previously-studied variants of common knowledge, such as continuous common knowledge [18]) are defined as fixed points of functions on $\mathcal{F}_R$, this is not the case with timely common knowledge, which we define as a fixed point of a function on ${\mathcal{F}_R}^I$. Intuitively, as noted above, this stems from the asymmetry of timely coordination with regard to the requirements posed on the various agents. Given this intuition, one may expect δ-common knowledge to reduce, for constant δ (i.e. symmetric constraints), to a non-tuple fixed point in some way, and to coincide in some sense with the previously-studied variants of common knowledge surveyed above. To show this, we first note that $C_I^\delta \psi = (K_i(\psi \cap \xi_i))_{i\in I}$, where $\bar{\xi}$ is the greatest fixed point of the function $\tilde{f}_\psi^\delta : \mathcal{F}_R^I \to \mathcal{F}_R^I$ given by

$$\tilde{f}_\psi^\delta : \ (x_i)_{i\in I} \ \mapsto \ \left(\bigcap_{j\in I\setminus\{i\}} \circledcirc^{\leq \delta(i,j)} K_j(\psi \cap x_j) \right)_{i\in I}.$$

[19] Throughout their analysis, Ben-Zvi and Moses implicitly assume that the problems they consider are solvable.

Next, we note that if indeed δ is a constant function attaining a nonnegative value, then it is straightforward to verify that $K_i(\psi \cap \xi_i) = K_i(\psi \cap (\cap\bar{\xi}))$ for every $i \in I$ (for $\bar{\xi}$ as defined above), yielding $C_I^\delta\psi = (K_i(\psi \cap (\cap\bar{\xi})))_{i\in I}$. Moreover, in this case $\cap\bar{\xi}$ is the greatest fixed point of $\cap\tilde{f}_\psi^\delta$. We now review the previously studied non-tuple variants of common knowledge surveyed above, and discuss when, and how, the above-described special case of δ-common knowledge for constant δ generalizes them.

When $\delta \equiv \infty$, then by definition, δ-coordination is equivalent to eventual coordination, $\cap\tilde{f}_\psi^\delta$ is the function presented in Theorem B.5(1), and thus $C_I^\delta\psi = (K_i(\psi \cap C_I^{\diamondsuit}\psi))_{i\in I}$. In addition, in this case Theorem 4.6(1,4,5) implies Theorem B.5.

Reducing our results for timely common knowledge to ε-common knowledge is somewhat more delicate. Assume, for the remainder of this section, that $\delta \equiv \varepsilon$ for some finite $\varepsilon \geq 0$. (Recall that for $\varepsilon = 0$, ε-coordination is equivalent to perfect coordination and Theorem B.6 reduces to Theorem B.4.)

In general, ε-coordination is a stricter condition than δ-coordination.[20] However, for a (coordinate-wise) stable ensemble, as well as for an ensemble consisting at most of one point per agent per run, δ-coordination is equivalent to ε-coordination — this follows from observing that given a δ-coordinated ensemble, taking only the first point (or in a continuous-time model, the infimal point) of each agent in each run (and no points for runs in which the original ensemble contained no points for said agent) yields an ε-coordinated (and hence also δ-coordinated) ensemble. If we restrict ourselves to stable ψ and to protocols exhibiting perfect recall, then by Claim C.25, every coordinate of $C_I^\delta\psi$ is stable. Under these conditions, it may be verified that $K_iC_I^\varepsilon\psi$, for every $i \in I$, is stable as well.[21] In this case, by Lemma C.27, $C_I^\delta\psi$ is the greatest fixed point of g_ψ^δ and thus, $C_I^\delta\psi = (K_i(\psi \cap \xi))_{i\in I}$, where ξ is the greatest fixed point of $\cap\tilde{g}_\psi^\delta$, where $\tilde{g}_\psi^\delta$ is defined analogously to $\tilde{f}_\psi^\delta$, but using $\circledcirc^{\delta(i,j)}$ in lieu of $\circledcirc^{\leq\delta(i,j)}$. Analogously to the proof of Lemma C.27, but in a less cumbersome way (as $\delta < \infty$), it may be shown that in this case $C_I^\varepsilon\psi$ is the greatest fixed point of $\cap\tilde{g}_\psi^\delta$ as well, and thus $C_I^\delta\psi = (K_i(\psi \cap C_I^\varepsilon\psi))_{i\in I}$,[22] and hence Theorem 4.6(1,4,5) reduces to Theorem B.6. In the absence of stability of ψ, or in the absence of perfect recall (at least of the "relevant events"), things stop working so well. Indeed, as noted above, in such cases δ-coordination does not necessarily coincide with ε-coordination, and consequently, examples may be constructed in which $C_I^\delta\psi \neq (K_i(\psi \cap C_I^\varepsilon\psi))_{i\in I}$.

The above discussion raises an interesting question: why have we not defined $C_I^\delta\psi$ as $(K_i\xi_i)_{i\in I}$ instead of defining it as $(K_i(\psi \cap \xi_i))_{i\in I}$? (for $\bar{\xi}$ the greatest fixed point of $\tilde{f}_\psi^\delta$.) Indeed, the connection between such a definition and the previously-studied variants of common knowledge is much cleaner to describe [16, Chapter 10], and it yields results broadly similar to those presented in this paper [16, Chapters 7,8]. Nonetheless, much like $(K_iC_I^\varepsilon\psi)_{i\in I}$, and somewhat like $(K_iC_I^{\diamondsuit}\psi)_{i\in I}$, such a definition does not seem to naturally lend to a characterisation along the lines of "the greatest δ-coordinated ensemble contained in ψ",[23] making it more cumbersome to use than the definition we presented in Section 4.

[20] This stems from two main "reasons":

1. δ-coordination is defined using $\circledcirc^{\leq\delta(i,j)}$ rather than $\circledcirc^{[-\delta(j,i),\delta(i,j)]}$, which we define to mean "at some time no earlier than $-\delta(j,i)$ from now and no later than $\delta(i,j)$ from now". It may be readily verified that all the results in this paper hold for such a definition as well, as long as this replacement is performed in the definition of f_ψ^δ as well. The only difference is that Claim C.25, stating that δ-common knowledge is stable, yields to different proof strategies in this case, e.g. showing that $(\circledcirc^{\leq 0}(C_I^\delta\psi)_i)_{i\in I} \leq f_\psi^\delta((\circledcirc^{\leq 0}(C_I^\delta\psi)_i)_{i\in I})$ and applying the induction rule for timely common knowledge.
2. Timely coordination is based on pairwise constraints. The results presented in this paper may be quite readily generalized to deal with arbitrary timing constraints of various natures, such as, e.g. for some $J \subseteq I$, "For every $i \in J$ and for every $(r,t) \in \boldsymbol{e}_i$, there exists a time interval $T \subseteq \mathbb{T}$ of length at most δ_J, s.t. $t \in T$ and s.t. there exist $(t_j)_{j\in J} \in T^J$ satisfying $(r,t_j) \in \boldsymbol{e}_j$ for every $j \in J$". (Whatever the timing constraints are, the generalized definition of $f_{\psi_i}^\delta$ simply intersects on all constraints pertaining to i.) Under such a generalization, ε-coordination is equivalent to δ-coordination, when setting $\delta_I \equiv \varepsilon$ in the above constraint example, and when providing no further constraints. Furthermore, in this case the generalization of $\tilde{f}_\psi^\delta$ satisfies that $\cap\tilde{f}_\psi^\delta$ is the function presented in Theorem B.6(1), and thus the appropriate generalization of Theorem 4.6(1,4,5) reduces to Theorem B.6.

It remains to be seen whether such generalizations as described in this footnote are of any real added value.

[21] This may be proved by showing that given stability of ψ and perfect recall, it holds that $\circledcirc^{\leq 0}C_I^\varepsilon\psi \subseteq E_I^\varepsilon(\psi \cap \circledcirc^{\leq 0}C_I^\varepsilon\psi)$.

[22] Another way to derive this equality is by using [12, Exercise 11.17(d)], which shows that, for every $i \in I$, if ψ is stable and given perfect recall, $K_iC_I^\varepsilon\psi = K_i(\cap_{n\in\mathbb{N}}(\circledcirc^\varepsilon E_I)^n\psi)$, and to apply (2). It should be noted, though, that the proof hinted to by [12, Exercise 11.17(d)] strongly relies on a discrete modeling of time, and breaks down in a continuous-time model, unlike the proof that we sketch above.

[23] While the ensemble defined by eventual common knowledge of an event of the form $\diamondsuit\psi$ is the greatest eventually-coordinated I-ensemble $\bar{\boldsymbol{e}}$ satisfying $\cup\bar{\boldsymbol{e}} \subseteq \diamondsuit\psi$ (the proof of this statement is left to the reader), we note that analogous characterisations for the ensembles defined by ε-common knowledge and by eventual common knowledge (of events not necessarily of the form $\diamondsuit\psi$) are, however, more elusive to phrase. (Moreover, parts 2–4 of Theorems B.5 and B.6 do not uniquely define these variants of common knowledge either.) In contrast, we note that $(K_i(\psi \cap C_I^\varepsilon\psi))_{i\in I}$ (resp. $(K_i(\psi \cap C_I^{\diamondsuit}\psi))_{i\in I}$) may be naturally characterised as the greatest ε-coordinated (resp. eventually-coordinated) I-ensemble whose union is contained in ψ.

Ceteris Paribus Structure in Logics of Game Forms

Davide Grossi
Department of Computer Science
University of Liverpool

Emiliano Lorini
IRIT-CNRS, Université Paul Sabatier
Toulouse, France

François Schwarzentruber
ENS Cachan - Brittany extension - IRISA

ABSTRACT

The article introduces a ceteris paribus modal logic interpreted on the equivalence classes induced by sets of propositional atoms. This logic is used to embed two logics of agency and games, namely atemporal STIT and the coalition logic of propositional control (CL−PC). The embeddings highlight a common ceteris paribus structure underpinning the key modal operators of both logics, they clarify the relationship between STIT and CL−PC, and enable the transfer of complexity results to the ceteris paribus logic.

Keywords

Ceteris Paribus Reasoning, Game Logics, STIT logic, coalition logic of propositional control, satisfiability problem, complexity.

1. INTRODUCTION

In a strategic game, the α-effectivity of a set of players consists in those sets of outcomes of the game for which the players have some collective action which forces the game to end up in that set, no matter what the other players do [MP82]. So, if a set of outcomes X belongs to the α-effectivity of a set of players J, then each agent in J can fix an individual action such that, for all actions of the other players, the game will end up in X.

It was already observed in [vBGR09] that the sort of reasoning underlying the notion of α-effectivity is of a ceteris paribus nature. Evaluating the outcomes that can be reached in a game once a set of players J has fixed their actions, amounts to considering what necessarily will be the case under the ceteris paribus condition 'all current actions of J being equal'. It has been shown in [vBGR09] how this intuition can be used, for instance, to give a modal formulation of Nash equilibria.

The present paper builds on that idea and systematically explores the ceteris paribus structure of two main logics of agency and games based on the α-effectivity concept: STIT [BPX01, Hor01] (the logic of *seeing to it that*) in its atemporal version [HS08], and the coalitional logic of propositional control (CL−PC) [vdHW05]. To articulate the analysis, whose main tool will consist of embedding results, the paper introduces and studies a simple ceteris paribus logic based on propositional equivalence.

TARK 2013, Chennai, India.

Structure of the paper.
Section 2 introduces a logic called *propositional equivalence ceteris paribus logic* (PECP in short), which will be used as yardstick to analyze the game logics addressed in the paper. The logic will be axiomatized and briefly compared with existing modal logics of *ceteris paribus* reasoning.
Section 3 provides a study of the relationship between the atemporal version of STIT and PECP. We show that PECP embeds atemporal group STIT—the fragment of atemporal STIT in which both actions of individuals and groups are represented—under the assumption that the agents' choices are bounded. We call the latter atemporal 'bounded' group STIT. Moreover, we show that PECP embeds atemporal individual STIT—the variant of atemporal STIT in which only the actions of individuals are represented. The former embedding is used to transfer complexity results to PECP. We also present an embedding in PECP of a variant of atemporal group STIT in which groups are nested (i.e., given two sets of agents J and J' either $J \subseteq J'$ or viceversa).
Section 4 provides an embedding of coalition logic of propositional control into atemporal 'bounded' group STIT and, indirectly, it provides an embedding of coalition logic of propositional control into PECP.
We conclude in Section 5. Longer proofs are collected in a technical appendix at the end of the paper.

2. A CETERIS PARIBUS LOGIC BASED ON PROPOSITIONAL EQUIVALENCE

2.1 Equivalence modulo a set of atoms

Consider a structure (W, V) where W is a set of states, and $V : \mathbf{P} \longrightarrow 2^W$ a valuation function from a countable set of atomic propositions $\mathbf{P}$ to subsets of W. We define a simple notion of propositional equivalence between states in W, modulo subsets of $\mathbf{P}$.

DEFINITION 1. *(Equivalence modulo X) Given a pair (W, V), $X \subseteq \mathbf{P}$ and $|X| < \omega$, the relation $\sim_X \subseteq W^2$ is defined as:*

$$w \sim_X^V w' \iff \forall p \in X : \big(w \in V(p) \Longleftrightarrow w' \in V(p)\big)$$

When X is a singleton (e.g. p), we will often write $\sim_p^V$ instead of $\sim_{\{p\}}^V$. Also, in order to avoid clutter, we will often drop the reference to V in $\sim_X^V$.

Intuitively, two states w and w' are equivalent up to set X, or X-equivalent, if and only if they satisfy the same atoms in X (according to a given valuation V). The finiteness of X is clearly not essential in the definition. It is assumed

because, as we will see, each set X will be taken to model a set of actions of some agent in a game form and sets of actions are always assumed to be finite.

We state the following simple fact without proof.

FACT 1. *(Properties of $\sim_P$) The following holds for any set of states W, valuation $V : \mathbf{P} \longrightarrow 2^W$ and finite sets $X, Y \subseteq \mathbf{P}$:*

(i) $\sim_X$ is an equivalence relation on W;

(ii) if $X \subseteq Y$ then $\sim_Y \subseteq \sim_X$;

(iii) if X is a singleton, $\sim_X$ induces a bipartition of W;

(iv) $\sim_X \cap \sim_Y = \sim_{X \cup Y}$;

(v) $\sim_\emptyset = W^2$.

2.2 A modal logic of $\sim_X$

In this section we consider a simple modal language interpreted on relations $\sim_X$ and axiomatize its logic on the class of structures (W, V). The key modal operator of the language will be $\langle X \rangle$, whose intuitive meaning is 'φ is the case in some state which is X-equivalent to the current one' or, to stress a ceteris paribus reading, 'φ is possible *all things expressed in X being equal*'. We call the resulting logic *propositional equivalence ceteris paribus logic*, PECP in short.

2.2.1 Syntax of PECP.

Let $\mathbf{P}$ be a countable set of atomic propositions. The language $\mathcal{L}_{\mathsf{PECP}}(\mathbf{P})$ is defined by the following BNF:

$$\mathcal{L}_{\mathsf{PECP}}(\mathbf{P}) : \varphi ::= p \mid \neg\varphi \mid (\varphi \wedge \varphi) \mid \langle X \rangle \varphi$$

where p ranges over $\mathbf{P}$ and X is a finite subset of atomic propositions ($X \subseteq \mathbf{P}$ and X finite). Note that as the set of finite subsets of atomic propositions is countable, the language $\mathcal{L}_{\mathsf{PECP}}(\mathbf{P})$ is also countable. The Boolean connectives $\top, \vee, \rightarrow, \leftrightarrow$ and the dual operators $[X]$ are defined as usual.

The set $SF(\varphi)$ of subformulas of a formula φ is defined inductively as follows:

- $SF(p) = \{p\}$;
- $SF(\neg\varphi) = \{\neg\varphi\} \cup SF(\varphi)$;
- $SF(\varphi \wedge \psi) = \{\varphi \wedge \psi\} \cup SF(\varphi) \cup SF(\psi)$;
- $SF(\langle X \rangle \varphi) = \{\langle X \rangle \varphi\} \cup SF(\varphi)$.

We say that a signature X appears in φ if there exists a formula ψ such that $\langle X \rangle \psi \in SF(\varphi)$.

2.2.2 Semantics of PECP

This is the class of models we will be working with:

DEFINITION 2. *(PECP-models) Given a countable set $\mathbf{P}$, a PECP-model for $\mathcal{L}_{\mathsf{PECP}}(\mathbf{P})$ is a tuple $\mathcal{M} = (W, V)$ where:*

- *W is a non-empty set of states;*
- *$V : \mathbf{P} \longrightarrow 2^W$ is a valuation function.*

Intuitively, a PECP-model consists just of a state-space and a valuation function for a given set of atoms. The satisfaction relation is defined as follows:

DEFINITION 3. *(Satisfaction for PECP-models) Let $\mathcal{M} = (W, V)$ be an PECP-model for $\mathcal{L}_{\mathsf{PECP}}(\mathbf{P})$, $w \in W$ and $\varphi, \psi \in \mathcal{L}_{\mathsf{PECP}}(\mathbf{P})$:*

$$\begin{array}{rcl} \mathcal{M}, w \models p & \iff & w \in V(p); \\ \mathcal{M}, w \models \neg\varphi & \iff & \mathcal{M}, w \not\models \varphi; \\ \mathcal{M}, w \models \varphi \wedge \psi & \iff & \mathcal{M}, w \models \varphi \text{ AND } \mathcal{M}, w \models \psi; \\ \mathcal{M}, w \models \langle X \rangle \varphi & \iff & \exists w' \in W : w \sim_X^V w' \text{ AND } \mathcal{M}, w' \models \varphi \end{array}$$

Formula φ is PECP-satisfiable, if and only if there exists a model $\mathcal{M}$ and a state w such that $\mathcal{M}, w \models \varphi$. Formula φ is valid in $\mathcal{M}$, noted $\mathcal{M} \models \varphi$, if and only if for all $w \in W$, $\mathcal{M}, w \models \varphi$. Finally, φ is PECP-valid, noted $\models_{\mathsf{PECP}} \varphi$, if and only if it is valid in all PECP-models. The logical consequence of formula φ from a set of formulae, noted $\Phi \models_{\mathsf{PECP}} \varphi$, is defined as usual.

So, modal operators are interpreted on the equivalence relations $\sim_X$ induced by the valuation of the model. It is worth observing that the logic of this class of models is not invariant under uniform substitution, suffice it to mention a validity such as $[\{p\}]p \vee [\{p\}]\neg p$.

2.2.3 Axiomatics of PECP

We can obtain an axiom system for PECP by a reduction technique. Let X, Y range over finite elements of $2^{\mathbf{P}}$, φ, ψ over $\mathcal{L}_{\mathsf{PECP}}(\mathbf{P})$, and p over $\mathbf{P}$:

(P)	all tautologies of propositional calculus
(K)	$[\emptyset](\varphi \rightarrow \psi) \rightarrow ([\emptyset]\varphi \rightarrow [\emptyset]\psi)$
(T)	$\varphi \rightarrow \langle\emptyset\rangle\varphi$
(4)	$\langle\emptyset\rangle\langle\emptyset\rangle\varphi \rightarrow \langle\emptyset\rangle\varphi$
(5)	$\langle\emptyset\rangle\varphi \rightarrow [\emptyset]\langle\emptyset\rangle\varphi$
(Reduce)	$[X]\varphi \leftrightarrow \bigwedge_{\pi \subseteq X} \left(\left(\bigwedge_{p \in \pi} p \wedge \bigwedge_{p \in X \setminus \pi} \neg p \right) \rightarrow [\emptyset] \left(\left(\bigwedge_{p \in \pi} p \wedge \bigwedge_{p \in X \setminus \pi} \neg p \right) \rightarrow \varphi \right) \right)$

And it is closed under the following inference rules ($\vdash_{\mathsf{PECP}}$ has its usual meaning):

(MP)	IF $\vdash_{\mathsf{PECP}} \varphi$ AND $\vdash_{\mathsf{PECP}} \varphi \rightarrow \psi$ THEN $\vdash_{\mathsf{PECP}} \psi$
(N)	IF $\vdash_{\mathsf{PECP}} \varphi$ THEN $\vdash_{\mathsf{PECP}} [\emptyset]\varphi$

The first thing to notice is that the system consists of S5 plus the Reduce axiom. Logic S5 is known to be sound and strongly complete for the class of models where the accessibility relation is the total relation W^2 [BdRV01], and modality $[\emptyset]$ is here axiomatized as one would axiomatize the global modality (cf. properties *(i)* and *(v)* in Fact 1).

Having said this, soundness and strong completeness of the above system are easy to establish. For soundness, it suffices to show that Reduce is PECP-valid, which follows straightforwardly from Definition 1. Intuitively, the axiom reduces $[X]\varphi$ by taking care of all the possible truth-value combinations of the atoms in X. If a given combination, e.g., $\left(\bigwedge_{p \in \pi} p \wedge \bigwedge_{p \in X \setminus \pi} \neg p\right)$, is true at a given state (for some π), then in all accessible states, if that combination is true, then φ is also true.

To obtain completeness we proceed as customary in DEL [vKv07], by using axiom `Reduce` and the following rule of substitution of provable equivalents (`REP`) to remove the occurrences of those $\langle X\rangle$ and $[X]$ operators from formulae where $X \neq \emptyset$:

$$(\texttt{REP}) \qquad \text{IF } \vdash_{\mathsf{PECP}} \varphi \leftrightarrow \varphi' \text{ THEN } \vdash_{\mathsf{PECP}} \psi \leftrightarrow \psi[\varphi/\varphi']$$

where $\psi[\varphi/\varphi']$ is the formula that results from ψ by replacing zero or more occurrences of φ, in ψ, by φ'.

One can show that `REP` is derivable for every operator $[X]$ as follows: first one can show that each $[X]$ operator satisfies the Axiom `K` and the rule of necessitation `N`. Let us provide the syntactic proofs of this. For notational convenience we use the following abbreviation:

$$\widehat{\pi} \stackrel{\texttt{def}}{=} \left(\bigwedge_{p\in\pi} p \wedge \bigwedge_{p\in X\setminus\pi} \neg p \right)$$

Derivation of `K` for $[X]$:

1. $\vdash [X](\varphi \to \psi) \leftrightarrow \bigwedge_{\pi\subseteq X} (\widehat{\pi} \to [\emptyset](\widehat{\pi} \to (\varphi \to \psi)))$
   by `Reduce`
2. $\vdash (\widehat{\pi} \to (\varphi \to \psi)) \to ((\widehat{\pi} \to \varphi) \to (\widehat{\pi} \to \psi))$
   by `P`
3. $\vdash \bigwedge_{\pi\subseteq X} (\widehat{\pi} \to [\emptyset](\widehat{\pi} \to (\varphi \to \psi))) \to \bigwedge_{\pi\subseteq X} (\widehat{\pi} \to [\emptyset]((\widehat{\pi} \to \varphi) \to (\widehat{\pi} \to \psi)))$
   by `P`, 2 and `RM` for $[\emptyset]$ (if $\vdash \varphi \to \psi$ then $\vdash [\emptyset]\varphi \to [\emptyset]\psi$)
4. $\vdash \bigwedge_{\pi\subseteq X} (\widehat{\pi} \to [\emptyset]((\widehat{\pi} \to \varphi) \to (\widehat{\pi} \to \psi))) \to \bigwedge_{\pi\subseteq X} (\widehat{\pi} \to ([\emptyset](\widehat{\pi} \to \varphi) \to [\emptyset](\widehat{\pi} \to \psi)))$
   by `K` and `P`
5. $\vdash \bigwedge_{\pi\subseteq X} (\widehat{\pi} \to ([\emptyset](\widehat{\pi} \to \varphi) \to [\emptyset](\widehat{\pi} \to \psi))) \to (\bigwedge_{\pi\subseteq X} (\widehat{\pi} \to [\emptyset](\widehat{\pi} \to \varphi)) \to \bigwedge_{\pi\subseteq X} (\widehat{\pi} \to [\emptyset](\widehat{\pi} \to \psi)))$
   by `P`
6. $\vdash (\bigwedge_{\pi\subseteq X} (\widehat{\pi} \to [\emptyset](\widehat{\pi} \to \varphi)) \to \bigwedge_{\pi\subseteq X} (\widehat{\pi} \to [\emptyset](\widehat{\pi} \to \psi))) \leftrightarrow ([X]\varphi \to [X]\psi)$
   by `Reduce`
7. $\vdash [X](\varphi \to \psi) \to ([X]\varphi \to [X]\psi)$
   from 1 and 3-6

Derivation of `N` for $[X]$:

1. $\vdash \varphi$
   hypothesis
2. $\vdash [\emptyset]\varphi$
   from 1 by `N` for $[\emptyset]$
3. $\vdash \bigwedge_{\pi\subseteq X} [\emptyset](\widehat{\pi} \to \varphi)$
   from 2 by the S5 theorem $[\emptyset]\varphi \to [\emptyset](\psi \to \varphi)$
4. $\vdash \bigwedge_{\pi\subseteq X} (\widehat{\pi} \to [\emptyset](\widehat{\pi} \to \varphi))$
   from 3 by `P`
5. $\vdash [X]\varphi$
   from 4 by `Reduce` and `MP`

Then one proves that `REP` is derivable by an induction routine analogous to the one used in [Che80, Th. 4.7].

We opted for this axiomatization in virtue of its simplicity, but alternative systems are of course possible. One in particular is worth mentioning. It first reduces $\langle p\rangle$ operators by axiom:

$$\langle p\rangle\varphi \leftrightarrow ((p \wedge \langle\emptyset\rangle(p \wedge \varphi)) \vee (\neg p \wedge \langle\emptyset\rangle(\neg p \wedge \varphi))) \quad (1)$$

This states that $\langle p\rangle\varphi$ is equivalent to either the case in which the current state satisfies p and there exists a (possibly different) p-state where φ is true, or the case where $\neg p$ is true and there exists a (possibly different) $\neg p$-state where φ is true (recall property *(iii)* in Fact 1). Given the above reduction, one can then use axioms to enforce the appropriate behavior of $\sim_X$ relations where X consists of more than one atom. To this aim, axioms can be used that are known to be canonical for properties *(ii)* and *(iv)* of Fact 1, namely:

$$\langle X \cup Y\rangle\varphi \to \langle X\rangle\varphi \quad (2)$$
$$\langle X\rangle i \wedge \langle Y\rangle i \to \langle X \cup Y\rangle i \quad (3)$$

where i ranges over a set of nominals. A complete system could then be obtained by axiomatizing the behavior of nominals—through axioms and rules used in hybrid logic [AT06]. From that system, a named canonical model could be built (i.e., a canonical model where all maximal consistent sets contain exactly one nominal) where the axioms in Formulae 1-3 would enforce the desirable properties on the canonical relations.

2.3 Exponentially embedding PECP into S5

The property expressed by axiom `Reduce` enables a truth-preserving translation of PECP into S5. This translation is, however, such that the translated formula is exponentially larger by a tower of exponents of height equal to the modal depth of the original formula.

In this section we propose a translation that is single exponential and preserves satisfiability. Take the standard modal language $\mathcal{L}_{\Box}(\mathbf{P})$ with one modal operator $\Box$ defined on the set of atoms $\mathbf{P}$. S5-models are structures $\mathcal{M} = (W, V)$ where W is a set of states, and $V : \mathbf{P} \longrightarrow 2^W$ a valuation function. Given an S5-model $\mathcal{M} = (W, V)$ and a state $w \in W$, the truth conditions are defined as follows:

$$\mathcal{M}, w \models \Box\varphi \iff \forall u \in W : \mathcal{M}, u \models \varphi$$

S5-satisfiability is defined as usual. It is possible to define an exponential truth-preserving reduction $tr : \mathcal{L}_{\mathsf{PECP}}(\mathbf{P}) \longrightarrow \mathcal{L}_{\Box}(\mathbf{P})$ as follows:

- $tr(\varphi_0) = p_{\varphi_0} \wedge \bigwedge_{\varphi\in SF(\varphi_0)} \Box(p_\varphi \leftrightarrow tr_1(\varphi))$

where p_φ are fresh atomic propositions and tr_1 is defined as follows:

$$\begin{aligned} tr_1(p) &= p \quad \text{FOR } p \in \mathbf{P} \\ tr_1(\neg\varphi) &= \neg tr_1(\varphi) \\ tr_1(\varphi \wedge \psi) &= tr_1(\varphi) \wedge tr_1(\psi) \\ tr_1([\emptyset]\varphi) &= \Box p_\varphi \\ tr_1([X]\varphi) &= \bigwedge_{\pi \subseteq X} \left(\left(\bigwedge_{p \in \pi} p \wedge \bigwedge_{p \in X \setminus \pi} \neg p \right) \rightarrow \right. \\ &\quad \left. \Box \left(\left(\bigwedge_{p \in \pi} p \wedge \bigwedge_{p \in X \setminus \pi} \neg p \right) \rightarrow p_\varphi \right) \right) \end{aligned}$$

Intuitively, the translation is designed to operate like axiom `Reduce` but avoiding exponential blow-up to pile up with the modal depth of the formula. The atomic propositions p_φ in $tr_1([X]\varphi)$ avoid the non-elementary size of $tr(\varphi_0)$. The definition of $tr_1([\emptyset]\varphi)$ corresponds to the degenerated case of $tr_1([X]\varphi)$ where $X = \emptyset$.The following theorem states the satisfiability preservation. The proof is given in Appendix A.

THEOREM 1. *(tr preserves satisfiability) Let φ_0 be a* PECP*-formula. We have equivalence between φ_0 is* PECP*-satisfiable and $tr(\varphi_0)$ is* S5*-satisfiable.*

As a consequence, we also obtain the following result.

COROLLARY 1. *(Decidability) The satisfiability problem for* PECP *is decidable and in NEXPTIME.*

PROOF. The satisfiability problem for S5 is decidable and in NP [BdRV01]. The result follows from Theorem 1 and a decision procedure may work as follows: in order to check that φ is satisfiable we compute the formula $tr(\varphi)$ and we apply a NP-decision procedure to check whether $tr(\varphi)$ is S5-satisfiable or not. □

Notice that if the cardinality of each X that appears in operators $[X]$ of φ is bounded by a fixed integer, then the translation tr becomes polynomial in the size of φ. Thus, as S5-satisfiability problem is NP-complete, the PECP-satisfiability problem with a bounded cardinality restrictions over set of atomic propositions in modal operators is in NP. As it is trivially NP-hard, it is NP-complete.

In Section 3, we will embed the atemporal version of STIT (the logic of *seeing to it that*) into PECP thereby obtaining lower bounds results.

2.4 PECP and modal ceteris paribus logics

Before moving to the next section, we briefly compare PECP with two works in the modal logic of ceteris paribus reasoning: release logic, and the logic of ceteris paribus preference.

Release logic has been introduced and studied in [KM03, KM00] in order to provide a modal logic characterization of a general notion of irrelevancy. Modal operators in release logic are S5 operators indexed by subsets of a finite set `Iss` of abstract elements denoting the issues that are taken to be irrelevant, or that can be *released*, while evaluating the formula in the scope of the operator. A release model is therefore a tuple $(W, \{\sim^r_X\}_{X \subseteq \texttt{Iss}}, V)$ where all $\sim^r_X$ are a equivalence relations with the additional constraint that if $X \subseteq Y$ then $\sim^r_X \subseteq \sim^r_Y$, that is, by releasing more issues one obtains a more granular relation. This is, more precisely, the semantics of release operators:

$$\mathcal{M}, w \models \Diamond_X \varphi \iff \exists w' \in W : w \sim^r_X w' \text{ AND } \mathcal{M}, w' \models \varphi$$

where $X \subseteq \texttt{Iss}$.

One can easily observe that, by Fact 1 (clause *(ii)*), PECP models are release models where $\texttt{Iss} = \mathbf{P}$ and where the release relation $\sim^r_X = \sim_{-X}$. Vice versa, for $\texttt{Iss} = \mathbf{P}$, not all release models are PECP models. As a consequence, the logic of $\langle -X \rangle$ operators in PECP is a conservative extension of the logic of $\Diamond_X$ release operators.

Preference logic has also long been concerned with so-called ceteris paribus preferences, that is, preferences incorporating an "all other things being equal" condition. A first logical analysis of such preferences dates back to [Von63], where dyadic modal operators are studied representing statements like 'φ is preferred to ψ, ceteris paribus'. More recently, [vBGR09] has provided a modal logic of ceteris paribus preferences based on standard unary modal operators. Leaving the preferential component of such logic aside, its ceteris paribus fragment concerns sentences of the form $\langle\Gamma\rangle\varphi$ whose intuitive meaning is 'there exists a state which is equivalent to the evaluation state with respect to all the formulae in the finite set Γ and which satisfies φ', where the formulae in Γ are drawn from the full language. It is easy to see that logic PECP is, in fact, the fragment of the ceteris paribus logic where Γ is allowed to consist only of a finite set of atoms.

3. PECP EMBEDDING OF ATEMPORAL STIT

In this section, we investigate the possibility of embedding the logic of agency STIT into PECP. STIT logic (the logic of *seeing to it that*) [BPX01, Hor01] is one of the most prominent logical accounts of agency. It is the logic of constructions of the form "agent i (or group J) sees to it that φ". STIT has a non-standard modal semantics based on the concepts of *moment* and *history*. However, as shown by [BHT08, HS08], the basic STIT language without temporal operators can be 'simulated' in a standard Kripke semantics.

3.1 Atemporal group STIT

First let us recall the syntax and the semantics of atemporal group STIT. The language of this logic is built from a countable set of atomic propositions $\mathbf{P}$ and a finite set of agents $AGT = \{1, \ldots, n\}$ and is defined by the following BNF:

$$\mathcal{L}_{G-\mathsf{STIT}}(\mathbf{P}, AGT) : \varphi ::= p \mid \neg\varphi \mid (\varphi \wedge \varphi) \mid [J : stit]\varphi$$

where p ranges over $\mathbf{P}$ and J ranges over 2^{AGT}. The construction $[J : stit]\varphi$ is read "group J sees to it that φ is true regardless of what the other agents choose". We define the dual operator $\langle J : stit\rangle\varphi \stackrel{\text{def}}{=} \neg[J : stit]\neg\varphi$. When $J = \emptyset$, the construction $[\emptyset : stit]\varphi$ is read "φ is true regardless of what every agent chooses" or simply "φ is necessarily true".

DEFINITION 4 (STIT-KRIPKE MODEL [HS08]). *A* STIT*-Kripke model $\mathcal{M} = (W, \{R_J\}_{J \subseteq AGT}, V)$ is a 3-tuple where:*

- *W is a non-empty set of worlds;*
- *for all $J \subseteq AGT$, R_J is an equivalence relation such that:*

i) $R_J \subseteq R_\emptyset$;

ii) $R_J = \bigcap_{j \in J} R_{\{j\}}$;

iii) for all $w, u_1, \ldots, u_n \in W$, *if* $u_1 \in R_\emptyset(w), \ldots, u_n \in R_\emptyset(w)$ *then* $\bigcap_{1 \leq j \leq n} R_{\{j\}}(u_j) \neq \emptyset$;

- $V : \mathbf{P} \to 2^W$ *is a valuation function for atomic propositions;*

with $R_J(w) = \{u \in W : (w, u) \in R_J\}$ *for any* $J \in 2^{AGT}$.

The partition induced by the equivalence relation R_J is the set of possible choices of the group J.[1] Indeed, in STIT a choice of a group J at a given world w is identified with the set of possible worlds $R_J(w)$. We call $R_J(w)$ the set of possible outcomes of group J's choice at world w, in the sense that group J's current choice at w forces the possible worlds to be in $R_J(w)$. The set $R_\emptyset(w)$ is simply the set of possible outcomes at w, or said differently, the set of outcomes of the current game at w. According to Condition *(i)*, the set of possible outcomes of a group J's choice is a subset of the set of possible outcomes. Condition *(ii)*, called *additivity*, means that the choices of the agents in a group J is made up of the choices of each individual agent and no more. Condition *(iii)* corresponds to the property of *independence of agents*: whatever each agent decides to do, the set of outcomes corresponding to the joint action of all agents is non-empty. More intuitively, this means that agents can never be deprived of choices due to the choices made by other agents. In [LS11] we supposed determinism for the group AGT, that is to say that the set of outcomes corresponding to a joint action of all agents is a singleton. Horty's group STIT logic [Hor01] does not suppose this. Here we deal with Horty's version of STIT. So a STIT model is a game form in which a joint action of all agents might determine more than one outcome.

EXAMPLE 1. *The tuple* $\mathcal{M} = (W, R_\emptyset, R_{\{1\}}, R_{\{2\}}, R_{\{1,2\}}, V)$ *defined by:*

- $W = \{w, u, v, r, s, t, z\}$;
- $R_\emptyset = W \times W$;
- $R_{\{1\}} = \{w, u, v\}^2 \cup \{r, s\}^2 \cup \{t, z\}^2$;
- $R_{\{2\}} = \{w, r, t\}^2 \cup \{u, v, s, z\}^2$;
- $R_{\{1,2\}} = \{(w,w), (r,r), (s,s), (t,t), (z,z), (u,u), (v,v), (u,v), (v,u)\}$;
- *for all* $p \in \mathbf{P}$, $V(p) = \emptyset$.

is a STIT*-Kripke model. Figure 1 shows the model* $\mathcal{M}$. *The equivalence classes induced by the equivalence relation* $R_{\{1\}}$ *are represented by ellipses and correspond to the choices of agent* 1. *The equivalence classes induced by the equivalence relation* $R_{\{2\}}$ *are represented by rectangles and correspond to the choices of agent* 2. *The choice of group* $\{1, 2\}$ *at a given world is determined by the intersection of the choice of agent* 1 *and the choice of agent* 2 *at this world. For example, the choice of agent* 1 *at world* u *is* $\{w, u, v\}$ *whereas the choice of agent* 2 *at world* u *is* $\{u, v, s, z\}$. *The choice of group* $\{1, 2\}$ *at* u *is* $\{u, v\}$. *Note that Condition* (iii) *of Definition 4 ensures that for any choice of agent* 1 *and for any choice of agent* 2 *the intersection between these two choices is non-empty. That is, for any equivalence class induced by the relation* $R_{\{1\}}$ *and for any equivalence class induced by the relation* $R_{\{2\}}$, *the intersection between these two equivalence classes is non-empty.*

[1] One can also see the partition induced by the equivalence relation R_j as the set of actions that agent j can *try*, where the notion of *trying* corresponds to the notion of *volition* studied in philosophy of action [O'S74, McC74].

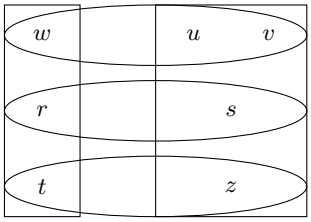


Figure 1: The model $\mathcal{M}$

Given a STIT-Kripke model $\mathcal{M} = (W, \{R_J\}_{J \subseteq AGT}, V)$ and a world w in $\mathcal{M}$, the truth conditions of STIT formulae are the following:

$$\begin{aligned}
\mathcal{M}, w \models p &\iff w \in V(p);\\
\mathcal{M}, w \models \neg\varphi &\iff \mathcal{M}, w \not\models \varphi;\\
\mathcal{M}, w \models \varphi \wedge \psi &\iff \mathcal{M}, w \models \varphi \text{ AND } \mathcal{M}, w \models \psi;\\
\mathcal{M}, w \models [J : stit]\varphi &\iff \forall v \in R_J(w) : \mathcal{M}, v \models \varphi
\end{aligned}$$

where $R_J(w) = \{u \in W \mid (w, u) \in R_J\}$.

We are not able to embed group STIT into PECP because of many reasons. The first one is that the group STIT satisfiability problem is undecidable if there are more than 3 agents [HS08].[2] The second one is that group STIT does not have the finite model property. Indeed in [HS08], a translation from the product logic $\mathsf{S5}^n$ to group STIT logic is given and as $\mathsf{S5}^n$ does not have the finite model property [GKWZ03], atemporal group STIT will also not have it. On the contrary PECP inherits the finite model property from S5. Indeed, if a formula φ is PECP-satisfiable, Theorem 1 says that $tr(\varphi)$ is S5-satisfiable. But as S5 has the polynomial model property, there exists a polynomial-sized S5-model for $tr(\varphi)$ in the size of $tr(\varphi)$. In other words, there exists an exponential S5-model for $tr(\varphi)$ in the size of φ. Theorem 1 ensures that there exists an exponential PECP-model for φ in the size of φ.

We will nevertheless embed a variant of group STIT under the assumption that every agent has a finite and bounded number of actions in his repertoire. For every agent j, a R_j-equivalence class $R_j(u)$ corresponds to an action of agent j. We say that agent j has k_j actions in a STIT model if and only if there are exactly k_j R_j-equivalence classes in $\mathcal{M}$.

The game structure in STIT-models should be enforced in PECP-models. That is why we introduce special atomic propositions to encode the game structure. Without loss of generality, we assume that the set $\mathbf{P}$ contains special atomic propositions $\mathfrak{rep}_1^j, \mathfrak{rep}_2^j, \ldots$ for all agents j which are used to represent the actions of the agents. Let k be the maximal number of actions: $k = max_{j \in AGT} k_j$. For every agent, we

[2] See [LS11] for a study of some decidable fragments of group STIT.

represent its actions by numbers ℓ in $\{0,\ldots,k-1\}$ and some atomic propositions encode the binary representation of ℓ. Let m be an integer that represents the number of digits we need to represent an action. For instance let $m = \lceil log_2 k \rceil$ (the ceiling of the logarithm of k). For a given agent j, $\mathfrak{R}_m^j = \{\mathfrak{rep}_1^j, \ldots, \mathfrak{rep}_m^j\}$ is the set atomic propositions that represent the binary digits of an action of agent j. We suppose that if $j \neq i$ then $\mathfrak{R}_m^j \cap \mathfrak{R}_m^i = \emptyset$.

EXAMPLE 2. *For example, in the model of Example 1, agent 1 has $k_1 = 3$ actions and agent 2 has $k_2 = 2$ actions. So $k = 3$ and $m = \lceil log_2 3 \rceil = 2$. We have $\mathfrak{R}_m^1 = \{\mathfrak{rep}_1^1, \mathfrak{rep}_2^1\}$ and $\mathfrak{R}_m^2 = \{\mathfrak{rep}_1^2, \mathfrak{rep}_2^2\}$. Then for instance, we may represent the action of agent 1 corresponding to $R_{\{1\}}(w) = \{w,u,v\}$ by the valuation $\neg\mathfrak{rep}_1^1 \wedge \neg\mathfrak{rep}_2^1$, the action of agent 1 corresponding to $\{r,s\}$ by $\mathfrak{rep}_1^1 \wedge \neg\mathfrak{rep}_2^1$, the action of agent 1 corresponding to $\{t,z\}$ by $\neg\mathfrak{rep}_1^1 \wedge \mathfrak{rep}_2^1$, the action of agent 2 corresponding to $\{w,r,t\}$ by $\neg\mathfrak{rep}_1^2 \wedge \neg\mathfrak{rep}_2^2$ and the action of agent 2 corresponding to $\{u,v,s,z\}$ by $\mathfrak{rep}_1^2 \wedge \neg\mathfrak{rep}_2^2$.*

Let $\mathfrak{R}_m = \bigcup_{j \in AGT} \mathfrak{R}_m^j$ be the set of all atomic propositions used to denote actions. Let us define the following PECP formula:

$$GRID_m \stackrel{\text{def}}{=} \bigwedge_{x \in \mathfrak{R}_m} [\emptyset]((x \rightarrow \langle \mathfrak{R}_m \setminus \{x\}\rangle \neg x) \wedge (\neg x \rightarrow \langle \mathfrak{R}_m \setminus \{x\}\rangle x))$$

This formula enforces a STIT model to contain all possible valuations over $\mathfrak{R}_m$. A model that satisfies $GRID_m$ is then interpreted as a game form where each valuation of $\mathfrak{R}_m^j$ represents an action of player j.

We now define a translation from $\mathcal{L}_{G-\mathsf{STIT}}$ to $\mathcal{L}_{\mathsf{PECP}}(\mathbf{P})$ as follows:

$$\begin{aligned} tr_2(p) &= p \quad \text{for } p \in \mathbf{P} \\ tr_2(\neg\varphi) &= \neg tr_2(\varphi) \\ tr_2(\varphi \wedge \psi) &= tr_2(\varphi) \wedge tr_2(\psi) \\ tr_2([J : stit]\varphi) &= [\bigcup_{j \in J} \mathfrak{R}_m^j] tr_2(\varphi) \end{aligned}$$

The translation tr_2 should be parameterized by m. For notational convenience, in what follows we write tr_2 instead of tr_2^m leaving implicit the parameter m.

The set $\bigcup_{j \in J} \mathfrak{R}_m^j$ represents all the atomic propositions used to represented actions of the coalition J. We then have the following theorem whose proof is given in Appendix B at the end of the paper.

THEOREM 2. *Let us consider a group STIT formula φ. Let m be an integer. Then the following items are equivalent:*

1. *φ is STIT-satisfiable in a STIT-model where each agent has at most 2^m actions;*
2. *φ is STIT-satisfiable in a STIT-model where each agent has exactly 2^m actions;*
3. *$GRID_m \wedge tr_2(\varphi)$ is PECP-satisfiable.*

3.2 Atemporal individual STIT

In this subsection, we consider the following fragment of STIT called atemporal individual STIT [3]:

$$\mathcal{L}_{I-\mathsf{STIT}}(\mathbf{P}, AGT) : \varphi ::= p \mid \neg\varphi \mid (\varphi \wedge \varphi) \mid [\{j\} : stit]\varphi$$

where p ranges over $\mathbf{P}$ and j ranges over AGT.

This fragment of STIT, axiomatized by Xu in [Xu98], has the exponential finite model property (see Lemma 7 in [BHT08]). Moreover, as the following theorem highlights, it can be embedded in the logic PECP.

THEOREM 3. *Let us consider a STIT formula φ of the individual STIT fragment. Let m be the length of φ. Then the following three items are equivalent:*

1. *φ is STIT-satisfiable*
2. *φ is STIT-satisfiable in a model where each agent has at most 2^m actions;*
3. *$GRID_m \wedge tr_2(\varphi)$ is PECP-satisfiable.*

PROOF. $\boxed{1 \Rightarrow 2}$ Consider a STIT formula φ of the individual STIT fragment. If φ is STIT-satisfiable and m is the length of φ, then φ is STIT-satisfiable in a model where there are at most 2^m worlds (see Lemma 7 in [BHT08]). This implies that there are at most 2^m actions in that model.

The implications $2 \Rightarrow 3$ and $3 \Rightarrow 1$ come from Theorem 2. □

Thanks to Theorem 3, we reduce the NEXPTIME-complete satisfiability problem of individual STIT [BHT08] to the PECP-satisfiability problem. As the reduction is polynomial, we obtain the following lower bound complexity result for the PECP-satisfiability problem.

COROLLARY 2. *The PECP-satisfiability problem is NEXPTIME-hard.*

3.3 Group STIT where coalitions are nested

In this subsection we address the satisfiability problem of the fragment of PECP consisting of formulae φ of $\mathcal{L}_{\mathsf{PECP}}$ such that the sets of atomic propositions that appear in any operator $[X]$ occurring in φ form a linear set of sets of atomic propositions. More formally, if $[X]$ and $[X']$ are two operators occurring in φ then either $X \subseteq X'$ or $X' \subseteq X$. For instance, the formula $[\{p,q\}](\psi \wedge [\{p\}][\{p,q,r,s\}]\varphi)$ belongs to the fragment because $\{p\} \subseteq \{p,q\} \subseteq \{p,q,r,s\}$. On the contrary, the formula $[\{p\}]p \wedge [\{q\}]p$ is not an element of this fragment of PECP.

We call the satisfiability problem of this fragment of PECP the PECP-nested satisfiability problem. Due to the embedding proposed in Theorem 2 of STIT into PECP, we provide the following lower bound complexity result for the PECP-nested satisfiability problem. The proof is given in Appendix C.

THEOREM 4. *The PECP-nested satisfiability problem is PSPACE-hard.*

The following theorem provides an upper bound complexity result for this fragment of PECP. The proof is given in Appendix D.

[3] Some authors ([Bro08a, Wan06]) use the term 'multi-agent STIT' to designate the logic where operators are of the form $[\{j\} : stit]$. Here we prefer to use the more explicit term 'individual STIT' as in [HS08].

THEOREM 5. *The* PECP*-nested satisfiability problem is in PSPACE.*

This concludes our analysis of STIT logics via PECP. In the next section we move to the coalition logic of propositional control.

4. RELATING STIT WITH CL−PC, AND PECP WITH CL−PC

In this section we study the relationships between PECP, atemporal 'bounded' group STIT, and another well-known game logic, the logic CL−PC (*coalition logic of propositional control*).[4] Specifically, we show that CL−PC can be embedded into atemporal 'bounded' group STIT and, by the fact that atemporal 'bounded' group STIT can be embedded into PECP (Section 3.1), we indirectly show that CL−PC can be embedded into PECP.

CL−PC was introduced by [vdHW05] as a formal language for reasoning about capabilities of agents and coalitions in multiagent environments. In this logic the notion of capability is modeled by means of the concept of *control*. In particular, it is assumed that each agent i is associated with a specific finite subset $\mathbf{P}_i$ of the finite set of all propositions $\mathbf{P}$. $\mathbf{P}_i$ is the set of propositions *controlled* by the agent i. That is, the agent i has the ability to assign a (truth) value to each proposition $\mathbf{P}_i$ but cannot affect the truth values of the propositions in $\mathbf{P} \setminus \mathbf{P}_i$. In the variant of CL−PC studied by [vdHW05] it is also assumed that control over propositions is exclusive, that is, two agents cannot control the same proposition (i.e., if $i \neq j$ then $\mathbf{P}_i \cap \mathbf{P}_j = \emptyset$). Moreover, it is assumed that control over propositions is complete, that is, every proposition is controlled by at least one agent (i.e., for every $p \in \mathbf{P}$ there exists an agent i such that $p \in \mathbf{P}_i$).

The preceding concepts and assumptions are precisely formulated in the following section, which illustrates the syntax and the formal semantics of CL−PC.

4.1 Syntax and semantics of CL−PC

The *language of* CL−PC is built from a *finite* set of atomic propositions $\mathbf{P}$ and a finite set of agents $AGT = \{1, \ldots, n\}$, and is defined by the following BNF:

$$\mathcal{L}_{\mathsf{CL-PC}}(\mathbf{P}, AGT) : \varphi ::= p \mid \neg\varphi \mid (\varphi \wedge \varphi) \mid \Diamond_J \varphi$$

where p ranges over $\mathbf{P}$ and J ranges over 2^{AGT}. Operator $\Diamond_J$ is called *cooperation modality*, and the construction $\Diamond_J \varphi$ means that "group J has the contigent ability to achieve φ".

DEFINITION 5 (CL−PC MODEL). *A model for* CL−PC *is a tuple* $\mathcal{M} = (\mathbf{P}_1, \ldots, \mathbf{P}_n, X)$ *where:*

- $\mathbf{P}_1, \ldots, \mathbf{P}_n$ *is a partition of* $\mathbf{P}$ *among the agents in* AGT*;*
- $X \subseteq \mathbf{P}$ *is the set of propositions which are true in the initial state.*

For every group of agents $J \subseteq AGT$, let $\mathbf{P}_J = \bigcup_{i \in J} \mathbf{P}_i$ be the set of atomic propositions controlled by the group J. Moreover, for every group $J \subseteq AGT$ and for every set of atomic propositions $X \subseteq \mathbf{P}$, let $X_J = X \cap \mathbf{P}_J$ be the set of atomic propositions in X controlled by the group J. Sets X_J are called J-valuations.

Given a CL−PC model $\mathcal{M} = (\mathbf{P}_1, \ldots, \mathbf{P}_n, X)$, the truth conditions of CL−PC formulae are the following:

$$\begin{aligned}
\mathcal{M} \models p &\iff p \in X;\\
\mathcal{M} \models \neg\varphi &\iff \mathcal{M} \not\models \varphi;\\
\mathcal{M} \models \varphi \wedge \psi &\iff \mathcal{M} \models \varphi \text{ AND } \mathcal{M} \models \psi;\\
\mathcal{M} \models \Diamond_J \varphi &\iff \exists X'_J \subseteq \mathbf{P}_J : \mathcal{M} \bigoplus X'_J \models \varphi
\end{aligned}$$

where $\mathcal{M} \bigoplus X'_J$ is the CL−PC model $(\mathbf{P}_1, \ldots, \mathbf{P}_n, X'')$ such that:

$$\begin{aligned}
X''_{AGT \setminus J} &= X_{AGT \setminus J}\\
X''_J &= X'_J
\end{aligned}$$

That is, $\Diamond_J \varphi$ is true at a given model $\mathcal{M}$ if and only if, the coalition J can change the truth values of the atoms that it controls in such a way that φ will be true afterwards (i.e., given the actual truth-value combination of the atoms which are not controlled by J, there exists a truth-value combination of the atoms controlled by J which ensures φ).

Let us illustrate the CL−PC semantics with an example.

EXAMPLE 3. *Let* $AGT = \{1, 2, 3\}$, $\mathbf{P} = \{p, q, r\}$, $\mathbf{P}_1 = \{p\}$, $\mathbf{P}_2 = \{q\}$ *and* $\mathbf{P}_3 = \{r\}$.

Consider the CL−PC *model* $\mathcal{M} = (\mathbf{P}_1, \mathbf{P}_2, , \mathbf{P}_3, \{r\})$. *We have that:*

$$\mathcal{M} \models \Diamond_{\{1,2\}}((p \wedge q \wedge r) \vee (p \wedge \neg q \wedge r)).$$

Indeed, there exists a set of atoms $X'_{\{1,2\}} \subseteq \mathbf{P}_{\{1,2\}}$ *controlled by* $\{1, 2\}$ *such that* $\mathcal{M} \bigoplus X'_{\{1,2\}} \models ((p \wedge q \wedge r) \vee (p \wedge \neg q \wedge r))$.

For example, we have $\{p\} \subseteq \mathbf{P}_{\{1,2\}}$ *and* $(\mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3, \{p, r\}) \models ((p \wedge q \wedge r) \vee (p \wedge \neg q \wedge r))$, *where* $(\mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3, \{p, r\}) = \mathcal{M} \bigoplus \{p\}$.

4.2 Embedding CL−PC into STIT

The aim of this section is to provide an embedding of CL−PC into the variant of atemporal group STIT with bounded choices (atemporal 'bounded' group STIT) that have been presented in Section 3.1.

Let us provide the following STIT formulae which catpure four basic assumptions of CL−PC:

$$EXC^{+} \stackrel{\text{def}}{=} \bigwedge_{p \in \mathbf{P}} \bigwedge_{i,j \in AGT : i \neq j} (\langle \emptyset : stit \rangle [\{i\} : stit] p \rightarrow \neg \langle \emptyset : stit \rangle [\{j\} : stit] p)$$

$$EXC^{-} \stackrel{\text{def}}{=} \bigwedge_{p \in \mathbf{P}} \bigwedge_{i,j \in AGT : i \neq j} (\langle \emptyset : stit \rangle [\{i\} : stit] p \rightarrow \neg \langle \emptyset : stit \rangle [\{j\} : stit] \neg p)$$

$$COMPL \stackrel{\text{def}}{=} \bigwedge_{p \in \mathbf{P}} \bigvee_{i \in AGT} [\emptyset : stit]([\{i\} : stit] p \vee [\{i\} : stit] \neg p)$$

$$GRID^{*} \stackrel{\text{def}}{=} \bigwedge_{X \subseteq \mathbf{P}} \langle \emptyset : stit \rangle (\bigwedge_{p \in X} p \wedge \bigwedge_{p \in \mathbf{P} \setminus X} \neg p)$$

Formulae EXC^{+} and EXC^{-} mean that control over atomic propositions in $\mathbf{P}$ is exclusive (i.e., there is no proposition

[4] In [Ger06] generalizations of some of the assumptions underlying CL−PC have been studied. Here we only consider the original version of CL−PC proposed by van der Hoek & Wooldridge.

in $\mathbf{P}$ which can be forced to be true or false by more than one agent), whereas formula *COMPL* means that exercise of control over atomic propositions in $\mathbf{P}$ is complete (i.e., for every proposition in $\mathbf{P}$ there exists at least one agent who either forces it to be true or forces it to be false). Finally, formula $GRID^*$ means that all the possible truth-value combinations of the atomic propositions in $\mathbf{P}$ are possible. Note that EXC^+, EXC^-, *COMPL* and $GRID^*$ are well-formed STIT formulae because of the assumption that the set $\mathbf{P}$ is finite.[5]

We define the following translation from $\mathcal{L}_{\mathsf{CL-PC}}(\mathbf{P}, AGT)$ to $\mathcal{L}_{\mathsf{STIT}}(\mathbf{P}, AGT)$:

$$\begin{aligned} tr_3(p) &= p \quad \text{FOR } p \in \mathbf{P} \\ tr_3(\neg\varphi) &= \neg tr_3(\varphi) \\ tr_3(\varphi \wedge \psi) &= tr_3(\varphi) \wedge tr_3(\psi) \\ tr_3(\Diamond_J \varphi) &= \langle AGT \setminus J : stit \rangle tr_3(\varphi) \end{aligned}$$

The following theorem highlights that 'bounded' group STIT embeds CL−PC. The proof is given in Appendix E.

THEOREM 6. *Let $m = |\mathbf{P}|$. Then, a* CL−PC *formula φ is* CL−PC*-satisfiable if and only if $(EXC^+ \wedge EXC^- \wedge COMPL \wedge GRID^*) \wedge tr_3(\varphi)$ is satisfiable in a* STIT *model where each agent has at most 2^m actions.*

As PECP embeds atemporal 'bounded' group STIT (Theorem 2 in Section 3.1), from Theorem 6 it follows that PECP also embeds CL−PC. Indeed, given a CL−PC-satisfiable formula φ, one can use the translation tr_2 given in Section 3.1 in order to find a corresponding STIT formula which is STIT-satisfiable. Then, one uses the preceding translation tr_3 in order to find a corresponding PECP formula which is PECP-satisfiable.

COROLLARY 3. *Let $m = |\mathbf{P}|$. Then, a* CL−PC *formula φ is* CL−PC*-satisfiable if and only if $GRID_m \wedge tr_2((EXC^+ \wedge EXC^- \wedge COMPL \wedge GRID^*) \wedge tr_3(\varphi))$ is* PECP*-satisfiable.*

5. CONCLUSIONS

The paper has introduced a modal logic that arises by interpreting modal operators on the equivalence relations induced by finite sets of propositional atoms. This logic, called PECP, has been axiomatized, embedded (exponentially) into S5, and its relation to existing formalisms has been briefly discussed. PECP has then been used as a tool to compare two logics of agency and games—atemporal STIT and the coalitional logic of propositional control CL−PC—showing that CL−PC can be embedded in STIT and that, in turn, STIT can be embedded in PECP. These embedding preserve satisfiability and the paper has taken stock of them to provide a complexity analysis of logic PECP.

Moreover, via logic S5, one can easily show that embeddings in the other directions are also possible. S5, we have seen, embeds PECP, but is also directly embeddable in all the mentioned logic, which all contain the universal modality, in the following forms: $\langle\emptyset\rangle$ in PECP, $\langle AGT \setminus \emptyset : stit\rangle$ in atemporal STIT and $\Diamond_{AGT}$ in CL−PC. All in all, this illustrates a nice uniformity in the logical tools that seem to be needed to talk about α-effectivity and, we believe, that PECP offers a good paradigm for systematizing existing logics of game forms.

Directions of future work are manifold. First of all, we plan to look for principled generalizations of some of the assumptions underlying the logics studied: e.g., independence of agents in STIT such as "agent j and agent i are independent as far as the set of atomic propositions X is concerned", restriction to control over atomic propositions in CL−PC. Secondly, we intend to push further our study of the relationship between ceteris paribus logics and existing logics of agency and cooperation including the logic of "bringing it about that" [GR05], the logic STIT with time [Bro08b, Lor12] and Coalition Logic [Pau02]. Finally, in this paper we have shown that PECP and atemporal individual STIT have the same high complexity of the satisfiability problem when we consider the whole languages. The study of efficient syntactic fragments is then important and we intend to pursue this study in parallel both for PECP and for atemporal individual STIT. We expect that several complexity results about fragments of atemporal STIT may be transferred to fragments of PECP and viceversa.

[5]This assumption is also made by van der Hoek & Wooldridge in [vdHW05].

6. REFERENCES

[AT06] C. Areces and B. Ten Cate. Hybrid logics. In P. Blackburn, J. van Benthem, and F. Wolter, editors, *Handbook of Modal Logic*, pages 821–868. Elsevier, 2006.

[BdRV01] P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic*. Cambridge University Press, Cambridge, 2001.

[BHT08] P. Balbiani, A. Herzig, and N. Troquard. Alternative axiomatics and complexity of deliberative stit theories. *Journal of Philosophical Logic*, 37(4):387–406, 2008.

[BPX01] N.D. Belnap, M. Perloff, and M. Xu. *Facing the future: agents and choices in our indeterminist world*. Oxford University Press, USA, 2001.

[Bro08a] J. Broersen. A complete STIT logic for knowledge and action, and some of its applications. In *Proceedings of the 6th International Workshop on Declarative Agent Languages and Technologies (DALT 2008)*, volume 5397 of *LNCS*, pages 47–59. Springer-Verlag, 2008.

[Bro08b] J. Broersen. A logical analysis of the interaction between 'obligation-to-do' and 'knowingly doing'. In *Proceedings of the Ninth International Conference on Deontic Logic in Computer Science (DEON'08)*, volume 5076 of *LNCS*, pages 140–154. Springer-Verlag, 2008.

[Che80] B. F. Chellas. *Modal Logic. An Introduction*. Cambridge University Press, Cambridge, 1980.

[Ger06] J. Gerbrandy. Logics of propositional control. In *Proceedings of AAMAS'06*, pages 193–200. ACM, 2006.

[GKWZ03] D. M. Gabbay, A. Kurucz, F. Wolter, and M. Zakharyaschev. *Many-dimensional modal logics: theory and applications*. Elsevier, 2003.

[GR05] G. Governatori and A. Rotolo. On the axiomatization of Elgesem's logic of agency and ability. *Journal of Philosophical Logic*, 34:403–431, 2005.

[Hor01] J. F. Horty. *Agency and Deontic Logic*. Oxford University Press, Oxford, 2001.

[HS08] A. Herzig and F. Schwarzentruber. Properties of logics of individual and group agency. *Advances in modal logic*, 7:133–149, 2008.

[KM00] J. Krabbendam and J.-J.Ch. Meyer. Release logics for temporalizing dynamic logic, orthogonalising modal logics. In M. Barringer, M. Fisher, D. Gabbay, and G. Gough, editors, *Advances in Temporal Logic*, pages 21–45. Kluwer Academic Publisher, 2000.

[KM03] J. Krabbendam and J.-J. Ch. Meyer. Contextual deontic logics. In P. McNamara and H. Prakken, editors, *Norms, Logics and Information Systems*, pages 347–362, Amsterdam, 2003. IOS Press.

[Lor12] E. Lorini. A STIT-logic analysis of commitment and its dynamics. *Journal of Applied Logic*, 2012. to appear.

[LS11] E. Lorini and F. Schwarzentruber. A logic for reasoning about counterfactual emotions. *Artificial Intelligence*, 175(3-4):814–847, 2011.

[McC74] H. J. McCann. Volition and basic action. *The Philosophical Review*, 83:451–473, 1974.

[MP82] H. Moulin and B. Peleg. Cores of effectivity functions and implementation theory. *Journal of Mathematical Economics*, 10:115–145, 1982.

[O'S74] B. O'Shaughnessy. Trying (as the mental pineal gland). *The Journal of Philosophy*, 70:365–386, 1974.

[Pau02] M. Pauly. A modal logic for coalitional power in games. *Journal of Logic and Computation*, 12(1):149–166, 2002.

[Sch12] F. Schwarzentruber. Complexity results of stit fragments. *Studia logica*, 2012. to appear.

[vBGR09] J. van Benthem, P. Girard, and O. Roy. Everything else being equal: A modal logic for ceteris paribus preferences. *Journal of Philosophical Logic*, 38:83–125, 2009.

[vdHW05] W. van der Hoek and M. Wooldridge. On the logic of cooperation and propositional control. *Artificial Intelligence*, 164:81–119, 2005.

[vKv07] H. van Ditmarsch, B. Kooi, and W. van der Hoek. *Dynamic Epistemic Logic*, volume 337 of *Synthese Library Series*. Springer, 2007.

[Von63] G. H. Von Wright. *The Logic of Preference*. Edinburgh University Press, 1963.

[Wan06] H. Wansing. Tableaux for multi-agent deliberative-STIT logic. In G. Governatori, I. Hodkinson, and Y. Venema, editors, *Advances in Modal Logic, Volume 6*, pages 503–520. King's College Publications, 2006.

[Xu98] M. Xu. Axioms for deliberative STIT. *Journal of Philosophical Logic*, 27:505–552, 1998.

APPENDIX

A. PROOF OF THEOREM 1

Let φ_0 be a PECP-formula. We have equivalence between φ_0 is PECP-satisfiable and $tr(\varphi_0)$ is S5-satisfiable.

PROOF. $\boxed{\Rightarrow}$ Suppose that there exists a PECP-model $\mathcal{M} = (W, V)$ and a world $w \in W$ such that $\mathcal{M}, w \models \varphi_0$. Let V' be the valuation V modified such that p_φ is true in exactly all worlds u such that $\mathcal{M}, u \models \varphi$. Let $\mathcal{M}'$ be the S5-model defined as (W, V'). A standard induction provides that $\mathcal{M}', w \models tr(\varphi_0)$. More precisely, let us prove by induction that for all $\varphi \in SF(\varphi_0)$, we have $\mathcal{M}, u \models \varphi$ iff $\mathcal{M}', u \models tr_1(\varphi)$ for all $u \in W$.

- Propositional case: for all atomic propositions p, we have $\mathcal{M}, u \models p$ iff $u \in V(p)$ iff $u \in V'(p)$ iff $\mathcal{M}', u \models tr_1(p)$.
- Negation: $\mathcal{M}, u \models \neg\varphi$ iff $\mathcal{M}, u \not\models \varphi$ iff $\mathcal{M}', u \not\models \varphi$ iff $\mathcal{M}', u \models \neg\varphi$.
- Conjunction: $\mathcal{M}, u \models \varphi \wedge \psi$ iff $\mathcal{M}, u \models \varphi$ and $\mathcal{M}, u \models \psi$ iff $\mathcal{M}', u \models tr_1(\varphi)$ and $\mathcal{M}', u \models tr_1\psi)$ iff $\mathcal{M}, u \models tr_1(\varphi \wedge \psi)$.
- Case of a formula of the form $[X]\varphi$:

  $\mathcal{M}, u \models [X]\varphi$

  iff for all $v \in W$, $u \sim_X^V v$ implies $\mathcal{M}, v \models \varphi$

  iff for all $v \in W$, $u \sim_X^V v$ implies $\mathcal{M}', v \models p_\varphi$

  (by construction of V')

  iff $\mathcal{M}', u \models tr_1([X])\varphi$

By construction of V', we have $\mathcal{M}', w \models \bigwedge_{\varphi \in SF(\varphi_0)} \Box(p_\varphi \leftrightarrow tr_1(\varphi))$. As $\mathcal{M}, w \models \varphi_0$ we have $\mathcal{M}, w \models tr_1(\varphi_0)$ thus $\mathcal{M}, w \models p_{\varphi_0}$ by construction of V'. As a result, $\mathcal{M}, w \models tr(\varphi_0)$.

$\boxed{\Leftarrow}$ Suppose that there exists a S5 model $\mathcal{M}' = (W, V)$ and a world $w \in W$ such that $\mathcal{M}', w \models tr(\varphi_0)$. We define the relations $\sim_X$ where $X \subseteq \mathbf{P}$ as in the Definition 1. Let $\mathcal{M}$ be the PECP-model equal to (W, V). A standard induction provides that $\mathcal{M}, w \models \varphi_0$. More precisely, let us prove by induction that for all $\varphi \in SF(\varphi_0)$, we have $\mathcal{M}, u \models \varphi$ iff $\mathcal{M}', u \models tr_1(\varphi)$ for all $u \in W$.

- Propositional case: for all atomic propositions p, we have $\mathcal{M}, u \models p$ iff $u \in V(p)$ iff $u \in V'(p)$ iff $\mathcal{M}', u \models tr_1(p)$.
- Negation: $\mathcal{M}, u \models \neg\varphi$ iff $\mathcal{M}, u \not\models \varphi$ iff $\mathcal{M}', u \not\models \varphi$ iff $\mathcal{M}', u \models \neg\varphi$.
- Conjunction: $\mathcal{M}, u \models \varphi \wedge \psi$ iff $\mathcal{M}, u \models \varphi$ and $\mathcal{M}, u \models \psi$ iff $\mathcal{M}', u \models tr_1(\varphi)$ and $\mathcal{M}', u \models tr_1\psi)$ iff $\mathcal{M}, u \models tr_1(\varphi \wedge \psi)$.
- Case of a formula of the form $[X]\varphi$:

  $\mathcal{M}, u \models [X]\varphi$

  iff for all $v \in W$, $u \sim_X^V v$ implies $\mathcal{M}, v \models \varphi$

  iff for all $v \in W$, $u \sim_X^V v$ implies $\mathcal{M}', v \models tr_1(\varphi)$

  (by induction)

  iff for all $v \in W$, $u \sim_X^V v$ implies $\mathcal{M}', v \models p_\varphi$

  (because, as $\mathcal{M}', w \models tr(\varphi_0)$ we have that

  for all $v \in W$, $\mathcal{M}', v \models (p_\varphi \leftrightarrow tr_1(\varphi))$)

  iff $\mathcal{M}', u \models tr_1([X])\varphi$

As $\mathcal{M}', w \models tr(\varphi_0)$, we have that $\mathcal{M}', w \models (p_{\varphi_0} \leftrightarrow tr_1(\varphi_0))$ and $\mathcal{M}', w \models p_{\varphi_0}$. Thus, $\mathcal{M}', w \models tr_1(\varphi_0)$. Hence $\mathcal{M}, w \models \varphi_0$. □

B. PROOF OF THEOREM 2

Let us consider a group STIT formula φ. Let m be an integer. Then the following items are equivalent:

1. φ is satisfiable in a model where each agent has at most 2^m actions;
2. φ is satisfiable in a model where each agent has exactly 2^m actions;
3. $GRID_m \wedge tr_2(\varphi)$ is PECP-satisfiable.

PROOF. $\boxed{1 \Rightarrow 2}$ Let $\mathcal{M}^0 = (W^0, \{R^0_J\}_{J \subseteq AGT}, V^0)$ be a STIT-model with at most 2^m actions per agent and $w \in W^0$ such that $\mathcal{M}^0, w \models \varphi$. We construct a sequence of models $\mathcal{M}^j = (W^j, \{R^j_J\}_{J \subseteq AGT}, V^j)$ such that all agents $j' \in \{1, \ldots, j\}$ have exactly 2^m actions in $\mathcal{M}_j$ and such that $\mathcal{M}^j$ is bisimilar to $\mathcal{M}^{j-1}$. We construct $\mathcal{M}^j$ from $\mathcal{M}^{j-1}$ as follows. Let $R^{j-1}_{\{j\}}(w_1), \ldots, R^{j-1}_{\{j\}}(w_k)$ be an enumeration of $R^{j-1}_{\{j\}}$- classes (that is, actions for agents j), where $k \leq 2^m$. Let $(Copy_\ell)_{\ell \in \{k+1,\ldots,2^m\}}$ be a family of disjoint copies of $R^{j-1}_{\{j\}}(w_1)$. We write $u\mathbf{C}v$ to say that $u = v$ or v is a copy of u or u is a copy of v. The model $\mathcal{M}^j = (W^j, \{R^j_J\}_{J \subseteq AGT}, V^j)$ is defined as follows:

- $W^j = W^{j-1} \cup \bigcup_{\ell \in \{k+1,\ldots,2^m\}} Copy_\ell$;
- $R^j_{\{j\}} = R^{j-1}_{\{j\}} \cup \bigcup_{\ell \in \{k+1,\ldots,2^m\}} \{(u, v) \mid u, v \in Copy_\ell\}$
- $R^j_{\{j'\}} = \mathbf{C} \circ R^{j-1}_{\{j'\}} \circ \mathbf{C}$ for all $j' \neq j$;
- $V^j(p) = \{v \in W^j \mid v\mathbf{C}u \text{ and } u \in V^{j-1}(p)\}$.

This construction makes that $\mathcal{M}^j$ and $\mathcal{M}^{j-1}$ are bisimilar and by induction we have that all agents $j' \in \{1, \ldots, j\}$ have exactly 2^m actions in $\mathcal{M}_j$. Finally, we have $\mathcal{M}^n, w \models \varphi$ and each agent has exactly 2^m actions in $\mathcal{M}^n$.

$\boxed{2 \Rightarrow 3}$ Let us consider a STIT model $\mathcal{M} = (W, \{R_J\}_{J \subseteq AGT}, V)$ in which each agent has exactly 2^m actions. Let $w \in W$ be such that $\mathcal{M}, w \models \varphi$. For all $j \in AGT$, let $R_{\{j\}}(w_{j,1}), \ldots, R_{\{j\}}(w_{j,2^m})$ be an enumeration of all $R_{\{j\}}$-classes in $\mathcal{M}$. Let us extend V such that in all worlds of $R_{\{j\}}(w_{j,i})$ the valuations of the atomic propositions in $\mathfrak{R}^j$ correspond to the binary digits in the binary representation of i. For all $d \in \{1, \ldots, m\}$:

- $V(\mathfrak{rep}^j_d) = \bigcup_{i=1..2^m \mid \text{ the } d^{th} \text{ digit of } i \text{ is } 1} R_{\{j\}}(w_{j,i})$

Independence of agents in $\mathcal{M}$ ensures that $\mathcal{M}, w \models GRID_m$. We prove that $\mathcal{M}, u \models tr_2(\psi)$ iff $\mathcal{M}, u \models \psi$ by induction over all subformulae ψ of φ.

$\boxed{3 \Rightarrow 1}$ Let $\mathcal{M} = (W, V)$ be a PECP-model and $w \in W$ such that $\mathcal{M}, w \models GRID_m \wedge tr_2(\varphi)$. We define $R_J = \sim_{\bigcup_{j \in J} \mathfrak{R}^j}$. The resulting Kripke-model $\mathcal{M}' = (W, \{R_J\}_{J \subseteq AGT}, V)$ is a STIT-model where each agent has exactly 2^m actions. In particular, it satisfies the independence of agents because $\mathcal{M}, w \models GRID_m$. We prove that $\mathcal{M}, u \models tr_2(\psi)$ iff $\mathcal{M}', u \models \psi$ by induction over all subformulae ψ of φ. □

C. PROOF OF THEOREM 4

The PECP-nested satisfiability problem is PSPACE-hard.

PROOF. We reduce the satisfiability problem of STIT-formulae where coalitions are taken from a linear set of coalitions, which is PSPACE-complete [Sch12] to the PECP-nested satisfiability problem: we use the translation tr_2 of Subsection 3.1. Let φ be a STIT-formula. We have φ is STIT-satisfiable iff $tr_2(\varphi)$ is PECP-satisfiable.

$\boxed{\Rightarrow}$ As it stated in [Sch12], the STIT where coalitions are taken from a linear set of coalitions has the exponential model property. So the result of Theorem 2 is true. Hence if φ is STIT-satisfiable then $GRID_m \wedge tr_2(\varphi)$ is PECP-satisfiable (where m is the length of φ). Hence $tr_2(\varphi)$ is PECP-satisfiable.

$\boxed{\Leftarrow}$ Suppose that there exists a PECP-model $\mathcal{M} = (W, V)$ and $w \in W$ such that $\mathcal{M}, w \models tr_2(\varphi)$. We define $R_J = \sim_{\bigcup_{j \in J}}$. Then the STIT model $\mathcal{M}' = (W, (R_J)_{J \in \varphi}, V)$ is such that $\mathcal{M}', w \models \varphi$. Remark that we do not need to specify all the relations R_J for all J. As long as R_J is specified for all coalitions J that appear in φ and that $R_J \subseteq R_{J'}$ if $J' \subseteq J$, we can extend the Kripke model $\mathcal{M}'$ to a completely specified STIT-model also satisfying φ.[6] □

D. PROOF OF THEOREM 5

The PECP-nested satisfiability problem is in PSPACE.

PROOF. We reduce the PECP-nested satisfiability problem to the satisfiability problem of STIT where coalitions are taken from a linear set of coalitions. We define the set $A_X = \{j_p \text{ such that } p \in X\}$ where j_p is a fresh agent corresponding to the atomic proposition p. Let us define the following translation:

- $tr_4(p) = p$;
- $tr_4(\neg\varphi) = \neg tr_4(\varphi)$;
- $tr_4(\varphi \wedge \psi) = tr_4(\varphi) \wedge tr_4(\psi)$;
- $tr_4([X]\varphi) = [A_X : stit] tr_4(\varphi)$.

Let us consider a fixed PECP-formula φ. We recall that a signature X appears in φ if there exists a formula ψ such that $\langle X \rangle \psi \in SF(\varphi)$. We have also to define the following formula

$$\begin{aligned} CONTROL = \quad & [\emptyset : stit] \\ & \bigwedge_{X \text{ appearing in } \varphi} \\ & \bigwedge_{p \in X} (p \leftrightarrow [A_X : stit]p) \wedge \\ & \quad (\neg p \leftrightarrow [A_X : stit]\neg p). \end{aligned}$$

$tr_4(\varphi) \wedge CONTROL$ is a STIT-formula which is computable in polynomial time and which satisfies the condition of nesting over groups (i.e., for any two operators $[J : stit]$ and $[J' : stit]$ occurring in the formula either $J \subseteq J'$ or $J' \subseteq J$). We also have that φ is PECP-satisfiable iff $tr_4(\varphi) \wedge CONTROL$ is satisfiable in a STIT-model.

$\boxed{\Rightarrow}$ Suppose that there exists an PECP-model $\mathcal{M} = (W, V)$ and $w \in W$ such that $\mathcal{M}, w \models \varphi$. We define $R_{A_X} = \sim_X$. Then the STIT model $\mathcal{M}' = (W, (R_{A_X})_{X \in \varphi}, V)$ is such that $\mathcal{M}', w \models tr_4(\varphi) \wedge CONTROL$. Remark that we do not need to specify all the relations R_J for all J. As long as R_J is specified for all coalitions J that appear in $tr_4(\varphi) \wedge CONTROL$ and that $R_J \subseteq R_{J'}$ if $J' \subseteq J$, we can extend the Kripke

[6] See [Sch12] for more details about this construction.

model $\mathcal{M}'$ to a completely specified STIT-model also satisfying $tr_4(\varphi) \wedge CONTROL$.[7]

$\boxed{\Leftarrow}$ Suppose that there exists a STIT-model $\mathcal{M}' = (W, (R_{A_X})_{X\in\varphi}, V)$ and a world $w \in W$ such that $\mathcal{M}', w \models tr_4(\varphi) \wedge CONTROL$. As $\mathcal{M}', w \models CONTROL$, we have $\sim_X = R_{A_X}$. This is the reason why if we define $\mathcal{M} = (W, \{\sim_X\}_{X\in 2^{\mathbf{P}}}, V)$. Consequently, we have $\mathcal{M}, w \models \varphi$. $\square$

E. PROOF OF THEOREM 6

Let $m = |\mathbf{P}|$. Then, a CL−PC formula φ is CL−PC satisfiable if and only if $(EXC^+ \wedge EXC^- \wedge COMPL \wedge GRID^*) \wedge tr_3(\varphi)$ is satisfiable in a STIT model where each agent has at most 2^m actions.

Proof. Let us suppose $|\mathbf{P}| = m$.

$\boxed{\Rightarrow}$ Let $\mathcal{M}^* = (\mathbf{P}_1, \ldots, \mathbf{P}_n, X^*)$ be a CL−PC model such that $\mathcal{M}^* \models \varphi$, where $\mathbf{P}_1, \ldots, \mathbf{P}_n$ is a partition of $\mathbf{P}$ among the agents in AGT. We build the STIT model $\mathcal{M} = (W, \{R_J\}_{J\subseteq AGT}, V)$ as follows:

- $W = \{X : X \subseteq \mathbf{P}\}$,
- for all $J \subseteq AGT$ and for all $X, X' \in W$, $(X, X') \in R'_J$ if and only if $X_J = X'_J$,
- for all $p \in \mathbf{P}$ and for all $X \in W$, $X \in V(p)$ if and only if $p \in X$,

where for any $X \subseteq \mathbf{P}$ and for any $J \subseteq AGT$, $X_J = X \cap \mathbf{P}_J$ (with $\mathbf{P}_J = \bigcup_{i\in J} \mathbf{P}_i$). The size of $\mathcal{M}$ is 2^m. It follows that the number of R_{AGT}-equivalence classes (*alias* joint actions) is equal or lower than 2^m. Consequently, the number of actions for every agent is bounded by 2^m.

It is straightforward to prove that for all $X \in W$ we have $\mathcal{M}, X \models EXC^+ \wedge EXC^- \wedge COMPL \wedge GRID^*$. Moreover, by induction on the structure of φ, we prove that $\mathcal{M}, X^* \models tr_3(\varphi)$. The only interesting case is $\varphi = \Diamond_J\psi$:

$$\begin{aligned}\mathcal{M}^* \models \Diamond_J\psi \text{ iff there exists } X_J \subseteq \mathbf{P}_J \text{ s.t. } \mathcal{M}^* \textstyle\bigoplus X_J \models \psi\\ \text{iff there exists } X_J \subseteq \mathbf{P}_J \text{ s.t.}\\ \mathcal{M}, X_J \cup X^*_{AGT\setminus J} \models tr_3(\psi) \text{ (by I.H.)}\\ \text{iff } \mathcal{M}, X^* \models \langle AGT \setminus J : stit\rangle tr_3(\psi)\end{aligned}$$

$\boxed{\Leftarrow}$ Let $\mathcal{M} = (W, \{R_J\}_{J\subseteq AGT}, V)$ be a STIT model where the number of actions for every agent is bounded by 2^m and $w_0 \in W$ such that $\mathcal{M}, w_0 \models (EXC^+ \wedge EXC^- \wedge COMPL \wedge GRID^*) \wedge tr_3(\varphi)$.

For any $i \in AGT$, let

$$Ctrl_i = \{p \in \mathbf{P} : \forall v \in W, \begin{array}{l}\mathcal{M}, v \models [\{i\} : stit]p \text{ or}\\ \mathcal{M}, v \models [\{i\} : stit]\neg p\end{array}\}$$

be the set of atoms in $\mathbf{P}$ controlled by agent i. For any $J \subseteq AGT$, let $Ctrl_J = \bigcup_{i\in J} Ctrl_i$.

Lemma 1. *For all $J \subseteq AGT$, $X \subseteq \mathbf{P}$, $\pi_X \subseteq X$ and $w \in W$ we have:*

(i) if $Ctrl_J = X$ then $Ctrl_{AGT\setminus J} = \mathbf{P} \setminus X$,

(ii) if $\mathcal{M}, w \models \bigwedge_{p\in\pi^+_X} p \wedge \bigwedge_{p\in\pi^-_X} \neg p$ and $Ctrl_J = X$ then, for all $v \in R_J(w)$, we have $\mathcal{M}, v \models \bigwedge_{p\in\pi^+_X} p \wedge \bigwedge_{p\in\pi^-_X} \neg p$,

(iii) if $Ctrl_J = X$ then, for all $\pi'_{\mathbf{P}\setminus X} \subseteq \mathbf{P} \setminus X$, there exists $v \in R_J(w)$ such that $\mathcal{M}, v \models \bigwedge_{p\in\pi'^+_{\mathbf{P}\setminus X}} p \wedge \bigwedge_{p\in\pi'^-_{\mathbf{P}\setminus X}} \neg p$.

where for any $X \subseteq \mathbf{P}$ and for any $\pi_X \subseteq X$, $\pi^+_X = \pi_X$ and $\pi^-_X = X \setminus \pi_X$.

Proof. (i) Let us suppose that $p \notin Ctrl_J$. We are going to prove that $p \in Ctrl_{AGT\setminus J}$. From $p \notin Ctrl_J$ it follows that for all $w \in W$ we have $\mathcal{M}, v \models \neg p$ for some $v \in R_J(w)$. This implies that for all $i \in J$ and for all $w \in W$ we have $\mathcal{M}, w \models \neg[\{i\} : stit]p \wedge \neg[\{i\} : stit]\neg p$. From $\mathcal{M}, w_0 \models COMPL$ it follows that there is $i \in AGT \setminus J$ such that $\mathcal{M}, w \models [\{i\} : stit]p \vee [\{i\} : stit]\neg p$ for all $w \in W$. The latter implies that $p \in Ctrl_{AGT\setminus J}$. The other direction (i.e., $p \in Ctrl_J$ implies $p \notin Ctrl_{AGT\setminus J}$) follows from $\mathcal{M}, w_0 \models EXC^+ \wedge EXC^-$.

(ii) Let us suppose that $\mathcal{M}, w \models \bigwedge_{p\in\pi^+_X} p \wedge \bigwedge_{p\in\pi^-_X} \neg p$ and $Ctrl_J = X$. By the fact that relations R_J are reflexive, it follows that, for all $p \in \pi^+_X$, there exists $i \in J$ such that $\mathcal{M}, w \models [\{i\} : stit]p$ and for all $p \in \pi^-_X$ there exists $i \in J$ such that $\mathcal{M}, v \models [\{i\} : stit]\neg p$. From the latter it follows that for all $p \in \pi^+_X$ we have $\mathcal{M}, w \models [J : stit]p$ and for all $p \in \pi^-_X$ we have $\mathcal{M}, v \models [J : stit]\neg p$. Therefore, for all $v \in R_J(w)$, we have $\mathcal{M}, v \models \bigwedge_{p\in\pi^+_X} p \wedge \bigwedge_{p\in\pi^-_X} \neg p$.

(iii) Let us suppose that $Ctrl_J = X$ and let us consider an arbitrary $\pi'_{\mathbf{P}\setminus X} \subseteq \mathbf{P} \setminus X$ and $w \in W$. From $\mathcal{M}, w_0 \models GRID^*$ it follows that there exists $v \in W$ such that $\mathcal{M}, v \models \bigwedge_{p\in\pi'^+_{\mathbf{P}\setminus X}} p \wedge \bigwedge_{p\in\pi'^-_{\mathbf{P}\setminus X}} \neg p$. By item (ii), the latter implies that there exists $v \in W$ such that $\mathcal{M}, v \models [AGT \setminus J : stit](\bigwedge_{p\in\pi'^+_{\mathbf{P}\setminus X}} p \wedge \bigwedge_{p\in\pi'^-_{\mathbf{P}\setminus X}} \neg p)$. From the constraint of *independence of agents* it follows that there exists $v \in R_J(w)$ such that $\mathcal{M}, v \models \bigwedge_{p\in\pi'^+_{\mathbf{P}\setminus X}} p \wedge \bigwedge_{p\in\pi'^-_{\mathbf{P}\setminus X}} \neg p$. $\square$

We transform the STIT model $\mathcal{M}$ in a CL−PC model $\mathcal{M}^* = (\mathbf{P}_1, \ldots, \mathbf{P}_n, X^*)$ as follows:

- for all $p \in \mathbf{P}$, $p \in X^*$ if and only if $w_0 \in V(p)$,
- for all $p \in \mathbf{P}$ and for all $i \in AGT$, $p \in \mathbf{P}_i$ if and only if $p \in Ctrl_i$.

By the item (i) of Lemma 1 it is easy to check that $\mathcal{M}^*$ is indeed a CL−PC model. In particular, $\mathbf{P}_1, \ldots, \mathbf{P}_n$ is a partition of $\mathbf{P}$ among the agents in AGT.

By induction on the structure of φ and by using Lemma 1 it is straightforward to prove that $\mathcal{M}^* \models \varphi$. The only interesting case is $\varphi = \Diamond_J\psi$:

$$\begin{aligned}\mathcal{M}, w_0 \models &\langle AGT \setminus J : stit\rangle tr_3(\psi)\\ &\text{iff } \mathcal{M}, v \models tr_3(\psi) \text{ for some } v \in R_{AGT\setminus J}(w_0)\\ &\text{iff there exists } X_J \subseteq \mathbf{P}_J \text{ s.t.}\\ &\quad (\mathbf{P}_1, \ldots, \mathbf{P}_n, X_J \cup X^*_{AGT\setminus J}) \models \psi\\ &\quad \text{(by I.H., and items (ii) and (iii)}\\ &\quad \text{of Lemma 1)}\\ &\text{iff } \mathcal{M}^* \models \Diamond_J\psi\end{aligned}$$

This completes the proof.

[7] Again see [Sch12] for more details about this construction.

Deludedly Agreeing to Agree

Ziv Hellman*
Department of Statistics and Operations Research
The School of Mathematical Sciences
Tel Aviv University
Tel Aviv, 69978, Israel
zivhellman@post.tau.ac.il

ABSTRACT

We study conditions relating to the impossibility of agreeing to disagree in models of interactive KD45 belief (in contrast to models of S5 knowledge, which are used in nearly all the agreements literature). We show that even when the truth axiom is not assumed it turns out that players will find it impossible to agree to disagree under fairly broad conditions.[1]

General Terms

Theory

Keywords

agreeing to agree, beliefs, KD45

1. INTRODUCTION

One of the strongest assumptions underpinning the standard model of knowledge, known as S5, is the *truth axiom*, which essentially states that 'everything that a player knows is true'. This is equivalent, from one perspective, to asserting that no mistakes are ever made in the processing of signals.

Mistakes, of course, abound around us, and sometimes such mistakes can have significant consequences. Consider, for example the following scenario (a variation of an example appearing in [Hart and Tauman (2004)]): There are two traders. They trade on a daily basis, and since a trade involves one trader selling and the other buying, they can at least observe each others' willingness to trade. We may imagine that these two traders are the 'market leaders', in the sense that their actions are followed by others in the market and copied.

Let Ω be the set of all states of the world, with Ω containing nine states; $\Omega = \{1, 2, \ldots, 9\}$. For simplicity we will assume that there is a common prior p over Ω, with $p(\omega) = 1/9$ for all states ω. The private information of the two traders, Anne and Bob are summarized by partitions Π_A and Π_B respectively, with

$$\Pi_A = 1234|5678|9$$

and

$$\Pi_B = 123|456|789.$$

One standard interpretation of the structure of such partitional knowledge is that Anne and Bob receive signals. If the true state is 2, for example, Anne receives a signal that enables her to rule out the states $5, 6, 7, 8, 9$, and she therefore knows that the true state is one of $1, 2, 3, 4$. Bob, at the true state 3, receives a signal that enables him to rule out the states $4, 5, 6, 7, 8, 9$, and he therefore knows that the true state is one of $1, 2, 3$. Specifically, suppose that Bob may receive any one of three signals, $\sigma_1, \sigma_2, \sigma_3$, where σ_1 informs Bob that the true state is one of $1, 2, 3$, σ_2 informs Bob that the true state is one of $4, 5, 6$, and σ_3 informs Bob that the true state is one of $7, 8, 9$ (we will be less interested in this example with specifying Anne's possible signals).

Signal		States
σ_1	$\rightarrow$	$\{1, 2, 3\}$
σ_2	$\rightarrow$	$\{4, 5, 6\}$
σ_3	$\rightarrow$	$\{7, 8, 9\}$

Figure 1: Bob's signals and their interpretation when there are no processing errors.

So far, so standard. Now consider the possibility of a mistake in signals processing on the part of Bob. Suppose that Bob inputs the signals he receives into a black box that he has been assured outputs $1, 2, 3$, $4, 5, 6$, or $7, 8, 9$ if the input is σ_1,σ_2, or σ_3 respectively. Unbeknownst to Bob (and to Anne), however, Bob's black box is defective; when either σ_1 or σ_2 are given as input, the box outputs $4, 5, 6$ (hence even if, e.g., the true state is 1 Bob thinks the true state is one of $\{4, 5, 6\}$.

Signal		States
σ_1	$\rightarrow$	$\{4, 5, 6\}$
σ_2	$\rightarrow$	$\{4, 5, 6\}$
σ_3	$\rightarrow$	$\{7, 8, 9\}$

Figure 2: Bob's signal processing error.

Consider next the event $E = \{4, 9\}$. This event will be interpreted as a 'good' outcome (e.g., company earnings are about to rise), with the complement representing a 'bad' event that ought to trigger the sale of shares. Suppose that

*Research supported in part by the European Research Council under the European Commission's Seventh Framework Programme (FP7/2007 - 2013)/ERC grant agreement no. 249159, and in part by Israel Science Foundation grants 538/11 and 212/09.

[1] What follows is an extended abstract for TARK, not a full paper.

TARK 2013, Chennai, India.

the true state is 2, and that each one of the two traders behaves each day according to the following rule:

$$\begin{cases} \text{Buy} & \text{if the probability of } E \text{ is 0.3 or more;} \\ \text{Sell} & \text{if the probability of } E \text{ is less than 0.3.} \end{cases}$$

Given these assumptions, the following sequence of actions transpires. On Day 1, Anne, who processes signals correctly, supposes that the true state is one of $1, 2, 3, 4$, judges the probability of E to be $1/4$ and seeks to sell shares. Bob erroneously supposes that the true state is one of $4, 5, 6$, judges the probability of E to be $1/3$, and therefore buys shares from Anne.

Since Bob was willing to buy on Day 1, Anne 'learns' that the true state is not in $1, 2, 3$. She therefore erroneously supposes on Day 2 that the true state is 4 and offers to buy on Day 2. Bob does the same. By Day 3, it is 'common knowledge' that 4 is the 'true state' – Bob's error has now become Anne's error. Both traders seek to buy as many shares as they can, to their detriment, and a bubble has developed.

[Geanakoplos (1989)] and [Morris (1996)] show that in knowledge models that satisfy the truth axiom (but are not necessarily S5) more information is always beneficial for a player, in the sense that with more information a rational player will never choose an action that gives him less in expectation than an action that he chooses when he has less information. Without the truth axiom, that no longer holds true. Indeed, as the example here shows, without the truth axiom, not only is the 'mistaken' player in danger of choosing detrimental actions, his errors can cascade and 'infect' other players to their detriment: in Day 1 above, Anne makes the right decision in seeking to sell shares, but on Day 2, due to Bob's mistake, she is buying shares. Arguably, Anne has been mistaken all along, in accepting Bob's reports at face value, without considering the possibility that Bob might be mistaken.

The above story motivates the study of agreement and disagreement in models of *belief* as opposed to models of *knowledge*, which is the standard setting of most of the agreement literature.

2. PRELIMINARIES

2.1 Belief Structures

Fix a finite set of players I and a finite set of *states of the world*[2] denoted by Ω. Subsets of Ω are called *events*. The set of probability distributions over Ω is denoted by $\Delta(\Omega)$.

A *type function* t_i over Ω for player i is defined by assigning, for each ω, a probability distribution $t_i(\omega) \in \Delta(\Omega)$ representing player i's beliefs at ω. We associate with each type function t_i a partition Π_i of Ω defined[3] by $\Pi_i(\omega) = \{\omega' \mid t_i(\omega') = t_i(\omega)\}$. If we impose on a type function the property that $t_i(\omega)(\Pi_i(\omega)) = 1$, then the type functions is *partitional*. A *probabilistic belief structure* over Ω is then a set of partitional type functions $(t_i)_{i\in I}$ over Ω.

A function $b_i : \Omega \to 2^\Omega \setminus \emptyset$ is a *possibility function*. The event $b_i(\omega)$ is interpreted as the set of states that are considered possible for i at ω, while all other states are excluded by i at ω. We will call a possibility function $b_i : \Omega \to 2^\Omega \setminus \{\emptyset\}$ that is measurable with respect to a partition Π_i and satisfies $b_i(\omega) \subseteq \Pi_i(\omega)$ for each $\omega \in \Omega$ a *KD45 possibility function*.

A *belief structure* over Ω is a set of pairs $\mathbf{\Pi} = (\Pi_i, b_i)_{i\in I}$, where each b_i is a KD45 possibility function with respect to the partition Π_i of Ω. We will sometimes also call such a structure a KD45 belief structure.

The general structure of a model of KD45 belief of a player i is of an over-arching partition Π_i, with each partition element $\pi \in \Pi_i$ furthermore partitioned into $b_i(\omega)$ and $f_i(\omega)$ (using an arbitrary $\omega \in \pi$). Every element $\omega' \in f_i(\omega)$ is mapped by b_i into $b_i(\omega)$, where it is 'trapped', in the sense that $b_i(b_i(\omega')) = b_i(\omega') = b_i(\omega)$.

A probabilistic belief structure $(t_i)_{i\in I}$ over Ω induces a belief structure $(\Pi_i, b_i)_{i\in I}$ over Ω, where Π_i is the partition of Ω into the types of player i and $b_i(\omega)$ is the set of states in $\Pi_i(\omega)$ that have positive $t_i(\omega)$ probability. Conversely, every belief structure over Ω is induced by a probabilistic belief structure over Ω. We will sometimes make use of this by choosing, for a given belief structure $\mathbf{\Pi} = (\Pi_i, b_i)_{i\in I}$, an arbitrary probabilistic belief structure $(t_i^b)_{i\in I}$ that induces $\mathbf{\Pi}$.

2.2 Delusion

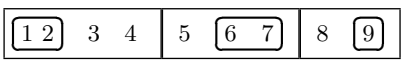
Let $\mathbf{\Pi} = (\Pi_i, b_i)_{i\in I}$ be a belief structure. If $\omega \in b_i(\omega)$ then b_i is *non-deluded* at ω. If $\omega \notin b_i(\omega)$ then b_i is *deluded* at ω; in this case we will also sometimes say that ω is a deluded state for player i. This is where we are dropping the 'truth axiom': if one accepts the truth axiom there are never any deluded states for any player.

If there is at least one state at which b_i is deluded, then b_i is *delusional*, and we will similarly say that the corresponding belief operator B_i is delusional if this is the case. It is straight-forward to show that a belief structure $\mathbf{\Pi}$ is non-delusional for all players if and only if it is an S5 structure, and it is similarly straight-forward to show that a state ω is non-deluded for player i if and only if $t_i^b(\omega) = 0$ for any probabilistic belief structure $(t_i^b)_{i\in I}$ that induces $\mathbf{\Pi}$.

Definition 1. A KD45 belief structure at which at all states $\omega \in \Omega$ either a) every player i is deluded at ω or b) every player i is non-deluded at ω will be called a *non-singular* structure. ♦

In examples, we will compactly express KD45 belief structures by separating states in different partition elements of Π_i by the square boxes. Within each partition element we will denote states that are in the same component of $b_i(\omega)$ by an oval box.

For example, if we write



then the intention is, for example, that 5, 6 and 7 are all in the same partition element, i.e., $\Pi_i(5) = \{5, 6, 7\}$, but 5 is a delusional state such that $b_i(5) = \{6, 7\}$.

3. BELIEF REVISION

The general approach we will follow is: in standard S5 concepts and formulae, replace Π_i by b_i and see what happens. We will apply this now to Bayesian belief revision.

[2] In the basic definitions of elements of belief structures we largely follow [Samet (2011)].

[3] The presentation here reverses most presentations of belief structures, in which partitions are given and used to define type functions; here we are starting with type functions and using them to define the partitions.

3.1 Standard belief revision and priors

Let μ be a probability distribution over Ω, and let Π_i be a partition of of Ω. The *(standard) revision* of μ at ω according to Π_i is the probability distribution $\widehat{\mu}(\omega)$ such that

$$\widehat{\mu}(\omega)(\omega') = \begin{cases} \frac{\mu(\omega')}{\mu(\Pi_i(\omega))} & \text{if } \omega' \in \Pi_i(\omega) \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

if $\mu(\Pi_i(\omega)) > 0$; otherwise it is undefined.

We may interpret this as follows: ex ante the player has a prior probability distribution of full support. When updating following a signal, the player excludes states outside $b_i(\omega)$, i.e. gives them zero probability. Since the player mistakes the reading of the signal, it is possible that he or she ends up giving the true state ω zero probability.

Let f be a random variable over Ω, μ be a probability distribution over Ω, and Π_i a partition of Ω. Then the *conditional expected value of f at ω* is

$$E_i^{\mu}(f \mid \Pi_i(\omega)) := \frac{1}{\mu(\Pi_i(\omega))} \sum_{\omega' \in \Pi_i(\omega)} f(\omega')\mu(\omega)(\omega'), \quad (2)$$

if $\mu(\Pi_i(\omega)) \neq 0$ (otherwise it is not defined).

Let $(t_i)_{i\in I}$ be a probabilistic belief structure over Ω, with $(\Pi_i)_{i\in I}$ the corresponding partition. A *(standard) prior* for t_i is a probability distribution $\mu \in \Delta(\Omega)$, such that $\widehat{\mu}(\omega) = t_i(\omega)$ at each ω, where $\widehat{\mu}(\omega)$ is the standard revision of μ at ω according to Π_i as defined in Equation (1). A *(standard) common prior* for $(t_i)_{i\in I}$ is a probability distribution $\mu \in \Delta(\Omega)$ that is a prior for each t_i.

Given a probabilistic belief structure $(t_i)_{i\in I}$ with corresponding partition $(P_i)_{i\in I}$, player i's *posterior expected value of f at ω* is

$$E_i^{t_i}(f \mid \Pi_i(\omega)) := \sum_{\omega' \in \Pi_i(\omega)} t_i(\omega')f(\omega'). \quad (3)$$

If there is a common prior μ, then for any random variable f the posterior expected value of each player equals the conditional expected value of f relative to μ and Π_i, i.e., $E_i^{t_i}(f \mid \Pi_i(\omega)) = E_i^{\mu}(f \mid \Pi_i(\omega))$.

3.2 Delusional belief revision

Now replace Π_i by b_i in Equations (1) and (3).

Let μ be a probability distribution over Ω, and let b_i be a belief structure over Ω with corresponding partition Π_i. We introduce here the *delusional revision* of μ at ω according to b_i, defining it as the probability distribution $\widehat{\mu}(\omega)$ such that

$$\widehat{\mu}(\omega)(\omega') = \begin{cases} \frac{\mu(\omega')}{\mu(b_i(\omega))} & \text{if } \omega' \in b_i(\omega) \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

if $\mu(b_i(\omega)) > 0$; otherwise it is undefined.

Let f be a random variable over Ω, let μ be a probability distribution over Ω, and let b_i be a belief structure over Ω with corresponding partition Π_i. Then the *delusional conditional expected value of f at ω* according to b_i is

$$E_i^{\mu}(f \mid b_i(\omega)) := \frac{1}{\mu(b_i(\omega))} \sum_{\omega' \in b_i(\omega)} f(\omega')\mu(\omega)(\omega'), \quad (5)$$

if $\mu(\Pi_i(\omega)) \neq 0$ (otherwise it is not defined).

Let $(t_i)_{i\in I}$ be a probabilistic belief structure over Ω, with $(\Pi_i)_{i\in I}$ the corresponding partition. Let b_i be the belief structure induced by t_i. A *delusional prior* for t_i is a probability distribution $\mu \in \Delta(\Omega)$, such that $\widehat{\mu}(\omega) = t_i(\omega)$ at each ω, where $\widehat{\mu}(\omega)$ is the delusional revision of μ at ω according to b_i as defined in Equation (4). A *common delusional prior* for $(t_i)_{i\in I}$ is a probability distribution $\mu \in \Delta(\Omega)$ that is a prior for each t_i.

Let ϕ_i be a standard prior for t_i, and suppose that for a state ω, $t_i(\omega)(\omega) = 0$, and therefore that $\omega \in \Pi_i(\omega)$ but $\omega \notin b_i(\omega)$. Then by Equation (1) it must be the case that $\phi_i(\omega) = 0$. The same reasoning does not hold for a delusional prior; a standard prior is a delusional prior, but the converse is not necessarily true.

EXAMPLE 1. *Consider a one-player probabilistic belief structure over a state space $\Omega = \{\omega_1, \omega_2, \omega_3\}$ defined by*

$$t(\omega_k)(\omega_1) = 0;\ t(\omega_k)(\omega_2) = \frac{1}{2};\ t(\omega_k)(\omega_3) = \frac{1}{2}$$

for $k \in \{1, 2, 3\}$:

$$t = \boxed{\underbrace{0}_{\omega_1}\ \underbrace{\tfrac{1}{2}}_{\omega_2}\ \underbrace{\tfrac{1}{2}}_{\omega_3}}.$$

This induces a belief structure

$$b(\omega_1) = b(\omega_2) = b(\omega_3) = \{\omega_2, \omega_3\},$$

with ω_1 a deluded state, visualised as

$$\boxed{\omega_1 \quad \boxed{\omega_2 \quad \omega_3}}$$

The probability structure has only one (standard) prior, $\mu = (0, 1/2, 1/2)$, but it has an infinite number of delusional priors. The set of delusional priors includes, for example, $(0, 1/2, 1/2)$ and $(1/3, 1/3, 1/3)$. ♦

3.3 Interpersonal Belief Credibility

S5 knowledge structures, by dint of satisfying the truth axiom, satisfy the property that $\bigcap_{i\in I} b_i(\omega) \neq \emptyset$ for all states $\omega \in \Omega$.

In KD45 belief structures there may be states at which $\bigcap_{i\in I} b_i(\omega) = \emptyset$. When

$$\bigcap_{i\in I} b_i(\omega) \neq \emptyset$$

for all states ω we will say that the belief structure satisfies *interpersonal belief credibility.*

4. COMMON BELIEF

Denote $b(\omega) = \bigcup_{i\in I} b_i(\omega)$ and let b^m be the composition of the function k repeated m times. Furthermore, define for each ω the *common belief set* $b^Q(\omega)$ of ω in Ω by

$$b^Q(\omega) := \bigcup_{m\geq 1} b^m(\omega) \quad (6)$$

S5 knowledge structures are naturally partitioned into common knowledge components. Let $\{\Omega, (k_i)_{i\in I})\}$ be a knowledge structure. The *meet* is the finest common coarsening of the players' partitions. Each element of the meet of $\mathbf{\Pi}$ is called a *common knowledge component* of $\mathbf{\Pi}$. Denote by $C(\omega)$ the common knowledge component of a state ω in a knowledge structure.

Let $T \subseteq \Omega$ be a common knowledge component. T can be characterised in several ways. One way is by knowledge chains. Defining $k : \Omega \to 2^{\Omega}$ by $k(\omega) := \bigcup_{i \in I} \Pi_i(\omega)$ and for $m \geq 0$ letting k^m be the composition of the function k repeated m times, it is well known that $T = \bigcup_{m \geq 1} k^m(\omega)$ for any $\omega \in T$.

In addition, in S5 knowledge structures, a common knowledge component at ω can be characterised by the fact that

$$C(\omega) = \bigcup_{\omega \in T} \Pi_i(\omega).$$

for *all* players $i \in I$

The corresponding statement in KD45 does not hold, i.e., it is not always the case that $b^Q(\omega) = \bigcup_{\omega' \in \Omega_0} b_i(\omega')$ for some $\omega_0 \subseteq \Omega$. When it does we will want to take note of this.

Definition 2. There is strong common belief in truth at a state ω if there exists $\Omega_0 \subseteq \Omega$ such that $b^Q(\omega) = \bigcup_{\omega' \in \Omega_0} b_i(\omega')$ for all $i \in I$. ♦

PROPOSITION 1. *There is strong common belief in truth at every state iff the belief structure is non-singular.*

5. AGREEMENT IN BELIEF STRUCTURES

5.1 Standard No Betting

Definition 3. An n-tuple of random variables $\{f_1, \ldots, f_n\}$ is a *bet* if $\sum_{i=1}^{n} f_i = 0$. ♦

Definition 4. Let $(t_i)_{i \in I}$ be a probabilistic belief structure. Then a bet is an *agreeable bet* at ω (relative to (t_i)) if $E_i^{t_i}(f \mid \Pi_i(\omega)) > 0$ for all $i \in I$. A bet f is a *common knowledge agreeable bet at* ω if it is common knowledge at ω that f is an agreeable bet. ♦

The main characterisation of the existence of common priors in S5 knowledge models in the literature is what is sometimes known as the No Betting Theorem: a finite type space has a common prior if and only if there does not exist a common knowledge agreeable bet at any ω. In the special case of a two-player probabilistic belief structure where the random variable is the characteristic function

$$1^H(\omega) = \begin{cases} 1 & \text{if } \omega \in H \\ 0 & \text{if } \omega \notin H \end{cases}$$

where H is an event, this characterisation implies the seminal Aumann Agreement Theorem ([Aumann (1976)]), which states that if it is common knowledge at a state of the world that player 1 ascribes probability η_1 to event H and player 2 ascribes probability η_2 to the same event, then $\eta_1 = \eta_2$.

5.2 KD45 No Betting

Definition 5. Let $(t_i)_{i \in I}$ be a probabilistic belief structure and $(b_i)_{i \in I}$ a belief structure induced by $(t_i)_{i \in I}$. A bet f is a *common belief agreeable bet at* ω if it is common belief at ω that f is an agreeable bet. ♦

With these definitions, we can now ask whether an analogue to the No Betting Theorem of S5 models holds in the KD45 setting. Given a probabilistic belief structure $(t_i)_{i \in I}$, does the existence of a common delusional prior imply that there is no common belief agreeable bet?

The answer to this question is no, as the following example[4] shows.

EXAMPLE 2. *Let $\Omega = \{\omega_1, \omega_2, \omega_3\}$. Consider the two-player probabilistic belief structure (t_1, t_2) defined by*

$$t_1(\omega_k)(\omega_1) = \frac{1}{3}; t_1(\omega_k)(\omega_2) = \frac{1}{3}; t_1(\omega_k)(\omega_3) = \frac{1}{3},$$

and

$$t_2(\omega_k)(\omega_1) = 0; t_2(\omega_k)(\omega_2) = \frac{1}{2}; t_2(\omega_k)(\omega_3) = \frac{1}{2}$$

for $k \in \{1, 2, 3\}$:

$$t_1 = \boxed{\underbrace{\tfrac{1}{3}}_{\omega_1} \; \underbrace{\tfrac{1}{3}}_{\omega_2} \; \underbrace{\tfrac{1}{3}}_{\omega_3}},$$

$$t_2 = \boxed{\underbrace{0}_{\omega_1} \; \underbrace{\tfrac{1}{2}}_{\omega_2} \; \underbrace{\tfrac{1}{2}}_{\omega_3}}.$$

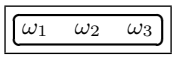
This induces the belief structure (b_1, b_2)

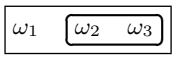
$$b_1(\omega_1) = b_1(\omega_2) = b_1(\omega_3) = \{\omega_1, \omega_2, \omega_3\},$$

and

$$b_2(\omega_1) = b_2(\omega_2) = b_2(\omega_3) = \{\omega_2, \omega_3\},$$

visualised as





For this belief structure, $\mu = (1/3, 1/3, 1/3)$ is a common delusional prior. Let $H = \{\omega_1, \omega_2\}$. Then it is common belief at every state ω that $E_1^{t_1}(1^H \mid b_1(\omega)) = 2/3$, while $E_2^{t_2}(1^H \mid b_2(\omega)) = 1/2$. ♦

To recapitulate something resembling the No Betting Theorem in belief structures, we add a new definition.

Definition 6. There is *weak common belief in truth*[5] at a state ω if there exists a state $\omega' \in b^Q(\omega)$ at which there is strong common belief in truth. ♦

An equivalent way of stating the content of Definition 6 is as follows: there is weak common belief in truth at ω iff there exists a state $\omega' \in b^Q(\omega)$ such that

$$\bigcup_{\omega'' \in b^Q(\omega')} b_i(\omega'') = \bigcup_{\omega'' \in b^Q(\omega')} b_j(\omega'')$$

for all $i, j \in I$. This can be read intuitively as the players 'eventually' getting to strong common belief in truth as they follow chains in the common belief set.

A belief structure version of the No Betting Theorem can be attained if we assume weak common belief in truth.

[4] This example is inspired by an example in [Collins (1997)].

[5] Although weak common belief in truth may seem abstract at first reading, it arises naturally in the study of interactive belief models. Concepts very similar to that of weak common belief in truth are introduced and used in [Battigalli and Bonanno (1999)] and [Tarbush (2011)].

THEOREM 1. *Let $(t_i)_{i\in I}$ be a probabilistic belief structure over Ω and let ω be a state at which there is weak common belief in truth. Then there is a common delusional prior if and only if there is no common belief agreeable bet at ω.*

Since strong common belief in truth implies weak common belief in truth, and in a non-singular probabilistic belief structure there is strong common belief in truth at every state, Theorem 2 (which is close in content to a result appearing in [Bonanno and Nehring (1999)]) follows from Theorem 1 as a corollary.

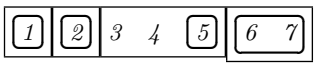
THEOREM 2. *Let $(t_i)_{i\in I}$ be a non-singular probabilistic belief structure over Ω. Then there is a common delusional prior if and only if there is no common belief agreeable bet at any state $\omega \in \Omega$.*

EXAMPLE 3. *The state space consists of $\{0,1,2,3,4,5,6,7\}$. There are two players, i and j. The belief structure*

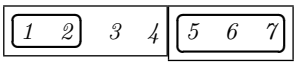
$$((\Pi_i, b_i), (\Pi_j, b_j))$$

is as follows:

Player i's beliefs are



Player j's beliefs are



The states 3 and 4 are delusional states for both player i and player j, hence they perceive the same world. Note also that $b_i(3) = \{5\}$ while $b_j(3) = \{1,2\}$, and this structure therefore does not satisfy interpersonal belief credibility. In fact, the structure can naturally be divided into two 'certainty components', $\{1,2\}$ and $\{5,6,7\}$; at states 3 and 4, player i is certain that the true component is $\{5,6,7\}$ while player j is certain that the true component is $\{1,2\}$.

The above belief structure can be induced by the following non-singular probabilistic belief structure (t_i, t_j):

$$t_i = \overbrace{\overbrace{1}^{1}\ \overbrace{2}^{1}}\ \overbrace{\overbrace{3}^{0}\ \overbrace{4}^{0}\ \overbrace{5}^{1}}\ \overbrace{\overbrace{6}^{1/2}\ \overbrace{7}^{1/2}}$$

$$t_j = \overbrace{\overbrace{1}^{1/2}\ \overbrace{2}^{1/2}\ \overbrace{3}^{0}\ \overbrace{4}^{0}}\ \overbrace{\overbrace{5}^{1/3}\ \overbrace{6}^{1/3}\ \overbrace{7}^{1/3}}$$

This probabilistic belief structure has an infinite number of common delusional priors; for example,

$$\mu = (\frac{1}{7}, \frac{1}{7}, \frac{1}{14}, \frac{1}{14}, \frac{1}{7}, \frac{1}{7}, \frac{1}{7}).$$

There can therefore be no common belief disagreement.

We close by noting the following. Suppose that are working in the standard S5 knowledge model (hence that the players make 'no mistakes', that is, they revise beliefs perfectly correctly), and that the players start out with two separate priors, given by

$$\mu_i = (\frac{1}{7}, \frac{1}{7}, \frac{1}{28}, \frac{3}{28}, \frac{1}{7}, \frac{1}{7}, \frac{1}{7})$$

and

$$\mu_j = (\frac{1}{7}, \frac{1}{7}, \frac{1}{14}, \frac{1}{14}, \frac{1}{7}, \frac{1}{7}, \frac{1}{7}).$$

Then the players will revise their beliefs into the following posteriors

$$\widehat{t}_i = \overbrace{\overbrace{1}^{1}\ \overbrace{2}^{1}}\ \overbrace{\overbrace{3}^{1/8}\ \overbrace{4}^{3/8}\ \overbrace{5}^{1/2}}\ \overbrace{\overbrace{6}^{1/2}\ \overbrace{7}^{1/2}}$$

$$\widehat{t}_j = \overbrace{\overbrace{1}^{1/3}\ \overbrace{2}^{1/3}\ \overbrace{3}^{1/6}\ \overbrace{4}^{1/6}}\ \overbrace{\overbrace{5}^{1/3}\ \overbrace{6}^{1/3}\ \overbrace{7}^{1/3}}.$$

Defining a bet $(f_i, -f_i)$ by

$$f_i = (1/4, 1/4, -6, 3, -1/8, 1/32, 1/32),$$

it can be checked that this bet is common knowledge agreeable at every state. But if the players make mistakes, using delusional revision with both players having deluded states at 3 and 4, then instead of $\widehat{t}_i$ and $\widehat{t}_j$ they will derive the posteriors t_i and t_j, which as we have seen have a common delusional prior precluding disagreement. ♦

6. REFERENCES

[Aumann (1976)] Aumann, R. J. (1976), Agreeing to Disagree, *Ann. Statist.*, 4(6), 1236–1239.

[Battigalli and Bonanno (1999)] Battigalli, P., and G. Bonanno (1999), Recent Results on Belief, Knowledge and the Epistemic Foundations of Game Theory *Review of Economic Studies*, 64, 23–46.

[Bonanno and Nehring (1999)] Bonanno, G., and K. Nehring (1999), How to Make Sense of the Common Prior Assumption under Incomplete Information, *Int. J. Game Theory* 28, 409–434.

[Collins (1997)] Collins, J., (1997), How We can Agree to Disagree. *Working paper.*

[Geanakoplos (1989)] Geanakoplos, J. (1989), Game Theory Without Partitions and Applications to Speculation and Consensus, *Cowles Foundation Discussion Paper 914.*

[Geanakoplos and Polemarchakis (1982)] Geanakoplos, J., and H. Polemarchakis (1982), We Can't Disagree Forever, *Journal of Economic Theory*, (28), 192–200.

[Hart and Tauman (2004)] Hart, S., and Y. Tauman (2004), Market Crashes without External Shocks, *Journal of Business*, (77), 1–8.

[Morris (1996)] Morris, S. (1996), The Logic of Belief and Belief Change: A Decision Theoretic Approach, *Journal of the Economic Theory*, (69), 1–23.

[Samet (2011)] Samet, D. (2011), Common Belief of Rationality in Games of Perfect Information, *Working Paper.*

[Tarbush (2011)] Tarbush, B. (2011), Agreeing to disagree with generalised decision functions, *University of Oxford, Department of Economics, Working Paper.*

APPENDIX

Proof of Theorem 1. We first add a definition and a lemma, for the sake of proving the theorem.

Definition 7. Let $(t_i)_{i\in I}$ be a probabilistic belief structure over Ω with corresponding partition profile $\Pi := (\Pi_i)_{i\in I}$, and let $X \subset \Omega$ be a subset of Ω. Define Π *restricted to* X, denoted Π^X, to be the partition profile over X given by

$\Pi_i^X(\omega) := \Pi_i(\omega) \cap X$ for any state ω. Further, for each $i \in I$ let t_i^X be any type function over $(X, \mathbf{\Pi}^X)$ that satisfies the property that for any $\omega \in \Omega$, $t_i(\omega)(\Pi_i^X)t_i^X(\omega) = t_i(\omega)$. ♦

Intuitively, Π_i^X is the partition of X derived from the partition Π_i of Ω by 'ignoring all states outside of X'. It then follows intuitively that $t_i^X(\omega)$, for each state $\omega \in X$, is $t_i(\omega)$ scaled relative to the other states in $\Pi_i^X(\omega)$ in such a way that $\sum_{\omega \in X} t_i^X(\omega) = 1$.

For a random variable f, denote

$$E_i^X(f \mid \Pi_i^X(\omega)) := \sum_{\omega' \in \Pi_i^X(\omega)} t_i^X(\omega') f(\omega').$$

A bet $\{f_1, \ldots, f_n\}$ is an *agreeable bet relative to* $(t_i^X)_i$ *at* $\omega \in X$ if $E_i^X(f \mid \omega) > 0$ for all $i \in I$. We will say that it is simply an agreeable relative to $(t_i^X)_i$ if it is an agreeable bet relative to $(t_i^X)_i$ at all states $\omega \in X$.

LEMMA 1. *Let $(t_i)_{i \in I}$ be a probabilistic belief structure over Ω, let $\omega \in \Omega$ and let X be a non-empty subset of $b^Q(\omega)$, the common belief set of ω. Suppose that there exists an agreeable bet relative to $(t_i^X)_i$. Then there exists an agreeable bet relative to $b^Q(\omega)$.*

Proof. Let f be an agreeable bet relative to $(t_i^X)_i$. If $X = b^Q(\omega)$, there is nothing to prove.

Otherwise, we distinguish a few cases:

1. Suppose that there exists a state $\omega'' \in X$ such that $b_i(\omega'') \setminus X \neq \emptyset$ for some $i \in I$. Let $\omega' \in b_i(\omega'') \setminus X$ (hence $t_i(\omega') > 0$), and let $\varepsilon := E_i^X(f_i \mid \Pi_i^X(\omega')) = E_i^X(f_i \mid \Pi_i^X(\omega''))$. By assumption, $\varepsilon > 0$ (since f is an agreeable bet relative to $(t_i^X)_i$). Set $Y := X \cup \omega'$.

   Next, let $\overline{f}_i(\omega')$ be a negative real number satisfying

   $$0 > \overline{f}_i(\omega') > \frac{-(1 - t_i^Y(\omega'))}{t_i^Y(\omega')}\varepsilon \ ,$$

   and for $j \neq i$, set $\overline{f}_j(\omega') := -\overline{f}_i(\omega')/(n-1) > 0$, where $n = |I|$.

   Clearly, by construction, $\sum_{j \in I} \overline{f}_j(\omega') = 0$. Complete the definition of $\overline{f}$ by letting $\overline{f}(\omega''') := f(\omega''')$ for all $\omega''' \in X$. It is straightforward to check that $\overline{f}$ is an agreeable bet relative to $(t_i^Y)_{i \in I}$.

2. Suppose that there is a state $\omega' \in b^Q(\omega) \setminus X$ such that $b_i(\omega') \cap X \neq \emptyset$. Set $Y := X \cup \omega'$.

   We distinguish two sub-cases:

   (a) If $t_i(\omega') = 0$, then for all $j \in I \setminus i$ let $\overline{f}_j(\omega')$ be any arbitrary positive number, and set $\overline{f}_i(\omega') = -\sum_{j \in I \setminus i} \overline{f}_j(\omega')$. Then $\overline{f}$ is an agreeable bet relative to $(t_i^Y)_{i \in I}$.

   (b) If $t_i(\omega') > 0$, let $\varepsilon := E_i^X(f_i \mid \Pi_i^X(\omega'))$. By assumption, $\varepsilon > 0$ (since $b_i(\omega') \cap X \neq \emptyset$ and f is an agreeable bet relative to $(t_i^X)_i$). From this point, define $\overline{f}_j$ for all $j \in I$ exactly as in Case 1 above, yielding an agreeable bet relative to $(t_i^Y)_{i \in I}$.

Now simply repeat this procedure as often as necessary to extend the agreeable bet to every state in the finite set $b^Q(\omega)$. ∎

Completion of the proof of Theorem 1. Let $(t_i)_{i \in I}$ be a probabilistic belief structure over Ω, and let ω be a state at which there is weak common belief in truth, and hence there is $\omega' \in b^Q(\omega)$ at which there is strong common belief in truth, i.e.,

$$\bigcup_{\omega'' \in b^Q(\omega')} b_i(\omega'') = \bigcup_{\omega'' \in b^Q(\omega')} b_j(\omega'')$$

for all $i, j \in I$. If we restrict attention solely to the states in $b^Q(\omega')$, we can consider the operators b_i for all i to constitute an S5 knowledge structure over $b^Q(\omega)$.

In one direction, suppose that there is a common delusional prior μ. Then μ restricted to $b^Q(\omega')$ is a common (standard) prior over $b^Q(\omega')$ regarded as a knowledge structure, hence there can be no common knowledge agreeable bet at any state in $b^Q(\omega')$. If there were a common belief agreeable bet at ω, then that bet would be a common knowledge agreeable bet over $b^Q(\omega')$ regarded as a knowledge structure, which we just showed cannot happen. The contradiction establishes that there is no common belief agreeable bet at ω.

In the other direction, suppose that there is no common delusional prior. Then there can be no common (standard) prior over $b^Q(\omega')$ regarded as a knowledge structure, because if there were such a prior μ, it could be extended to a common delusional prior $\widehat{\mu}$ over all of $b^Q(\omega)$ simply by setting

$$\widehat{\mu}(\omega'') = \begin{cases} \mu(\omega'') & \text{if } \omega'' \in b^Q(\omega') \\ 0 & \text{otherwise.} \end{cases}$$

We can then apply the standard No Betting Theorem for knowledge structures to conclude that there is a common knowledge agreeable bet $\{f_1, \ldots, f_n\}$ over $b^Q(\omega')$ as a knowledge structure, which is a common belief agreeable bet over $b^Q(\omega')$ as a belief structure. Applying Lemma 1, this can be extended to a common belief agreeable bet over all of $b^Q(\omega)$, which is what was needed to be shown. ∎

The Complexity of Online Manipulation of Sequential Elections

Edith Hemaspaandra
Dept. of Computer Science
Rochester Inst. of Technology
Rochester, NY 14623, USA
www.cs.rit.edu/∼eh

Lane A. Hemaspaandra
Dept. of Computer Science
University of Rochester
Rochester, NY 14627, USA
www.cs.rochester.edu/u/lane

Jörg Rothe
Institut für Informatik
Universität Düsseldorf
40225 Düsseldorf, Germany
rothe@cs.uni-duesseldorf.de

ABSTRACT

Most work on manipulation assumes that all preferences are known to the manipulators. However, in many settings elections are open and sequential, and manipulators may know the already cast votes but may not know the future votes. We introduce a framework, in which manipulators can see the past votes but not the future ones, to model online coalitional manipulation of sequential elections, and we show that in this setting manipulation can be extremely complex even for election systems with simple winner problems. Yet we also show that for some of the most important election systems such manipulation is simple in certain settings. This suggests that when using sequential voting, one should pay great attention to the details of the setting in choosing one's voting rule.

Among the highlights of our classifications are: We show that, depending on the size of the manipulative coalition, the online manipulation problem can be complete for each level of the polynomial hierarchy or even for PSPACE. We obtain the most dramatic contrast to date between the nonunique-winner and unique-winner models: Online weighted manipulation for plurality is in P in the nonunique-winner model, yet is coNP-hard (constructive case) and NP-hard (destructive case) in the unique-winner model. And we obtain what to the best of our knowledge are the first $P^{NP[1]}$-completeness and P^{NP}-completeness results in the field of computational social choice, in particular proving such completeness for, respectively, the complexity of 3-candidate and 4-candidate (and unlimited-candidate) online weighted coalition manipulation of veto elections.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*Multiagent systems*; F.1.2 [**Computation by Abstract Devices**]: Modes of Computation; F.2.2 [**Analysis of Algorithms and Problem Complexity**]: Nonnumerical Algorithms and Problems

General Terms

Theory

Keywords

Computational complexity, computational social choice, elections, manipulation, online algorithms, preferences, sequential voting

TARK 2013, Chennai, India.

1. INTRODUCTION

Voting is a widely used method for preference aggregation and decision-making. In particular, *strategic* voting (or *manipulation*) has been studied intensely in social choice theory (starting with the celebrated work of Gibbard [19] and Satterthwaite [29]) and, in the rapidly emerging area of *computational* social choice, also with respect to its algorithmic properties and computational complexity (starting with the seminal work of Bartholdi, Tovey, and Trick [3]; see the surveys [15, 16]). This computational aspect is particularly important in light of the many applications of voting in computer science, ranging from meta-search heuristics for the internet [14], to recommender systems [18] and multiagent systems in artificial intelligence (see the survey by Conitzer [11]).

Most of the previous work on manipulation, however, is concerned with voting where the manipulators know the nonmanipulative votes. Far less attention has been paid (see the related work below) to manipulation in the midst of elections that are modeled as dynamic processes.

We introduce a novel framework for online manipulation, where voters vote in sequence and the current manipulator, who knows the previous votes and which voters are still to come but does not know their votes, must decide—right at that moment—what the "best" vote to cast is. So, while other approaches to sequential voting are stochastic, game-theoretic (yet different from our approach, see Footnote 1), or axiomatic in nature (again, see the related work), our approach to manipulation of sequential voting is shaped by the area of "online algorithms" [8], in the technical sense of a setting in which one (for us, each manipulative voter) is being asked to make a manipulation decision just on the basis of the information one has in one's hands at the moment even though additional information/system evolution may well be happening down the line. In this area, there are different frameworks for evaluation. But the most attractive one, which pervades the area as a general theme, is the idea that one may want to "maxi-min" things—*one may want to take the action that maximizes the goodness of the set of outcomes that one can expect regardless of what happens down the line from one time-wise.* For example, if the current manipulator's preferences are Alice > Ted > Carol > Bob and if she can cast a (perhaps insincere) vote that ensures that Alice or Ted will be a winner no matter what later voters do, and there is no vote she can cast that ensures that Alice

will always be a winner, this maxi-min approach would say that that vote is a "best" vote to cast.

It will perhaps be a bit surprising to those familiar with online algorithms and competitive analysis that in our model of online manipulation we will not use a (competitive) *ratio*. The reason is that voting commonly uses an *ordinal* preference model, in which preferences are total orders of the candidates. It would be a severely improper step to jump from that to assumptions about intensity of preferences and utility, e.g., to assuming that everyone likes her nth-to-least favorite candidate exactly n times more than she likes her least favorite candidate.

Related Work.

Conitzer and Xia [37] (see also the related paper by Desmedt and Elkind [13]) define and study the Stackelberg voting game (also quite naturally called, in an earlier paper that mostly looked at two candidates, the roll-call voting game [30]). This basically is an election in which the voters vote in order, *and the preferences are common knowledge—everyone knows everyone else's preferences, everyone knows that everyone knows everyone else's preferences, and so on out to infinity.* Their analysis of this game is game-theoretically shaped; they compute a subgame perfect Nash equilibrium from the back end forward. Under their work's setting and assumptions, for bounded numbers of manipulators manipulation is in P, but we will show that in our model even with bounded numbers of manipulators manipulation sometimes (unless P = NP) falls beyond P.[1]

The interesting "dynamic voting" work of Tennenholtz [33] investigates sequential voting, but focuses on axioms and voting rules rather than on coalitions and manipulation. Much heavily Markovian work studies sequential decision-making and/or dynamically varying preferences; our work in contrast is nonprobabilistic and focused on the complexity of coalitional manipulation. Also somewhat related to, but quite different from, our work is the work on possible and necessary winners. The seminal paper on that is due to Konczak and Lang [25], and more recent work includes [36, 7, 1, 5, 6, 10, 4, 27]; the biggest difference is that those are, loosely, one-quantifier settings, but the more dynamic setting of online manipulation involves numbers of quantifiers that can grow with the input size. Another related research line studies multi-issue elections [38, 39, 40, 41]; although there the separate issues may run in sequence, each issue typically is voted on simultaneously and with preferences being common knowledge.

[1]Our work too is game-theoretically connected. Although in our model we are asking whether we can reach our goal no matter what the future nonmanipulators do, if one thinks about what the actual effect of this is, one can see that our setting is in effect well-captured by what is known as a 2-player combinatorial game (combinatorial games are a particular type of complete-information sequential game). In our setting, the goal of one player in this game will be to ensure that the winner set (which of course heavily depends on what moves have occurred already and on the election system) will have nonempty intersection with a certain subset of the candidates, and the goal of the other player will be to ensure that that does not happen. Of course, the former player is in effect the currently-under-consideration and still-to-vote members of the manipulative coalition, and the latter player is capturing the same except regarding nonmanipulators. So, the key differences between [37] and our work regard goals and coalitionality. For them, each player (and they may have many players) is in effect a completely separate agent, with a preference order, and is trying to see if a change as an individual will make a more preferred candidate win. For us, the manipulative voters function as a coalition, and one that has an all-or-nothing goal, and there are no gradations within that goal in terms of our analysis (despite the fact that we use a preference order when speaking of the coalition), and we are in effect a two-player combinatorial game.

2. PRELIMINARIES

Elections.

A *(standard, i.e., simultaneous) election* (C, V) is specified by a set C of candidates and a list V, where we assume that each element in V is a pair (v, p) such that v is a voter name and p is v's vote. How the votes in V are represented depends on the election system used—we assume, as is required by most systems, votes to be total preference orders over C. For example, if $C = \{a, b, c\}$, a vote of the form $c > a > b$ means that this voter (strictly) prefers c to a and a to b.

We introduce election snapshots to capture sequential election scenarios as follows. Let C be a set of candidates and let u be (the name of) a voter. An *election snapshot for C and u* is specified by a triple $V = (V_{<u}, u, V_{u<})$ consisting of all voters in the order they vote, along with, for each voter before u (i.e., those in $V_{<u}$), the vote she cast, and for each voter after u (i.e., those in $V_{u<}$), a bit specifying if she is part of the manipulative coalition (to which u always belongs). That is, $V_{<u} = ((v_1, p_1), (v_2, p_2), \ldots, (v_{i-1}, p_{i-1}))$, where the voters named $v_1, v_2, \ldots, v_{i-1}$ (including perhaps manipulators and nonmanipulators) have already cast their votes (preference order p_j being cast by v_j), and $V_{u<} = ((v_{i+1}, x_{i+1}), (v_{i+2}, x_{i+2}), \ldots, (v_n, x_n))$ lists the names of the voters still to cast their votes, in that order, and where $x_j = 1$ if v_j belongs to the manipulative coalition and $x_j = 0$ otherwise.

Scoring Rules.

A *scoring rule* for m candidates is given by a scoring vector $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_m)$ of nonnegative integers such that $\alpha_1 \geq \alpha_2 \geq \cdots \geq \alpha_m$. For an election (C, V), each candidate $c \in C$ scores α_i points for each vote that ranks c in the ith position. Let $score(c)$ be the total score of $c \in C$. All candidates scoring the most points are winners of (C, V). Some of the most popular voting systems are *k-approval* (especially *plurality*, aka 1-approval) and *k-veto* (especially *veto*, aka 1-veto). Their m-candidate, $m \geq k$, versions are defined by the scoring vectors $(\underbrace{1, \ldots, 1}_{k}, \underbrace{0, \ldots, 0}_{m-k})$ and $(\underbrace{1, \ldots, 1}_{m-k}, \underbrace{0, \ldots, 0}_{k})$. When m is not fixed, we omit the phrase "m-candidate."

Manipulation.

The *(standard) weighted coalitional manipulation problem* [12], $\mathcal{E}$-Weighted-Coalitional-Manipulation (abbreviated by $\mathcal{E}$-WCM), for any election system $\mathcal{E}$ is defined as follows: Given a candidate set C, a list S of nonmanipulative voters each having a nonnegative integer weight, a list T of the nonnegative integer weights of the manipulative voters (whose preferences over C are unspecified), with $S \cap T = \emptyset$, and a distinguished candidate $c \in C$, can the manipulative votes T be set such that c is a (or the) $\mathcal{E}$ winner of $(C, S \cup T)$?

Asking whether c can be made "a winner" is called the

nonunique-winner model and is the model of all notions in this paper unless mentioned otherwise. If one asks whether c can be made a "one and only winner," that is called the unique-winner model. We also use the *unweighted* variant, where each vote has unit weight, and write $\mathcal{E}$-UCM as a shorthand. Note that $\mathcal{E}$-UCM with a *single* manipulator (i.e., $\|T\| = 1$ in the problem instance) is the manipulation problem originally studied in [3, 2]. Conitzer, Sandholm, and Lang [12] also introduced the *destructive* variants of these manipulation problems, where the goal is not to make c win but to ensure that c is not a winner, and we denote the corresponding problems by $\mathcal{E}$-DWCM and $\mathcal{E}$-DUCM. Finally, we write $\mathcal{E}$-WC$_{\neq\emptyset}$M, $\mathcal{E}$-UC$_{\neq\emptyset}$M, $\mathcal{E}$-DWC$_{\neq\emptyset}$M, and $\mathcal{E}$-DUC$_{\neq\emptyset}$M to indicate that the problem instances are required to have a nonempty coalition of manipulators.

Complexity-Theoretic Background.

We assume the reader is familiar with basic complexity-theoretic notions such as the complexity classes P and NP, the class FP of polynomial-time computable functions, polynomial-time many-one reducibility ($\leq_m^p$), and hardness and completeness with respect to $\leq_m^p$ for a complexity class.

Meyer and Stockmeyer [28] and Stockmeyer [31] introduced and studied the polynomial hierarchy, PH $= \bigcup_{k\geq 0} \Sigma_k^p$, whose levels are inductively defined by $\Sigma_0^p = \mathrm{P}$ and $\Sigma_{k+1}^p = \mathrm{NP}^{\Sigma_k^p}$, and their co-classes, $\Pi_k^p = \mathrm{co}\Sigma_k^p$ for $k \geq 0$. They also characterized these levels by polynomially length-bounded alternating existential and universal quantifiers. P^{NP} is the class of problems solvable in deterministic polynomial time with access to an NP oracle. $\mathrm{P}^{\mathrm{NP}[1]}$ is the restriction of P^{NP} where only one oracle query is allowed. $\mathrm{P} \subseteq \mathrm{NP} \cap \mathrm{coNP} \subseteq \mathrm{NP} \cup \mathrm{coNP} \subseteq \mathrm{P}^{\mathrm{NP}[1]} \subseteq \mathrm{P}^{\mathrm{NP}} \subseteq \Sigma_2^p \cap \Pi_2^p \subseteq \Sigma_2^p \cup \Pi_2^p \subseteq \mathrm{PH} \subseteq \mathrm{PSPACE}$, where PSPACE is the class of problems solvable in polynomial space. The *quantified boolean formula problem*, QBF, is a standard PSPACE-complete problem. QBF_k ($\widetilde{\mathrm{QBF}}_k$) denotes the restriction of QBF with at most k quantifiers that start with $\exists$ ($\forall$) and then alternate between $\exists$ and $\forall$, and we assume that each $\exists$ and $\forall$ quantifies over a set of boolean variables. For each $k \geq 1$, QBF_k is Σ_k^p-complete and $\widetilde{\mathrm{QBF}}_k$ is Π_k^p-complete [32, 35].

3. OUR MODEL OF ONLINE MANIPULATION

The core of our model of online manipulation in sequential voting is what we call the *magnifying-glass moment*, namely, the moment at which a manipulator u is the one who is going to vote, is aware of what has happened so far in the election (and which voters are still to come, but in general not knowing what they want, except in the case of voters, if any, who are coalitionally linked to u). In this moment, u seeks to "figure out" what the "best" vote to cast is. We will call the information available in such a moment an *online manipulation setting* (*OMS*, for short) and define it formally as a tuple (C, u, V, σ, d), where C is a set of candidates; u is a distinguished voter; $V = (V_{<u}, u, V_{u<})$ is an election snapshot for C and u; σ is the preference order of the manipulative coalition to which u belongs; and $d \in C$ is a distinguished candidate. Given an election system $\mathcal{E}$, define the problem online-$\mathcal{E}$-Unweighted-Coalitional-Manipulation (abbreviated by online-$\mathcal{E}$-UCM), as follows: Given an OMS (C, u, V, σ, d) as described above, does there exist some vote that u can cast (assuming support from the manipulators coming after u) such that no matter what votes are cast by the nonmanipulators coming after u, there exists some $c \in C$ such that $c \geq_\sigma d$ and c is an $\mathcal{E}$ winner of the election? By "support from the manipulators coming after u" we mean that u's coalition partners coming after u, when they get to vote, will use their then-in-hand knowledge of all votes up to then to help u reach her goal: By a joint effort u's coalition can ensure that the $\mathcal{E}$ winner set will always include a candidate liked by the coalition as much as or more than d, even when the nonmanipulators take their strongest action so as to prevent this. Note that this candidate, c in the problem description, may be different based on the nonmanipulators' actions. (Nonsequential manipulation problems usually focus on whether a single candidate can be made to win, but in our setting, this "that person or better" focus is more natural.) For the case of weighted manipulation, each voter also comes with a nonnegative integer weight. We denote this problem by online-$\mathcal{E}$-WCM.

We write online-$\mathcal{E}$-UCM[k] in the unweighted case and online-$\mathcal{E}$-WCM[k] in the weighted case to denote the problem when the number of manipulators from u onward is restricted to be at most k.

Denote the corresponding destructive problems by online-$\mathcal{E}$-DUCM, online-$\mathcal{E}$-DWCM, online-$\mathcal{E}$-DUCM[k], and online-$\mathcal{E}$-DWCM[k]. In online-$\mathcal{E}$-DUCM we ask whether the given current manipulator u (assuming support from the manipulators after her) can cast a vote such that no matter what votes are cast by the nonmanipulators after u, no $c \in C$ with $d \geq_\sigma c$ is an $\mathcal{E}$ winner of the election, i.e., u's coalition can ensure that the $\mathcal{E}$ winner set never includes d or any even more hated candidate. The other three problems are defined analogously.

Note that online-$\mathcal{E}$-UCM generalizes the original unweighted manipulation problem with a single manipulator as introduced by Bartholdi, Tovey, and Trick [3]. Indeed, their manipulation problem in effect is the special case of online-$\mathcal{E}$-UCM when restricted to instances where there is just one manipulator, she is the last voter to cast a vote, and d is the coalition's most preferred candidate. Similarly, online-$\mathcal{E}$-WCM generalizes the (standard) coalitional weighted manipulation problem (for nonempty coalitions of manipulators). Indeed, that traditional manipulation problem is the special case of online-$\mathcal{E}$-WCM, restricted to instances where only manipulators come after u and d is the coalition's most preferred candidate. If we take an analogous approach except with d restricted now to being the most hated candidate of the coalition, we generalize the corresponding notions for the destructive cases. We summarize these observations as follows.

Proposition 1. *For each election system $\mathcal{E}$, it holds that (1) $\mathcal{E}$-UC$_{\neq\emptyset}$M $\leq_m^p$ online-$\mathcal{E}$-UCM, (2) $\mathcal{E}$-WC$_{\neq\emptyset}$M $\leq_m^p$ online-$\mathcal{E}$-WCM, (3) $\mathcal{E}$-DUC$_{\neq\emptyset}$M $\leq_m^p$ online-$\mathcal{E}$-DUCM, and (4) $\mathcal{E}$-DWC$_{\neq\emptyset}$M $\leq_m^p$ online-$\mathcal{E}$-DWCM.*

Corollary 2 below follows immediately.

Corollary 2. *(1) For each election system $\mathcal{E}$ such that the (unweighted) winner problem is solvable in polynomial time, it holds that $\mathcal{E}$-UCM $\leq_m^p$ online-$\mathcal{E}$-UCM. (2) For each election system $\mathcal{E}$ such that the weighted winner problem is solvable in polynomial time, it holds that $\mathcal{E}$-WCM $\leq_m^p$ online-$\mathcal{E}$-WCM. (3) For each election system $\mathcal{E}$ such that*

the winner problem is solvable in polynomial time, it holds that $\mathcal{E}$-DUCM $\leq_m^p$ online-$\mathcal{E}$-DUCM. *(4) For each election system* $\mathcal{E}$ *such that the weighted winner problem is solvable in polynomial time, it holds that* $\mathcal{E}$-DWCM $\leq_m^p$ online-$\mathcal{E}$-DWCM.

We said above that, by default, we will use the *nonunique-winner model* and all the above problems are defined in this model. However, we will also have some results in the *unique-winner model*, which will, here, sharply contrast with the corresponding results in the nonunique-winner model. To indicate that a problem, such as online-$\mathcal{E}$-UCM, is in the unique-winner model, we write online-$\mathcal{E}$-UCM_{UW} and ask whether the current manipulator u (assuming support from the manipulators coming after her) can ensure that there exists some $c \in C$ such that $c \geq_\sigma d$ and c is *the unique* $\mathcal{E}$ *winner* of the election.

4. GENERAL RESULTS

THEOREM 3. *(1) For each election system* $\mathcal{E}$ *whose weighted winner problem can be solved in polynomial time,*[2] *the problem* online-$\mathcal{E}$-WCM *is in* PSPACE. *(2) For each election system* $\mathcal{E}$ *whose winner problem can be solved in polynomial time, the problem* online-$\mathcal{E}$-UCM *is in* PSPACE. *(3) There exists an election system* $\mathcal{E}$ *with a polynomial-time winner problem such that the problem* online-$\mathcal{E}$-UCM *is* PSPACE*-complete. (4) There exists an election system* $\mathcal{E}$ *with a polynomial-time weighted winner problem such that the problem* online-$\mathcal{E}$-WCM *is* PSPACE*-complete.*

The proof of Theorem 3 is deferred to the appendix. The following theorem shows that for bounded numbers of manipulators the complexity crawls up the polynomial hierarchy. The theorem's proof is based on the proof given above, except we need to use the alternating quantifier characterization due to Meyer and Stockmeyer [28] and Stockmeyer [31] for the upper bound and to reduce from the Σ_{2k}^p-complete problem QBF_{2k} rather than from QBF for the lower bound.

THEOREM 4. *Fix any* $k \geq 1$. *(1) For each election system* $\mathcal{E}$ *whose weighted winner problem can be solved in polynomial time, the problem* online-$\mathcal{E}$-WCM[k] *is in* Σ_{2k}^p. *(2) For each election system* $\mathcal{E}$ *whose winner problem can be solved in polynomial time, the problem* online-$\mathcal{E}$-UCM[k] *is in* Σ_{2k}^p. *(3) There exists an election system* $\mathcal{E}$ *with a polynomial-time winner problem such that the problem* online-$\mathcal{E}$-UCM[k] *is* Σ_{2k}^p*-complete. (4) There exists an election system* $\mathcal{E}$ *with a polynomial-time weighted winner problem such that the problem* online-$\mathcal{E}$-WCM[k] *is* Σ_{2k}^p*-complete.*

Note that the (constructive) online manipulation problems considered in Theorems 3 and 4 are about ensuring that the winner set always contains some candidate in the σ segment stretching from d up to the top-choice. Now consider "pinpoint" variants of these problems, where we ask whether the distinguished candidate d herself can be guaranteed to be a winner (for nonsequential manipulation, that version indeed is the one commonly studied). Denote the *pinpoint* variant of, e.g., online-$\mathcal{E}$-UCM[k] by pinpoint-online-$\mathcal{E}$-UCM[k]. Since our hardness proofs in Theorems 3 and 4 make all or no one a winner (and as the upper bounds in these theorems also can be seen to hold for the pinpoint variants), they establish the corresponding completeness results also for the pinpoint cases. We thus have completeness results for PSPACE and Σ_{2k}^p for each $k \geq 1$. What about the classes Σ_{2k-1}^p and Π_k^p, for each $k \geq 1$? We can get completeness results for all these classes by defining appropriate variants of online manipulation problems. Let OMP be any of the online manipulation problems considered earlier, including the pinpoint variants mentioned above. Define freeform-OMP to be just as OMP, except we no longer require the distinguished voter u to be part of the manipulative coalition—u can be in or can be out, and the input must specify, for u and all voters after u, which ones are the members of the coalition. The question of freeform-OMP is whether it is true that for all actions of the nonmanipulators at or after u (for specificity as to this problem: if u is a nonmanipulator, it will in the input come with a preference order) there will be actions (each taken with full information on cast-before-them votes) of the manipulative coalition members such that their goal of making some candidate c with $c \geq_\sigma d$ (or exactly d, in the pinpoint versions) a winner is achieved. Then, whenever Theorem 4 establishes a Σ_{2k}^p or Σ_{2k}^p-completeness result for OMP, we obtain a Π_{2k+1}^p or Π_{2k+1}^p-completeness result for freeform-OMP and for $k = 0$ manipulators we obtain Π_1^p = coNP or coNP-completeness results. Similarly, the PSPACE and PSPACE-completeness results for OMP we established in Theorem 3 also can be shown true for freeform-OMP.

On the other hand, if we define a variant of OMP by requiring the final voter to always be a manipulator, the PSPACE and PSPACE-completeness results for OMP from Theorem 3 remain true for this variant; the Σ_{2k}^p and Σ_{2k}^p-completeness results for OMP from Theorem 4 change to Σ_{2k-1}^p and Σ_{2k-1}^p-completeness results for this variant; and the above Π_{2k+1}^p and Π_{2k+1}^p-completeness results for freeform-OMP change to Π_{2k}^p and Π_{2k}^p-completeness results for this variant, $k \geq 1$.

Finally, as an open direction (and related conjecture), we define for each of the previously considered variants of online manipulation problems a *full profile* version. For example, fullprofile-online-$\mathcal{E}$-UCM[k] (for a given election system $\mathcal{E}$) is the function problem that, given an OMS *without* any distinguished candidate, (C, u, V, σ), returns a length $\|C\|$ bit-vector that for each candidate $d \in C$ says if the answer to "$(C, u, V, \sigma, d) \in$ online-$\mathcal{E}$-UCM[k]?" is "yes" (1) or "no" (0). The function problem fullprofile-pinpoint-online-$\mathcal{E}$-UCM[k] is defined analogously, except regarding pinpoint-online-$\mathcal{E}$-UCM[k].

It is not hard to prove, as a corollary to Theorem 4, that:

THEOREM 5. *For each election system* $\mathcal{E}$ *whose winner problem can be solved in polynomial time, (1) the problem* fullprofile-online-$\mathcal{E}$-UCM[k] *is in* $\text{FP}^{\Sigma_{2k}^p[\mathcal{O}(\log n)]}$, *the class of functions computable in polynomial time given Turing access to a* Σ_{2k}^p *oracle with* $\mathcal{O}(\log n)$ *queries allowed on size* n *inputs; (2)* fullprofile-pinpoint-online-$\mathcal{E}$-UCM[k] *is in* $\text{FP}_{\text{tt}}^{\Sigma_{2k}^p}$, *the class of functions computable in polynomial time given truth-table access to a* Σ_{2k}^p *oracle.*

[2]We mention in passing here, and henceforward we will not explicitly mention it in the analogous cases, that the claim clearly remains true even when "polynomial time" is replaced by the larger class "polynomial space."

We conjecture that both problems are complete for the corresponding class under metric reductions [26], for suitably defined election systems with polynomial-time winner problems.

If the full profile version of an online manipulation problem can be computed efficiently, we clearly can also easily solve each of the decision problems involved by looking at the corresponding bit of the length $\|C\|$ bit-vector. Conversely, if there is an efficient algorithm for an online manipulation decision problem, we can easily solve its full profile version by running this algorithm for each candidate in turn. Thus, we will state our later results only for online manipulation decision problem.

PROPOSITION 6. *Let* OMP *be any of the online manipulation decision problems defined above. Then* fullprofile-OMP *is in* FP *if and only if* OMP *is in* P.

5. RESULTS FOR SPECIFIC NATURAL VOTING SYSTEMS

The results of the previous section show that, simply put, even for election systems with polynomial-time winner problems, online manipulation can be tremendously difficult. But what about *natural* election systems? We will now take a closer look at important natural systems. We will show that online manipulation can be easy for them, depending on which particular problem is considered, and we will also see that the constructive and destructive cases can differ sharply from each other and that it really matters whether we are in the nonunique-winner model or the unique-winner model. Finally, in studying the complexity of online manipulation of veto elections, we obtain (as Theorems 11 and 12) what to the best of our knowledge are the first $\mathrm{P}^{\mathrm{NP}[1]}$-completeness and P^{NP}-completeness results in the field of computational social choice.

THEOREM 7. *(1)* online-plurality-WCM *(and thus also* online-plurality-UCM*) is in* P. *(2)* online-plurality-DWCM *(and thus also* online-plurality-DUCM*) is in* P.

Theorem 7 refers to problems in the nonunique-winner model. By contrast, we now show that online manipulation for weighted plurality voting in the *unique-winner* model is coNP-hard in the *constructive* case and is NP-hard in the *destructive* case. This is perhaps the most dramatic, broad contrast yet between the nonunique-winner model and the unique-winner model, and is the first such contrast involving plurality. The key other NP-hardness versus P result for the nonunique-winner model versus the unique-winner model is due to Faliszewski, Hemaspaandra, and Schnoor [17], but holds only for (standard) weighted manipulation for Copeland$^{\alpha}$ elections ($0 < \alpha < 1$) with exactly three candidates; for fewer than three both cases there are in P and for more than three both are NP-complete. In contrast, the P results of Theorem 7 hold for all numbers of candidates, and the NP-hardness and coNP-hardness results of Theorem 8 hold whenever there are at least two candidates.

THEOREM 8. *(1)* online-plurality-$\mathrm{DWCM}_{\mathrm{UW}}$ *is* NP*-hard, even when restricted to only two candidates (and this also holds when restricted to three, four, ... candidates). (2)* online-plurality-$\mathrm{WCM}_{\mathrm{UW}}$ *is* coNP*-hard, even when restricted to only two candidates (and this also holds when restricted to three, four, ... candidates).*

PROOF. For the first statement, we prove NP-hardness of online-plurality-$\mathrm{DWCM}_{\mathrm{UW}}$ by a reduction from the NP-complete problem Partition: Given a nonempty sequence $(w_1, w_2, \ldots, w_z)$ of positive integers such that $\sum_{i=1}^{z} w_i = 2W$ for some positive integer W, does there exist a set $I \subseteq \{1, 2, \ldots, z\}$ such that $\sum_{i \in I} w_i = W$? Let $m \geq 2$. Given an instance $(w_1, w_2, \ldots, w_z)$ of Partition, construct an instance $(\{c_1, \ldots, c_m\}, u_1, V, c_1 > c_2 > \cdots > c_m, c_1)$ of online-plurality-$\mathrm{DWCM}_{\mathrm{UW}}$ such that V contains $m + z - 2$ voters $v_1, \ldots, v_{m-2}, u_1, \ldots, u_z$ who vote in that order. For $1 \leq i \leq m-2$, v_i votes for c_i and has weight $(m-1)W - i$, and for $1 \leq i \leq z$, u_i is a manipulator of weight $(m-1)w_i$. If $(w_1, w_2, \ldots, w_z)$ is a yes-instance of Partition, the manipulators can give $(m-1)W$ points to both c_{m-1} and c_m, and zero points to the other candidates. So c_{m-1} and c_m are tied for the most points and there is no unique winner. On the other hand, the only way to avoid having a unique winner in our online-plurality-$\mathrm{DWCM}_{\mathrm{UW}}$ instance is if there is a tie for the most points. The only candidates that can tie are c_{m-1} and c_m, since all other pairs of candidates have different scores modulo $m-1$. It is easy to see that c_{m-1} and c_m tie for the most points only if they both get exactly $(m-1)W$ points. It follows that $(w_1, w_2, \ldots, w_z)$ is a yes-instance of Partition.

For the second part, we adapt the above construction to yield a reduction from Partition to the complement of online-plurality-$\mathrm{WCM}_{\mathrm{UW}}$. Given an instance $(w_1, w_2, \ldots, w_z)$ of Partition, construct an instance $(\{c_1, \ldots, c_m\}, \widehat{u}, V, c_1 > c_2 > \cdots > c_m, c_m)$ of online-plurality-$\mathrm{WCM}_{\mathrm{UW}}$ such that V contains $m + z - 1$ voters $v_1, \ldots, v_{m-2}, \widehat{u}, u_1, \ldots, u_z$ who vote in that order. For $1 \leq i \leq m-2$, v_i has the same vote and the same weight as above, $\widehat{u}$ is a manipulator of weight 0, and for $1 \leq i \leq z$, u_i has the same weight as above, but in contrast to the case above, u_i is now a nonmanipulator. By the same argument as above, it follows that $(w_1, w_2, \ldots, w_z)$ is a yes-instance of Partition if and only if the nonmanipulators can ensure that there is no unique winner, which in turn is true if and only if the manipulator can not ensure that there is a unique winner. ❑

THEOREM 9. *For each scoring rule* $\alpha = (\alpha_1, \ldots, \alpha_m)$, online-$\alpha$-WCM *is in* P *if* $\alpha_2 = \alpha_m$ *and is* NP*-hard otherwise.*

THEOREM 10. *For each* k, online-k-approval-UCM *and* online-k-veto-UCM *are in* P.

PROOF. Consider 1-veto. Given an online-1-veto-UCM instance (C, u, V, σ, d), the best strategy for the manipulators from u onward (let n_1 denote how many of these there are) is to minimize $\max_{c <_\sigma d} score(c)$. Let n_0 denote how many nonmanipulators come after u. We claim that (C, u, V, σ, d) is a yes-instance if and only if d is ranked last in σ or there exists a threshold t such that (1) $\sum_{c <_\sigma d} (maxscore(c) \ominus t) \leq n_1$ (so those manipulators can ensure that all candidates ranked $<_\sigma d$ score at most t points), where "$\ominus$" denotes proper subtraction ($x \ominus y = \max(x - y, 0)$) and $maxscore(c)$ is c's score when none of the voters from u onward veto c, and

(2) $\sum_{c \geq_\sigma d}(maxscore(c) \ominus (t-1)) > n_0$ (so those nonmanipulators cannot prevent that some candidate ranked $\geq_\sigma d$ scores at least t points).

For 1-veto under the above approach, in each situation where the remaining manipulators can force success against all actions of the remaining nonmanipulators, u (right then as she moves) can set her *and all future manipulators' actions* so as to force success regardless of the actions of the remaining nonmanipulators. For k-approval and k-veto, $k \geq 2$, that approach provably cannot work (as will be explained right after this proof); rather, we sometimes need later manipulators' actions to be shaped by intervening nonmanipulators' actions. Still, the following P-time algorithm, which works for all k, tells whether success can be forced. As a thought experiment, for each voter v from u onwards in sequence do this: Order the candidates in $\{c \mid c \geq_\sigma d\}$ from most to least current approvals, breaking ties arbitrarily, and postpend the remaining candidates ordered from least to most current approvals. Let ℓ be k for k-approval and $\|C\| - k$ for k-veto. Cast the voter's ℓ approvals for the first ℓ candidates in this order if v is a manipulator, and otherwise for the last ℓ candidates in this order. Success can be forced against perfect play if and only if this P-time process leads to success. ❑

In the above proof we said that the approach for 1-veto (in which the current manipulator can set her and all future manipulators' actions so as to force success independent of the actions of intervening future nonmanipulators) provably cannot work for k-approval and k-veto, $k \geq 2$. Why not? Consider an OMS (C, u, V, σ, d) with candidate set $C = \{c_1, c_2, \ldots, c_{2k}\}$, σ being given by $c_1 >_\sigma c_2 >_\sigma \cdots >_\sigma c_{2k}$, and $d = c_1$. So, u's coalition wants to enforce that c_1 is a winner. Suppose that v_1 has already cast her vote, now it's $v_2 = u$'s turn, and the order of the future voters is $v_3, v_4, \ldots, v_{2j}$, where all v_{2i}, $2 \leq i \leq j$, belong to u's coalition, and all v_{2i-1} do not. Suppose that v_1 was approving of the k candidates in $C_1 \subseteq \{c_2, c_3, \ldots, c_{2k}\}$, $\|C_1\| = k$. Then u must approve of the k candidates in $\overline{C_1}$, to ensure that c_1 draws level with the candidates in C_1 and none of these candidates can gain another point. Next, suppose that nonmanipulator v_3 approves of the k candidates in $C_3 \subseteq \{c_2, c_3, \ldots, c_{2k}\}$, $\|C_3\| = k$. Then v_4, the next manipulator, must approve of all candidates in $\overline{C_3}$, to ensure that c_1 draws level with the candidates in C_3 and none of these candidates can gain another point. This process is repeated until the last nonmanipulator, v_{2j-1}, approves of the candidates in $C_{2j-1} \subseteq \{c_2, c_3, \ldots, c_{2k}\}$, $\|C_{2j-1}\| = k$, and v_{2j}, the final manipulator, is forced to counter this by approving of all candidates in $\overline{C_{2j-1}}$, to ensure that c_1 is a winner. This shows that there can be arbitrarily long chains such that the action of each manipulator after u depends on the action of the preceding intervening nonmanipulator.

We now turn to online weighted manipulation for veto when restricted to three candidates. We denote this restriction of online-veto-WCM by online-veto$_{|3}$-WCM.

THEOREM 11. online-veto$_{|3}$-WCM *is* $\mathrm{P}^{\mathrm{NP}[1]}$*-complete.*

Moving from three to four candidates increases the complexity, namely to P^{NP}-completeness, and that same bound holds for unlimitedly many candidates. Although this is a strict increase in complexity from $\mathrm{P}^{\mathrm{NP}[1]}$-completeness (unless the polynomial hierarchy collapses [24]), membership in P^{NP} still places this problem far below the general PSPACE bound from earlier in this paper. The proof of Theorem 12 is deferred to the appendix. Immediately from Theorems 10 and 12, we have that the full profile variants of online-k-veto-UCM and online-k-approval-UCM are in FP and that fullprofile-online-veto-WCM is in $\mathrm{FP}^{\mathrm{NP}}$.

THEOREM 12. online-veto-WCM *is* P^{NP}*-complete, even when restricted to only four candidates.*

6. UNCERTAINTY ABOUT THE ORDER OF FUTURE VOTERS

So far, we have been dealing with cases where the order of future voters was fixed and known. But what happens if the order of future voters itself is unknown? Even here, we can make claims. To model this most naturally, our "magnifying-glass moment" will focus not on one manipulator u, but will focus at a moment in time when some voters are still to come (as before, we know who they are and which are manipulators; as before, we have a preference order σ, and know what votes have been cast so far, and have a distinguished candidate d). And the question our problem is asking is: Is it the case that our manipulative coalition can ensure that the winner set will always include d or someone liked more than d with respect to σ (i.e., the winner set will have nonempty intersection with $\{c \in C \mid c \geq_\sigma d\}$), *regardless of what order the remaining voters vote in.* We will call this problem the *schedule-robust online manipulation problem*, and will denote it by SR-online-$\mathcal{E}$-UCM. (We will add a "[1,1]" suffix for the restriction of this problem to instances when at most one manipulator and at most one nonmanipulator have not yet voted.) One might think that this problem captures both a Σ_2^p and a Π_2^p issue, and so would be hard for both classes. However, the requirement of schedule robustness tames the problem (basically what underpins that is simply that exists-forall-predicate implies forall-exists-predicate), bringing it into Σ_2^p. Further, we can prove, by explicit construction of such a system, that for some simple election systems this problem is complete for Σ_2^p.

THEOREM 13. *(1) For each election system $\mathcal{E}$ whose winner problem is in* P*,* SR-online-$\mathcal{E}$-UCM *is in* Σ_2^p*. (2) There exists an election system $\mathcal{E}$, whose winner problem is in* P*, such that the problem* SR-online-$\mathcal{E}$-UCM *(indeed, even* SR-online-$\mathcal{E}$-UCM$[1,1]$*) is* Σ_2^p*-complete.*

7. CONCLUSIONS AND OPEN QUESTIONS

We introduced a novel framework for online manipulation in sequential voting, and showed that manipulation there can be tremendously complex even for systems with simple winner problems. We also showed that among the most important election systems, some have efficient online manipulation algorithms but others (unless P = NP) do not. It will be important to, complementing our work, conduct typical-case complexity studies (although we mention in passing that unless the polynomial hierarchy collapses, no heuristic algorithm for any NP-hard problem can have a subexponential error rate, see the discussion in the survey [23]). We have extended the scope of our investigation by studying online control [22, 21] and will also study online bribery.

8. ACKNOWLEDGMENTS

We thank the anonymous reviewers for helpful comments. This work was supported in part by NSF grants CCF-{0915792, 1101452, 1101479}, ARC grant DP110101792, DFG grant RO-1202/15-1, SFF grant "Cooperative Norm-setting" of HHU Düsseldorf, Friedrich Wilhelm Bessel Research Awards to Edith Hemaspaandra and Lane A. Hemaspaandra from the Alexander von Humboldt Foundation, and a DAAD grant for a PPP project in the PROCOPE program.

APPENDIX. DEFERRED PROOFS

We provide here deferred proofs of two of our results that were not proven in the paper's body. Most other proofs not in the body can be found in the technical report version [20].

Proof of Theorem 3. The proof of the first statement (which is analogous to the proof of the first statement in Theorem 4) follows from the easy fact that online-$\mathcal{E}$-WCM can be solved by an alternating Turing machine in polynomial time, and thus, due to the characterization of Chandra, Kozen, and Stockmeyer [9], by a deterministic Turing machine in polynomial space. The proof of the second case is analogous.

We construct an election system $\mathcal{E}$ establishing the third statement. Let (C, u, V, σ, d) be a given input. $\mathcal{E}$ will look at the lexicographically least candidate name in C. Let c represent that name string in some fixed, natural encoding. $\mathcal{E}$ will check if c represents a *tiered* boolean formula, by which we mean one whose variable names are all of the form $x_{i,j}$ (which really means a direct encoding of a string, such as "$x_{4,9}$"); the i, j fields must all be positive integers. If c does not represent such a tiered formula, everyone loses on that input. Otherwise (i.e., if c represents a tiered formula), let *width* be the maximum j occurring as the second subscript in any variable name ($x_{i,j}$) in c, and let *blocks* be the maximum i occurring as the first subscript in any variable name in c. If there are fewer than *blocks* voters in V, everyone loses. Otherwise, if there are fewer than $1 + 2 \cdot \mathit{width}$ candidates in C, everyone loses (this is so that each vote will involve enough candidates that it can be used to set all the variables in one block). Otherwise, if there exists some i, $1 \leq i \leq \mathit{blocks}$, such that for no j does the variable $x_{i,j}$ occur in c, then everyone loses. Otherwise, order the voters from the lexicographically least to the lexicographically greatest voter name. If distinct voters are allowed to have the same name string (e.g., John Smith), we break ties by sorting according to the associated preference orders within each group of tied voters (second-order ties are no problem, as those votes are identical, so any order will have the same effect). Now, the first voter in this order will assign truth values to all variables $x_{1,*}$, the second voter in this order will assign truth values to all variables $x_{2,*}$, and so on up to the *blocks*th voter, who will assign truth values to all variables $x_{\mathit{blocks},*}$.

How do we get those assignments from these votes? Consider a vote whose total order over C is σ' (and recall that $\|C\| \geq 1 + 2 \cdot \mathit{width}$). Remove c from σ', yielding σ''. Let $c_1 <_{\sigma''} c_2 <_{\sigma''} \cdots <_{\sigma''} c_{2 \cdot \mathit{width}}$ be the $2 \cdot \mathit{width}$ least preferred candidates in σ''. We build a vector in $\{0,1\}^{\mathit{width}}$ as follows: The ℓth bit of the vector is 0 if the string that names $c_{1+2(\ell-1)}$ is lexicographically less than the string that names $c_{2\ell}$, and this bit is 1 otherwise.

Let b_i denote the vector thus built from the ith vote (in the above ordering), $1 \leq i \leq \mathit{blocks}$. Now, for each variable $x_{i,j}$ occurring in c, assign to it the value of the jth bit of b_i, where 0 represents *false* and 1 represents *true*. We have now assigned all variables of c, so c evaluates to either *true* or *false*. If c evaluates to *true*, everyone wins, otherwise everyone loses. This completes the specification of the election system $\mathcal{E}$. $\mathcal{E}$ has a polynomial-time winner problem, as any boolean formula, given an assignment to all its variables, can easily be evaluated in polynomial time.

To show PSPACE-hardness, we $\leq_m^p$-reduce the PSPACE-complete problem QBF to the problem online-$\mathcal{E}$-UCM. Let y be an instance of QBF. We transform y into an instance of the form $(\exists x_{1,1}, x_{1,2}, \ldots, x_{1,k_1})\,(\forall x_{2,1}, x_{2,2}, \ldots, x_{2,k_2}) \cdots (Q_\ell\, x_{\ell,1}, x_{\ell,2}, \ldots, x_{\ell,k_\ell})\,[\Phi(x_{1,1}, x_{1,2}, \ldots, x_{1,k_1}, x_{2,1}, x_{2,2}, \ldots, x_{2,k_2}, \ldots, x_{\ell,1}, x_{\ell,2}, \ldots, x_{\ell,k_\ell})]$ in polynomial time, where $Q_\ell = \exists$ if ℓ is odd and $Q_\ell = \forall$ if ℓ is even, the $x_{i,j}$ are boolean variables, Φ is a boolean formula, and for each i, $1 \leq i \leq \ell$, Φ contains at least one variable of the form $x_{i,*}$. This quantified boolean formula is $\leq_m^p$-reduced to an instance (C, u, V, σ, c) of online-$\mathcal{E}$-UCM as follows:

1. C contains a candidate whose name, c, encodes Φ, and in addition C contains $2 \cdot \max(k_1, \ldots, k_\ell)$ other candidates, all with names lexicographically greater than c—for specificity, let us say their names are the $2 \cdot \max(k_1, \ldots, k_\ell)$ strings that immediately follow c in lexicographic order.

2. V contains ℓ voters, $1, 2, \ldots, \ell$, who vote in that order, where $u = 1$ is the distinguished voter and all odd voters belong to u's manipulative coalition and all even voters do not. The voter names will be lexicographically ordered by their number, 1 is least and ℓ is greatest.

3. The manipulators' preference order σ is to like candidates in the opposite of their lexicographic order. In particular, c is the coalition's most preferred candidate.

This is a polynomial-time reduction. It follows immediately from this construction and the definition of $\mathcal{E}$ that y is in QBF if and only if (C, u, V, σ, c) is in online-$\mathcal{E}$-UCM.

To prove the last statement, simply let $\mathcal{E}$ be the election system that ignores the weights of the voters and then works exactly as the previous election system. ❑ Theorem 3

Proof of Theorem 12. We first show that online-veto-WCM is in P^{NP}. The proof is reminiscent of the proof for 1-veto in Theorem 10. Let (C, u, V, σ, d) be a given instance of online-veto-WCM with $C = \{c_1, c_2, \ldots, c_m\}$ and $c_1 >_\sigma c_2 >_\sigma \cdots >_\sigma c_m$. Suppose $d = c_i$. Our P^{NP} algorithm proceeds as follows:

1. Compute the minimal threshold t_1 such that there exists a partition $(A_{i+1}, \ldots, A_m)$ of the weights of the manipulators from u onward such that for each j, $i+1 \leq j \leq m$, $\mathit{maxscore}(c_j) - \sum A_j \leq t_1$, where $\mathit{maxscore}(c_j)$ is c_j's score when none of the voters from u onward veto c. That is, by having manipulators from u onward with weights in A_j veto c_j, the manipulators from u onward can ensure that none of the candidates they dislike more than d exceeds a score of t_1.

2. Compute the minimal threshold t_2 such that there exists a partition $(A_1, \ldots, A_i)$ of the weights of the non-manipulators after u such that for each j, $1 \leq j \leq i$,

$maxscore(c_j) - \sum A_j \leq t_2$. That is, if the nonmanipulators after u with weights in A_j veto c_j, none of the candidates that the manipulators like as least as much as d exceeds a score of t_2.

3. Accept if and only if $t_1 \leq t_2$.

Note that the first two steps of the algorithm can both be done in $\mathrm{FP}^{\mathrm{NP}}$ by using an NP oracle that checks whether there exists a partition of the specified kind.

It remains to show that online-veto$_{|4}$-WCM is P^{NP}-hard. We will reduce from the standard P^{NP}-complete problem MAXSATASG$_=$, which is the set of pairs of 3cnf formulas[3] that have the same maximal satisfying assignment [34]. To be precise, we will assume that our propositional variables are $x_1, x_2, \ldots$. If x_n is the largest propositional variable occurring in ϕ, we often write $\phi(x_1, \ldots, x_n)$ to make that explicit. An assignment for $\phi(x_1, \ldots, x_n)$ is an n-bit string α such that α_i gives the assignment for variable x_i. We will sometimes identify α with the binary integer it represents. For ϕ a formula, $maxsatasg(\phi)$ is the lexicographically largest satisfying assignment for ϕ. If ϕ is not satisfiable, $maxsatasg(\phi)$ is not defined. And we define MAXSATASG$_=$ as the set of pairs of 3cnf formulas $(\phi(x_1, \ldots, x_n), \psi(x_1, \ldots, x_n))$ such that ϕ and ψ are satisfiable 3cnf formulas, and $maxsatasg(\phi) = maxsatasg(\psi)$.

The OMS that we will construct will have four candidates, $a >_\sigma b >_\sigma c >_\sigma d$, and the distinguished candidate will be b. Looking at the P^{NP} algorithm above, we can see that determining whether the OMS can be manipulated basically amounts to determining whether the nonmanipulator weights have a "better" partition than the manipulator weights.

So, we will associate formulas with multisets of positive integers, and their satisfying assignments with subset sums. This already happens in the standard reduction from 3SAT to SubsetSum. However, we also want larger satisfying assignments to correspond to "better" subset sums. In order to do this, we use Wagner's variation of the 3SAT to SubsetSum reduction [34]. Wagner uses this reduction to prove that determining whether the largest subset sum up to a certain bound is odd is a P^{NP}-hard problem.

LEMMA 14. *Let $\phi(x_1, \ldots, x_n)$ be a 3cnf formula. Wagner's reduction maps this formula to an instance $(k_1, \ldots, k_t, L)$ of SubsetSum with the following properties:*

1. *For all assignments α, $\phi[\alpha]$ if and only if there exists a subset of $k_1, \ldots, k_t$ that sums to $L + \alpha$.*
2. *For all K such that $2^n \leq K \leq 2(2^n - 1)$, no subset of $k_1, \ldots, k_t$ sums to $L + K$.*

Proof of Lemma 14. The first claim is immediate from the proof of Theorem 8.1(3) from [34]. For the second claim, note that $L + K \leq L + 2(2^n - 1) < L + 6^n$. In Wagner's construction, $L = \underbrace{3\cdots3}_{m}\underbrace{1\cdots1}_{n}\underbrace{0\cdots0}_{n}$ in base 6, where m is the number of clauses in ϕ. So, $(L + K)$'s representation in base 6 is $\underbrace{3\cdots3}_{m}\underbrace{1\cdots1}_{n}$ followed by n digits. It is easy to see from Wagner's construction that the subset sums of this form that can be realized are exactly $L + \beta$, where β is a satisfying assignment of ϕ. Since $K \geq 2^n$, K is not even an assignment, and thus no subset of $k_1, \ldots, k_t$ sums to $L + K$. ❑ Lemma 14

Let $\phi(x_1, \ldots, x_n)$ and $\psi(x_1, \ldots, x_n)$ be 3cnf formulas, and consider instance (ϕ, ψ) of MAXSATASG$_=$. Without loss of generality, we assume that x_1 does not actually occur in ϕ or ψ. We will define an OMS (C, u, V, σ, b) with $C = \{a, b, c, d\}$ and $\sigma = a > b > c > d$ such that $(\phi, \psi) \in$ MAXSATASG$_=$ if and only if (C, u, V, σ, b) is a positive instance of online-veto-WCM. Note that MAXSATASG$_=$ corresponds to optimal solutions being equal, while online-veto-WCM corresponds to one optimal solution being at least as good as the other. We will first modify the formulas such that we also look at the optimal solution for one formula being at least as good as the optimal solution for the other. The following is immediate.

CLAIM 15. *$(\phi, \psi) \in MAXSATASG_=$ if and only if $\phi \wedge \psi$ is satisfiable and $maxsatasg(\phi \wedge \psi) \geq maxsatasg(\phi \vee \psi)$.*

It will also be very useful if one of the formulas is always satisfiable. We can easily ensure this by adding an extra variable that will correspond to the highest order bit of the satisfying assignment. Recall that x_1 does not occur in ϕ or ψ.

CLAIM 16. *$(\phi, \psi) \in MAXSATASG_=$ if and only if $\phi \wedge \psi \wedge x_1$ is satisfiable and*

$$maxsatasg(\phi \wedge \psi \wedge x_1) \geq maxsatasg(\phi \vee \psi \vee \overline{x_1}).$$

Now we would like to apply the reduction from Lemma 14 on $\phi \wedge \psi \wedge x_1$ and $\phi \vee \psi \vee \overline{x_1}$. But wait! This reduction is defined for 3cnf formulas, and $\phi \vee \psi \vee \overline{x_1}$ is not in 3cnf. Since ϕ and ψ are in 3cnf, it is easy to convert $\phi \vee \psi \vee \overline{x_1}$ into cnf in polynomial time. Let g be the standard reduction from CNF-SAT to 3SAT. We can rename the variables such that g has the following property: For $\xi(x_1, \ldots, x_n)$ a cnf formula, $g(\xi)(x_1, \ldots, x_n, x_{n+1}, \ldots, x_{\hat{n}})$ is a 3cnf formula such that $\hat{n} > n$ and such that for all assignments $\alpha \in \{0,1\}^n$, $\xi[\alpha]$ if and only if there exists an assignment $\beta \in \{0,1\}^{\hat{n}-n}$ such that $g(\xi)[\alpha\beta]$.

Let $\widehat{\psi}(x_1, \ldots, x_{\hat{n}}) = g(\phi \vee \psi \vee \overline{x_1})$. Let $\widehat{\phi}(x_1, \ldots, x_{\hat{n}}) = \phi \wedge \psi \wedge (x_1 \vee x_1 \vee x_1) \wedge (x_{\hat{n}} \vee x_{\hat{n}} \vee \overline{x_{\hat{n}}})$.

CLAIM 17.
- *$\widehat{\phi}$ and $\widehat{\psi}$ are in 3cnf and $\widehat{\psi}$ is satisfiable.*
- *$(\phi, \psi) \in MAXSATASG_=$ if and only if $\widehat{\phi}$ is satisfiable and $maxsatasg(\widehat{\phi}) \geq maxsatasg(\widehat{\psi})$.*

Proof of Claim 17. From the previous claim we know that if $(\phi, \psi) \in$ MAXSATASG$_=$, then $\phi \wedge \psi \wedge x_1$ is satisfiable and thus $\widehat{\phi}$ is satisfiable. Also from the previous claim, if $(\phi, \psi) \in$ MAXSATASG$_=$, then $maxsatasg(\phi \wedge \psi \wedge x_1) \geq maxsatasg(\phi \vee \psi \vee \overline{x_1})$. Let α be the maximal satisfying assignment of $\phi \wedge \psi \wedge x_1$. Then $\alpha 1^{\hat{n}-n}$ is the maximal satisfying assignment of $\widehat{\phi}$. Let α' be the maximal satisfying assignment of $\phi \vee \psi \vee \overline{x_1}$. Then $\alpha'\beta$ is the maximal satisfying assignment of $\widehat{\psi}$ for some β. Since $\alpha \geq \alpha'$, it follows that $\alpha 1^{\hat{n}-n} \geq \alpha'\beta$.

For the converse, suppose that $\widehat{\phi}$ is satisfiable and $maxsatasg(\widehat{\phi}) \geq maxsatasg(\widehat{\psi})$. Let γ be the maximal satisfying assignment of $\widehat{\phi}$ and let γ' be the maximal satisfying

[3] We denote a formula in conjunctive normal form by *cnf formula*, and a *3cnf formula* is a cnf formula with exactly three literals per clause.

assignment of $\widehat{\psi}$. Then the length-n prefix of γ is the maximal satisfying assignment of $\phi \wedge \psi \wedge x_1$ and the length-n prefix of γ' is the maximal satisfying assignment of $\phi \vee \psi \vee \overline{x_1}$. Since $\gamma \geq \gamma'$, the n-bit prefix of γ is greater than or equal to the n-bit prefix of γ'. ❑ Claim17

We now apply Wagner's reduction from Lemma 14 to $\widehat{\phi}$ and $\widehat{\psi}$. Let $k_1, \ldots, k_t, L$ be the output of Wagner's reduction on $\widehat{\phi}$ and let $k'_1, \ldots, k'_{t'}, L'$ be the output of Wagner's reduction on $\widehat{\psi}$.

As mentioned previously, we will define an OMS (C, u, V, σ, b) with $C = \{a, b, c, d\}$ and $\sigma = a > b > c > d$ such that $(\phi, \psi) \in \text{MAXSATASG}_=$ if and only if (C, u, V, σ, b) is a positive instance of online-veto-WCM. Because we are looking at veto, when determining the outcome of an election, it is easiest to simply count the number of vetoes for each candidate. Winners have the fewest vetoes. For $\hat{c}$ a candidate, we will denote the total weight of the voters that veto $\hat{c}$ by $vetoes(\hat{c})$.

There are four voters in $V_{<u}$: one voter of weight L vetoing a, one voter of weight $L + 2L' + 2(2^{\hat{n}} - 1) - \sum k'_i$ vetoing b, one voter of weight L' vetoing c, and one voter of weight $L' + 2L + 2(2^{\hat{n}} - 1) - \sum k_i$ vetoing d. Let $u = u_1$. $V_{u<}$ consists of $t - 1$ further manipulators $u_2, \ldots, u_t$ followed by t' nonmanipulators $u'_1, \ldots, u'_{t'}$. The weight of manipulator u_i is k_i and the weight of nonmanipulator u'_i is k'_i.

It remains to show that the reduction is correct. First suppose that (ϕ, ψ) is in $\text{MAXSATASG}_=$. By Claim 17, this implies that $\widehat{\phi}$ and $\widehat{\psi}$ are satisfiable 3cnf formulas such that $maxsatasg(\widehat{\phi}) \geq maxsatasg(\widehat{\psi})$. Let $\alpha = maxsatasg(\widehat{\phi})$. We know from Lemma 14 that there exists a subset of $k_1, \ldots, k_t$ that sums to $L + \alpha$. The manipulators corresponding to this subset will veto c, so that c receives $L + \alpha$ vetoes from the manipulators. The remaining manipulators will veto d, i.e., d receives $(\sum k_i) - L - \alpha$ vetoes from the manipulators. After the manipulators have voted, $vetoes(a) = L$, $vetoes(b) = L + 2L' + 2(2^{\hat{n}} - 1) - \sum k'_i$, $vetoes(c) = L' + L + \alpha$, and $vetoes(d) = L' + L + 2(2^{\hat{n}} - 1) - \alpha$. Since $\alpha \leq 2^{\hat{n}} - 1$, $vetoes(c) \leq vetoes(d)$. We will show that no matter how the nonmanipulators vote, a or b is a winner. Suppose for a contradiction that after the nonmanipulators have voted, $vetoes(a) > vetoes(c)$ and $vetoes(b) > vetoes(c)$. If that were to happen, there would be a subset of $k'_1, \ldots, k'_{t'}$ summing to K such that $L + K = vetoes(a) > vetoes(c) = L + L' + \alpha$ and $L + 2L' + 2(2^{\hat{n}} - 1) - K = vetoes(b) > vetoes(c) = L + L' + \alpha$. It follows that $\alpha < K - L' < 2(2^{\hat{n}} - 1)$ and there exists a subset of $k'_1, \ldots, k'_{t'}$ that sums to $L' + (K - L')$. It follows from Lemma 14 that $K - L'$ is a satisfying assignment for $\widehat{\psi}$. But that contradicts the assumption that $maxsatasg(\widehat{\phi}) \geq maxsatasg(\widehat{\psi})$.

The proof of the converse is very similar. Suppose that $(\phi, \psi) \notin \text{MAXSATASG}_=$. By Claim 17, $\widehat{\psi}$ is satisfiable. Let $\alpha = maxsatasg(\widehat{\psi})$. By Claim 17, either $\widehat{\phi}$ is not satisfiable or $maxsatasg(\widehat{\phi}) < \alpha$. Suppose the manipulators vote such that c receives K vetoes from some of them. Without loss of generality, assume all other manipulators veto d, so that d receives $(\sum k_i) - K$ vetoes from the manipulators. We know from Lemma 14 that there exists a subset of $k'_1, \ldots, k'_{t'}$ that sums to $L' + \alpha$. After the manipulators have voted, the nonmanipulators will vote such that a receives $L' + \alpha$ vetoes from the nonmanipulators and the remaining nonmanipulators will veto b, i.e., b receives $(\sum k'_i) - L' - \alpha$ vetoes from the nonmanipulators. So, $vetoes(a) = L + L' + \alpha$, $vetoes(b) = L + L' + 2(2^{\hat{n}} - 1) - \alpha$, $vetoes(c) = L' + K$, and $vetoes(d) = L' + 2L + 2(2^{\hat{n}} - 1) - K$. We will show that neither a nor b is a winner. Since $\alpha \leq 2^{\hat{n}} - 1$, $vetoes(a) \leq vetoes(b)$. So it suffices to show that a is not a winner. If a were a winner, $vetoes(a) \leq vetoes(c)$ and $vetoes(a) \leq vetoes(d)$. This implies that $\alpha \leq K - L \leq 2(2^{\hat{n}} - 1)$. It follows from Lemma 14 that $K - L$ is a satisfying assignment for $\widehat{\phi}$. But that contradicts the assumption that either $\widehat{\phi}$ is not satisfiable or $maxsatasg(\widehat{\phi}) < \alpha$. ❑ Theorem 12

9. REFERENCES

[1] Y. Bachrach, N. Betzler, and P. Faliszewski. Probabilistic possible winner determination. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence*, pages 697–702. AAAI Press, July 2010.

[2] J. Bartholdi, III and J. Orlin. Single transferable vote resists strategic voting. *Social Choice and Welfare*, 8(4):341–354, 1991.

[3] J. Bartholdi, III, C. Tovey, and M. Trick. The computational difficulty of manipulating an election. *Social Choice and Welfare*, 6(3):227–241, 1989.

[4] D. Baumeister and J. Rothe. Taking the final step to a full dichotomy of the possible winner problem in pure scoring rules. *Information Processing Letters*, 112(5):186–190, 2012.

[5] N. Betzler. On problem kernels for possible winner determination under the k-approval protocol. In *Proceedings of the 35th International Symposium on Mathematical Foundations of Computer Science*, pages 114–125. Springer-Verlag *Lecture Notes in Computer Science #6281*, August 2010.

[6] N. Betzler and B. Dorn. Towards a dichotomy of finding possible winners in elections based on scoring rules. *Journal of Computer and System Sciences*, 76(8):812–836, 2010.

[7] N. Betzler, S. Hemmann, and R. Niedermeier. A multivariate complexity analysis of determining possible winners given incomplete votes. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence*, pages 53–58. AAAI Press, July 2009.

[8] A. Borodin and R. El-Yaniv. *Online Computation and Competitive Analysis*. Cambridge University Press, 1998.

[9] A. Chandra, D. Kozen, and L. Stockmeyer. Alternation. *Journal of the ACM*, 26(1), 1981.

[10] Y. Chevaleyre, J. Lang, N. Maudet, J. Monnot, and L. Xia. New candidates welcome! Possible winners with respect to the addition of new candidates. *Mathematical Social Sciences*, 64(1):74–88, 2012.

[11] V. Conitzer. Making decisions based on the preferences of multiple agents. *Communications of the ACM*, 53(3):84–94, 2010.

[12] V. Conitzer, T. Sandholm, and J. Lang. When are elections with few candidates hard to manipulate? *Journal of the ACM*, 54(3):Article 14, 2007.

[13] Y. Desmedt and E. Elkind. Equilibria of plurality voting with abstentions. In *Proceedings of the 11th ACM Conference on Electronic Commerce*, pages 347–356. ACM Press, June 2010.

[14] C. Dwork, R. Kumar, M. Naor, and D. Sivakumar. Rank aggregation methods for the web. In *Proceedings of the 10th International World Wide Web Conference*, pages 613–622. ACM Press, Mar. 2001.

[15] P. Faliszewski, E. Hemaspaandra, and L. Hemaspaandra. Using complexity to protect elections. *Communications of the ACM*, 53(11):74–82, 2010.

[16] P. Faliszewski, E. Hemaspaandra, L. Hemaspaandra, and J. Rothe. A richer understanding of the complexity of election systems. In S. Ravi and S. Shukla, editors, *Fundamental Problems in Computing: Essays in Honor of Professor Daniel J. Rosenkrantz*, pages 375–406. Springer, 2009.

[17] P. Faliszewski, E. Hemaspaandra, and H. Schnoor. Copeland voting: Ties matter. In *Proceedings of the 7th International Conference on Autonomous Agents and Multiagent Systems*, pages 983–990. International Foundation for Autonomous Agents and Multiagent Systems, May 2008.

[18] S. Ghosh, M. Mundhe, K. Hernandez, and S. Sen. Voting for movies: The anatomy of recommender systems. In *Proceedings of the 3rd Annual Conference on Autonomous Agents*, pages 434–435. ACM Press, 1999.

[19] A. Gibbard. Manipulation of voting schemes. *Econometrica*, 41(4):587–601, 1973.

[20] E. Hemaspaandra, L. Hemaspaandra, and J. Rothe. The complexity of online manipulation of sequential elections. Technical Report arXiv:1202.6655 [cs.GT], Computing Research Repository, arXiv.org/corr/, Feb. 2012. Revised, September 2012.

[21] E. Hemaspaandra, L. Hemaspaandra, and J. Rothe. Controlling candidate-sequential elections. In *Proceedings of the 20th European Conference on Artificial Intelligence*, pages 905–906. IOS Press, Aug. 2012.

[22] E. Hemaspaandra, L. Hemaspaandra, and J. Rothe. Online voter control in sequential elections. In *Proceedings of the 20th European Conference on Artificial Intelligence*, pages 396–401. IOS Press, Aug. 2012.

[23] L. Hemaspaandra and R. Williams. An atypical survey of typical-case heuristic algorithms. *SIGACT News*, 43(4), Dec. 2012. To appear.

[24] J. Kadin. The polynomial time hierarchy collapses if the boolean hierarchy collapses. *SIAM Journal on Computing*, 17(6):1263–1282, 1988. Erratum appears in the same journal, 20(2):404.

[25] K. Konczak and J. Lang. Voting procedures with incomplete preferences. In *Proceedings of the Multidisciplinary IJCAI-05 Workshop on Advances in Preference Handling*, pages 124–129, July/August 2005.

[26] M. Krentel. The complexity of optimization problems. *Journal of Computer and System Sciences*, 36(3):490–509, 1988.

[27] J. Lang, M. Pini, F. Rossi, D. Salvagnin, K. Venable, and T. Walsh. Winner determination in voting trees with incomplete preferences and weighted votes. *Journal of Autonomous Agents and Multi-Agent Systems*, 25(1):130–157, 2012.

[28] A. Meyer and L. Stockmeyer. The equivalence problem for regular expressions with squaring requires exponential space. In *Proceedings of the 13th IEEE Symposium on Switching and Automata Theory*, pages 125–129. IEEE Press, Oct. 1972.

[29] M. Satterthwaite. Strategy-proofness and Arrow's conditions: Existence and correspondence theorems for voting procedures and social welfare functions. *Journal of Economic Theory*, 10(2):187–217, 1975.

[30] B. Sloth. The theory of voting and equilibria in noncooperative games. *Games and Economic Behavior*, 5(1):152–169, 1993.

[31] L. Stockmeyer. The polynomial-time hierarchy. *Theoretical Computer Science*, 3(1):1–22, 1976.

[32] L. Stockmeyer and A. Meyer. Word problems requiring exponential time. In *Proceedings of the 5th ACM Symposium on Theory of Computing*, pages 1–9. ACM Press, 1973.

[33] M. Tennenholtz. Transitive voting. In *Proceedings of the 5th ACM Conference on Electronic Commerce*, pages 230–231. ACM Press, July 2004.

[34] K. Wagner. More complicated questions about maxima and minima, and some closures of NP. *Theoretical Computer Science*, 51(1–2):53–80, 1987.

[35] C. Wrathall. Complete sets and the polynomial-time hierarchy. *Theoretical Computer Science*, 3(1):23–33, 1976.

[36] L. Xia and V. Conitzer. Determining possible and necessary winners under common voting rules given partial orders. In *Proceedings of the 23rd AAAI Conference on Artificial Intelligence*, pages 196–201. AAAI Press, July 2008.

[37] L. Xia and V. Conitzer. Stackelberg voting games: Computational aspects and paradoxes. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence*, pages 697–702. AAAI Press, July 2010.

[38] L. Xia and V. Conitzer. Strategy-proof voting rules over multi-issue domains with restricted preferences. In *Proceedings of the 6th International Workshop On Internet And Network Economics*, pages 402–414. Springer-Verlag *Lecture Notes in Computer Science #6484*, Dec. 2010.

[39] L. Xia, V. Conitzer, and J. Lang. Aggregating preferences in multi-issue domains by using maximum likelihood estimators. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems*, pages 399–406. International Foundation for Autonomous Agents and Multiagent Systems, May 2010.

[40] L. Xia, V. Conitzer, and J. Lang. Strategic sequential voting in multi-issue domains and multiple-election paradoxes. In *Proceedings of the 12th ACM Conference on Electronic Commerce*, pages 179–188. ACM Press, 2011.

[41] L. Xia, J. Lang, and V. Conitzer. Hypercubewise preference aggregation in multi-issue domains. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, pages 158–163. AAAI Press, July 2011.

Symbolic Synthesis of Knowledge-based Program Implementations with Synchronous Semantics

X. Huang
xiaoweih@cse.unsw.edu.au

R. van der Meyden
meyden@cse.unsw.edu.au

ABSTRACT

This paper deals with the automated synthesis of implementations of knowledge-based programs with respect to two synchronous semantics (clock and synchronous perfect recall). An approach to the synthesis problem based on the use of symbolic representations is described. The method has been implemented as an extension to the model checker MCK. Two applications of the implemented synthesis system are presented: the muddy children puzzle (where performance is compared to an explicit state method for a related problem implemented in the model checker DEMO), and a knowledge-based program for a dynamic leader election problem in a ring of processes.

Categories and Subject Descriptors

D.2.4 [**Software Engineering**]: Software/Program Verification; F.4.1 [**Mathematical Logic**]: Modal Logic

General Terms

Theory, Verification

Keywords

Synthesis, Logic of Knowledge, Knowledge-based Programs

1. INTRODUCTION

One of the main motivations for the application of epistemic logic in computer science has been the observation that it provides a beneficial level of abstraction through which to view distributed systems. A range of problems in distributed computing have been studied from this perspective, including protocols for agreement [17, 8, 9, 27], message transmission [19], atomic commitment [15, 25], clock synchronization [26, 28], leader election [18], and secure communication [35, 1, 2, 24, 36].

Many of these analyses are based on first expressing the solution to a problem in terms of the relation between an agent's actions and its knowledge, and then seeking to understand the conditions under which the agent has the requisite knowledge. A first codification of the approach was the semantic notion of *knowledge-based protocols* of [16], and the idea was refined and given a syntactic basis in the *knowledge-based programs* of [11, 10]. The latter provide a simple guarded command programming notation (in the style of Unity [7]), in which the guards in conditional statements are not just expressions over the agent's local variables, but may also contain formulas of epistemic logic asserting some property of the agent's knowledge.

Knowledge-based programs resemble standard proagrams, but they do not have a straightforward operational semantics. Instead, they are semantically more like a specification, in that they stand in an *implementation* relation to standard programs. To obtain an implementation, one must replace the knowledge conditions in the program by expressions in the agents' local variables that are equivalent, when running the resulting standard program. Because of the *fixpoint* nature of this semantics, in general, a knowledge-based program could have no, one, or many behaviourally distinct implementations. There are, however, some syntactic and semantic conditions under which implementations are guaranteed to be unique [11]. One of these is that the formulas appearing in the knowledge conditions are free of temporal operators and that the semantics of the knowledge operators is *synchronous*, in the sense that agents always know the current time.

The early literature on knowledge in distributed computing and knowledge-based programs is confined to "pencil and paper" analyses. In recent years, automated tool support for knowledge-based analysis has begun to be developed, in the form of *epistemic model checkers* [13, 23, 30, 20, 37], which are able to automatically verify whether standard programs satisfy epistemic specifications. These model checkers have been applied to a number of case studies in which it is verified that a proposed implementation of a knowledge-based program is indeed an implementation [3, 2, 24]. However, the approach applied in these studies still requires that the proposed implementation be derived manually. Since the implementations may make use of subtle sources of information, this can be a highly nontrivial task, although the examples automatically constructed by the model checker when checking an incorrect implementation can provide useful information to guide the search [3, 1].

Our contribution in this paper is to develop the first practical tool for automated synthesis of knowledge-based program implementations, by extending methods from epistemic model checking. The main contributions of the paper are as follows:

1. We develop a practical syntax for knowledge-based programs that extends the Unity style programs of [11] to encompass use of knowledge in assignment statements, as well as sequential structure.
2. We show how existing symbolic techniques for epistemic model checking may be extended to yield an approach to automated synthesis of knowledge-based program implementations. Our techniques work for the special case of atemporal knowledge-based programs with respect to two distinct synchronous semantics for knowledge, the *clock* and *synchronous perfect recall semantics*, in which, as noted above, unique implementations are guaranteed to exist.

TARK 2013, Chennai, India.

3. We have implemented these algorithms as an extension of the epistemic model checker MCK. One benefit of building on the existing model checking technology is that properties of the implementation derived can directly be verified, with many of the computational steps required for verification already performed by the synthesis procedure.

4. We conduct a number of validation case studies of knowledge-based program implementation using the resulting tool, considering two types of examples. In the first, the muddy children puzzle, we compare the performance of our symbolic synthesis approach to the performance of the model checker DEMO. DEMO does not synthesize implementations of knowledge-based programs, but solves a closely related model checking problem using an explicit state rather than symbolic technique. The second example we consider is a knowledge-based program for a leader election protocol in a ring of processes.

The structure of the paper is as follows. In Section 2, we review the basics of epistemic model checking and a symbolic technique used in the implementation of such systems. We develop a syntax and semantics for knowledge-based programs in Section 3. Section 4 describes the basis for a symbolically implementable procedure for synthesis of knowledge-based program implementations. The application of this procedure to a number of examples is discussed in Section 5. Section 6 discusses related work, and we make some concluding remarks in Section 7.

2. EPISTEMIC MODEL CHECKING

In this section, we recall the background we require from epistemic logic (following [10]) and epistemic model checking (following [34]).

Let $Prop$ be a set of atomic proposition and let Ags be a finite set of agents. The temporal-epistemic logic that we work with has the syntax

$$\phi ::= p \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid X\phi \mid K_i\phi$$

where $p \in Prop$ and $i \in Ags$. Intuitively, formula $X\phi$ expresses that ϕ holds at the next time, and $K_i\phi$ expresses that agent i knows that ϕ holds. A formula is *atemporal* if it does not make use of the temporal operator X.

At all times, each agent i is assumed to be in some local state that records all the information that it can access at that time. The environment e records "everything else that is relevant". Let S be the set of environment states and let L_i be the set of local states of agent i. A *global state* s of a multi-agent system is a tuple $(s_e, s_1, ..., s_n)$ such that $s_e \in S$ and $s_i \in L_i$ for all $i \in Ags$.

A *run* r is a function from time to global states, i.e., $r : \mathbb{N} \to S \times L_1 \times ... \times L_n$. A pair (r, m) consisting of a run r and a time m is called a point. A *system* $\mathcal{R}$ is a set of runs. We call $\mathcal{R} \times \mathbb{N}$ the set of points of $\mathcal{R}$. If $r(m) = (s_e, s_1, .., s_n)$ then for $x \in Ags \cup \{e\}$ we write $r_x(m)$ for s_x and $r_x(0..m)$ for $r_x(0)...r_x(m)$. Relative to a system $\mathcal{R}$, we define the set $\mathcal{K}_i(r, m) = \{(r', m') \in \mathcal{R} \times \mathbb{N} \mid r_i(m) = r'_i(m')\}$ to be the set of points that are indistinguishable from the point (r, m) for agent i.

An interpreted system $\mathcal{I}$ is a tuple $(\mathcal{R}, \pi)$ such that $\mathcal{R}$ is a system and $\pi : \mathcal{R} \times \mathbb{N} \to \mathcal{P}(Prop)$ is an assignment giving an interpretation to the atomic propositions at each point. Given an interpreted system $\mathcal{I}$, a point (r, m), and a formula ϕ, we define the relation $\mathcal{I}, (r, m) \models \phi$ inductively by

- $\mathcal{I}, (r, m) \models p$ if $p \in \pi(r, m)$
- $\mathcal{I}, (r, m) \models \neg\phi$ if not $\mathcal{I}, (r, m) \models \phi$
- $\mathcal{I}, (r, m) \models \phi_1 \wedge \phi_2$ if $\mathcal{I}, (r, m) \models \phi_1$ and $\mathcal{I}, (r, m) \models \phi_2$
- $\mathcal{I}, (r, m) \models X\phi$ if $\mathcal{I}, (r, m + 1) \models \phi$
- $\mathcal{I}, (r, m) \models K_i\phi$ if $\mathcal{I}, (r', m') \models \phi$ for all points $(r', m') \in \mathcal{K}_i(r, m)$

Since interpreted systems are infinite structures and for model checking we require a finite input, we generate interpreted systems from finite structures. A *(finite state) transition model* M for agents Ags is a tuple $M = (S, I, \{O_i\}_{i \in Ags}, \to, \pi)$, where S is a (finite) set of states, $I \subseteq S$ is the set of initial states, each $O_i : S \to \mathcal{O}$ is a function representing the observation that agent i makes at each state, $\to \subseteq S^2$ is a serial transition relation over states in S, and $\pi : S \to \mathcal{P}(Prop)$ is a propositional assignment. Let $k_i(s) = \{s' \in S \mid O_i(s) = O_i(s')\}$ be the set of states that are indistinguishable from state s for agent i, based on its observation.

A *path* ρ from a state s of M is a finite or infinite sequence of states $s_0 s_1 ...$, such that $s_0 = s$ and $s_k \to s_{k+1}$ for all $k < |\rho| - 1$, where $|\rho|$ is the total number of states in ρ. Given such a path ρ, we use $\rho(m)$ to denote the state s_m. A *fullpath* from a state s is an infinite path from s. A *path* ρ is initialized if $\rho(0) \in I$.

To obtain a system from a finite state transition model M, we treat the states of M as the states of the environment, and obtain runs from paths by adding local states at each point. This can be done in a variety of ways, representing different levels to which agents recall their observations. We call the level of recall a *view* and deal with the views `obs`, `clk` and `spr` representing recall only of the current observation, recall of the current observation and the time and synchronous perfect recall, respectively.

For each initialized fullpath ρ and view $\mathcal{V} \in \{\texttt{obs}, \texttt{clk}, \texttt{spr}\}$, we define a run $\rho^{\mathcal{V}}$. The state of the environment at time m is given by $\rho_e^{\mathcal{V}}(m) = \rho(m)$ in each case, and the agents' local states are assigned as follows:

- $\mathcal{V} = \texttt{obs}$: the local state of agent i at time m is $\rho_i^{\texttt{obs}}(m) = O_i(\rho(m))$;
- $\mathcal{V} = \texttt{clk}$: the local state of agent i at time m is $\rho_i^{\texttt{clk}}(m) = (m, O_i(\rho(m)))$;
- $\mathcal{V} = \texttt{spr}$: the local state of agent i at time m is $\rho_i^{\texttt{spr}}(m) = O_i(\rho(0))...O_i(\rho(m))$.

Given a system M and a view $\mathcal{V}$, we write $\mathcal{R}^{\mathcal{V}}(M)$ for the set of runs $\rho^{\mathcal{V}}$ where ρ is an initialized full-path of M. The interpretation π of M lifts to an interpretation $\pi^{\mathcal{V}}$ on the global states in $\mathcal{R}^{\mathcal{V}}(M)$, defined by $\pi^{\mathcal{V}}((s, l_1, \ldots, l_n)) = \pi(s)$. We define the interpreted system obtained from M using view $\mathcal{V}$ by $\mathcal{I}^{\mathcal{V}}(M) = (\mathcal{R}^{\mathcal{V}}(M), \pi^{\mathcal{V}})$. Given a finite model M, a view $\mathcal{V}$, and a formula ϕ, we write $M \models^{\mathcal{V}} \phi$ if $\mathcal{I}^{\mathcal{V}}(M), (r, 0) \models \phi$ for all $r \in \mathcal{R}^{\mathcal{V}}(M)$.

The *model checking problem* is to determine, given a finite state transition model M, a view $\mathcal{V}$ and a temporal epistemic formula ϕ, whether $\mathcal{I}^{\mathcal{V}}(M) \models \phi$. *Epistemic model checkers* are software systems that solve this problem. A number of such systems have been implemented. MCK [13] supports all three views, MCMAS [23], Verics [20] and MCTK [30] work with the observational view. These systems use a variety of temporal logics for the temporal expressiveness in formulas. MCK supports a superset of the language defined above.

Before concluding this section, we define a presentation of standard S5$_n$ Kripke-structures that will be used later. An *epistemic model* is a tuple $\mathtt{M} = (S, \{O_i\}_{i \in Ags}, \pi)$, where the components are of the same type as the similarly named components in state transition models. Given a model $\mathtt{M}$, a state $s \in S$, and an *atemporal* formula ϕ, the relation $\mathtt{M}, s \models \phi$ can be recursively defined as follows:

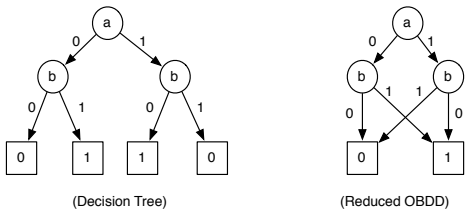


Figure 1: A decision tree and its reduced OBDD

- $\mathtt{M}, s \models p$ if $p \in \pi(s)$
- $\mathtt{M}, s \models \neg\phi$ if not $M, s \models \phi$
- $\mathtt{M}, s \models \phi_1 \wedge \phi_2$ if $M, s \models \phi_1$ and $M, s \models \phi_2$
- $\mathtt{M}, s \models K_i\phi$ if $M, s' \models \phi$ for all states s' such that $O_i(s) = O_i(s')$

It is easily seen that for atemporal formulas ψ that are boolean combinations of formulas of the form $K_i\phi$ (for a fixed i), for all $s, s' \in S$ with $O_i(s) = O_i(s')$ we have $\mathtt{M}, s \models \psi$ iff $\mathtt{M}, s' \models \psi$, i.e., the truth value of ψ depends only on $O_i(s)$. For $o \in O_i(S)$, we may therefore define the relation $M, o \models_i \psi$ if $M, s \models \psi$ for some (equivalently, all) $s \in S$ with $O_i(s) = o$.

2.1 Symbolic Data Structures

MCK supports a number of different algorithmic approaches to solving the epistemic model checking problem. One of these is based on symbolic model checking using (reduced) ordered binary decision diagrams (BDD) [6]. These are data structures defined as follows.

Let V be a set of variables. A *V-assignment* is a function $s : V \to \{0, 1\}$. Write $Assgts(V)$ for the set of all V-assignments, and $s[v \mapsto x]$ for the function that is identical to s except that it takes value x on input v. A *V-indexed boolean function* is a mapping $f : Assgts(V) \to \{0, 1\}$. Note that such functions are able to represent sets $X \subseteq Assgts(V)$ by their characteristic functions f_X, mapping s to 1 just in case $s \in X$. One way to represent such a function f is using a *binary tree* of height n, with each level corresponding to one of the variables in V, and leaves labelled from $\{0, 1\}$. This tree can in turn be thought of as a finite state automaton on alphabet $\{0, 1\}$. Reduced ordered binary decision diagrams (BDD's in the sequel) more compactly represent such a function as a *dag* of height n, with binary branching, by applying the usual finite state automaton minimization algorithm. A very simple example of this for the function $f(a, b, c) = a$ `xor` b is illustrated in Figure 1. In some cases, the degree of compaction obtained in the minimal dag representation is considerable. We note that the amount of compaction obtained is sensitive to the variable ordering used, and finding a variable ordering that minimizes the result is NP-hard, though there exist good heuristics, such as *sifting* [29].

Given this minimal representation of V-indexed boolean functions, it is moreover possible to compute (in practice, often in reasonable time) some operations on these functions, by means of algorithms that take as input the BDD representation of the input functions and returns the BDD representation of the result. The operations for which this can be done include the following:

- Boolean operations $\wedge, \neg$, defined pointwise on functions. E.g., if $f, g : Assgts(V) \to \{0, 1\}$, then $f \wedge g : Assgts(V) \to \{0, 1\}$ is defined by $(f \wedge g)(s) = f(s) \wedge g(s)$.
- Boolean quantification $\exists, \forall$, e.g., if $f : Assgts(V) \to \{0, 1\}$ and $v \in V$ then $\exists v(f) : Assgts(V \setminus \{v\}) \to \{0, 1\}$ maps $s \in Assgts(V \setminus \{v\})$ to $f(s[v \mapsto 0]) \vee f(s[v \mapsto 1])$.
- variable substitution: if $f : Assgts(V) \to \{0, 1\}$ and $U \subseteq V$ and U' are sets with $U' \cap (V \setminus U) = \emptyset$, and $\sigma : U \to U'$ is a bijection, then $f_\sigma : Assgts((V \setminus U) \cup U') \to \{0, 1\}$ maps $s : Assgts((V \setminus U) \cup U')$ to s', where $s'(v)$ is $s(v)$ when $v \in V \setminus U$ and $s(\sigma^{-1}(v))$ when $v \in U'$.

Symbolic model checking, as implemented in MCK, then proceeds using BDD representations of sets and relations relevant to model checking. For example, the set I of initial states of a system can be represented as a BDD-encoded boolean function f_I indexed by the state variables V. Relations can be represented using "primed" versions of the state variables V, defined by $V' = \{v' \mid v \in V\}$. A relation such as the transition relation $\to$ of a model can then be represented as a function $f_\to$ indexed by variables $V \cup V'$, such that if s and t are assignments to V, we have $s \to t$ iff $f_\to(s \cup t') = 1$, where t' is obtained from t by renaming each variable v to its primed counterpart v'. Operations such as the composition of a relation and a set can then be performed at the level of the BDD representation, e.g. $\{t \in S \mid s \in I \wedge s \to t\}$ is represented by the function $\exists V(f_I \times 1_{V'} \wedge f_\to)\sigma$ on V, where $f_I \times 1_{V'}$ trivially extends f_I by adding (irrelevant) variables V', and σ renames the variables V' to the variables V by removing the prime symbol.

3. KNOWLEDGE-BASED PROGRAMS

Knowledge-based analyses of systems typically concern the interaction between agents' knowledge and their actions. *Knowledge-based programs* [11, 10] have been proposed to capture such relationships in a program-like notation, with actions chosen according to conditions expressed in epistemic logic.

The original presentation of knowledge-based programs used a very simplified (Unity style [7]) programming notation, consisting of a single infinitely repeated do loop containing a set of guarded statements of the form $\phi \to a$ where ϕ is an epistemic formula and a an action. We develop here a slightly richer and more structured notation, using sequential composition and an epistemic assignment statement. The notation is based on the modelling notation already employed by MCK. We focus on *atemporal* programs with a synchronous semantics for knowledge (either the clock or synchronous perfect recall semantics), since this is a case in which unique implementations are guaranteed to exist.

Since, in general, even atemporal knowledge-based programs may not have finite state implementations under the perfect recall semantics, we also limit ourselves to terminating programs, so omit looping from the language. Our handling of parallelism and actions (signals) is somewhat in the spirit of synchronous languages such as Esterel [5]. To give the semantics of knowledge-based programs, we use a formulation based on [32], which allows flexibility in choice of view based on a notion of *environment* that replaces the notion of *context* of [11, 10].

3.1 Standard Programs: Syntax

Define a *standard program* over a set V of variables and a set A of atomic statements to be either the terminated program ϵ or a sequence P of the form $stat_1 ; \ldots ; stat_k$, where the $stat_i$ are simple statements and ';' denotes sequential composition. Each simple statement $stat_i$ is either an atomic statement in A or a nondeterministic branching statement of the form

$$\text{if } g_1 \to a_1 \,[]\, g_2 \to a_2 \,[] \ldots [] \, g_k \to a_k \text{ fi}$$

where each a_i is an atomic statement in A and the g_i are boolean expressions over V called *guards*. Intuitively, a nondeterministic branching statement executes by performing one of the assignments a_i for which the corresponding guard g_i is true. If several guards hold simultaneously, one of the corresponding actions is selected nondeterministically. We treat P as identical to $P; \epsilon$. The *length* of a program is the number of simple statements it contains. We use standard programs to describe both the behavior of agents and the environment in which they operate. The type of atomic statements used in these two cases is different.

Environment models are used to represent how states of the environment are affected by actions of the agents. Formally, we define an *environment model* to be a tuple $\mathcal{M}_e = (Ags, Acts, Var_e, Init_e, \tau)$ where Ags is a set of agents, $Acts$ is a set of actions available to the agents, Var_e is a set of (boolean) environment variables[1], $Init_e$ is an initial condition, in the form of a boolean formula over Var_e, and τ is a *transitions clause* for the environment e, expressed in the form of a standard program.

In addition to the environment variables Var_e, an additional set $ActVar(\mathcal{M}_e) = \{i.a \mid i \in Ags,\ a \in Acts\}$ of (boolean) *action variables* are generated for each model $\mathcal{M}_e$. Intuitively, $i.a$ represents that agent i performs action a. The transitions clause is a standard program over the set of variables in $Var_e \cup ActVar(\mathcal{M}_e)$ and the set of atomic actions of the form $x := expr$, where $x \in Var_e$ and $expr$ is a boolean expresssion over $Var_e \cup ActVar(\mathcal{M}_e)$. The statement $x := expr$ represents that the value of the expression $expr$, is assigned to the variable x.

Protocols are used to describe the behaviour of the agents. A *protocol for agent i in environment model $\mathcal{M}_e$ (of length m)* is a tuple $Prot_i = (PVar_i, LVar_i, OVar_i, Init_i, Acts_i, Prog_i)$ where $PVar_i \subseteq Var_e$ is a set of *parameter variables*[2], $LVar_i$ is a set of *local variables*, $OVar_i \subseteq PVar_i \cup LVar_i$ is a set of *observable variables*, $Init_i$ is an initial condition, in the form of a formula over $LVar_i$, and $Prog_i$ is standard program of length m. The guards in $Prog_i$ are over the set of variables $PVar_i \cup LVar_i$. The atomic statements in P have the form

$$\ll a \mid x_1 := expr_1, ..., x_m := expr_m \gg$$

where $a \in Acts \cup \{nil\}$ and each $x_1 := e_i$ is an assignment statement with x_i in $LVar_i$ and e_i an expression over $PVar_i \cup LVar_i$. Intuitively, such an atomic statement is executed by emitting action a as a signal to the environment: when agent i performs the action, the variable $i.a$ is set to be true (and all other action variables $i.b$ set to be false.) The environment transition clause then runs to update the environment variables. Concurrently, the statement performs the *simultaneous assignment* $x_1 := expr_1, ..., x_m := expr_m$ in a single step of computation. That is, the expressions e_i are first evaluated in the state from which the atomic statement is performed, and their values are then simultaneously assigned to the variables x_i. We abbreviate an atomic statement of the form $\ll nil \mid x := expr \gg$ to $x := expr$, and also abbreviate $\ll nil \mid \gg$ to *skip*.

A *joint protocol (of length m)* is a tuple **Prot** associating a protocol $\mathbf{Prot}_i$ (of length at most m) with each $i \in Ags$. A *system model* is a pair $\mathcal{S} = (\mathcal{M}_e, \mathbf{Prot})$ consisting of an environment model $\mathcal{M}_e$ and a joint protocol **Prot** for $\mathcal{M}_e$. This represents a set of agents running particular protocols in the context of a given environment.

[1]To simplify the presentation we assume here that all variables are boolean; our implementation also allows variables to have a declared finite type.

[2]In the concrete MCK syntax these may be given using new variables in a parameter declaration as aliases for environment variables. This allows sharing of protocol code between agents running similar programs but with different parameter bindings; see the examples below.

3.2 Standard Programs: Semantics

We now show how a system model generates a finite state transition model. To do so, we first convert the system model into a simple form of parallel program and provide these programs with an operational semantics.

We assume we are given a system model $\mathcal{S} = (\mathcal{M}_e, \mathbf{Prot})$, where $\mathcal{M}_e = (Ags, Acts, Var_e, Init_e, \tau)$ and

$$\mathbf{Prot}_i = (PVar_i, LVar_i, OVar_i, Init_i, Acts_i, Prog_i)$$

for $i \in Ags$. We define global states with respect to this model to be boolean assignments s over the set of variables $Var_e \cup \bigcup_{i \in Ags} LVar_i$. We also define the *parallel program*

$$Prog(\mathcal{S}) = \tau \parallel_{i \in Ags} Prog_i\ .$$

This is an expression representing $|Ags| + 1$ components, i.e., the specially identified environment component τ, a program over environment variables, and the $|Ags|$ components $Prog_i$, representing programs associated to the agents.

Intuitively, the operational semantics of these parallel statements defines a transition relation on global states. The definition of the transition relation is given in three stages, captured in the following rule:

$$\frac{\bigwedge_{i \in Ags}\ (s, P_i) \hookrightarrow \ \ll \mathbf{a}_i \mid \alpha_i \gg; P'_i, \quad (s \cup \mathbf{a}, \tau) \longrightarrow^* (s' \cup \mathbf{a}, \epsilon), \quad \theta = \{x \mapsto e(s) \mid \text{“}x := e\text{”} \in \alpha_i,\ i \in Ags\}}{(s, \tau \parallel_{i \in Ags} P_i) \rightarrow (s'\theta, \tau \parallel_i P'_i)}$$

The explanation of this statement is as follows: in the first stage, given a global state s, with its remaining computation represented by program P_i, each agent i first generates an atomic statement $a = \ll \mathbf{a}_i \mid \alpha_i \gg$ as well as a program P'_i, to be run after this atomic statement has executed. This is represented formally by a relation $(s, P_i) \hookrightarrow a; P'_i$. In the second stage, the actions in these statements are then combined into a joint atomic action $\mathbf{a}$, viewed as an assignment making the action variables $i.\mathbf{a}_i$ true for $i \in Ags$, and all other action variables false. This assignment is added to the current global state, and the environment program τ then causes a transition of the environment state, expressed by the statement $(\tau, s \cup \mathbf{a}) \longrightarrow^* (s' \cup \mathbf{a}, \epsilon)$ that represents that the environment program τ, when executed with respect to joint action $\mathbf{a}$, runs to termination having caused the global state to change from s to s' (only the environment variables change during the running of τ). Finally, the local states are updated, by executing the assignments α_i locally at each agent. This is captured by first defining the substitution θ that defines the update to be performed, based on the values $e(s)$ of expressions e in the state s, and then applying that substitution to the global state s' (represented by $s'\theta$).

The relations used above are defined by the following rules:

$$\frac{}{(s, \epsilon) \hookrightarrow skip; \epsilon} \qquad \frac{}{(s, a; P) \hookrightarrow a; P}$$

$$\frac{s \models g_i}{(s, \mathbf{if}\ g_1 \rightarrow a_1\ [] \ldots []\ g_m \rightarrow a_m\ \mathbf{fi}; P) \hookrightarrow a_i; P}$$

$$\frac{s \models \bigwedge_{i \in Ags} \neg g_i}{(s, \mathbf{if}\ g_1 \rightarrow a_1\ [] \ldots []\ g_m \rightarrow a_m\ \mathbf{fi}; P) \hookrightarrow skip; P}$$

$$\frac{(s, P) \hookrightarrow x := e; P', \quad \theta = [x \mapsto e(s)]}{(s, P) \rightarrow (s\theta, P')}$$

(The rules for $\hookrightarrow$ apply to both the environment program and the agent protocols; the last rule applies only to steps of the environment computation.)

We can now define a model $M(\mathcal{S}) = (S, I, \{O_i\}_{I \in Ags}, \rightarrow, \pi)$ for each system model $\mathcal{S}$. The components are given as follows: S is

the set of pairs $(s, \tau\|_{i\in Ags}P_i)$, where s is a global state of $\mathcal{S}$ and each P_i is a protocol for agent i, the set I is the set of pairs $(s, Prog(\mathcal{S}))$ such that $s \models Init_e \wedge \bigwedge_{i\in Ags} Init_i$, the function O_i is defined by $O_i((s, \tau\|_{i\in Ags}P_i)) = s \upharpoonright OVar_i$, the transition relation $\rightarrow$ is as defined above, and π associates each variable with its value, i.e. $v \in \pi(s)$ iff $s(v) = 1$.

Note that, given a view $\mathcal{V}$, we obtain from $M(\mathcal{S})$ an interpreted system $\mathcal{I}^{\mathcal{V}}(M(\mathcal{S}))$. We use this construction of interpreted systems to give semantics to knowledge based programs.

3.3 Knowledge-based protocols

The syntax of knowledge-based protocols is given as a generalization of the definitions above. A *knowledge-based protocol for agent i in environment $\mathcal{M}_e$*, is a tuple

$$\mathtt{P}_i = (PVar_i, LVar_i, OVar_i, Init_i, Acts_i, \mathtt{Prog}_i) ,$$

where the components are exactly as for a protocol for agent i in environment $\mathcal{M}_e$, except that in the program $\mathtt{Prog}_i$, both the guards g in conditional statements and the expressions e in the assignments in atomic statements may be formulas of the logic of knowledge. Figure 2(a) gives an example of such a program, (corresponding to a stage of the well-known "Muddy Children" problem, which we discuss in more detail in Section 5). A *joint knowledge-based protocol* is a tuple $\mathtt{P} = \{\mathtt{P}_i\}_{i\in Ags}$ consisting of a knowledge-based protocol $\mathtt{P}_i$ for each agent i.

To give semantics to knowledge-based protocols, we define a relation of *implementation* between knowledge-based protocols and standard protocols. Intuitively, an implementation is a standard protocol that is structurally similar to the knowledge-based protocol, except that knowledge formulas have been replaced by expressions in the local variables, where such expressions are equivalent to the knowledge formulas. To make sense of this equivalence we need to evaluate the knowledge formulas in an interpreted system: for this we take the system generated by the standard protocol.

We first give the semantics with respect to the clock view. Note that since programs are sequences $stat_1; \ldots; stat_m$ of simple statements, each such simple statement can be associated with a time of occurrence, viz., $stat_i$ occurs at time $i-1$. In case $stat_i$ is a conditional statement **if** $g_1 \rightarrow a_1 \,[]\ldots[]\, g_k \rightarrow a_k$ **fi** we also say that each of the atomic statements a_j occur at time $i-1$. (Intuitively, it takes no time to evaluate the guard g_j.) Given a knowledge-based program $\mathtt{Prog}_i$, we transform it into its *skeleton*, denoted $skell(\mathtt{Prog}_i)$, by replacing each knowledge formula ϕ in a guard g or assigned expression e, occurring at time t, by a new variable v^t_ϕ, whose name indicates both the time t and the formula being replaced. (More precisely, we replace the maximal subformulas ϕ that contain knowledge operators but do not contain "non-observable" variables in $PVar_i \setminus OVar_i$.) Let $skellVar(\mathtt{Prog}_i)$ be the set of such new variables in $skell(Prog_i)$. We define $skell(\mathtt{P}) = \{skell(Prog_i)\}_{i\in Ags}$ and $skellVar(\mathtt{P}) = \cup_{i\in Ags} skellVar(Prog_i)$.

Next, let θ be a substitution mapping each skeleton variable $v^t_\phi \in skellVar(\mathtt{Prog}_i)$, for $i \in Ags$, to a boolean expression on the observable variables of agent i's protocol $\mathtt{P}_i$. If we apply this substitution to $skell(\mathtt{Prog}_i)$, we obtain a standard program $skell(\mathtt{Prog}_i)\theta$. We write $\mathtt{P}_i\theta$ for the result of replacing the knowledge-based program $\mathtt{Prog}_i$ in $\mathtt{P}_i$ by $\mathtt{Prog}_i\theta$. This is a standard protocol for agent i.

Similarly, if $\mathtt{P} = \{\mathtt{P}_i\}_{i\in Ags}$ is a *joint knowledge-based protocol*, and θ is a substitution satisfying the condition above for all agents i, we write $\mathtt{P}\theta$ for the joint standard protocol $\{\mathtt{P}_i\theta\}_{i\in Ags}$. We now define $\mathtt{P}\theta$ to be an *implementation* of the joint knowledge-based protocol $\mathtt{P}$ with respect to the view $\mathtt{clk}$ if $\mathcal{I}^{\mathtt{clk}}(M(\mathcal{M}_e, \mathtt{P}\theta)) \models X^t(\phi \Leftrightarrow \theta(v^t_\phi))$ for all $v^t_\phi \in skellVar(\mathtt{P})$. That is, in the system obtained with respect to the view $\mathtt{clk}$ by running the standard protocol $\mathtt{P}\theta$ in the environment $\mathcal{M}_e$, each knowledge formula ϕ in $\mathtt{P}$ is equivalent to the concrete expression $\theta(v^t_\phi)$ on the local state variables that replaces it in the standard protocol (at the time t that this formula is relevant to the behaviour of the program).

Since the definition of implementation of a knowledge-based program is stated as a constraint on substitutions, it is not clear whether there exist any substitutions satisfying this constraint, or whether such substitutions are unique. The following theorem states that in fact, given our assumptions, there is essentially a unique implementation.

THEOREM 1. *If $\mathtt{P}$ is a joint atemporal knowledge-based protocol for environment $\mathcal{M}_e$, then there exists a substitution θ such that $skell(\mathtt{P})\theta$ is an implementation of $\mathtt{P}$ in $\mathcal{M}_e$ with respect to $\mathtt{clk}$. Moreover, for all substitutions θ, θ' such that $skell(\mathtt{P})\theta$ and $skell(\mathtt{P})\theta'$ are implementations of $\mathtt{P}$ in $\mathcal{M}_e$ with respect to $\mathtt{clk}$, we have that $\mathcal{I}^{\mathtt{clk}}(M(\mathcal{M}_e, \mathtt{P}\theta)) \models X^t(\theta(v^t_\phi) \Leftrightarrow \theta'(v^t_\phi))$ for all $v^t_\phi \in skellVar(\mathtt{P})$.*

The result is similar to a result of [11]. Note that although $\theta(v^t_\phi)$ and $\theta'(v^t_\phi)$ may be distinct formulas, they are equivalent, in the context of any implementation, at the time of their relevance to the behaviour of the implementation. It follows that the systems $\mathcal{I}^{\mathtt{clk}}(M(\mathcal{M}_e, \mathtt{P}\theta))$ and $\mathcal{I}^{\mathtt{clk}}(M(\mathcal{M}_e, \mathtt{P}\theta'))$ are isomorphic with respect to the variables of $\mathcal{M}_e$ and $\mathtt{P}$.

We now consider the synchronous perfect recall semantics. In this case, an agent's knowledge is semantically defined using not just the agent's current observation, but also using its past observations. Implementations of knowledge-based programs with respect to this semantics are therefore permitted to refer to these past observations. To enable this, we first introduce some new "history" variables to represent the past observations, and then state the perfect recall semantics as an application of the clock semantics.

Given a joint knowledge-based program $\mathtt{P}$ of length m, let $\mathtt{P}^h$ be the knowledge based program obtained after making the following modifications to $\mathtt{P}$:

1. if $OVar_i$ is the set of observable variables for agent i, replace this by the set $OVar^h_i = OVar_i \cup \{v@k \mid v \in OVar_i,\ 0 \le k < m\}$;
2. replace $LVar_i$ by $LVar_i \cup \{v@k \mid v \in OVar_i,\ 0 \le k < m\}$;
3. replace each atomic statement $\ll a \mid \alpha \gg$ at time k in $Prog_i$ by the statement $\ll a \mid \alpha, \beta \gg$, where β is the collection of assignments $v@k := v$ for $v \in OVar_i$.[3]

Intuitively, each variable $v@k$ is a new local observable variable that records the value of the original observable variable v of agent i at time k. We now define an implementation of $\mathtt{P}$ in $\mathcal{M}_e$ with respect to the synchronous perfect recall semantics to be an implementation of $\mathtt{P}^h$ in $\mathcal{M}_e$ with respect to the clock semantics. By Theorem 1, such implementations are also guaranteed to exist and are behaviourally unique.

4. SYNTHESIS

The semantics for knowledge-based programs requires that the (semantically unique) implementing substitution θ for all knowledge conditions be given, and then verified for correctness. We now describe an incremental construction of this substitution that serves as the basis for our symbolic synthesis procedure. For the remainder of this section we fix an environment model $\mathcal{M}_e$ and a joint knowledge-based program $\mathtt{P}$. Let N be the maximal time of

[3]Our implementation optimizes this by sharing history of observed environment variables between agents.

(a) **if** $K_i muddy_i \vee K_i \neg muddy_i \rightarrow \ll$ SayYes $\mid \gg$
[] $\neg(K_i muddy_i \vee K_i \neg muddy_i) \rightarrow \ll$ SayNo $\mid \gg$ **fi**

(b) **if** $v^0_{K_i muddy_i \vee K_i \neg muddy_i} \rightarrow \ll$ SayYes $\mid \gg$
[] $v^0_{\neg(K_i muddy_i \vee K_i \neg muddy_i)} \rightarrow \ll$ SayNo $\mid \gg$ **fi**

Figure 2: A knowledge-based program (a) and its skeleton (b)

occurrence of any knowledge condition in P. Let $skell(\mathtt{Prog}_i) = stat^i_1; \ldots stat^i_m$.

For the clock semantics, we work with epistemic Kripke structures $\mathtt{M}(S) = (S, \{O_i\}, \pi)$, where S is a set of assignments to $Var_e \cup \bigcup_{i \in Ags} LVar_i$, the observation functions are just restrictions to the observable variables, i.e., $O_i(s) = s \restriction OVar_i$, and and π is just the trivial interpretation on S, i.e., $v \in \pi(s)$ iff $s(v) = 1$.

In particular, for $k = 0 \ldots N$ we define structures $\mathtt{M}_k = \mathtt{M}(S_k)$ by defining the sets S_k. At the same time, we define the substitution θ. These definitions proceed inductively, as follows. First, we define S_0 to be the set of assignments s such that $s \models Init_e \wedge \bigwedge_{i \in Ags} Init_i$. This determines $\mathtt{M}_0$.

Assuming that $\mathtt{M}_k$ has been constructed, we next define the implementation $\theta(v^k_\phi)$ of each knowledge condition ϕ in $\mathtt{Prog}_i$ at time k. This implementation is required to be a boolean expression over the set $OVar_i$ of observable variables for agent i. Rather than give this formula explicitly, we characterize it by describing the assignments o to these variables on which the formula is satisfied. For $v^k_\phi \in skellVar(\mathtt{Prog}_i)$, we let $\theta(v^k_\phi)$ be any formula such that for all $o = O_i(s)$ with $s \in S_k$, we have $o \models \theta(v^k_\phi)$ iff $\mathtt{M}_k, o \models_i \phi$. (This does not necessarily uniquely define $\theta(v^k_\phi)$ on all possible observations, but leaves some flexibility to optimize the size of the formula by choosing its value appropriately on the "don't-care" observations, applying ideas familiar from digital circuit design theory [21].)

Next, we define

$$S_{k+1} = \{t \mid \exists s \in S_k((s, \tau \parallel_{i \in Ags} stat^i_k\theta) \longrightarrow (t, \tau \parallel_{i \in Ags} \epsilon))\} .$$

That is, we run the k-th step of the knowledge-based programs using the implementations of the knowledge conditions as just defined from $\mathtt{M}_k$, using the operational semantics $\longrightarrow$ for standard programs. (Note that the substitution θ has not yet been completely defined, but it has already been sufficiently defined to provide a value for each v^k_ϕ in $stat^i_k$, so that $stat^i_k\theta$ is a standard program not containing any skeleton variables v^j_ψ.) This now gives the structure $\mathtt{M}_{k+1} = \mathtt{M}(S_{k+1})$.

The following result states that the substitution obtained by this process provides an implementation of P.

THEOREM 2. *Let θ be the substitution defined above. Then* $\mathtt{P}\theta$ *implements* P *in* $\mathcal{M}_e$ *with respect to the view* clk.

The iteration using the epistemic structures $\mathtt{M}_k$ in this construction is a generalization of an algorithm already in use in MCK for model checking standard programs with respect to specifications of the form $X^k\phi$, with ϕ an atemporal formula, interpreted with respect to the clock semantics. In the case of standard programs, the substitution θ is the empty substitution, and the construction simplifies to the existing algorithm in that case. The existing algorithm was already implemented symbolically using BDD's (see section 2.1) to represent the structures M_k, and the implementation is easily generalized to cover the extensions above. The main change is that it is now required at each stage to evaluate the applicable knowledge formulas ϕ in the structures M_k. This is done using an existing algorithm that computes a BDD representation of the set of states of M_k satisfying ϕ, given the BDD representation of M_k. The concrete condition $\theta(v^k_\phi)$ is then extracted as a boolean expression over observable variables that holds in M_k at the same assignments to observable variables as the formula ϕ. (Note that since ϕ is a boolean combination of observable variables and formulas $K_i\psi$, its satisfaction depends only on observable variables.)

Since the semantics of knowledge-based programs with respect to the synchronous perfect recall semantics has been introduced above by means of a reduction to the clock case, we note that we also obtain a procedure for synthesis of implementations with respect to the synchronous perfect recall semantics. The only change required is the introduction of history variables as described above.

5. EXAMPLES

In this section we discuss the performance of the symbolic synthesis approach on a number of simple examples, and compare it to an explicit state approach to a closely related problem.

The explicit state approach is essentially that implemented in DEMO [37], which is the only other model epistemic checker that presently has the expressive power to handle a problem close to the knowledge-based program synthesis problem that our system is able to handle. However, compared to our formulation, DEMO does not include knowledge-based programs as an explicit construct, it does not attempt to synthesize a concrete implementation of such a programs, and it can handle only situations where the atomic propositions do not change value over time.

DEMO deals with the evaluation of statements $(M, S) \models [U, T]\phi$, where M is an epistemic model, S is a set of states of that model, U is an epistemic update and T is a set of states of U. More precisely, $M = (W, \{\sim_i\}_{i \in Ags}, \pi)$ where W is a set of worlds, each $\sim_i$ is an equivalence relation on W, and $\pi : W \rightarrow Prop$ is an interpretation of the atomic propositions. The update structure U has the form $(E, \{\sim^E_i\}_{i \in Ags}, pre)$, where E is a set of *events*, each $\sim^E_i$ is an equivalence relation on E representing events that agent i is not able to distinguish, and pre maps E to formulas (since it is all we will need, we assume here that these formulas are atemporal but possibly epistemic formulas in our language). Intuitively, $pre(e)$ is a pre-condition for the occurrence of event e. The *update of M by U* is then defined to be the epistemic structure $M \circ U = (W', \{\sim'_i\}, \pi')$ where $W' = \{(w, e) \in W \times E \mid M, w \models pre(e)\}$, the relation $\sim'_i$ is defined by $(w_1, e_1) \sim'_i (w_2, e_2)$ if $w_1 \sim_i w_2$ and $e_1 \sim^E_i e_2$, and $\pi'(w, e) = \pi(w)$. The statement $(M, S) \models [U, T]\phi$, where $S \subseteq W$ and $T \subseteq E$, holds just when $M \circ U, (w, e) \models \phi$ for all $w \in S$ and $e \in T$ with $w \models pre(e)$.

In the special case where actions do not change the values of propositions (one example where this holds is the Muddy Children problem, discussed below) we can encode each stage of a knowledge-based program as an update. Suppose that each agent $i = 1 \ldots n$ has atomic statement

$$\textbf{if } g^i_1 \rightarrow a^i_1 [] \ldots [] g^i_{k_i} \rightarrow a^i_{k_i} \textbf{ fi}$$

where the g^i_j are atemporal epistemic formulas. Then the parallel composition of these statements corresponds to a set $E = \Pi^n_{i=1}\{1 \ldots k_i\}$ with $pre(j_1, \ldots j_n) = \bigwedge_{i=1 \ldots n} g^i_{j_i}$. If the effect of the actions a^i_j on observable variables (as described in the environment model) can be captured by indistinguishability relations on E, then we can encode the stage of the knowledge-based program as an update.

One difference is immediately apparent however: in DEMO, the epistemic model M, the update structure U, and the structure $M \circ U$ are all represented by an explicit enumeration of their states. Be-

```
muddy: Bool[Agent]
info: Bool[Agent]

init_cond = (Exists x:Agent() (muddy[x])) /\ Forall x:Agent() (info[x] == muddy[x])

agent Child0 "child" ( info[Child1], info[Child2], info[Child3] )
agent Child1 "child" ( info[Child2], info[Child3], info[Child0] )
agent Child2 "child" ( info[Child3], info[Child0], info[Child1] )
agent Child3 "child" ( info[Child0], info[Child1], info[Child2] )


transitions
begin
info[Child0] := Child0.SayYes; info[Child1] := Child1.SayYes;
info[Child2] := Child2.SayYes; info[Child3] := Child3.SayYes
end

protocol "child" ( info1: observable Bool, info2: observable Bool, info3: observable Bool )
begin
 if (Knows Self muddy[Self]) \/ (Knows Self neg muddy[Self]) -> << SayYes >>
 [] otherwise -> skip  fi;
 if (Knows Self muddy[Self]) \/ (Knows Self neg muddy[Self]) -> << SayYes >>
 [] otherwise -> skip  fi;
 if (Knows Self muddy[Self]) \/ (Knows Self neg muddy[Self]) -> << SayYes >>
 [] otherwise -> skip  fi;
 if (Knows Self muddy[Self]) \/ (Knows Self neg muddy[Self]) -> << SayYes >>
 [] otherwise -> skip  fi
end
```

Figure 3: A knowledge-based program for muddy children (perfect recall version)

cause of the cartesian products in the definition, the size of these state spaces potentially grows exponentially in the number of agents (and the number of updates applied to an initial structure), although DEMO applies a quotient under a maximal bisimulation that may reduce the size of these spaces in some cases. Our symbolic representation, on the other hand, has the potential to avoid this exponential blow-up. (The benefit is only potential because BDD representations, though they often prove to be small in practice, are also not guaranteed to be small in all cases.) It is therefore interesting to investigate whether this potential benefit is realized in interesting cases. We now explore this question for the well-known Muddy children problem.

5.1 Muddy children

The muddy children puzzle [10] can be stated as follows:

> A group of n children have been playing outside, and some have mud on their foreheads. Each child can see the forehead of the others but cannot see his or her own forehead. Father says to group, “At least one of you has mud on your forehead". He then repeated asks following question: “Do you know whether or not you have mud on your forehead?" The children give their answers (‘Yes" or “No") simultaneously each time the question is asked, and each child observes the answers given by the other children.

Assuming that the children are perfect reasoners, have perfect recall and are honest, the expected behaviour is that if k out of n of the children are muddy, then all children will answer “No" until round k, in which all the muddy children answer “Yes” and the clean children answer “No". We note also that from round $k + 1$ all children answer “Yes".

The puzzle can be represented as a knowledge-based program. Figure 3 gives the representation of the environment and the children’s protocol in the concrete syntax of our MCK implementation, for the perfect recall case with $n = 4$. Father’s statement is captured by means of the statement `init_cond`, which defines the set of initial states. Since these are common knowledge, it is initially common knowledge that there is at least one muddy child. (The existential/universal quantifiers are restricted to finite types and are just a syntactic sugar for disjunction/conjunction.) An agent’s observable variables $OVar_i$ are declared using the keyword `observable`. Each observable variable adds complexity to the BDD computation and, in the perfect recall semantics, is moreover replicated at each moment of time. To minimize these costs, we use a variable `info[x]` that represents the new information concerning agent x at each step. Initially this variable is used to represent whether agent x is muddy, and at later steps it represents whether agent x has just said “Yes". Each agent observes all the variables `info[x]` for the *other* agents (it can always deduce whether it would itself have said “Yes" at the previous step).

Symmetry of the children’s behaviour is handled by giving a general description, the knowledge-based protocol `"child"`. The knowledge-based program consists of a repeated sequence of `if` statements, in which the keyword `otherwise` represents the negation of the disjunction of all the preceding guards. The `agent` statements create fresh instances of the general protocol, in which the the parameters of the instance are aliased to the corresponding environment variables, and the keyword `Self` in the protocol is interpreted as the agent being defined. In particular, each such statement says that a child can observe (via the observable variables $\texttt{see}_i$) the new information about the other children. The environment’s transitions clause simply stores the childrens’ answers to the boolean variables `info[x]` for $x \in Ags$.

The representation we use for a clock semantics version is slightly different. Here, we cannot rely on agents to remember whether they initially observed other children to be muddy, or whether they said “Yes" in the previous step. We therefore make observable to all agents an array `said`, representing the previous statements of all the agents, and also include an observable variable for the muddiness of each other child. Interpreting this version with respect to the clock semantics yields exactly the same behaviour of the children in the implementation as the perfect recall version, viz., if there are k muddy children then these children first say “Yes" at stage k.

To represent the puzzle in DEMO, each step of the knowledge based program needs to be represented as an update structure $U = (E, \{\sim_i\}_{i\in Ags}, pre)$, in which the events E correspond to possible observations that are made by the children after they reply to father's question. Thus, for n agents, the set of events $E = \{0,1\}^n$, with $e \sim_i e'$ iff $e = e'$, and for each tuple $e = (e_1, \ldots, e_n) \in E$ we have that $pre(e) = \bigwedge_{i=1}^{n} \phi_i$ where $\phi_i = (K_i\texttt{muddy[i]}) \vee (K_i\neg\texttt{muddy[i]})$ if $e_i = 1$ and $\phi_i = \neg((K_i\texttt{muddy[i]}) \vee (K_i\neg\texttt{muddy[i]}))$ if $e_i = 0$. Since DEMO runs in the Haskell interpreter, it is possible to represent U succinctly as a Haskell program, but to perform the update calculation DEMO needs to construct the set E explicitly, so necessarily performs an exponential amount of work.

The experimental results[4] comparing the performance of our symbolic approach to the DEMO modelling are shown in Table 1. In addition to synthesis, we check, for the MCK program, the formula $X^n\phi$ where $\phi = \bigwedge_{i\in Ags}(K_i\texttt{muddy[i]} \vee K_i\neg\texttt{muddy[i]})$ which expresses that after n rounds, all the children know whether they are muddy. For the DEMO program, we check ϕ after updating n times.

Note that for n muddy children, we are dealing with an initial state space of $2^n - 1$ states and a deterministic solution protocol that runs for n steps, giving $n \cdot (2^n - 1)$ points in the relevant part of the interpreted system. The results demonstrate that our symbolic approach does in practice scale significantly better when dealing with this exponentially growing problem, particularly in the case of the clock semantics version. (Recall that the synthesized behaviour is exactly the same as for the perfect recall interpretation.) DEMO's performance rapidly degrades as the number of agents increases, whereas our symbolic approach to the clock semantics is able to very efficiently handle problems of larger scale. On the other hand, DEMO is implicitly computing the perfect recall solution, so an arguably the fairer comparison is with the MCK perfect recall model. Here too our approach scales better, e.g., handling 10 agents in time comparable to DEMO's time for 7 agents. However, after initially lagging DEMO's explicit state approach, the total running time for the larger cases that can be handled becomes more than two orders of magnitude better.

5.2 Leader Election

The second example we consider concerns maintaining knowledge of the leader on a ring of agents. We suppose that there are n agents numbered $i = 1 \ldots n$, with agent i able to send messages to agent $(i \mod n)+1$. Each agent has an observable input buffer that is able to store one message. The agent also observes its own agent number. An agent can crash at any time, and once crashed, remains crashed. The leader at time t is defined as the highest numbered agent that has not crashed by time t (and otherwise 0).

In each round, the environment first crashes a subset of the agents. All uncrashed agents may send a message. The network delivers any message that an agent is trying to send to the intended recipient, provided that the sender has not crashed. If agent i has crashed, then the network detects this, and in place of the message that agent i would have sent, the network delivers the message that was in agent i's buffer to agent $(i \mod n) + 1$. (Intuitively, if the network cannot deliver a message to an agent it sends it to the next agent in the ring.) Each message is reliably marked with a "from" field, so that the recipient can determine the original sender of the message. (Note that this means also that when it receives a message that is marked as from an agent other than the ("lower" numbered) agent to its left, an agent can deduce that all agents between itself and the original sender of the message have crashed.)

Note that the definition of the leader is a global property. Since it takes at least n rounds of communication for any information about a node to reach all other nodes, if the leader crashes then another distant non-crashed agent cannot know about the crash for several steps: agents distant from the leader therefore cannot know whether the actual leader is still alive, so cannot know who is the leader. We therefore focus on a weaker property than knowledge of who is the leader. We say that agent i's presumed leader is the largest agent number m_i for which agent i considers it possible that m_i is the leader. To help acquire and spread this knowledge, the agents inform each other about their presumed leader: in each round, each (noncrashed) agent i sends its neighbour a message "*from i: j*" stating that its presumed leader is j.

Figure 4 shows our MCK representation of the knowledge-based protocol. (For space reasons we omit the program for the environment.) To help identify crashes in the first step, we set the message in agent i's buffer at time 0 to be *from i:* 0. The atomic statement $\ll$ `Sendj | presumed := j` $\gg$, performed by agent i, has the effect (encoded in the program for the environment) of causing the message "*from i: j*" to be delivered to the right neighbour of agent i provided that the agent has not crashed. The assignment to local variable `presumed` stores the current presumed value.

It can be seen that the state space for this problem grows rapidly: since a run is determined by the time at which each agent crashes, if at all, for n agents there are $(k + 1)^n$ runs of length k. Table 2 shows the performance of our symbolic synthesis procedure as we increase the number of steps of the protocol. (We do not give a comparison to DEMO. This problem is beyond the scope of DEMO, because it handles only static propositions, whereas in this problem the propositions change value over time.)

We have confirmed by model checking a manual solution for the 3 agent case that that under both the clock and perfect recall semantics, at each step, an agent knows that the leader is not $A3$ just when one of the following holds:

1. it knew this already in the previous step, because it already had `presumed` < 3,
2. it receives a message from another agent that must have passed through a chain of failures including A3, or
3. it receives a message "from j: y" with $y < 3$, which implies that agent j knows that $A3$ is not the leader, or
4. it receives the message "from 3: 0", which implies that A3 failed in the first step.

Essentially the same predicate (with 2 in place of 3) captures the circumstances under which an agent knows that the leader is not 2, provided it also knows that the leader is not 3.

Manually verifying a nonterminating protocol that uses the above predicates at each step to determine the current presumed leader, we can verify that the following properties holds in the resulting protocol (for the case of 3 agents): at all times all noncrashed agents are greater than or equal to the actual leader, and if there are no more crashes after time t, then within 2 steps all non-crashed agent's presumed leaders are the same as the actual leader.

6. RELATED WORK

Our focus in this paper has been on the pragmatics of knowledge-based program syntax and on synthesis using a particular data structure for the symbolic representation of knowledge-based program implementations. A number of works have approached the problem

[4]Our experiments were conducted on a Ubuntu Linux system (3.06GHz Intel Core i3 with 4G memory). Each process is allocated up to 500M memory.

```
protocol "elect" (crashed : Bool,  my_num: observable LeaderNum,
                  from_field: observable LeaderNum, message: observable LeaderNum)

presumed: LeaderNum

init_cond = presumed == 3

begin
if  (neg crashed) /\ neg Knows  Self neg leader == 3 ->  <<Send3 | presumed := 3 >>
[] (neg crashed) /\ (Knows  Self neg leader == 3 )
                 /\ neg Knows  Self neg leader == 2 ->  <<Send2 | presumed := 2 >>
[] (neg crashed) /\ (Knows  Self neg leader == 3 )
                 /\ (Knows  Self neg leader == 2 )
                 /\  neg Knows  Self neg leader == 1 -> <<Send1 | presumed := 1 >>
[] otherwise -> skip fi;
(repeat if statement)
end
```

Figure 4: A knowledge-based protocol for leader election

No. of Children	4	5	6	7	8	9	10
DEMO	0.54	5.79	71.11	897.28	9,995.10	> 36,000	
MCK `clk`	0.32	0.88	2.03	6.32	9.09	20.23	57.30
MCK `spr`	1.14	5.96	13.13	58.92	96.12	484.22	1,239.60

Table 1: Running Times (seconds) of Muddy Children

No. of Agents, Semantics	Length of Run								
	2	3	4	5	6	7	8	9	10
3, MCK clk	2.35	4.42	8.58	9.95	21.42	29.56	28.39	33.92	35.58
3, MCK spr	11.26	10.43	63.01	170.16	1,607.82	5,971.44	26,624.59	> 36,000	

Table 2: Performance of synthesis on election protocol (seconds)

of constructing implementations from a more theoretical perspective.

Besides identifying the synchronous atemporal case, that we have treated here, as one in which unique implementations exist, in [11] it is shown that deciding the existence of an implementation with respect to the observational view in a finite state environment is PSPACE complete, even when the knowledge conditions are expressed using linear time temporal logic operators. Since model checking LTL is also PSPACE complete but is still considered practical, this might suggest that this knowledge-based program implementation problem should also be tractable; unfortunately the algorithm in question requires guessing an implementation from an exponentially large set and then verifying it, so it is not clear that this is the case.

We have focussed on programs of bounded length. It is shown in [32] that the problem of determining whether an atemporal formula of the form $K_i\phi$ where ϕ is propositional, holds at a given view of length n in the implementation of a knowledge-based program with respect to the synchronous perfect recall view can be as hard as PSPACE-complete. Besides indicating that we cannot expect to always obtain tractable implementations in the perfect recall case even for programs of bounded length, this result also has implications for nonterminating knowledge-based programs: it implies that implementations of such programs are not finite state encodable in general. However, this does not preclude the practicality of synthesis in particular cases.

For example, it is shown in [31] that finite state implementations of nonterminating knowledge-based programs are guaranteed to exist in the case of the clock view, as well as broadcast environments and environments with a single agent with synchronous perfect recall. A formal verification of these results is described in [12]. The implementation approach we have considered in the present paper can in principle be extended to construct such implementations, but we have not yet experimented with this.

A general scheme that constructs a finite state implementation with respect to the perfect recall semantics in the (undecidable) situation that one exists is described in [33]. The construction exploits a quotient by the maximal bisimulation on temporal slices, that is similar to the optimization used in the DEMO implementation. We refer to Section 5 for a comparison of our approach to the epistemic update logic problem considered in DEMO.

A number of papers have also applied model checking of knowledge properties to synthesize distributed control strategies [4, 14, 22]. However, the approach taken in these works is weaker than that in knowledge-based programs. Roughly, it corresponds to taking just one iteration of the fixpoint operator for a knowledge-based program, so that it is not guaranteed that the implementing condition is equivalent to the desired knowledge property in the protocol synthesized.

7. CONCLUSION

Our contribution in this paper has been to take the first step towards the goal of a practical tool, based on symbolic methods, for knowledge-based program implementation. We have demonstrated that the approach works on two modest scale examples. In future work, we plan to undertake further application case studies. We also intend to develop optimizations of our initial implementation: we believe that many avenues remain open for improvement of the performance of our system. We also plan to extend it in directions such as handling non-termination and probabilistic knowledge.

8. REFERENCES

[1] O. A. Bataineh and R. van der Meyden. Epistemic model checking for knowledge-based program implementation: an

application to anonymous broadcast. In *SecureComm'10, 6th Int. ICST Conf. on Security and Privacy in Communication Networks*, 2010.

[2] O. A. Bataineh and R. van der Meyden. Abstraction for epistemic model checking of dining-cryptographers based protocols. In *Proc. Conf. on Theoretical Aspects of Knowledge and Rationality*, 2011.

[3] K. Baukus and R. van der Meyden. A knowledge based analysis of cache coherence. In *Proc. 6th Int. Conf. on Formal Engineering Methods*, pages 99–114, 2004.

[4] S. Bensalem, D. Peled, and J. Sifakis. Knowledge based scheduling of distributed systems. In *Time for Verification, Essays in Memory of Amir Pnueli*, volume 6200 of *Lecture Notes in Computer Science*, pages 26–41. Springer, 2010.

[5] G. Berry and G. Gonthier. The esterel synchronous programming language: Design, semantics, implementation. *Sci. Comput. Program.*, 19(2):87–152, 1992.

[6] R. Bryant. Graph-based algorithms for boolean function manipulation. *IEEE Transactions on Computers*, C-35(8):677–691, 1986.

[7] K. M. Chandy and J. Misra. *Parallel Program Design: A Foundation.* Addison-Wesley, 1988.

[8] C. Dwork and Y. Moses. Knowledge and common knowledge in a Byzantine environment : crash failures. In *Proc. Conf. on Theoretical Aspects of Reasoning about Knowledge*, pages 149–169, 1986.

[9] C. Dwork and Y. Moses. Knowledge and common knowledge in a Byzantine environment: crash failures. *Information and Computation*, 88(2):156–186, 1990.

[10] R. Fagin, J. Y. Halpern, Y. Moses, and M. Y. Vardi. *Reasoning about Knowledge*. MIT Press, Cambridge, Mass., 1995.

[11] R. Fagin, J. Y. Halpern, Y. Moses, and M. Y. Vardi. Knowledge-based programs. *Distributed Computing*, 10(4):199–225, 1997.

[12] P. Gammie. Verified synthesis of knowledge-based programs in finite synchronous environments. In *Proc. 2nd Int. Conf on Interactive Theorem Proving*, pages 87–102, 2011.

[13] P. Gammie and R. van der Meyden. MCK: Model checking the logic of knowledge. In *Proc. 16th Int. Conf. on Computer Aided Verification (CAV'04)*, pages 479–483, 2004.

[14] S. Graf, D. Peled, and S. Quinton. Achieving distributed control through model checking. *Formal Methods in System Design*, 40(2):263–281, 2012.

[15] V. Hadzilacos. A knowledge-theoretic analysis of atomic commitment protocols. In *PODS '87: Proc. 6th ACM Symp. on Principles of Database Systems*, pages 129–134, 1987.

[16] J. Y. Halpern and R. Fagin. Modelling knowledge and action in distributed systems. *Distributed Computing*, 3(4):159–177, 1989.

[17] J. Y. Halpern and Y. Moses. Knowledge and common knowledge in a distributed environment. *J. ACM*, 37(3):549–587, 1990.

[18] J. Y. Halpern and S. Petride. A knowledge-based analysis of global function computation. *Distributed Computing*, 23(3):197–224, 2010.

[19] J. Y. Halpern and L. D. Zuck. A little knowledge goes a long way: knowledge-based derivations and correctness proofs for a family of protocols. *J. ACM*, 39(3):449–478, 1992.

[20] M. Kacprzak, W. Nabialek, A. Niewiadomski, W. Penczek, A. Pólrola, M. Szreter, B. Wozna, and A. Zbrzezny. Verics 2007 - a model checker for knowledge and real-time. *Fundam. Inform.*, 85(1-4):313–328, 2008.

[21] M. Karnaugh. The map method for synthesis of combinational logic circuits. *Trans. of the American Institute of Electrical Engineers, part I*, 72(9):593âĂŞ599, 1953.

[22] G. Katz, D. Peled, and S. Schewe. Synthesis of distributed control through knowledge accumulation. In *Proc. Int. Conf on Computer Aided Verification*, pages 510–525, 2011.

[23] A. Lomuscio, H. Qu, and F. Raimondi. MCMAS: A model checker for the verification of multi-agent systems. In *Proc. Conf. on Computer Aided Verification*, pages 682–688, 2009.

[24] X. Luo, K. Su, M. Gu, L. Wu, and J. Yang. Symbolic model checking the knowledge in herbivore protocol. In *Proc. Model Checking and Artificial Intelligence*, 2010.

[25] M. S. Mazer. A knowledge theoretic account of recovery in distributed systems: The case of negotiated commitment. In *Proc. Conf. on Theoretical Aspects of Rationality and Knowledge,*, pages 309–323, 1988.

[26] Y. Moses and B. Bloom. Knowledge, timed precedence and clocks (preliminary report). In *Proc. IEEE Symp. on Principles of Distributed Computing*, pages 294–303, 1994.

[27] Y. Moses and M. R. Tuttle. Programming simultaneous actions using common knowledge. *Algorithmica*, 3:121–169, 1988.

[28] G. Neiger and S. Toueg. Simulating synchronized clocks and common knowledge in distributed systems. *J. ACM*, 40(2):334–367, 1993.

[29] R. Rudell. Dynamic variable ordering for ordered binary decision diagrams. In *Proc. IEEE/ACM Int. Conf. on Computer-aided design*, 1993.

[30] K. Su, G. Lv, and Y. Zhang. Model checking time and knowledge(mctk). http://www.cs.sysu.edu.cn/~skl/emck.html.

[31] R. van der Meyden. Finite state implementations of knowledge-based programs. In *Proc. Conf. on Foundations of Software Technology and Theoretical Computer Science*, pages 262–273, 1996.

[32] R. van der Meyden. Knowledge based programs: On the complexity of perfect recall in finite environments. In *Proc. Conf. on Theoretical Aspects of Rationality and Knowledge,*, pages 31–49, 1996.

[33] R. van der Meyden. Constructing finite state implementations of knowledge-based programs with perfect recall. In *PRICAI Workshop on Intelligent Agent Systems (1996)*, volume 1209 of *LNCS*, pages 135–151. Springer, 1997.

[34] R. van der Meyden and N. V. Shilov. Model checking knowledge and time in systems with perfect recall (extended abstract). In *Proc. Conf. on Foundations of Software Technology and Theoretical Computer Science*, pages 432–445, 1999.

[35] R. van der Meyden and K. Su. Symbolic model checking the knowledge of the dining cryptographers. In *Proc. 17th IEEE Computer Security Foundations Workshop*, pages 280–291, 2004.

[36] R. van der Meyden and T. Wilke. Preservation of epistemic properties in security protocol implementations. In *Proc. Conf. on Theoretical Aspects of Rationality and Knowledge*, pages 212–221, 2007.

[37] J. van Eijck. Dynamic epistemic modelling. Technical report, Centrum voor Wiskunde en Informatica, Amsterdam, 2004. CWI Report SEN-E0424.

Epistemic Logic for Communication Chains

Jeffrey Kane

Department of Mathematics
and Computer Science
McDaniel College
Westminster, Maryland, USA
jmk001@mcdaniel.edu

Pavel Naumov

Department of Mathematics
and Computer Science
McDaniel College
Westminster, Maryland, USA
pnaumov@mcdaniel.edu

ABSTRACT

The paper considers epistemic properties of linear communication chains. It describes a sound and complete logical system that, in addition to the standard axioms of S_5 in a multi-modal language, contains two non-trivial axioms that capture the linear structure of communication chains.

1. INTRODUCTION

In this paper we study epistemic properties of linear communication protocols that we call *communication chains*. An example of such a protocol is the Telephone game[1] depicted in Figure 1: person P picks a random four-letter word a and communicates it to Q. Person Q changes at most one letter in a, and communicates it to person R as b. Finally, R again changes at most one letter in b and communicates it to S as c. For instance, sequence (a, b, c) could be $(byte, bite, cite)$. We refer to such a sequence as a *run* of the protocol.

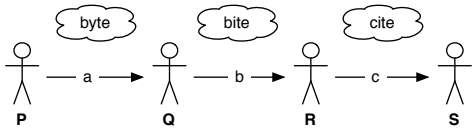


Figure 1: Telephone Game.

Note that anyone who knows the value of message a on the run $r_1 = (byte, bite, cite)$ will be able to conclude that $c \neq book$. We say that channel a on run r_1 "knows" that $c \neq book$ and write it as $r_1 \Vdash \Box_a(c \neq book)$. Note also that anyone who knows the value of a on the run r_1 will also be able to conclude that anyone knowing the value of b on the same run will be able to conclude that $c \neq book$. We write this as $r_1 \Vdash \Box_a\Box_b(c \neq book)$.

Formulas that are true on one run might not be true on another run of the same protocol. For example, if $r_2 = (toon, torn, tort)$, then $r_2 \nVdash \Box_a(c \neq book)$ since a person who only knows the value of a on run r_2 cannot distinguish this run from $(toon, boon, book)$. One can consider statements that are true on any run of the Telephone game protocol. Examples of such statements are:

$$\Box_b(a \neq book) \to \Box_b(c \neq book),$$

$$\Box_a(c \neq book) \to \Box_b(c \neq book).$$

[1]This game is also known as Chinese Whispers, Grapevine, Broken Telephone, Whisper Down the Lane, and Gossip.

The first of these statements is true due to the symmetry of the Telephone game: if (a, b, c) is a run then (c, b, a) is also a run. This property is not necessarily true for all protocols. The second statement, although it is written in the language specific to the Telephone game, can be stated in the form which is true on each run of each protocol over the communication chain depicted in Figure 1:

$$\Box_a p_c \to \Box_b p_c, \tag{1}$$

where p_c is an arbitrary atomic proposition about the value of the message c. In this paper we study that type of "universal" statements that are true on each run of each protocol.

As we will see later, runs can be viewed as Kripke worlds, so all formulas provable in multi-modal version of S_5 are "universal" statements in our sense. In addition to S_5 theorems, however, our logical system included many facts that reflect the linear structure of the communication chain. The above formula (1) is one of them. Other, less obvious examples are:

$$\Box_a\Diamond_c\varphi \to \Box_b\Diamond_c\varphi,$$

$$\Box_a\Box_c\varphi \to \Box_a\Box_b\Box_c\varphi,$$

$$\Box_b(\Box_a\varphi \vee \Box_c\psi) \to (\Box_b\varphi \vee \Box_b\psi),$$

where φ and ψ are arbitrary formulas and $\Diamond_c$, as usual in modal logic, stands for $\neg\Box_c\neg$. We will prove soundness of these principles in Section 4.

The main result of this paper is a sound and complete axiomatization of all properties that are true on each run of each protocol of a given communication chain.

A communication chain can also be interpreted as a timeline. Then, formula $\Box_k\varphi$ means that anyone, who has complete information about a moment k in history, knows that φ is true. For example, one can say,

$$\Box_{2012}(\text{In the past, dinosaurs roamed the Earth}) \to$$
$$\Box_{2011}(\text{In the past, dinosaurs roamed the Earth}).$$

This interpretation connects our work with other works on axiomatizations reasoning about time [2, 3, 8, 10, 11]. These works, however, are very different from ours in the syntax and semantics that they use. Properties like the the three formulas above cannot be expressed in their language.

Epistemic logic for reasoning about communication graphs, in a language significantly different from ours, was proposed by Pacuit and Parikh [9]. They prove decidability of their logical system, but do not give a complete explicit axiomatization.

This work is also connected to works on information flow on graphs [1, 4, 5, 6, 7], that study properties of nondeducibility, functional dependency, and fault tolerance predicates. Unlike these works, this paper is using modal language. We discuss possible generalization of our work to arbitrary communication graphs in the conclusion.

2. SYNTAX AND SEMANTICS

In the informal discussion above, we have implicitly assumed that communication chains have finite length. In the formal presentation through the rest of the paper we consider infinite chains whose communication channels are labeled by consecutive integer numbers (see Figure 2). This is

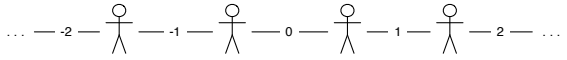


Figure 2: Infinite Chain.

done in order to simplify our presentation. Our results still hold for finite chains. Furthermore, any finite chain can be viewed as an infinite chain in which a fixed default message is sent through a cofinite number of channels.

We also assume that for each $k \in \mathbb{Z}$ there is a (possibly infinite) set P_k of "atomic propositions" about channel k and that sets P_k and P_m are disjoint for each $k \neq m$.

Next we define formulas in our language. The set of all formulas will be denoted by $\Phi(\mathbb{Z})$. By $\Phi(A)$ we denote the set of formulas whose "outermost" modalities have form $\Box_k$ for some $k \in A$ and "outermost" atomic propositions belong to P_k for some $k \in A$. Thus, for example,

$$\Box_k(\Box_m\varphi \rightarrow \Box_n\psi) \in \Phi(\{k\})$$

$$\Box_m\Box_k\varphi \rightarrow \Box_n\Box_k\psi \in \Phi(\{m,n\}).$$

DEFINITION 1. *For each $A \subseteq \mathbb{Z}$, set $\Phi(A)$ is the minimal set of formulas such that*

1. $\bot \in \Phi(A)$,
2. $P_k \subseteq \Phi(A)$, *for each* $k \in A$,
3. *if* $\varphi \in \Phi(A)$ *and* $\psi \in \Phi(A)$, *then* $\varphi \rightarrow \psi \in \Phi(A)$.
4. *if* $\varphi \in \Phi(\mathbb{Z})$, *then* $\Box_k\varphi \in \Phi(A)$, *for each* $k \in A$.

We assume that the boolean connectives $\wedge$, $\vee$, and $\neg$ are defined through $\rightarrow$ and $\bot$ in the standard way. As common in modal logic, by $\Diamond_k\varphi$ we denote formula $\neg\Box_k\neg\varphi$.

In the Telephone game example in the introduction, we have assumed that all messages are four-letter words. In general, we will allow each channel k to have its own set of possible values V_k. In the same example, we have assumed that each person changes at most one letter in the word. In general, we assume that there are *local conditions* that specify relations between values of the adjacent channels. In addition, for any $k \in \mathbb{Z}$, any $v \in V_k$, and any $p \in P_k$, we use predicate $Tr(v,p)$ to specify if an atomic proposition p is "true" when the value of the channel k is v.

DEFINITION 2. *A triple $(\{V_k\}_{k\in\mathbb{Z}}, \{L_k\}_{k\in\mathbb{Z}}, Tr)$ is called a protocol if*

1. *V_k is an arbitrary set (of "values"), for each $k \in \mathbb{Z}$.*
2. *$L_k \subseteq V_{k-1} \times V_k$ is an arbitrary ("local condition") predicate, for each $k \in \mathbb{Z}$.*
3. *Tr is a binary predicate such that $Tr \subseteq \bigcup_{k\in\mathbb{Z}}(V_k \times P_k)$.*

DEFINITION 3. *For any protocol $(\{V_k\}_{k\in\mathbb{Z}}, \{L_k\}_{k\in\mathbb{Z}}, Tr)$, a run is an arbitrary function $r(k)$ on $\mathbb{Z}$ such that $r(k) \in V_k$ and $(r(k-1), r(k)) \in L_k$ for each $k \in \mathbb{Z}$.*

Next is the core definition of this paper. It formally defines the semantics of the modality $\Box_k$.

DEFINITION 4. *For any given protocol*

$$\mathcal{P} = (\{V_k\}_{k\in\mathbb{Z}}, \{L_k\}_{k\in\mathbb{Z}}, Tr),$$

we define relation $\Vdash$ between an arbitrary run r of the protocol $\mathcal{P}$ and an arbitrary formula $\varphi \in \Phi(\mathbb{Z})$ as follows:

1. $r \nVdash \bot$,
2. $r \Vdash p$ *if* $Tr(r(k), p)$, *where* $p \in P_k$.
3. $r \Vdash \varphi \rightarrow \psi$ *if* $r \nVdash \varphi$ *or* $r \Vdash \psi$,
4. $r \Vdash \Box_k\varphi$ *if* $r' \Vdash \varphi$ *for each* r' *such that* $r'(k) = r(k)$.

Note that relation $r'(k) = r(k)$ between runs r' and r is an equivalence relation. Thus, the set of all runs of any given protocol acts as a set of possible worlds of an S_5 Kripke frame.

3. AXIOMS

Our logical system is an extension of the multi-modal version of S_5 by additional properties that deal with atomic propositions and topological structure of the communication chain. As will be shown in the next section, the traditional transitivity and S_5 axioms of the modal logic S_5 follow from a more general[2] Self-Awareness axiom below.

1. Distributivity: $\Box_k(\varphi \rightarrow \psi) \rightarrow (\Box_k\varphi \rightarrow \Box_k\psi)$,
2. Reflexivity: $\Box_k\varphi \rightarrow \varphi$,
3. Self-Awareness: $\varphi \rightarrow \Box_k\varphi$, where $\varphi \in \Phi(\{k\})$,
4. Gateway: $\Box_k\varphi \rightarrow \Box_n\varphi$, where $\varphi \in \Phi(A)$ and either $k < n \leq min(A)$ or $max(A) \leq n < k$,
5. Disjunction: $\Box_k(\varphi \vee \psi) \rightarrow \Box_k\varphi \vee \Box_k\psi$, where $\varphi \in \Phi(A)$, $\psi \in \Phi(B)$, and $max(A) \leq k \leq min(B)$.

We write $\vdash \varphi$ if $\varphi \in \Phi(\mathbb{Z})$ is provable from the axioms above and propositional tautologies in the language $\Phi(\mathbb{Z})$ using the Modus Ponens inference rule and the Necessitation inference rule:

$$\frac{\varphi}{\Box_k\varphi}.$$

We write $X \vdash \varphi$ if φ is provable from the theorems of our system and the additional set of axioms X using only Modus Ponens inference rule.

[2]The Self-Awareness axiom includes, for example, the principle $p \rightarrow \Box_k p$ for $p \in P_k$, which is not provable in S_5.

4. EXAMPLES

Soundness of our logical system will be shown in the next section. Here we give several examples of proofs in our formal system.

PROPOSITION 1 (TRANSITIVITY). *$\vdash \Box_k\varphi \to \Box_k\Box_k\varphi$ for each $\varphi \in \Phi(\mathbb{Z})$ and each $k \in \mathbb{Z}$.*

PROOF. Note that $\Box_k\varphi \in \Phi(\{k\})$. Thus, by the Self-Awareness axiom, $\vdash \Box_k\varphi \to \Box_k\Box_k\varphi$. □

PROPOSITION 2 (S5 AXIOM). *$\vdash \Diamond_k\varphi \to \Box_k\Diamond_k\varphi$, for each $\varphi \in \Phi(\mathbb{Z})$ and each $k \in \mathbb{Z}$.*

PROOF. Note that $\Diamond_k\varphi \in \Phi(\{k\})$. Thus, by the Self-Awareness axiom, $\vdash \Diamond_k\varphi \to \Box_k\Diamond_k\varphi$. □

PROPOSITION 3. *If $k \le m \le n$ and $\varphi \in \Phi(\mathbb{Z})$, then*

$$\vdash \Box_k\Diamond_n\varphi \to \Box_m\Diamond_n\varphi.$$

PROOF. Note that $\Diamond_n\varphi \in \Phi(\{n\})$. Thus, by the Gateway axiom, $\vdash \Box_k\Diamond_n\varphi \to \Box_m\Diamond_n\varphi$. □

PROPOSITION 4. *If $k \le m \le n$ and $\varphi \in \Phi(\mathbb{Z})$, then*

$$\vdash \Box_k\Box_n\varphi \to \Box_k\Box_m\Box_n\varphi.$$

PROOF. Note that $\Box_n\varphi \in \Phi(\{n\})$. Hence, by the Gateway axiom, $\vdash \Box_k\Box_n\varphi \to \Box_m\Box_n\varphi$. Thus, by the Necessitation rule, $\vdash \Box_k(\Box_k\Box_n\varphi \to \Box_m\Box_n\varphi)$. Then, by the Distributivity axiom, $\vdash \Box_k\Box_k\Box_n\varphi \to \Box_k\Box_m\Box_n\varphi$. Therefore, $\vdash \Box_k\Box_n\varphi \to \Box_k\Box_m\Box_n\varphi$ by Proposition 1. □

PROPOSITION 5. *If $k \le m \le n$ and $\varphi, \psi \in \Phi(\mathbb{Z})$, then*

$$\vdash \Box_m(\Box_k\varphi \vee \Box_n\psi) \to (\Box_m\varphi \vee \Box_m\psi).$$

PROOF. Note that $\Box_k\varphi \in \Phi(\{k\})$ and $\Box_n\psi \in \Phi(\{n\})$. Hence, by the Disjunction axiom,

$$\vdash \Box_m(\Box_k\varphi \vee \Box_n\psi) \to (\Box_m\Box_k\varphi \vee \Box_m\Box_n\psi). \qquad (2)$$

At the same time, by the Reflexivity axiom, $\vdash \Box_k\varphi \to \varphi$. Hence, by the Necessitation rule, $\vdash \Box_m(\Box_k\varphi \to \varphi)$. Thus, by the Distributivity axiom, $\vdash \Box_m\Box_k\varphi \to \Box_m\varphi$. One can similarly show that $\vdash \Box_m\Box_n\psi \to \Box_m\psi$. Therefore, from Statement (2), $\vdash \Box_m(\Box_k\varphi \vee \Box_n\psi) \to (\Box_m\varphi \vee \Box_m\psi)$. □

5. SOUNDNESS

Soundness of propositional tautologies and the Modus Ponens inference rule is straightforward. We will prove soundness of the Necessitation rule and of the remaining five axioms as separate lemmas.

LEMMA 1 (NECESSITATION). *If $r \Vdash \varphi$ for any run r of any protocol, then $r \Vdash \Box_k\varphi$ for any run r of any protocol.*

PROOF. Consider any run r. It will be sufficient to show that $r' \Vdash \varphi$ for each r' such that $r'(k) = r(k)$, which is true due to the assumption of the lemma. □

LEMMA 2 (DISTRIBUTIVITY). *For any run r of a protocol P, if $r \Vdash \Box_k(\varphi \to \psi)$ and $r \Vdash \Box_k\varphi$, then $r \Vdash \Box_k\psi$.*

PROOF. Let r' be any run of P such that $r'(k) = r(k)$. We will show that $r' \Vdash \psi$. Indeed, by the first assumption, $r' \Vdash \varphi \to \psi$. By the second assumption, $r' \Vdash \varphi$. Therefore, by Definition 4, $r' \Vdash \psi$. □

LEMMA 3 (REFLEXIVITY). *For any run r of a protocol P, if $r \Vdash \Box_k\varphi$, then $r \Vdash \varphi$.*

PROOF. Lemma follows from Definition 4 and the fact that $r(k) = r(k)$. □

In the proofs of the soundness of the next three axioms, we use the following auxiliary lemma:

LEMMA 4. *For any $A \subseteq \mathbb{Z}$, any formula $\varphi \in \Phi(A)$, and any runs r, r' of the protocol $(\{V_k\}_{k\in\mathbb{Z}}, \{L_k\}_{k\in\mathbb{Z}}, Tr)$ such that $r(a) = r'(a)$ for every $a \in A$, $r \Vdash \varphi$ if and only if $r' \Vdash \varphi$.*

PROOF. Induction on structural complexity of formula φ. If $\varphi \equiv \bot$, then the required follows from Definition 4.

If $\varphi \equiv p \in P_a$ is an atomic proposition for some $a \in A$, then $r \Vdash p$, by Definition 4 is equivalent to $Tr(r(a), p)$. At the same time, $Tr(r(a), p)$ is equivalent to $Tr(r'(a), p)$ due to the assumption that $r(a) = r'(a)$. Finally, again by Definition 4, $Tr(r'(a), p)$ is equivalent to $r' \Vdash p$.

If $\varphi \equiv \varphi_1 \to \varphi_2$, then $r \Vdash \varphi_1 \to \varphi_2$ is equivalent to disjunction of $r \nVdash \varphi_1$ and $r \Vdash \varphi_2$ by Definition 4. The disjunction, by the Induction Hypothesis, is equivalent to the disjunction of $r' \nVdash \varphi_1$ and $r' \Vdash \varphi_2$. Which, in turn, is equivalent to $r' \Vdash \varphi_1 \to \varphi_2$ by Definition 4.

Finally, assume that $\varphi \equiv \Box_a\psi$ for some $a \in A$. Without loss of generality, we suppose $r \Vdash \Box_a\psi$ and will prove $r' \Vdash \Box_a\psi$. Indeed, let r'' be any run of the protocol such that $r''(a) = r'(a)$. It will be sufficient to show that $r'' \Vdash \psi$. Note that $r''(a) = r'(a) = r(a)$. Thus, $r'' \Vdash \psi$ due to the assumption $r \Vdash \Box_a\psi$ and Definition 4. □

LEMMA 5 (SELF-AWARENESS). *For any run r of a protocol P, any $k \in \mathbb{Z}$, and any $\varphi \in \Phi(\{k\})$, if $r \Vdash \varphi$, then $r \Vdash \Box_k(\varphi)$.*

PROOF. Consider any run r' such that $r'(k) = r(k)$. It will be sufficient to show that $r' \Vdash \varphi$, which is true due to the assumption $r \Vdash \varphi$ and Lemma 4. □

LEMMA 6 (GATEWAY). *For any $A \subseteq \mathbb{Z}$, any $\varphi \in \Phi(A)$, any run r, and any $k, n \in \mathbb{Z}$ such that $k < n \le min(A)$ or $max(A) \le n < k$, if $r \Vdash \Box_k\varphi$, then $r \Vdash \Box_n\varphi$.*

PROOF. Without loss of generality, assume that $k < n \le min(A)$. Let r' be any run such that $r(n) = r'(n)$. We will show that $r' \Vdash \varphi$. Indeed, consider function $r^+(x)$ on $\mathbb{Z}$ such that

$$r^+(x) = \begin{cases} r(x) & \text{if } x < n, \\ r'(x) & \text{otherwise.} \end{cases}$$

We will show that r^+ is a run of the protocol. It trivially satisfies local conditions L_x for all $x \ne n$. To show that local condition L_n is satisfied notice that $L_n(r^+(n-1), r^+(n))$ is equivalent to $L_n(r(n-1), r'(n))$. Then it is also equivalent to $L_n(r(n-1), r(n))$ due to the assumption $r(n) = r'(n)$. Statement $L_n(r(n-1), r(n))$ is true because r is a run of the protocol.

Note that $r^+(k) = r(k)$ by the assumption $k < n$. Thus, $r^+ \Vdash \varphi$ by the assumption $r \Vdash \Box_k\varphi$. Hence, $r' \Vdash \varphi$ by Lemma 4 and due to the fact that $r^+(a) = r'(a)$ for each $a \in A$. □

LEMMA 7 (DISJUNCTION). *For any $A, B \subseteq \mathbb{Z}$, any $\varphi \in \Phi(A)$, any $\psi \in \Phi(B)$, any run r, and any integer $k \in \mathbb{Z}$ such that $max(A) \le k \le min(B)$, if $r \Vdash \Box_k(\varphi \vee \psi)$, then $r \Vdash \Box_k\varphi \vee \Box_k\psi$.*

PROOF. Suppose that $r \nVdash \Box_k\varphi \vee \Box_k\psi$. Thus, by Definition 4, $r \nVdash \Box_k\varphi$ and $r \nVdash \Box_k\psi$. Hence, by Definition 4, there are runs r_1 and r_2 where $r_1(k) = r(k) = r_2(k)$ such that $r_1 \nVdash \varphi$ and $r_2 \nVdash \psi$.

Consider function $r^+(x)$ on $\mathbb{Z}$ such that

$$r^+(x) = \begin{cases} r_1(x) & \text{if } x \leq k, \\ r_2(x) & \text{if } x \geq k. \end{cases}$$

This function is well defined since $r_1(k) = r_2(k)$. It satisfies local conditions of the protocol since runs r_1 and r_2 do. Thus, r^+ is a run of the protocol. Note that $r^+(a) = r_1(a)$ for each $a \in A$ and $r^+(b) = r_1(b)$ for each $b \in B$. Hence, by Lemma 4, $r^+ \nVdash \varphi$ and $r^+ \nVdash \psi$. Thus, by Definition 4, $r^+ \nVdash \varphi \vee \psi$. This is a contradiction with the assumption $r \Vdash \Box_k(\varphi \vee \psi)$ and the fact that $r^+(k) = r_1(k) = r(k)$. □

6. COMPLETENESS

In this section we will prove the completeness of our logical system with respect to the semantics defined above. We start with two technical lemmas.

LEMMA 8. $\vdash \Box_k(\varphi \wedge \psi) \rightarrow (\Box_k\varphi \wedge \Box_k\psi)$.

PROOF. It will be sufficient to prove that $\vdash \Box_k(\varphi \wedge \psi) \rightarrow \Box_k\varphi$. Note that $(\varphi \wedge \psi) \rightarrow \varphi$ is a propositional tautology. Thus, $\vdash \Box_k((\varphi \wedge \psi) \rightarrow \varphi)$ by the Necessitation rule. Hence, $\vdash \Box_k(\varphi \wedge \psi) \rightarrow \Box_k\varphi$, by the Distributivity axiom. □

LEMMA 9. *For any disjoint subsets $A \subseteq \mathbb{Z}$, $B \subseteq \mathbb{Z}$, any family of formulas $\{\varphi_i\}_{i \in A \cup B}$, and any $k \in \mathbb{Z}$ such that $max(A) \leq k \leq min(B)$,*

$$\vdash \Box_k\left(\bigvee_{i \in A \cup B} \varphi_i\right) \rightarrow \left(\Box_k\left(\bigvee_{i \in A} \varphi_i\right) \vee \Box_k\left(\bigvee_{i \in B} \varphi_i\right)\right).$$

PROOF. Note the the following formula is a propositional tautology in our language:

$$\bigvee_{i \in A \cup B} \varphi_i \rightarrow \left(\bigvee_{i \in A} \varphi_i \vee \bigvee_{i \in B} \varphi_i\right).$$

Hence, by the Necessitation Rule,

$$\vdash \Box_k\left(\bigvee_{i \in A \cup B} \varphi_i \rightarrow \left(\bigvee_{i \in A} \varphi_i \vee \bigvee_{i \in B} \varphi_i\right)\right).$$

Thus, by the Distributivity axiom,

$$\vdash \Box_k\left(\bigvee_{i \in A \cup B} \varphi_i\right) \rightarrow \Box_k\left(\bigvee_{i \in A} \varphi_i \vee \bigvee_{i \in B} \varphi_i\right).$$

Therefore,

$$\vdash \Box_k\left(\bigvee_{i \in A \cup B} \varphi_i\right) \rightarrow \Box_k\left(\bigvee_{i \in A} \varphi_i\right) \vee \Box_k\left(\bigvee_{i \in B} \varphi_i\right),$$

by the Disjunction axiom. □

THEOREM 1. *If $\nvdash \varphi$, then there is a protocol P and a run r of the protocol P such that $r \nVdash \varphi$.*

PROOF. Assume that $\nvdash \varphi$. Let X_0 be a maximal and consistent subset of $\Phi(\mathbb{Z})$ containing $\neg\varphi$. Let $\mathbb{X}$ be the set of all maximal consistent subsets of $\Phi(\mathbb{Z})$.

DEFINITION 5. *For any $X, Y \in \mathbb{X}$ let $X \sim_k Y$ mean that $\psi \in X$ if and only if $\psi \in Y$ for each $\psi \in \Phi(\{k\})$.*

LEMMA 10. *For any $X \in \mathbb{X}$ and any ψ such that $\Box_k\psi \notin X$, there is $Y \in \mathbb{X}$ such that $Y \sim_k X$ and $\neg\psi \in Y$.*

PROOF. We will first show that the following set is consistent: $\{\sigma \in \Phi(\{k\}) \mid \sigma \in X\} \cup \{\neg\psi\}$. Indeed, let there be $\sigma_1, \ldots, \sigma_n \in \Phi(\{k\}) \cap X$ such that

$$\vdash \sigma_1 \rightarrow (\sigma_2 \rightarrow \cdots \rightarrow (\sigma_n \rightarrow \psi)\ldots).$$

By the Necessitation rule,

$$\vdash \Box_k(\sigma_1 \rightarrow (\sigma_2 \rightarrow \cdots \rightarrow (\sigma_n \rightarrow \psi)\ldots)).$$

By multiple applications of the Distributivity axiom,

$$\vdash \Box_k\sigma_1 \rightarrow (\Box_k\sigma_2 \rightarrow \cdots \rightarrow (\Box_k\sigma_n \rightarrow \Box_k\psi)\ldots).$$

By multiple applications of the Self-Awareness axiom,

$$\vdash \sigma_1 \rightarrow (\sigma_2 \rightarrow \cdots \rightarrow (\sigma_n \rightarrow \Box_k\psi)\ldots).$$

Hence, by multiple applications of the Modus Ponens rule, $\sigma_1, \sigma_2, \ldots, \sigma_n \vdash \Box_k\psi$. Thus, $X \vdash \Box_k\psi$, which is a contradiction with maximality of X and the assumption $\Box_k\psi \notin X$. Let Y be a maximal consistent set containing $\{\sigma \in \Phi(\{k\}) \mid \sigma \in X\} \cup \{\neg\psi\}$.

We are only left to show that if $\sigma \in Y$, then $\sigma \in X$ for each $\sigma \in \Phi(\{k\})$. Indeed, assume that $\sigma \notin X$. Then, $\neg\sigma \in X$ by the maximality of X. Hence, $\neg\sigma \in Y$ due to the choice of Y. Therefore, $\sigma \notin Y$ due to consistency of Y. □

LEMMA 11. *$\sim_k$ is an equivalence relation on $\mathbb{X}$, for each $k \in \mathbb{Z}$.* □

We now will define protocol $P = (\{V_k\}_{k \in \mathbb{Z}}, \{L_k\}_{k \in \mathbb{Z}}, Tr)$.

DEFINITION 6. *Let V_k be the set of equivalence classes of $\mathbb{X}$ with respect to relation $\sim_k$.*

By $[X]_k$ we mean the equivalence class of element X with respect to the equivalence relation $\sim_k$.

DEFINITION 7. *$L_k(\alpha, \beta)$ if set $\alpha \cap \beta$ is not empty.*

DEFINITION 8. *For any $p \in P_k$ and any set $X \in \mathbb{X}$, $Tr([X]_k, p)$ is true if $p \in Y$ for each $Y \sim_k X$.*

In other words, $Tr([X]_k, p)$ iff $p \in \cap [X]_k$. This concludes the definition of the protocol P.

LEMMA 12. *For each $\psi \in \Phi(A)$, any run r of the protocol P, any $k \in \mathbb{Z}$, and any $X \in \mathbb{X}$, if $\Box_k\psi \in X$, $X \in r(k)$, and either $k \leq n \leq min(A)$ or $max(A) \leq n \leq k$, then $\Box_n\psi \in Z$ for each $Z \in r(n)$.*

PROOF. Without loss of generality, let $k \leq n \leq min(A)$. Induction on n. If $n = k$, then $\Box_k\psi \in X$ implies, by Definition 5, that $\Box_k\psi \in Z$ for each $Z \sim_k X$. Therefore, $\Box_k\psi \in Z$ for each $Z \in r(k)$.

Assume now that $k < n$. By L_n condition, there exists Y such that $Y \in r(n-1) \cap r(n)$. By the Induction Hypothesis, $\Box_{n-1}\psi \in Y$. Hence, by the Gateway axiom, $Y \vdash \Box_n\psi$. Hence, $\Box_n\psi \in Y$ by maximality of Y. Thus, by the Definition 5, $\Box_n\psi \in Z$ for each $Z \sim_n Y$. Therefore, $\Box_n\psi \in Z$ for each $Z \in r(n)$. □

Recall that value of any run r under protocol P is an equivalence class of $\mathbb{X}$. Thus, $\cap\, r(k)$ is a set of formulas. We will refer to this intersection in the next lemma.

LEMMA 13. *For any non-empty set $A \subseteq \mathbb{Z}$ and any set of formulas $\{\psi_a\}_{a\in A}$ such that $\psi_a \in \Phi(\{a\})$ for each $a \in A$ and any $X \in r(k)$, if*

$$\Box_k \bigvee_{a\in A} \psi_a \in X$$

and either $k \leq min(A)$ or $max(A) \leq k$, then there is $a_0 \in A$ such that $\psi_{a_0} \in \cap\, r(a_0)$.

PROOF. Without loss of generality, assume $k \leq min(A)$. We will prove the lemma by induction on the size of set A.

Base Case. Suppose that $A = \{a_0\}$. By Lemma 12, assuming $n = a_0$, we have $\Box_{a_0}\psi_{a_0} \in X$ for each $X \in r(a_0)$. Hence, due to maximality of the set X and the Reflexivity axiom, $\psi_{a_0} \in X$ for each $X \in r(a_0)$. Therefore, $\psi_{a_0} \in \cap\, r(a_0)$.

Induction Step. Suppose that $|A| > 1$. Let X_0 be any set from $r(min(A))$. By Lemma 12, assuming $n = min(A)$, we have

$$\Box_{min(A)} \bigvee_{a\in A} \psi_a \in X_0.$$

Hence, by Lemma 9 and due to maximality of X_0,

$$\Box_{min(A)} \left(\psi_{min(A)} \vee \bigvee_{a\in A\setminus\{min(A)\}} \psi_a \right) \in X_0.$$

By the Disjunction axiom,

$$X_0 \vdash \Box_{min(A)}\psi_{min(A)} \vee \Box_{min(A)} \bigvee_{a\in A\setminus\{min(A)\}} \psi_a.$$

Hence, due to maximality of the set X_0, one of the following statements is true:

$$\Box_{min(A)}\psi_{min(A)} \in X_0,$$

$$\Box_{min(A)} \bigvee_{a\in A\setminus\{min(A)\}} \psi_a \in X_0.$$

In either case, the required follows from the Induction Hypothesis. $\square$

LEMMA 14. *For any non-empty set $A \subseteq \mathbb{Z}$ and any set of formulas $\{\psi_a\}_{a\in A}$ such that $\psi_a \in \Phi(\{a\})$ for each $a \in A$ and any $X \in r(k)$, if*

$$\Box_k \bigvee_{a\in A} \psi_a \in X,$$

then there is $a_0 \in A$ such that $\psi_{a_0} \in \cap\, r(a_0)$.

PROOF. By Lemma 9 and due to maximality of X,

$$\Box_k \left(\bigvee_{\substack{a\in A\\ a\leq k}} \psi_a \vee \bigvee_{\substack{a\in A\\ a>k}} \psi_a \right) \in X.$$

By the Disjunction axiom,

$$X \vdash \Box_k \left(\bigvee_{\substack{a\in A\\ a\leq k}} \psi_a \right) \vee \Box_k \left(\bigvee_{\substack{a\in A\\ a>k}} \psi_a \right).$$

Hence, due to maximality of the set X, one of the following statements is true:

$$\Box_k \left(\bigvee_{\substack{a\in A\\ a\leq k}} \psi_a \right) \in X \quad \text{or} \quad \Box_k \left(\bigvee_{\substack{a\in A\\ a>k}} \psi_a \right) \in X.$$

In either case, the required follows from Lemma 13. $\square$

LEMMA 15. *$r \Vdash \psi$ if and only if $\psi \in \cap\, r(k)$, for each $k \in \mathbb{Z}$, each run r of the protocol P, and each $\psi \in \Phi(\{k\})$.*

PROOF. Induction on structural complexity of formula ψ. If $\psi \equiv \bot$, then the required follows from consistency of the set $r(k)$ and Definition 4. If ψ is an atomic proposition, then the required follows from Definition 8.

Assume that $\psi \equiv \sigma \to \sigma'$ for some $\sigma, \sigma' \in \Phi(\{k\})$.

($\Rightarrow$) : Suppose that $r \Vdash \sigma \to \sigma'$. Thus, $r \nVdash \sigma$ or $r \Vdash \sigma'$. In the first case, by the Induction Hypothesis, $\sigma \notin \cap\, r(k)$. Hence, there is $X \in r(k)$ such that $\sigma \notin X$. Thus, $\sigma \to \sigma' \in X$ due to maximality of the set X. Hence, by Definition 5, $\sigma \to \sigma' \in Y$, for each $Y \sim_k X$. Therefore, $\sigma \to \sigma' \in \cap\, r(k)$.

In the second case, by the Induction Hypothesis, $\sigma' \in \cap\, r(k)$. Thus, $\sigma' \in X$ for each $X \in r(k)$. Hence, $\sigma \to \sigma' \in X$ for each $X \in r(k)$ due to maximality of set X. Therefore, $\sigma \to \sigma' \in \cap\, r(k)$.

($\Leftarrow$) : Suppose that $r \nVdash \sigma \to \sigma'$. Thus, $r \Vdash \sigma$ and $r \nVdash \sigma'$. Then, by the Induction Hypothesis, $\sigma \in \cap\, r(k)$ and $\sigma' \notin \cap\, r(k)$. Hence, there is $X \in r(k)$ such that $\sigma \in X$ and $\sigma' \notin X$. Thus, by maximality of the set X and the Modus Ponens rule, $\sigma \to \sigma' \notin X$. Therefore, $\sigma \to \sigma' \notin \cap\, r(k)$.

Finally, assume that $\psi_k \equiv \Box_k\sigma$. Let $\bigwedge_i \bigvee_j \sigma^i_j$ be the conjunctive normal form of the formula σ such that $\sigma^i_j \in \Phi(\{j\})$ for each i and each j. Note that the following formula is provable in propositional logic without any additional modal axioms:

$$\sigma \to \bigwedge_i \bigvee_j \sigma^i_j.$$

Thus, by the Necessitation Rule,

$$\vdash \Box_k \left(\sigma \to \bigwedge_i \bigvee_j \sigma^i_j \right).$$

By the Distributivity axiom,

$$\vdash \Box_k\sigma \to \Box_k \left(\bigwedge_i \bigvee_j \sigma^i_j \right). \tag{3}$$

One can similarly show that

$$\vdash \Box_k \left(\bigwedge_i \bigvee_j \sigma^i_j \right) \to \Box_k\sigma. \tag{4}$$

($\Leftarrow$) : Suppose that $\Box_k\sigma \in \cap\, r(k)$. Let r' be any run of the protocol such that $r(k) = r'(k)$. We will show that $r' \Vdash \bigwedge_i \bigvee_j \sigma^i_j$.

Note that $\Box_k\sigma \in \cap\, r(k)$ implies that $\Box_k\sigma \in \cap\, r'(k)$, because of the assumption $r(k) = r'(k)$. Hence, $\Box_k\sigma \in X$ for each $X \in r'(k)$. Thus, taking into account Statement (3),

$$\Box_k \left(\bigwedge_i \bigvee_j \sigma^i_j \right) \in X.$$

Then, by Lemma 8,

$$\Box_k \left(\bigvee_j \sigma_j^i \right) \in X.$$

for each $X \in r'(k)$ and each i. Hence by Lemma 14, for each i there is j_0 such that $\sigma_{j_0}^i \in \cap r'(j_0)$. Thus, by the Induction Hypothesis, for each $X \in r'(k)$ and each i there is j_0 such that $r' \Vdash \sigma_{j_0}^i$. Hence, $r' \Vdash \bigwedge_i \bigvee_j \sigma_j^i$.
($\Rightarrow$) : Suppose that $\Box_k \sigma \notin \cap r(k)$. Thus, there is $X \in r(k)$ such that $\Box_k \sigma \notin X$. Then, due to Statement (4),

$$\Box_k \left(\bigwedge_i \bigvee_j \sigma_j^i \right) \notin X.$$

Hence, by Lemma 10, there is $Y \sim_k X$ such that

$$\neg \bigwedge_i \bigvee_j \sigma_j^i \in Y.$$

Thus, due to the maximality of Y, there is i_0 such that

$$\neg \bigvee_j \sigma_j^{i_0} \in Y.$$

Hence, due to the maximality of Y, for each j,

$$\neg \sigma_j^{i_0} \in Y. \tag{5}$$

Consider function r_Y such that $r_Y(n) = [Y]_n$ for each $n \in \mathbb{Z}$. Note that $Y \in [Y]_{n-1} \cap [Y]_n$. Thus, $[Y]_{n-1} \cap [Y]_n$ is not empty for any $n \in \mathbb{Z}$. Therefore, r is a run of the protocol P. By Definition 5, Statement (5) implies that $\neg \sigma_j^{i_0} \in Y'$ for each j and each $Y' \sim_j Y$. Hence, $\neg \sigma_j^{i_0} \in \cap\, r_Y(j)$ for each j. Thus, by the Induction Hypothesis, $r_Y \Vdash \neg \sigma_j^{i_0}$ for each j. Then,

$$r_Y \Vdash \neg \bigvee_j \sigma_j^{i_0}.$$

Hence,

$$r_Y \Vdash \neg \bigwedge_i \bigvee_j \sigma_j^i.$$

In other words, $r_Y \Vdash \neg\sigma$. Therefore, $r \nVdash \Box_k \sigma$. $\square$

To finish the proof of the theorem, assume that $\bigwedge_i \bigvee_j \varphi_j^i$ is the conjunctive normal form of the formula $\neg\varphi$ such that $\varphi_j^i \in \Phi(\{j\})$ for each i and each j. Consider r such that $r(n) = [X_0]_n$ for each $n \in \mathbb{Z}$. Note that $X_0 \in [X_0]_{n-1} \cap [X_0]_n$. Thus, $[X_0]_{n-1} \cap [X_0]_n$ is not empty for any $n \in \mathbb{Z}$. Therefore r is a run of the protocol P.

Recall that $\neg\varphi \in X_0$. Thus, $\bigwedge_i \bigvee_j \varphi_j^i \in X_0$. Hence, $\bigvee_j \varphi_j^i \in X_0$ for each i due to maximality of the set X_0. Hence, again due to maximality of X_0, for each i there is j_i such that $\varphi_{j_i}^i \in X_0$. Hence, by Lemma 15, $r \Vdash \varphi_{j_i}^i$ for each i. Thus, $r \Vdash \bigwedge_i \bigvee_j \varphi_j^i$. Therefore, $r \Vdash \neg\varphi$. In other words, $r \nVdash \varphi$. $\square$

7. CONCLUSIONS

7.1 Directed Chains

Although edges representing channels a, b, and c in Figure 1 are drawn as directed, none of our definitions so far have used them as such. The "directness" of these edges can be captured by restricting the class of all protocols to these that satisfy the additional *continuity condition* [1]: for each $v \in V_{k-1}$ there is $u \in V_k$ such that $L_k(v, u)$. This requirement, however, does not change any of our results and the existing proof of completeness still holds because the protocol constructed in the proof of completeness satisfies the continuity condition. Indeed, for any $[X]_{k-1} \in V_{k-1}$ one can just take $[X]_k \in V_k$ and notice that L_k is true because $X \in [X]_{k-1} \cap [X]_k$.

7.2 Communication Networks

Communication chains can be generalized to non-linear communication networks like the one depicted in Figure 3. Intuitively it is clear that if $\Box_a \Box_f \varphi$ on this network, then

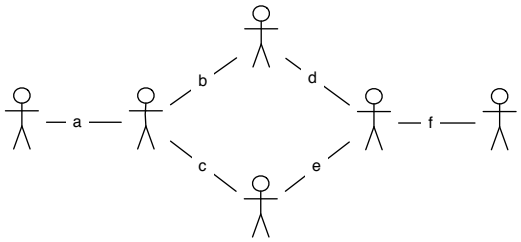


Figure 3: Communication Network.

this knowledge of a is acquired through channels b and c. This is an example of a more general form of the Gateway axiom for communication networks. However, straightforward formalization of this principle

$$\Box_a \Box_f \varphi \to (\Box_b \Box_f \varphi \vee \Box_c \Box_f \varphi)$$

is not true since the encrypted evidence of $\Box_f \varphi$ could have traveled through channels b and d and the encryption key through channels c and e. Thus, neither b nor c alone would have knowledge of $\Box_f \varphi$ under such a protocol. The right way to formalize the Gateway principle in this setting is

$$\Box_a \Box_f \varphi \to (\Box_{b,c} \Box_f \varphi),$$

where $\Box_{b,c}\psi$ means that anyone who knows values b and c will be able to conclude ψ. In general, Definition 4 can be modified to say

4. *$r \Vdash \Box_A \varphi$ if $r' \Vdash \varphi$ for each r' such that $r'(a) = r(a)$ for all $a \in A$.*

Then, the Gateway axiom can be stated as follows: if every path from each edge in set A to each edge in set B goes through an edge in set G, then

$$\Box_A \varphi \to \Box_G \varphi,$$

for each $\varphi \in \Phi(B)$. Similarly, the Disjunction axiom can be rephrased for communication networks as: if every path from each edge in set A to each edge in set B goes through an edge in set G, then

$$\Box_G(\varphi \vee \psi) \to (\Box_G \varphi) \vee (\Box_G \psi),$$

for each $\varphi \in \Phi(A)$ and each $\psi \in \Phi(B)$.

Both of these axioms are sound in the stated form. However, our proof of completeness heavily relies on equivalence relation $\sim_k$ and it is not clear how relations $\sim_A$ for multiple A all of which might contain k should work together. Thus, a complete axiomatization of epistemic logic for non-linear communication networks remains an open problem.

8. REFERENCES

[1] Michael S. Donders, Sara Miner More, and Pavel Naumov. Information flow on directed acyclic graphs. In Lev D. Beklemishev and Ruy de Queiroz, editors, *WoLLIC*, volume 6642 of *Lecture Notes in Computer Science*, pages 95–109. Springer, 2011.

[2] Tim French, Ron van der Meyden, and Mark Reynolds. Axioms for logics of knowledge and past time: Synchrony and unique initial states. In Renate A. Schmidt, Ian Pratt-Hartmann, Mark Reynolds, and Heinrich Wansing, editors, *Advances in Modal Logic*, pages 53–72. King's College Publications, 2004.

[3] Joseph Y. Halpern, Ron van der Meyden, and Moshe Y. Vardi. Complete axiomatizations for reasoning about knowledge and time. *SIAM J. Comput.*, 33(3):674–703, 2004.

[4] Sarah Holbrook and Pavel Naumov. Fault tolerance in belief formation networks. In Luis Fariñas del Cerro, Andreas Herzig, and Jérôme Mengin, editors, *JELIA*, volume 7519 of *Lecture Notes in Computer Science*, pages 267–280. Springer, 2012.

[5] Sara Miner More and Pavel Naumov. Hypergraphs of multiparty secrets. In *11th International Workshop on Computational Logic in Multi-Agent Systems CLIMA XI (Lisbon, Portugal), LNAI 6245*, pages 15–32. Springer, 2010.

[6] Sara Miner More and Pavel Naumov. The functional dependence relation on hypergraphs of secrets. In João Leite, Paolo Torroni, Thomas Ågotnes, Guido Boella, and Leon van der Torre, editors, *CLIMA*, volume 6814 of *Lecture Notes in Computer Science*, pages 29–40. Springer, 2011.

[7] Sara Miner More and Pavel Naumov. Logic of secrets in collaboration networks. *Ann. Pure Appl. Logic*, 162(12):959–969, 2011.

[8] Eric Pacuit. Some comments on history based structures. *J. Applied Logic*, 5(4):613–624, 2007.

[9] Eric Pacuit and Rohit Parikh. Reasoning about communication graphs. In Benedikt Löwe Dov Gabbay Johan van Benthem, editor, *Interactive Logic: Games and Social Software*, 2007.

[10] Rohit Parikh and Ramaswamy Ramanujam. A knowledge based semantics of messages. *Journal of Logic, Language and Information*, 12(4):453–467, 2003.

[11] Ron van der Meyden. Axioms for knowledge and time in distributed systems with perfect recall. In *LICS*, pages 448–457. IEEE Computer Society, 1994.

Knowledge-Based Programs as Plans: Succinctness and the Complexity of Plan Existence (Extended Abstract)*

Jérôme Lang
LAMSADE, CNRS, Université Paris-Dauphine, France
lang@lamsade.dauphine.fr

Bruno Zanuttini
GREYC, Université de Caen Basse-Normandie, CNRS, ENSICAEN, France
bruno.zanuttini@unicaen.fr

ABSTRACT

Knowledge-based programs (KBPs) are high-level protocols describing the course of action an agent should perform as a function of its knowledge. The use of KBPs for expressing action policies in AI planning has been surprisingly overlooked. Given that to each KBP corresponds an equivalent plan and *vice versa*, KBPs are typically more succinct than standard plans, but imply more on-line computation time. Here we make this argument formal, and prove that there exists an exponential succinctness gap between knowledge-based programs and standard plans. Then we address the complexity of plan existence. Some results trivially follow from results already known from the literature on planning under incomplete knowledge, but many were unknown so far.

1. INTRODUCTION

Knowledge-based programs (KBPs) [7] are high-level protocols which describe the actions an agent should perform as a function of its knowledge, such as, typically, **if** $\mathbf{K}\varphi$ **then** π **else** π', where $\mathbf{K}$ is an epistemic modality and π, π' are subprograms.

Thus, in a KBP, branching conditions are epistemically interpretable, and deduction tasks are involved at execution time (on-line). KBPs can be seen as a powerful language for expressing policies or plans, in the sense that epistemic branching conditions allow for exponentially more compact representations. In contrast, standard plans (as in contingent planning) or standard policies (as in POMDPs) either are sequential or branch on objective formulas, and hence can be executed efficiently, but they can be exponentially larger (see for instance [1]).

Having said this, KBPs have surprisingly been overlooked in the perspective of planning. Initially developed for distributed computing, they have been considered in AI for agent design [5] and game theory [10]. For planning, the only works we know of are by Reiter [17], who gives an implementation of KBPs in Golog; Classen and Lakemeyer [6], who implement KBPs in a decidable fragment of the situation calculus; Herzig *et al.* [9], who discuss KBPs for propositional planning problems, and Laverny and Lang [12, 13], who generalize KBPs to *belief*-based programs allowing for uncertain action effects and noisy observations. None of these papers really addresses computational issues.

A few papers in the AI planning literature have studied planning with incomplete knowledge where the agent's knowledge is represented by means of epistemic modalities, such as Petrick and Bacchus [16]. Another recent stream of work focuses on describing planning problems within the framework of Dynamic Epistemic Logic (Löwe *et al.* [14], Bolander and Andersen [3]). Nilogy and Ramanujam [15] also make use of epistemic logic for planning with "action trials", where action feedback corresponds to the action succeeding or failing. However, in all these papers, epistemic formulas are used only for representing the current knowledge state and the effects of actions, not in branching conditions, which bear on observations only.

Recently, [11] have started to address the computational issues of planning with knowledge-based programs, by identifying the complexity of plan verification under various assumptions on the available constructs for plans and the available actions. Even if they briefly address the succinctness of knowledge-based programs compared to standard plans, the discussion remains at an informal level; moreover they do not consider at all the plan existence problem, which is even more important for practical planning purposes than plan verification. This paper contributes to fill these two gaps.

We define knowledge-based programs and planning problems in Section 3. Section 4 formally relates KBPs to standard plans, by showing that both have the same expressivity, but that KBPs are exponentially more succinct than standard plans. Section 5 focuses on the plan existence problem. We could think that because KBPs and standard plans are equally expressive, KBP existence is equivalent to standard plan existence, the complexity of which has been investigated, especially by Rintanen [18]. This is partly true, and indeed some results about KBP existence directly follow from these earlier results. This is however not true for (a) "small" KBP existence problems, where the objective is to find a small enough KBP allowing to reach the goal; (b) purely epistemic plan existence, which have surprisingly been ignored. Our main results are the following: (a) the existence of a bounded-size solution KBP is EXPSPACE-complete, and falls down to Σ_3^p-complete if loops are disallowed, to Σ_2^p-complete for the restriction to ontic actions and the restriction to epistemic actions and positive goals; (b)

*This work was supported by the French National Research Agency under grant ANR-10-BLAN-0215 (LARDONS).

TARK 2013, Chennai, India.

purely epistemic plan existence is PSPACE-complete, and coNP-complete if the goal is a positive epistemic formula. Further issues are briefly evoked in the conclusion.

2. PRELIMINARIES

A KBP is executed by an agent in an environment. We model what the agent *knows* about the current state (of the environment and internal variables) in the propositional epistemic logic S_5. Let $X = \{x_1, \ldots, x_n\}$ be propositional symbols. A *state* is a valuation of X; *e.g.*, $\overline{x_1}x_2$ is the state where x_1 is false and x_2 is true. We sometimes use the notation x^ϵ with $x^1 = x$ and $x^0 = \bar{x}$. A *knowledge state* M for S_5 is a nonempty set of states (those the agent considers possible): at any point in time, the agent has a knowledge state $M \subseteq 2^X$ and the current state is some $s \in M$. For instance, $M = \{x_1\bar{x}_2, \bar{x}_1x_2\}$ means that the agent knows x_1 and x_2 have different values in the current state.

Formulas of S_5 are built up from X, the usual connectives, and the knowledge modality $\mathbf{K}$. An S_5 formula is *objective* if it does not contain any occurrence of $\mathbf{K}$. Objective formulas are denoted by φ, ψ, etc. whereas general S_5 formulas are denoted by Φ, Ψ etc. For an objective formula φ, we denote by $Mods(\varphi)$ the set of all states which satisfy φ (*i.e.*, $Mods(\varphi) = \{s \in 2^X, s \models \varphi\}$). The size $|\Phi|$ of an S_5 formula Φ is the total number of occurrences of propositional symbols, connectives and modality $\mathbf{K}$ in Φ. It is well-known (see, *e.g.*, [7]) that any S_5 formula is equivalent to a formula without nested $\mathbf{K}$ modalities; therefore we disallow them. An S_5 formula Φ is *purely subjective* if objective formulas occur only in the scope of $\mathbf{K}$, and a purely subjective S_5 formula is in *knowledge negative normal form (SKNNF)* if the negation symbol $\neg$ occurs only in objective formulas (in the scope of $\mathbf{K}$) or directly before a $\mathbf{K}$ modality. Note that any purely subjective S_5 formula Φ can be rewritten into an equivalent SKNNF of polynomial size using de Morgan's laws. An SKNNF formula is *positive* if the negation symbol never appears in front of a $\mathbf{K}$ modality. For instance, $\mathbf{K}\neg(p \wedge q) \vee \neg(\mathbf{K}r \vee \mathbf{K}\neg r)$ is not in SKNNF, but is equivalent to the SKNNF formula $\mathbf{K}\neg(p \wedge q) \vee (\neg\mathbf{K}r \wedge \neg\mathbf{K}\neg r)$, which is not a positive SKNNF, whereas $\mathbf{K}\neg(p \wedge q) \wedge (\mathbf{K}r \vee \mathbf{K}\neg\neg r)$ is a positive SKNNF.

The satisfaction of a purely subjective formulas depends only on a knowledge state M, not on the *actual* current state (see, *e.g.*, [7]): M satisfies an atom $\mathbf{K}\varphi$, written $M \models \mathbf{K}\varphi$, if for all $s \in M$, $s \models \varphi$, and the semantics for combinations of atoms with $\neg, \wedge, \vee$ is defined as usual.

3. KNOWLEDGE-BASED PROGRAMS AND PLANNING PROBLEMS

We briefly recall the essential definitions about KBPs [11]. Given a set A_O of ontic actions and a set A_E of epistemic actions, a *knowledge-based program* (KBP) is defined inductively as follows:

- the empty plan π_λ is a KBP;
- any action $\alpha \in A_O \cup A_E$ is a KBP;
- if π and π' are KBPs, then $\pi;\pi'$ is a KBP;
- if π, π' are KBPs and Φ is a formula in $SKNNF$, then **if** Φ **then** π **else** π' **endif** is a KBP;
- if π is a KBP and Φ is a formula in SKNNF, then **while** Φ **do** π **endwhile** is a KBP.

The class of *while-free* KBPs is obtained by omitting the **while** construct. The *size* $|\pi|$ of a KBP π is defined to be the number of occurrences of actions, plus the size of branching conditions, in π. Finally, we sometimes view while-free KBPs as trees, with some nodes labelled by actions and having one child (the KBP following this action), and some nodes labelled by an epistemic formula and having two children (for **if** constructs). Accordingly, we refer to *branches* of KBPs.

Let $X' = \{x' \mid x \in X\}$, denoting the values of variables after an action has been taken. An *ontic action* α is represented by its *theory* Σ_α, which is a propositional formula over $X \cup X'$ such that for all states $s \in 2^X$, the set $\{s' \in 2^{X'} \mid ss' \models \Sigma_\alpha\}$ is nonempty, and is exactly the set of possible states after α is performed in s. For instance, with $X = \{x_1, x_2\}$, the action α which nondeterministically reinitializes the value of x_1 has the theory $\Sigma_\alpha = (x'_2 \leftrightarrow x_2)$. Observe that ontic actions are nondeterministic in general; moreover, when taking such an action the agent does not know which outcome occurred. We sometimes omit the "'frame axioms" of the form $x'_i \leftrightarrow x_i$ from Σ_α, *e.g.*, we write $x'_1 \leftrightarrow \neg x_1$ for the action of switching x_1, whatever the other variables.

Now, an *epistemic action* α is represented by its *feedback theory* Ω_α, which is a list of positive epistemic atoms of the form $\Omega_\alpha = (\mathbf{K}\varphi_1, \ldots, \mathbf{K}\varphi_n)$. For instance, the epistemic action which senses the value of an objective formula φ is denoted by $\mathrm{test}(\varphi)$, and its feedback theory is $\Omega_{\mathrm{test}(\varphi)} = (\mathbf{K}\varphi, \mathbf{K}\neg\varphi)$. We require that feedbacks be exhaustive ($\varphi_1 \vee \cdots \vee \varphi_n$ is tautological), so that in any state an epistemic action yields a feedback, but we do not require them to be mutually exclusive; if several feedbacks are possible in some state, one is chosen nondeterministically at execution time.

Operational Semantics.

The agent executing a KBP starts in some knowledge state M^0, and at any timestep t until the execution terminates, it has a current knowledge state M^t. When execution comes to a branching condition Φ, Φ is evaluated in the current knowledge state (*i.e.*, the agent decides whether $M^t \models \Phi$ holds).

The knowledge state M^t is defined inductively as the *progression* of M^{t-1} by the action executed between $t-1$ and t. Formally, given a knowledge state $M \subseteq 2^X$ and an *ontic* action α, the *progression* of M by α is defined to be $\mathrm{Prog}(M, \alpha) = M' \subseteq 2^{X'}$ defined by $M' = \{s' \in 2^{X'} \mid s \in M, ss' \models \Sigma_\alpha\}$. Now given an *epistemic* action α, a knowledge state M, and a feedback $\mathbf{K}\varphi_i \in \Omega_\alpha$ with $M \not\models \mathbf{K}\neg\varphi_i$, the progression of M by $\mathbf{K}\varphi_i$ is defined to be $\mathrm{Prog}(M, \mathbf{K}\varphi_i) = \{s \in M \mid s \models \varphi_i\}$. The progression is undefined when $M \models \mathbf{K}\neg\varphi_i$.

EXAMPLE 1. *Consider a system composed of three components; for each $i = 1, 2, 3$, we have a propositional symbol ok_i meaning that component i is in working order, an action $repair(i)$ that makes ok_i true, and an action $\mathrm{test}(i)$ that returns the truth value of ok_i; for instance, $\Sigma_{repair(1)} = ok'_1 \wedge (ok'_2 \leftrightarrow ok_2) \wedge (ok'_3 \leftrightarrow ok_3)$ and $\Omega_{\mathrm{test}(1)} = (\mathbf{K}ok_1, \mathbf{K}\neg ok_1)$. Let $\pi = \pi_1;\pi_2;\pi_3$, where π_i is defined as*

if $\neg(\mathbf{K}ok_i \vee \mathbf{K}\neg ok_i)$ ***then*** $\mathrm{test}(i)$ ***endif*** ;
if $\mathbf{K}\neg ok_i$ ***then*** $repair(i)$ ***endif***

With $M^0 = Mods((ok_1 \leftrightarrow (ok_2 \wedge ok_3)) \wedge (\neg ok_2 \vee \neg ok_3))$,

$\mathrm{Prog}(M^0, \mathit{repair}(1))$ *is* $M^1 = \mathit{Mods}(ok_1 \wedge (\neg ok_2 \vee \neg ok_3))$, $\mathrm{Prog}(M^1, \mathbf{K}ok_2)$ *is* $M^2 = \mathit{Mods}(ok_1 \wedge ok_2 \wedge \neg ok_3))$, *and* $\mathrm{Prog}(M^2, \mathit{repair}(3))$ *is* $M^3 = \mathit{Mods}(ok_1 \wedge ok_2 \wedge ok_3)$.

Finally, a *trace* τ of a KBP π in a knowledge state M^0 is a sequence of knowledge states, either infinite, *i.e.*, $\tau = (M^i)^{i \geq 0}$, or finite, *i.e.*, $\tau = (M^0, M^1, \ldots, M^T)$, which corresponds to the iterated progression of M^0 by the actions in π, given an outcome $s \in 2^X$ (resp. a feedback $\mathbf{K}\varphi$) for each ontic (resp. epistemic) action encountered. We say that two KBPs π and π' are *equivalent* (resp. *equivalent in* M^0) if they have exactly the same traces in any initial knowledge state (resp. in M^0).

KBPs as Plans.

We define a *knowledge-based planning problem* P to be a tuple (I, A_O, A_E, G), where $I = \mathit{Mods}(\varphi^0)$ is the *initial knowledge state*, G is an SKNNF S_5 formula called the *goal*, and A_O (resp. A_E) is a set of ontic (resp. epistemic) actions together with their theories. Then a KBP π (using actions in $A_O \cup A_E$) is said to be a *(valid) plan* for P if all its traces in I are finite, and for all traces $(M^0, \ldots, M^T)$ of π with $M^0 = I$, $M^T \models G$ holds.

Interesting restrictions of knowledge-based planning problems are obtained either by restricting the form of KBPs (by disallowing loops, or by bounding the size of the KBP), by restricting the set of actions allowed (by requiring all actions to be ontic or all actions to be epistemic), or by adding a restriction on the goal (by requiring it to be a *positive* KNNF). The restriction to positive goals deserves some comments. After all, one may think that goals should *always* be positive – and in most of practical cases they will indeed be: why should a robot care about *not knowing* something? The more it knows, the easier it is to make accurate decisions. This is true in a single-agent environment. Now, even if our paper does not address full multi-agent environments (which are much more complex to handle), it allows to represent at least a simple class of multi-agent planning problems, where only one agent is able to act but other agents observe its actions and feedbacks. But there might be facts which the acting agent wants to avoid the other to learn, and under the assumption that observations are considered as public announcements, the acting agent will also want *not* to learn these facts[1].

4. SUCCINCTNESS

So as to measure the benefit of using KBPs as plans, we compare them to what we call *standard policies*. We define such policies exactly as KBPs, but allowing branching on feedbacks just obtained via an epistemic action, rather than on unrestricted epistemic formulas. What we have in mind here is to compare KBPs to *reactive* policies, for which the next action to take can be found efficiently at execution time.

DEFINITION 1 (STANDARD POLICY). *A* standard policy *is a KBP in which the last action executed before any branching* ***if*** Φ *or* ***while*** Φ *is an epistemic action* a *such that* Φ *is some* $\mathbf{K}\varphi_i \in \Omega_a$.

[1]The reader has certainly experienced the situation where the screen of her laptop, connected to a videoprojector, appears on a screen in front of everyone and each of her actions (reading email, inspecting the contents of a directory...) could possibly reveal some information she does not want everyone to see.

Hence evaluating a branching condition of a standard policy at execution time only requires to compare the feedback just obtained to the branching condition Φ. Particular cases of standard policies are policies for *partially observable Markov decision processes* (POMDPs), which alternate the following steps: (i) taking an (ontic) action, (ii) receiving an observation about the current state, and (iii) branching on the observation received. Observe however that our definition is more general, in that the alternation between decision and observation+branching steps is unrestricted, and that loops are allowed. For instance, our definition also encompasses sequential plans (of the form $a_1; a_2; \ldots; a_n$), but also controllers with finite memory [4].

Clearly, for every initial knowledge state M^O and every KBP π, there is a standard policy equivalent to π in M^0. Such a policy can be obtained by simulating all possible executions of π in M^0 and, for each one, evaluating all (epistemic) branching conditions. We only give an example here (a formal definition is given in the Appendix — Definition 4 and Proposition 11).

EXAMPLE 2. *The standard policy associated with* π *and* M^0 *in Example 1 is the following:*

---

repair(1); test(2);
if $\mathbf{K}\neg ok_2$ ***then***
  repair(2);
  test(3);
  if $\mathbf{K}\neg ok_3$ ***then*** *repair*(3) ***endif***
else *repair*(3)
endif

Such translations are of course not guaranteed to be polynomial, which raises the issue whether KBPs are more succinct than standard policies. We first give a formal definition of succinctness.

DEFINITION 2 (SUCCINCTNESS). *Let* $\mathcal{C} = (\mathcal{C}_n)_{n \in \mathbb{N}}$ *be a class of KBPs (or standard policies), and let* $\mathcal{P} = (P_n)_{n \in \mathbb{N}}$ *be a family of planning problems. Then* $\mathcal{C}$ *is said to be* succinct *for* $\mathcal{P}$ *if there is a polynomial* $p : \mathbb{N} \to \mathbb{N}$ *and a family* $(\pi_n)_{n \in \mathbb{N}}$ *of KBPs satisfying* $\pi_n \in \mathcal{C}_n$, $|\pi_n| \in O(p(n))$, *and such that* π_n *is a valid plan for* P_n.

A class $\mathcal{C}$ *is said to be* as succinct as *a class* $\mathcal{C}'$ *if for all families* $\mathcal{P}$ *of planning problems such that* $\mathcal{C}'$ *is succinct for* $\mathcal{P}$, $\mathcal{C}$ *is also succinct for* $\mathcal{P}$. *It is said to be* more succinct than $\mathcal{C}'$ *if in addition, there is a family* $\mathcal{P}$ *of planning problems for which* $\mathcal{C}$ *is succinct but* $\mathcal{C}'$ *is not.*

Note that our definition of being more succinct is quite demanding, since not only it requires that there is no polysize KBP in $\mathcal{C}'$ equivalent to $\pi \in \mathcal{C}$, but also it requires that there is no polysize KBP which is valid for the same problem (may it be nonequivalent to π).

Clearly, because standard policies are defined as particular cases of KBPs, the latter are always at least as succinct than the former. We now show that KBPs are *more succinct* than standard policies, even under several restrictions.

PROPOSITION 1. *If* $\mathsf{NP} \not\subseteq \mathsf{P/poly}$ *holds, while-free KBPs with atomic epistemic branching conditions are more succinct than while-free standard policies.*

PROOF. For all $n \in \mathbb{N}$, we exhibit a KBP π_n as in the claim which essentially reads a 3CNF formula over n variables (hidden in the initial state), and either makes sure that

it is unsatisfiable, or builds a model. This KBP has size polynomial in n. Now assume there is a while-free standard policy π' of size polynomial in $|\pi|$, and hence in n, which is a valid plan for the same problem. Then because standard policies can be executed with constant-time delay and because π' is while-free, execution of π' would be a (possibly nonuniform) polytime algorithm for 3SAT, yielding 3SAT $\in$ P/poly and hence, NP $\subseteq$ P/poly. The construction of the KBP π_n and the definition of the knowledge-based planning problem P_n are detailed in the Appendix (Proposition 12). □

Observe that the proof even shows that there are planning problems with succinct while-free KBPs (with atomic branching conditions) but with no compact while-free plan with polynomial-delay execution (cf. the notion of a *compact sequential-access representation* [1]). Observe however that if loops are allowed, then there does exist a compact standard policy for the 3SAT problem (for instance, the DPLL algorithm). However, it turns out that there are problems with succinct KBPs (with loops) but with no succinct standard policy at all (even with loops).

PROPOSITION 2. *KBPs are more succinct than standard policies.*

PROOF. There is a KBP π of size polynomial in n (in particular, manipulating a number of variables polynomial in n) with exactly one trace in some precise initial knowledge state M^0, of size $2^{2^n} - 1$ [11, Proposition 5]. Now Proposition 13 in the Appendix shows that given a KBP π, a planning problem P can be built efficiently, for which all valid plans are equivalent to π in M^0 (up to a polynomial number of void actions), and for which π is indeed valid. Towards a contradition, assume that there is a valid standard policy π' for P. Then π' has exactly one trace, of size $2^{2^n} - 1$ (up to a polynomial). But if π' has size polynomial in n, then it can manipulate at most n variables, and because it is a standard policy it can be in at most $2^n|\pi'|$ different configurations (values of variables plus control point). Hence it cannot have a terminating trace of length greater than $2^n|\pi|$, a contradiction. □

We conclude this section by considering the succintness gap induced by loops in KBPs.

PROPOSITION 3. *KBPs are more succinct than while-free KBPs.*

PROOF. Assume towards a contradiction that for each KBP π, there is an equivalent while-free KBP π' satisfying $|\pi'| \leq p(|\pi|)$. Then there is an algorithm showing that verifying a KBP (with loops) is a problem in Σ_3^{P} (Proposition 14 in the Appendix). Since on the other hand we know that verifying an unrestricted KBP is an EXPSPACE-hard problem [11, Proposition 6], we get a contradiction with $\Sigma_3^{\mathsf{P}} \subseteq$ PSPACE $\subsetneq$ EXPSPACE (Savitch's theorem). Finally, given a polynomial-size KBP π for which there is no equivalent polynomial-size while-free KBP, we build a problem which has only π and equivalent KBPs as valid plans (Proposition 13 in the Appendix), and this problem shows that KBPs are more succinct than while-free KBPs. □

5. COMPLEXITY OF PLAN EXISTENCE

We now consider the problem of deciding whether there exists a valid KBP for a given planning problem. Since the main benefit of using KBPs is to get succinct (and readable) plans, we insist on the "small KBP existence" problem, where we ask whether there exists a valid KBP within a given size bound.

DEFINITION 3 (EXISTENCE). *The* plan existence problem *takes as input a knowledge-based planning problem $P = (I, A_O, A_E, G)$ and asks whether there exists a valid KBP π for P. The* bounded size plan existence problem *takes as input a knowledge-based planning problem $P = (I, A_O, A_E, G)$ and an integer k encoded in unary, and asks whether there exists a KBP π for P satisfying $|\pi| \leq k$.*

We start with the complexity of plan existence, that is, without a size bound.

PROPOSITION 4. *Plan existence is* 2-EXPTIME*-complete. It is* EXPSPACE*-complete if only ontic actions are allowed.*

PROOF. The first two results follow from the fact that there is a valid KBP for a given knowledge-based planning problem P if and only if there is a valid standard policy for P (Proposition 11 in the Appendix), together with known results by Rintanen [18] and by Haslum and Jonsson [8]. □

PROPOSITION 5. *While-free KBP existence restricted to epistemic actions is PSPACE-complete.*

PROOF. Write WFE-EXISTENCE for the problem of while-free KBP existence. We introduce a variant, called WFOE-EXISTENCE (for "While-Free Ordered Epistemic"), in which a total order $<$ on A_E is given as an additional input, and the question is whether there is a valid KBP for P, in which actions occur in the order $<$ in any execution. Then we show QBF $\leq^P$ WFOE-EXISTENCE $\leq^P$ WFE-EXISTENCE.

The reductions are given in the Appendix (Propositions 17 and 18). Because QBF is PSPACE-complete, it follows that WFE-EXISTENCE is PSPACE-hard. Finally, because only epistemic actions are available, the state never changes, and hence executing the same epistemic action twice in an execution is useless. It follows that we are essentially searching for a tree of height at most $|A_E|$, and membership in PSPACE easily follows. □

PROPOSITION 6. *While-free KBP existence restricted to epistemic actions and positive goals is* coNP*-complete.*

PROOF. This proof is essentially by a reduction to validity in S_5 (Proposition 16 in the Appendix). □

PROPOSITION 7. *Bounded KBP existence is EXPSPACE-complete.*

PROOF. For hardness, we reduce the problem of verifying that a KBP π is valid for a planning problem $P = (I, A_O, A_E, G)$ to plan existence, by building a planning problem P' with bound $k = |\pi|$ for which π is valid if and only if it is valid for P, and every valid plan is equivalent to π. For this we use Proposition 13 with the construction initialized with I and G. Hence if π is valid for P, then P' has a plan of size at most k (namely, π), and if π is not valid for P, then P' has no valid plan. Because verification is an EXPSPACE-hard problem [11, Proposition 6], we get hardness. Membership follows from the fact that a plan π can be guessed, that verifying that it is valid is in EXPSPACE [11, Proposition 6 again], and from NEXPSPACE = EXPSPACE (Savitch's theorem). □

PROPOSITION 8. *While-free bounded KBP existence is Σ_3^p-complete. Hardness holds even if the goal is restricted to be a positive epistemic formula.*

PROOF. Since solutions have bounded size, membership in Σ_3^{P} follows from the fact that while-free KBP verification is in Π_2^{P} [11, Proposition 2]. For hardness, we give a reduction from $\mathrm{QBF}_{3,\exists}$ (Proposition 19 in the Appendix). □

PROPOSITION 9. *While-free bounded KBP existence restricted to ontic actions is Σ_2^{P}-complete.*

PROOF. Because there is no feedback, there is no need for branching, therefore there is a plan of size at most k if and only if there is a valid plan which is a sequence of at most k actions. The bounded KBP existence problem is therefore equivalent to the bounded plan existence problem, which is known to be Σ_2^{P}-complete [2] if the goal is positive atomic. Now membership in Σ_2^{P} in the general case follows from the fact that verifying a plan can be done by computing the memoryful progression [11] in polynomial time, then checking that it entails the goal using a coNP-oracle. □

As for purely epistemic planning problems, things are easy only in the case of positive goals.

PROPOSITION 10. *While-free bounded KBP existence restricted to epistemic actions and to positive goals is Σ_2^{P}-complete.*

PROOF. Since the goal Γ is positive epistemic and the state cannot change, executing more epistemic actions cannot render a valid plan invalid. In particular, removing all branching conditions and linearizing a valid plan gives a valid plan. Hence there is a valid plan of size $\leq k$ if and only if there is a sequence of k epistemic actions which is a valid plan. Hence the problem can be solved by guessing a plan $a_1; \ldots; a_k$ and checking $\bigwedge_{i=1}^{k}(\bigvee_{\mathbf{K}\varphi_j \in \Omega_{a_i}} \varphi_j) \models \Gamma$, which can be done by a call to a coNP-oracle. Now for hardness, we give a reduction from $\mathrm{QBF}_{2,\exists}$ (Proposition 20 in the Appendix). □

6. CONCLUSION

Our contributions are twofold. First, we have made formal the succinctness gap obtained by the possibility to branch on complex epistemic formulas instead of simply branching on observations. Second, we have obtained several nontrivial results on the complexity of KBP existence for a planning problem. The results are synthesized in the table below. Note that as far as unbounded KBP existence is concerned, whether loops are allowed or not does not make a difference: since valid plans are required to stop, every valid KBP with loops can be rewritten into an equivalent while-free KBP. This remark helps us having all cells of the left column filled.

	unbounded	bounded
general	2-EXPTIME-c.	EXPSPACE-c.
while-free (wf)	2-EXPTIME-c.	Σ_3^p-c.
ontic	EXPSPACE-c.	?
wf, ontic	EXPSPACE-c.	Σ_2^p-c.
wf, epistemic	PSPACE-c.	?
wf, epist.+pos. goals	coNP-c.	Σ_2^p-c.

We do not know the complexity of KBP existence for while-free epistemic actions and arbitrary (not necessarily positive) goals (we only know that it is Σ_2^p-hard, and in Σ_3^p). Neither do we know the complexity of bounded plan existence with ontic actions and loops (other than membership in EXPSPACE).

7. REFERENCES

[1] C. Bäckström and P. Jonsson. Limits for compact representations of plans. In *Proc. ICAPS 2011*, pages 146–153, 2011.

[2] Chitta Baral, Vladik Kreinovich, and Raul Trejo. Computational complexity of planning and approximate planning in presence of incompleteness. In *IJCAI*, pages 948–955, 1999.

[3] Thomas Bolander and Mikkel Birkegaard Andersen. Epistemic planning for single and multi-agent systems. *Journal of Applied Non-Classical Logics*, 21(1):9–34, 2011.

[4] B. Bonet, H. Palacios, and H. Geffner. Automatic derivation of finite-state machines for behavior control. In *Proc. AAAI-10*, 2010.

[5] R.I. Brafman, J.Y. Halpern, and Y. Shoham. On the knowledge requirements of tasks. *Journal of Artificial Intelligence*, 98(1–2):317–350, 1998.

[6] J. Claßen and G. Lakemeyer. Foundations for knowledge-based programs using es. In *KR*, pages 318–318, 2006.

[7] R. Fagin, J. Halpern, Y. Moses, and M. Vardi. *Reasoning about Knowledge*. MIT Press, 1995.

[8] P. Haslum and P. Jonsson. Some results on the complexity of planning with incomplete information. In *Proc. 5th European Conference on Planning (ECP 1999)*, pages 308–318, 1999.

[9] A. Herzig, J. Lang, and P. Marquis. Action representation and partially observable planning in epistemic logic. In *Proceedings of IJCAI03*, pages 1067–1072, 2003.

[10] J.Halpern and Y. Moses. Characterizing solution concepts in games using knowledge-based programs. In *Proceedings of IJCAI-07*, 2007.

[11] Jérôme Lang and Bruno Zanuttini. Knowledge-based programs as plans - the complexity of plan verification. In *ECAI*, pages 504–509, 2012.

[12] N. Laverny and J. Lang. From knowledge-based programs to graded belief-based programs part i: On-line reasoning. *Synthese*, 147(2):277–321, 2005.

[13] N. Laverny and J. Lang. From knowledge-based programs to graded belief-based programs, part ii: off-line reasoning. In *IJCAI*, pages 497–502, 2005.

[14] Benedikt Löwe, Eric Pacuit, and Andreas Witzel. Del planning and some tractable cases. In *LORI*, pages 179–192, 2011.

[15] Rajdeep Niyogi and Ramaswamy Ramanujam. An epistemic logic for planning with trials. In *LORI*, pages 238–250, 2009.

[16] Ronald P. A. Petrick and Fahiem Bacchus. Extending the knowledge-based approach to planning with incomplete information and sensing. In *ICAPS*, pages 2–11, 2004.

[17] R. Reiter. On knowledge-based programming with sensing in the situation calculus. *ACM Trans. Comput. Log.*, 2(4):433–457, 2001.

[18] Jussi Rintanen. Complexity of planning with partial observability. In *ICAPS*, pages 345–354, 2004.

APPENDIX

A. SUCCINCTNESS

DEFINITION 4. *Let π be a KBP and M^0 be an initial knowledge state. The* standard policy *$f(\pi, M^0)$ induced by π and M^0 is defined inductively as follows:*

- *if π is the empty KBP, then $f(\pi, M^0)$ is the empty standard policy,*
- *if π is $\alpha; \pi'$ for an ontic action $\alpha \in A_O$, then $f(\pi, M^0)$ is $\alpha; f(\pi', \mathrm{Prog}(M^0, \alpha))$,*
- *if π is $\alpha; \pi'$ for an epistemic action $\alpha \in A_E$, then $f(\pi, M^0)$ is*

  α;
  if $\mathbf{K}\varphi_1$ ***then*** $f(\pi', \mathrm{Prog}(M^0, \mathbf{K}\varphi_1))$
  else if $\mathbf{K}\varphi_2$ ***then*** $f(\pi', \mathrm{Prog}(M^0, \mathbf{K}\varphi_2))$
  else ...
  endif

  with $\{\mathbf{K}\varphi_1, \mathbf{K}\varphi_2, \dots\} = \Omega_\alpha$,
- *if π is* ***if*** Φ ***then*** π_1 ***else*** π_2 ***endif*** $; \pi'$, *then (i) if $M^0 \models \Phi$ holds then $f(\pi, M^0)$ is $f(\pi_1; \pi', M^0)$, and (ii) otherwise (*i.e., *$M^0 \not\models \Phi$) $f(\pi, M^0)$ is $f(\pi_2; \pi', M^0)$,*
- *if π is* ***while*** Φ ***do*** π_1 ***endwhile*** $; \pi'$, *then (i) if $M^0 \models \Phi$ holds then $f(\pi, M^0)$ is $f(\pi_1; \pi, M^0)$, and (ii) otherwise (*i.e., *$M^0 \not\models \Phi$) $f(\pi, M^0)$ is $f(\pi', M^0)$.*

PROPOSITION 11. *Let π be a KBP and M^0 be an initial knowledge state. Then π and the standard policy $f(\pi, M^0)$ are equivalent in M^0.*

PROOF. It is easily shown by induction on the structure of π that for every possible outcome (resp. feedback) of an ontic (resp. epistemic) action taken in π, the iterated progression of M^0 by π or $f(\pi, M^0)$ are the same. □

PROPOSITION 12. *There is a family of planning problems $\mathcal{P} = (P_n)_{n\in\mathbb{N}}$ for which there is a succinct family of while-free KBPs $(\pi_n)_{n\in\mathbb{N}}$, and any family of KBPs for $\mathcal{P}$ is a (possibly nonuniform) family of algorithms for* 3SAT.

PROOF. Let $n \in \mathbb{N}$, implicitly defining a set of n Boolean variables and the SAT problem for 3CNF formulas over n variables. The variables and actions involved in the construction of π_n are the following:

- n unobservable Boolean variables $x_1, \dots, x_n$, intuitively storing an assignment $\vec{x}$ to the variables of a 3CNF formula (this assignment is arbitrary and unknown to the agent),
- $O(n^3 \times 3 \log n)$ Boolean variables $\ell_{i,j,k}$ ($i = 1, \dots, n^3$, $j = 1, 2, 3$, $k = 1, \dots, \log n$), intuitively encoding a 3CNF formula φ ($\ell_{i,j,k}$ represents the kth bit of the encoding of the literal in position j in the ith clause) ; the value of these variables, i.e., the 3CNF formula, is arbitrary, but can be "read" by a KBP through epistemic actions $\mathrm{test}(\ell_{i,j,k})$,
- an unobservable variable s ("satisfied") which is necessarily false if $\vec{x}$ does not satisfy φ ; to model this, the initial knowledge state is defined to be

  $$M_n^0 = \bigwedge_{i=1,\dots,n^3} \neg\chi_i \to \neg s$$

  where χ_i is true if and only if $\vec{x}$ satisfies the ith clause of φ (that is, χ_i is

  $$\bigvee_{x\in\{x_1,\dots,x_n\}} \left((x \wedge \bigvee_j \ell_{i,j} = x) \vee (\bar{x} \wedge \bigvee_j \ell_{i,j} = \bar{x}) \right)$$

  where $\ell_{i,j} = x$ is appropriately encoded over the "bits" $\ell_{i,j,k}$),
- ontic actions x_i^+ and x_i^-, for $i = 1, \dots, n$, setting x_i to 1 or 0, respectively.

The goal G_n of the planning problem P_n is either to know that s is false ($\mathbf{K}\bar{s}$) or to know that $\vec{x}$ is a model of φ ($\mathbf{K}(\vec{x} \models \varphi)$, expressed using a formula using the variables χ_i as above).

We claim that the KBP π_n defined as follows is a valid plan for P_n:

$\mathrm{test}(\ell_{1,1,1}); \mathrm{test}(\ell_{1,1,2}); \dots; \mathrm{test}(\ell_{n^3,3,\log n})$;
if $\mathbf{K}\bar{s}$ **then** stop
else
  if $\mathbf{K}\neg(\varphi \wedge x_1)$ **then** x_1^- **else** x_1^+ **endif**
  ...
  if $\mathbf{K}\neg(\varphi \wedge x_n)$ **then** x_n^- **else** x_n^+ **endif**

where $\mathbf{K}\neg(\varphi \wedge x_i)$ is a shorthand for $\mathbf{K}\neg(\chi_1 \wedge \dots \wedge \chi_{n^3} \wedge x_i)$.

Indeed, because the value of s cannot change during the execution, s is *guaranteed* to be false if and only if the (arbitrary) initial assignment $\vec{x}$ does not satisfy φ. Because the initial value of $\vec{x}$ cannot be observed, this is true if and only if φ is unsatisfiable. Otherwise, by definition an assignment to $\vec{x}$ can be built which satisfies φ. Finally, P_n encodes 3SAT for formulas of n variables, and π_n is a valid plan for it. □

PROPOSITION 13. *Given a KBP π and an initial knowlege state M^0, one can build a knowledge-based planning problem $P = (I, A_O, A_E, G)$ in time polynomial in $|\pi|$, so that π is valid for P and all KBPs which are valid for P are equivalent to π (up to additional variables in P and to a polynomial number of void actions).*

PROOF. Using a polynomial number of void actions (with theory $\Sigma = \bigwedge_{x\in X} x' \leftrightarrow x$ for ontic actions and $\Omega = \{\mathbf{K}\top\}$ for epistemic actions), we first normalize π so that it starts with an ontic action, then epistemic and ontic actions alternate, and finally that only ontic actions occur right before and right after any occurrence of **if** Φ **then** , **else** , **endif** , **while** Φ **do** , and **endwhile** . By duplicating actions, we also ensure that any action is used at most once in π; for example, we duplicate a to a^1, a^2, a^3, with $\Sigma_{a^i} = \Sigma_a$, for the first, second, and third occurrences of a in π. All these steps can clearly be performed in polynomial time.

We now describe how I, A_O, A_E, and G are computed from π. The constructions are performed iteratively, starting with $I = M^0$, A_O (resp. A_E) being the set of ontic (resp. epistemic) actions occurring in π, and $G = \mathbf{K}\top$.

We describe in details how to handle the case when π is a sequence of actions. Handling of branching and loops will

be described more briefly, but relies on the same techniques. So let $\pi = a_1; \ldots; a_k$ with $a_1, a_3, \ldots$ being ontic actions and $a_2, a_4, \ldots$ being epistemic actions.

We first introduce two fresh variables, ok and s, and replace I with $I \wedge \mathbf{K}ok$ and G with $G \wedge \mathbf{K}ok \wedge \neg(\mathbf{K}s \vee \mathbf{K}\bar{s})$. Intuitively, ok is known to be true at the beginning and must be known to be true at the end, but taking any ontic action at another moment than π does will assign it to false as a side-effect. Now the value of s (standing for "secret") is not known initially and must not be known at the end, but taking any epistemic action at another moment than π does will reveal its value.

Now for each sequence of actions $a_i; a_{i+1}; a_{i+2}$ in π, where a_i, a_{i+2} are ontic and a_{i+1} is epistemic, we introduce two fresh variables, r_{i+1} (standing for "ready" to execute a_{i+1}) and p_{i+2}. Intuitively, a_i will assign r_{i+1} to 1, and a_{i+1} will reveal the value of p_{i+2} (only in case r_{i+1} is known to be true). Then a_{i+2} is duplicated into two actions, exactly one of which has to be chosen, depending on the value of p_{i+2}. In this manner, we force a_{i+2} to occur only after a sequence $a_i; a_{i+1}$ in any valid plan.

More precisely, in A_O and A_E we:

- replace Σ_{a_i} with $\Sigma_{a_i} \wedge r'_{i+1}$,
- replace $\Omega_{a_{i+1}}$ with $\{\mathbf{K}(\varphi \wedge r_{i+1} \rightarrow p^{\epsilon}_{i+2}), \mathbf{K}(\varphi \wedge \bar{r}_{i+1}) \mid \mathbf{K}\varphi \in \Omega_{a_{i+1}}, \epsilon = 0, 1\}$,
- replace a_{i+2} with two ontic actions, namely a^p_{i+2} and $a^{\bar{p}}_{i+2}$ defined by

$$\begin{cases} \Sigma_{a^p_{i+2}} & = & \Sigma_{a_{i+2}} \wedge (ok' \leftrightarrow ok \wedge p_{i+2}) \wedge \bar{r}'_{i+2} \\ \Sigma_{a^{\bar{p}}_{i+2}} & = & \Sigma_{a_{i+2}} \wedge (ok' \leftrightarrow ok \wedge \bar{p}_{i+2}) \wedge \bar{r}'_{i+2} \end{cases}$$

  and make them reinitialize p_{i+2}, that is, the frame axiom $p'_{i+2} \leftrightarrow p_{i+2}$ is *not* in $\Sigma_{a^p_{i+2}}, \Sigma_{a^{\bar{p}}_{i+2}}$.

Note that because the process is iterated, the first transformation is in fact applied to $\Sigma_{a^p_i}$ and $\Sigma_{a^{\bar{p}}_i}$.

Moreover, for any other epistemic action $a \neq a_{i+1}$, we

- replace Ω_a with $\{\mathbf{K}(\varphi \wedge (r_{i+1} \rightarrow s^{\epsilon})) \mid \mathbf{K}\varphi \in \Omega_a, \epsilon = 0, 1\}$ or, in the general case where this transformation has already been performed for $r_{i_1}, \ldots, r_{i_k}$, we replace it with

$$\{\mathbf{K}(\varphi \wedge (r_{i_1} \vee \cdots \vee r_{i_k} \vee r_{i+1}) \rightarrow s^{\epsilon}) \mid \mathbf{K}\varphi \in \Omega_a, \epsilon = 0, 1\}$$

Finally, for handling the last action we introduce a fresh variable $stop$, and we replace I with $I \wedge \overline{stop}$, G with $G \wedge \mathbf{K}stop$, Σ_{a_k} with $\Sigma_{a_k} \wedge stop'$, we replace $ok' \leftrightarrow ok \wedge p^{\epsilon}_i$ with $ok' \leftrightarrow ok \wedge p^{\epsilon}_i \wedge \overline{stop}$ in all other (ontic) action theories, and duplicate each feedback $\mathbf{K}\omega$ in other action theories into $\mathbf{K}(\omega \wedge (stop \rightarrow s^{\epsilon}))$, $\epsilon = 0, 1$. For handling the first action, we replace I with $I \wedge \mathbf{K}r_1$, add $\bar{r}_1$ to σ_{a_1}, and add feedbacks $\mathbf{K}r_1 \rightarrow s^{\epsilon}$, $\epsilon = 0, 1$, to all epistemic actions.

We now claim that P as defined above has (a plan equivalent to) π as a valid plan, and that any other valid plan for it is equivalent to π (in both cases, up to void actions and additional variables).

As regards validity of π, consider the plan π' obtained from π by replacing all subsequences $a_i; a_{i+1}; a_{i+2}$ with

$$a_i; a_{i+1};\ \textbf{if } \mathbf{K}p_{i+2} \textbf{ then } a^p_{i+2} \textbf{ else } a^{\bar{p}}_{i+2} \textbf{ endif}$$

Then clearly, when execution comes to a_{i+1}, r_{i+1} is true (and known to be so), hence one of the feedbacks $\mathbf{K}(r_{i+1} \rightarrow p^{\epsilon}_{i+2})$ is obtained, revealing the truth value of p_{i+2}. Hence a^p_{i+2} or $a^{\bar{p}}_{i+2}$ is correctly chosen for preserving achievement of the goal $\mathbf{K}ok$. Moreover, because for all $j < i$, the value of r_j has been reinitialized by action a_j, the feedback of a_{i+1} gives no clue about the value of s (through $\mathbf{K}(r_j \rightarrow s^{\epsilon})$), preserving the goal $\neg(\mathbf{K}s \vee \mathbf{K}\bar{s})$.

Now let π' be any plan which is valid for P, and consider a fixed sequence of outcomes for ontic actions and feedbacks for epistemic actions, with the aim of showing that π' takes (up to void actions) the same actions as π, in the same order. The proof works by induction.

First assume that π' takes an ontic action $a_i \neq a_1$ as its first action. Then because of the effect $ok' \rightarrow r_i$ and since the value of r_i is not known in the initial state I, the goal $\mathbf{K}ok$ is not preserved. Since no action allows to set it back, this is a contradiction with the validity of π'. Now assume that π' takes an epistemic action a_i as its first action. Then because r_1 is true in the initial state, a_i reveals the value of s, a contradiction again since this value cannot change along the execution. Moreover, by construction the knowledge state resulting from taking a_1 satisfies $\mathbf{K}\bar{r}_1$ and $\mathbf{K}r_2$, no variable r_i ($i \neq 2$) is known to be true in it, and the value of no variable p_i is known.

We now consider the second action taken by π'. Because r_1 is false this cannot be a_1, and because the value of p_i is known for no i, this cannot be a^p_i nor $a^{\bar{p}}_i$, for any ontic action a_i. Hence this is an epistemic action, but because r_2 is true this can only be a_2 (otherwise the value of s would be revealed). Now by construction, the resulting knowledge state satisfies $\mathbf{K}\bar{r}_1$ and $\mathbf{K}r_2$, no variable r_i ($i \neq 2$) is known to be true in it, the value of p_3 is known in it, and finally the value of no other p_i is known.

Finally consider the third action taken by π'. Taking any ontic action other than a^p_3 or $a^{\bar{p}}_3$ would result in a blind choice of a^p_i or $a^{\bar{p}}_i$ since the value of p_i ($i \neq 3$) is not known. Now taking an epistemic action other than a_2 would reveal the value of s (since r_2 is known to be true). Finally, either π' takes a_2 again, which amounts to a void action, or it takes a_3. Now by construction, after a_3 is taken the knowledge state satisfies $\mathbf{K}r_4$, no variable r_i ($i \neq 4$) is known to be true (since a_3 assigns r_3 to false), and the value of no p_i is known (since a_3 reinitializes p_3). Hence we are in the same situation as after the first action has been taken, and the induction goes on, which concludes for KBPs π which are simple sequences of actions.

We now briefly show how to handle subprograms of the form

$$a;\ \textbf{if } \mathbf{K}\varphi \textbf{ then } b; \ldots \textbf{ else } c; \ldots \textbf{ endif }; \ldots$$

We introduce a new fluent, f ("forbidden"), and add $\neg\mathbf{K}f$ to the initial knowledge state and to the goal. Recall that due to the normalization step, actions a, b, c are all ontic. Then we

- replace Σ_a with $\Sigma_a \wedge r'_{b,c}$,
- replace Σ_b with $\Sigma_b \wedge ok' \leftrightarrow (ok \wedge r_{b,c} \wedge \varphi) \wedge \bar{r}'_{b,c}$,
- replace Σ_c with $\Sigma_c \wedge ok' \leftrightarrow (ok \wedge r_{b,c} \wedge (f' \leftrightarrow f \vee \varphi)) \wedge \bar{r}'_{b,c}$.

and as in the case of sequences, we add feedbacks to all epistemic actions, so that they reveal the value of s if executed when $r_{b,c}$ is known to be true. The construction ensures that executing b while $K\varphi$ is not true results in $\neg\mathbf{K}ok$, hence violating the goal, and that executing c while $K\varphi$ is true results in $\mathbf{K}f$, again violating the goal.

Finally, subprograms of the form

$$\textbf{while } \mathbf{K}\varphi \textbf{ do } a; \ldots; b \textbf{ endwhile } ; c$$

are handled exactly as if they were

$$\textbf{if } \mathbf{K}\varphi \textbf{ then } (a; \ldots; b; \textbf{ if } \mathbf{K}\varphi \textbf{ then } a \textbf{ else } c) \textbf{ else } c;$$

The fact that the two occurrences of a refer to exactly the same action simulate a "goto" construct and hence, ensure that a valid plan loops when necessary. □

PROPOSITION 14. *If while-free KBPs are as succinct as KBPs (with loops), then verifying a KBP with loops is a problem in* Σ_3^{P}.

PROOF. Let p be a polynomial such that for all KBPs π, there is an equivalent while-free KBP π' satisfying $|\pi'| \leq p(|\pi|)$. Then given a KBP π and a planning problem P, verifying that π is valid for P can be done by the following algorithm, which essentially guesses an equivalent while-free π' and verifies it instead of directly verifying π:

1. guess a while-free KBP π' of size at most $p(|\pi|)$,
2. check that π' and π are equivalent; the complement can be decided as follows:
   (a) guess a trace τ of size $|\pi'|$ and the corresponding sequence of outcomes of ontic actions and feedbacks of epistemic actions,
   (b) from the outcomes and feedbacks, compute the corresponding trace of π,
   (c) check that at some point, π and π' are not in the same knowledge state,
3. verify that π' is valid for P.

The traces in Item 2 can be represented in space polynomial in $|\pi'|$ using memoryful progression [11]. Checking that π and π' are in different knowledge states at some point can be done by verifying that their memoryful progressions are not equivalent over the variables of this timepoint, which is a problem in Σ_2^{P}(guess a disagreeing assignment and check that it can be extended to a model of one progression but none of the other).

Finally, Item 2 can be solved by a call to a Σ_2^{P}-oracle. Moreover, verifying a while-free KBP (Item 3) is a problem in Π_2^{p} [11, Proposition 2]. Finally, we get a nondeterministic algorithm using a Σ_2^{P}-oracle (or a Π_2^{p}-oracle), hence the whole problem is in Σ_3^{P}. □

B. PLAN EXISTENCE

PROPOSITION 15. *Plan existence is* Σ_2^{P}*-hard if only epistemic actions are allowed.*

PROOF. We give a reduction from $\mathrm{QBF}_{2,\exists}$. Let

$$\forall a_1 \ldots a_n \exists b_1 \ldots \exists b_p \varphi$$

be a QBF formula. We define an epistemic planning problem $P = (I, \emptyset, A_E, G)$ by:

- $I = \mathbf{K}\top$,
- $A_E = \{\mathrm{test}(a_1), \ldots, \mathrm{test}(a_n)\}$,
- $G = \neg\mathbf{K}\neg\varphi \wedge \bigwedge_{i=1}^{n}(\mathbf{K}a_i \vee \mathbf{K}\bar{a}_i)$.

Clearly, any valid plan for P must perform all actions in all branches, since $\mathrm{test}(a_i)$ is the only action revealing the value of a_i. Hence, there is a valid plan for P if and only if performing all actions in sequence constitutes a valid plan π. Now this KBP π is valid for P if and o nly if for every $\vec{a} \in 2^{\{a_1,\ldots,a_n\}}$, it holds $\mathbf{K}\vec{a} \models \neg\mathbf{K}\neg\varphi$, that is, for every $\vec{a} \in 2^{\{a_1,\ldots,a_n\}}$, there is a $\vec{b} \in 2^{\{b_1,\ldots,b_p\}}$ with $\vec{a}\vec{b} \models \varphi$. □

PROPOSITION 16. *Plan existence is* coNP*-complete if only epistemic actions are allowed and the goal is restricted to be a positive epistemic formula.*

PROOF. We first show membership. Because the goal is positive, it is easy to see that adding epistemic actions cannot render a valid plan invalid, and hence the problem amounts to deciding whether performing all actions in sequence constitutes a valid plan π. Because there are no ontic actions, and hence the state never changes, this amounts to checking that the formula $\bigwedge_{a\in A_E}(\bigvee_{\mathbf{K}\varphi_i\in\Omega_a}\varphi_i)$ entails G. We conclude by observing that this formula has size polynomial in $|A_E|$ and that the entailment test is one in propositional logic, hence in coNP.

Hardness follows from the following reduction from UNSATISFIABILITY: a propositional formula φ is unsatisfiable if and only if the planning problem with no action, initial knowledge state $\mathbf{K}\top$ and goal $\mathbf{K}\neg\varphi$ has a plan. □

PROPOSITION 17. *There is a polynomial-time reduction from* QBF *to* WFOE-EXISTENCE.

PROOF. Let $\psi = \exists a_1 \forall b_1 \ldots \exists a_k \forall b_k \varphi$ be a QBF, where $a_1, \ldots, a_k$ and $b_1, \ldots, b_k$ are Boolean variables (restricting the quantifiers to scope over only one variable is without loss of generality, since any QBF can be rewritten in this manner by introducing dummy variables). We define the following instance $P = (I, \emptyset, A_E, G, <)$ of WFOE-EXISTENCE, where intuitively a_i (resp. $\bar{a}_i$) is encoded by "revealing the value of x_i" (resp. "not revealing the value of x_i"), and b_i (resp. $\bar{b}_i$) is encoded by "y_i is (known to be) true" (resp. false):

- $I = \mathbf{K}\top$,
- $A_E = \{\mathrm{test}(x_i) \mid i = 1, \ldots, k\} \cup \{\mathrm{test}(y_i) \mid i = 1, \ldots, k\}$,
- $G = \varphi$ with $\begin{cases} a_i \text{ replaced with } \mathbf{K}x_i \vee \mathbf{K}\bar{x}_i \\ \bar{a}_i \text{ replaced with } \neg\mathbf{K}x_i \wedge \neg\mathbf{K}\bar{x}_i \\ b_i \text{ replaced with } \mathbf{K}y_i \\ \bar{b}_i \text{ replaced with } \mathbf{K}\bar{y}_i \end{cases}$,
- $<$ is $(\mathrm{test}(x_1), \mathrm{test}(y_1), \mathrm{test}(x_2), \ldots, \mathrm{test}(x_k), \mathrm{test}(y_k))$.

Assume first that there is a strategy σ witnessing the validity of ψ, and build a KBP π from σ by:

- replacing any decision node $a_i \leftarrow 1$ with the action $\mathrm{test}(x_i)$,
- replacing any decision node $a_i \leftarrow 0$ with the empty KBP,
- replacing any branching node on b_i with 1-child σ_1 and 0-child σ_0 with the KBP

  $$\mathrm{test}(y_i);\ \textbf{if } \mathbf{K}y_i \textbf{ then } \pi_1 \textbf{ else } \pi_0 \textbf{ endif}$$

  where π_1 (resp. π_0) is obtained recursively from σ_1 (resp. σ_0).

Clearly, the order of actions in π follows $<$. Now by construction, $\mathrm{test}(x_i)$ (resp. $\mathrm{test}(y_i)$) is the only action revealing the value of x_i (resp. y_i), and validity of π for P follows.

Conversely, let π be a KBP for P, and let π_N be its normalized, equivalent KBP, obtained by

- removing all nonatomic branching conditions, *e.g.*, by replacing a test **if** $\Phi \wedge \Psi$ **then** ... **endif** with the test **if** Φ **then if** Ψ **then** ... **endif** ,
- replacing each negative atomic branching condition of the form $\neg\mathbf{K}\ell$ with $\mathbf{K}\bar{\ell}$ if it has test(ℓ) as an ancestor on its branch, and with $\mathbf{K}\top$ otherwise (then simplifying),
- removing any occurrence of test(ℓ) which has test(ℓ) as its parent,
- pushing up any test, e.g., **if** $\mathbf{K}\ell$, right after the action test(ℓ) on the same branch, and reorganizing the KBP as necessary (since we are not concerned with size bounds, it does not matter if this incurs an explosion in size).

Then define a strategy σ from π_N by

- replacing test(x_i) with a decision node $a_i \leftarrow 1$,
- ignoring actions test(y_i),
- replacing **if** $\mathbf{K}x_i$ **then** π_1 **else** π_0 **endif** with σ_1 or with σ_0, arbitrarily, where σ_1 (resp. σ_0) is obtained recursively from π_1 (resp. π_0),
- replacing **if** $\mathbf{K}y_i$ **then** π_1 **else** π_0 **endif** with a branching node on b_i, with 1-child σ_1 and 0-child σ_0.

Clearly, σ witnesses the validity of the QBF ψ. Why σ_1 or σ_0 can be chosen arbitrarily in the third item is because x_i and $\bar{x}_i$ play a symmetric role in P. □

Proposition 18. *There is a polynomial-time reduction from* WFOE-Existence *to* WFE-Existence.

Proof. Let $P = (I, \emptyset, A_E, G, <)$ be an instance of WFOE-Existence, and write $A_E = \{a_1, \ldots, a_n\}$ with $a_i < a_{i+1}$ for all i. We define an instance $P' = (I', \emptyset, A'_E, G')$ which forces the actions to occur in order in any valid plan. To do so, for each action $a_i \in A_E$ we essentially (i) duplicate a_i into two actions, a_i^p and a_i^n, and (ii) modify the feedback of a_{i-1} such that it reveals the value of an otherwise hidden variable p_{i-1}. Then we modify the goal G so that a_i^p must be taken if a_{i-1} yielded $\mathbf{K}p_{i-1}$, and a_i^n must be taken if a_{i-1} yielded $\mathbf{K}\bar{p}_{i-1}$ ("p" stands for "positive" and "n" for "negative"). In this manner, a valid plan must execute a_{i-1} before a_i, for otherwise it cannot choose between a_i^p and a_i^n.

More precisely, for each action $a_i \in A_E$ we introduce two fresh variables, p_i and n_i, and four more, $\mu_i^p, \mu_i^n, \mu_i^{\bar{p}}, \mu_i^{\bar{n}}$, which act as mutexes between the "twin" actions a_i^p and a_i^n. Then we define the following actions:

- a_i^p, representing the action to take when a_{i-1} yielded $\mathbf{K}p_{i-1}$ or $\mathbf{K}\bar{n}_{i-1}$, with feedback theory $\Omega_{a_i^p} = \{\mathbf{K}(\varphi \wedge p_i^\delta \wedge (\mu_i^p)^\epsilon) \mid \mathbf{K}\varphi \in \Omega_{a_i}, \delta, \epsilon = 0, 1\}$,
- a_i^n (dually), with feedback theory $\Omega_{a_i^n} = \{\mathbf{K}(\varphi \wedge n_i^\delta \wedge (\mu_i^n)^\epsilon) \mid \mathbf{K}\varphi \in \Omega_{a_i}, \delta, \epsilon = 0, 1\}$,
- $a_i^{\bar{p}}$, representing the "pass" action when a_{i-1} yielded $\mathbf{K}p_{i-1}$ or $\mathbf{K}\bar{n}_{i-1}$, with feedback theory $\Omega_{a_i^{\bar{p}}} = \{\mathbf{K}(p_i^\delta \wedge (\mu_i^{\bar{p}})^\epsilon) \mid \delta, \epsilon = 0, 1\}$,
- $a_i^{\bar{n}}$, with feedback theory $\Omega_{a_i^{\bar{n}}} = \{\mathbf{K}(n_i^\delta \wedge (\mu_i^{\bar{n}})^\epsilon) \mid \delta, \epsilon = 0, 1\}$.

We define A'_E to be $\{a_i^p, a_i^n, a_i^{\bar{p}}, a_i^{\bar{n}} \mid i = 1, \ldots, n\}$, and we define the goal G' to be:

$$G \wedge \left\{ \begin{array}{lll} & \bigwedge_{i=2}^{n} & (\mathbf{K}p_{i-1} \vee \mathbf{K}\bar{n}_{i-1}) \rightarrow (\mathbf{K}p_i \vee \mathbf{K}\bar{p}_i) \\ \wedge & \bigwedge_{i=2}^{n} & (\mathbf{K}\bar{p}_{i-1} \vee \mathbf{K}n_{i-1}) \rightarrow (\mathbf{K}n_i \vee \mathbf{K}\bar{n}_i) \\ \wedge & \bigwedge_{\substack{i=1,\ldots,n \\ a,b \in \{p,n,\bar{p},\bar{n}\} \\ a \neq b}} & (\neg\mathbf{K}\mu_i^a \wedge \neg\mathbf{K}\bar{\mu}_i^a) \vee (\neg\mathbf{K}\mu_i^b \wedge \neg\mathbf{K}\bar{\mu}_i^b) \end{array} \right.$$

Finally, we define $I' = I$, and we show that there is a valid KBP π for P if and only if there is a valid KBP π' for P'.

First let π be a valid KBP for P. We build a KBP π' as follows. We replace each occurrence of an action a_i in π with **if** $\mathbf{K}p_{i-1} \vee \mathbf{K}\bar{n}_{i-1}$ **then** a_i^p **else** a_i^n **endif** . Now for each nonoccurrence of a_i in π, *i.e.*, at each place where a_{i-1} occurs right before a_{i+d}, $d > 1$, we insert a "pass" action by inserting the KBP **if** $\mathbf{K}p_{i-1} \vee \mathbf{K}\bar{n}_{i-1}$ **then** $a_i^{\bar{p}}$ **else** $a_i^{\bar{n}}$ **endif** . It is easily shown by induction on π' that each time a variant of action a_i is taken, either the value of p_{i-1} or the value of n_{i-1} is indeed known, and validity of π' follows.

Conversely, let π' be a valid KBP for P'. Because of the mutexes μ_i^a, at most one variant of each action a_i can occur along any branch of π'. Moreover, if, say, a_i^p occurs twice along a branch, then the deepest occurrence can be removed without changing the validity of π', since there are only epistemic actions and hence, the state never changes. Finally, because of the first and second sets of clauses in G', starting from the first action in π' all other actions must follow in order. Hence a valid KBP π for P can be built by replacing a_i^p or a_i^n with a, ignoring all "pass" actions $a_i^{\bar{p}}, a_i^{\bar{n}}$, and finally removing all tests on fresh variables p_i, n_i, and μ_i^a's, keeping the "else" or "then" subprogram arbitrarily. By construction, the resulting KBP π is valid for P, and the order of actions in π respects $<$. □

Proposition 19. *While-free bounded KBP existence with a positive epistemic goal is Σ_3^p-hard.*

Proof. Let $\psi = \exists a_1 \ldots a_n \forall b_1 \ldots b_p \exists c_1 \ldots c_q \varphi$ be an instance of $\mathrm{QBF}_{3,\exists}$. Without loss of generality, we assume $n = p$ (otherwise we add dummy variables). We define an instance $P = (I, A_O, A_E, G)$ of while-free bounded KBP existence by:

- $I = \mathbf{K}\top$,
- $A_O = \{\alpha_i^+, \alpha_i^- \mid i = 1, \ldots, n\} \cup \{\gamma_j^+, \gamma_j^- \mid j = 1, \ldots, q\}$, where:
  - α_i^+ (resp. α_i^-) assigns 1 (resp. 0) to a_i and, as a side effect, nondeterministically reassigns all b_j's,
  - γ_i^+ (resp. γ_j^-) assigns 1 (resp. 0) to c_j,
- $A_E = \{\mathrm{test}(a_i \leftrightarrow b_i) \mid i = 1, \ldots, n\}$,
- $G = \mathbf{K}\varphi \wedge \bigwedge_{j=1}^{p} (\mathbf{K}b_j \vee \mathbf{K}\bar{b}_j)$,
- $k = 2n + (|\varphi| + 3)q$.

Assume that ψ is a positive instance of $\mathrm{QBF}_{3,\exists}$. Then there exists an assignment $\vec{a} \in 2^{\{a_1,\ldots,a_n\}}$ and a conditional assignment $f : 2^{\{b_1,\ldots,b_p\}} \rightarrow 2^{\{c_1,\ldots,c_q\}}$ such that for each $\vec{b} \in 2^{\{b_1,\ldots,b_p\}}$, $\vec{a}\vec{b}f(\vec{b})$ satisfies φ. Let $\alpha_i^* = \alpha_i^+$ if a_i is assigned 1 in $\vec{a}$ and $\alpha_i^* = \alpha_i^-$ if it is assigned 0. Let π be the following KBP:

$$\begin{array}{l} \alpha_1^*; \ldots; \alpha_n^*; \mathrm{test}(a_1 \leftrightarrow b_1); \ldots; \mathrm{test}(a_n \leftrightarrow b_n); \\ \quad \textbf{if } \mathbf{K}(\varphi \rightarrow c_1) \textbf{ then } \gamma_1^+ \textbf{ else } \gamma_1^-; \\ \ldots; \\ \quad \textbf{if } \mathbf{K}(\varphi \rightarrow c_q) \textbf{ then } \gamma_q^+ \textbf{ else } \gamma_q^-; \end{array}$$

Clearly, π is a valid plan for P, and its size is $2n + (|\varphi| + 3)q$.

Conversely, assume I is a negative instance of $\mathrm{QBF}_{3,\exists}$, that is, for every assignment $\vec{a} \in 2^{\{a_1,\ldots,a_n\}}$ there is an assignment $g(\vec{a}) \in 2^{\{b_1,\ldots,b_p\}}$ such that for each $\vec{c} \in 2^{\{c_1,\ldots,c_q\}}$, $\vec{a}g(\vec{a})\vec{c}$ satisfies $\neg\varphi$. We claim that there is no valid plan π for P — and *a fortiori*, no valid plan of size at most $\leq 2n + (|\varphi| + 3)q$. Indeed, assume there is a plan π for P.

First, the only way of knowing the truth value of the b_i's it to perform test$(a_i \leftrightarrow b_i)$ after an action α_i^+ or α_i^-. Therefore, every execution of π must contain at least an action α_i^+ or α_i^- and further on, test$(a_i \leftrightarrow b_i)$. Moreover, if another action α_j^+ or α_j^- appears later in the execution, after test$(a_i \leftrightarrow b_i)$ has been performed, then, because all variables $b_1, \ldots, b_p$ are nondeterministically reassigned, test$(a_i \leftrightarrow b_i)$ has to be performed again after that. Therefore, each execution of π must contain, in a first part, at least an action α_i^+ or α_i^- for every i, then, in a second part, all actions test$(a_i \leftrightarrow b_i)$ and no action α_i^+ nor α_i^- (but possibly some actions γ_i^+ or γ_i^-).

Now consider an execution e of π, and for $i = 1, \ldots, n$, let $v_i(e) = 1$ (resp. 0) if the last occurrence of an action α_i^+ or α_i^- is α_i^+ (resp. α_i^-), and let $\vec{a}(e) \in 2^{\{a_1,\ldots,a_n\}}$ be the corresponding assignment. Moreover, consider the point in the execution e just after the last action α_i^+ or α_i^- has been performed. After this point, all actions test$(a_i \leftrightarrow b_i)$ are executed. Consider the particular execution e' where the results of these actions are such that the revealed truth value of the variables $b_1, \ldots, b_p$ constitute exactly the assignment $g(\vec{a})$. The actions γ_i^+, γ_i^- taken (before or after this point or after it) result in an assignment $\vec{c}$ of $c_1, \ldots, c_q$. Now, by assumption, $\vec{a}g(\vec{a})\vec{c}$ does not satisfy φ, therefore this particular execution does not satisfy the goal, contradicting the validity of π. □

PROPOSITION 20. *While-free bounded KBP existence restricted to epistemic actions and to positive goals is Σ_2^{P}-hard.*

PROOF. We give a reduction from $\mathrm{QBF}_{2,\exists}$. Let $\psi = \exists a_1 \ldots a_n \forall b_1 \ldots \exists b_p \varphi$ be a QBF formula. We build a planning problem P as follows:

- we use propositional symbols $a_1, \ldots, a_n$, $b_1, \ldots, b_p$, c, $d_1, \ldots, d_n$,
- $A_E = \{\alpha_1, \ldots, \alpha_n, \beta_1, \ldots, \beta_n\}$ defined by the feedback theories

$$\Omega_{\alpha_i} = \{ \mathbf{K}(c \to a_i) \wedge d_i, \mathbf{K}(c \to a_i) \wedge \neg d_i, \mathbf{K}(c \wedge \neg a_i \wedge d_i), \mathbf{K}(c \wedge \neg a_i \wedge \neg d_i) \}$$

$$\Omega_{\beta_i} = \{ \mathbf{K}(c \to \neg a_i) \wedge d_i, \mathbf{K}(c \to \neg a_i) \wedge \neg d_i, \mathbf{K}(c \wedge a_i \wedge d_i), \mathbf{K}(c \wedge a_i \wedge \neg d_i) \}$$

- $G = \bigwedge_{i=1,\ldots,n} (\mathbf{K}d_i \vee \mathbf{K}\neg d_i) \wedge (\mathbf{K}c \vee \mathbf{K}(c \to \varphi))$,
- $k = n$.

If ψ is valid then let $\vec{a} \in 2^{\{a_1,\ldots,a_n\}}$ be an assignment which witnesses this fact. Let π the KBP $\gamma_1; \ldots; \gamma_n$, where γ_i is α_i if $\vec{a}$ assigns 1 to a_i, and γ_i is β_i if it assigns 0 to it. After every possible execution of π, either the agent knows c, or it knows $\bigwedge_i (c \to \vec{a})$; in the latter case, because $\vec{a}\vec{b} \models \varphi$ for all $\vec{b}$, the agent knows $c \to \varphi$, hence in both cases the second part of the goal is satisfied. Finally, by construction the agent knows the truth value of each d_i, and hence π is a valid plan containing exactly n actions.

Conversely, assume that there is a valid plan of size $\leq n$. Because the agent must learn the truth value of each d_i, π must contain α_i or β_i for each i, and since π is of size n, it contains exactly one of α_i or β_i for each i. Now consider the execution of π in which the sequence of observations is of the form $\mathbf{K}(c \to a_1^{\epsilon_1}) \wedge d_1^{\delta_1}, \ldots, \mathbf{K}(c \to a_n^{\epsilon_n}) \wedge d_n^{\delta_n}$. After this execution, the agent does not know c, therefore, since π is valid, it knows $c \to \varphi$. This means that $\bigwedge_i (c \to a_i^{\epsilon_i}) \wedge \bigwedge_i d_i^{\delta_i}$ entails $\mathbf{K}(c \to \varphi)$, which entails $\bigwedge_i (c \to a_i^{\epsilon_i}) \models c \to \varphi$, which is itself equivalent to $\bigwedge_i a_i^{\epsilon_i} \models \varphi$ and hence, $\exists a_1 \ldots a_n \forall b_1 \ldots \exists b_p \varphi$ is a valid instance of $\mathrm{QBF}_{2,\exists}$. □

R.E. Axiomatization of Conditional Independence

Pavel Naumov
Department of Mathematics
and Computer Science
McDaniel College
Westminster, Maryland, USA
pnaumov@mcdaniel.edu

Brittany Nicholls
Department of Mathematics
and Computer Science
McDaniel College
Westminster, Maryland, USA
brn002@mcdaniel.edu

ABSTRACT

The paper investigates properties of the conditional independence relation between pieces of information. This relation is also known in the database theory as embedded multivalued dependency. In 1980, Parker and Parsaye-Ghomi established that the properties of this relation can not be described by a finite system of inference rules. In 1995, Herrmann proved that the propositional theory of this relation is undecidable. The main result of this paper is a complete recursively enumerable axiomatization of this theory.

1. INTRODUCTION

In this paper, we study the properties of interdependencies between pieces of information. We call these pieces *secrets* to emphasize the fact that they might be unknown to some parties. For example, if secret a is the area of a triangle and secret p is the perimeter of the same triangle, then there is an interdependence between these secrets in the sense that not every value of secret a is compatible with every value of secret p. If there is no interdependence between two secrets, then we say that the two secrets are *independent*. In other words, secrets a and b are independent if each possible value of secret a is compatible with each possible value of secret b. We denote this relation between two secrets by $a \parallel b$. This relation was introduced by Sutherland [18] and is sometimes referred to as *nondeducibility*. Halpern and O'Neill [6] proposed a closely related notion called f-secrecy. Donders, More, and Naumov described properties of a multi-argument variation $a_1 \parallel a_2 \parallel \cdots \parallel a_n$ of the same relation under the assumption that the secrets are generated over an undirected graph [12], a directed acyclic graph [2], or a hypergraph [11] with a fixed topology.

Independence relation can be generalized to relate two sets of secrets. If A and B are two such sets, then $A \parallel B$ means that any consistent combination of values of secrets in set A is compatible with any consistent combination of values of secrets in set B. Note that "consistent combination" is an important condition here since some interdependence may exist between secrets in set A even while the entire set of secrets A is independent from the secrets in set B. A sound and complete axiomatization of this relation between sets of secrets was given by More and Naumov [10]:

1. *Empty Set*: $\varnothing \parallel A$,
2. *Monotonicity*: $A, B \parallel C \to A \parallel C$,
3. *Symmetry*: $A \parallel B \to B \parallel A$,
4. *Exchange*: $A, B \parallel C \to (A \parallel B \to A \parallel B, C)$,

where here and everywhere below by A, B we mean the union of the sets A and B. The same axioms were shown by Geiger, Paz, and Pearl [3] to provide a complete axiomatization of the independence relation between sets of random variables in probability theory. More recently, the same system was shown to be sound an complete with respect to concurrency [14] and game [15] semantics.

Suppose now that a, b, c, and d are four secrets with integer values such that $a+b+c+d \equiv 0 \pmod 2$. Note that $a \parallel b$ is true since every possible value of a is consistent with any possible value of b. At the same time, if values of c and d are fixed, then not every possible value of secret a is compatible with every possible value of secret b. We will say that secrets a and b are not independent conditionally on c, d and denote this by $\neg(a \parallel_{c,d} b)$. On the other hand, if only value of c is fixed, then any value of a is still consistent with any value of b. We write this as $a \parallel_c b$. In general, conditional independence relation $A \parallel_C B$ can be defined between any three disjoint sets of secrets. This relation, which is also known in the database theory as *embedded multivalued dependency*, has many non-trivial properties. For example, later we will show soundness of the following principles:

$$A \parallel_C B \wedge A \parallel_{B,C} D \to A \parallel_C B, D,$$

$$A, B \parallel_C D \to A \parallel_{B,C} D,$$

$$B \parallel_A C \wedge E \parallel_B D \wedge D \parallel_C F \wedge E \parallel_D F \wedge A \parallel_E F \to E \parallel_A F.$$

Parker and Parsaye-Ghomi [16] have shown that this relation can not be described by a finite system of inference rules. Herrmann [7, 8] proved the undecidability of the propositional theory of this relation. Lang, Liberatore, and Marquis [9] studied complexity of conditional independence between sets of propositional variables. Studený [17] has shown that the related conditional independence in probability theory has no complete finite characterization. More recently, Grädel and Väänänen discussed (incomplete) logical systems describing properties of the conditional independence in propositional and first order languages [4] and suggested model checking game semantics for these systems [5].

The main result of this paper is a complete infinite recursively enumerable axiomatization of the propositional theory of the relation $A \parallel_C B$. This work builds on the techniques from our previous TARK paper [13], where we gave a complete axiomatization of a different ternary knowledge relation. The "diagram" notion used in the current paper is a generalization of the "diamond" notations from the previous paper.

2. SYNTAX AND SEMANTICS

We assume a fixed alphabet of "secret" variables: $a, b, \ldots$.

DEFINITION 1. *By the set of formulas Φ we mean the minimal set of formulas such that*

1. $\bot \in \Phi$,
2. $A \parallel_C B \in \Phi$ *for each pairwise disjoint sets of secret variables A, B, and C,*
3. $\varphi_1 \to \varphi_2 \in \Phi$ *if* $\varphi_1, \varphi_2 \in \Phi$.

As usual, all other boolean connectives are assumed to be defined through the implication and the constant false.

DEFINITION 2. *A protocol is a pair $\mathcal{P} = \langle V, R\rangle$, where,*

1. *for any secret variable a, set $V(a)$ is an arbitrary set of "values" of secret a,*
2. *R is a set of functions r on secret variables such that $r(a) \in V(a)$ for any secret variable a. Elements of R will be called "runs" of the protocol.*

For any set of secret variables A and any runs r_1 and r_2, we write $r_1 \equiv_A r_2$ if $r_1(a) = r_2(a)$ for any $a \in A$. The next definition is the core definition of this paper. Item 3 below formally defines conditional independence relation between sets of secrets.

DEFINITION 3. *For any protocol $\mathcal{P} = \langle V, R\rangle$ and any formula $\varphi \in \Phi$, we define the binary relation $\mathcal{P} \vDash \varphi$ as follows:*

1. $\mathcal{P} \nvDash \bot$,
2. $\mathcal{P} \vDash A \parallel_C B$ *if and only if, for any $r_1, r_2 \in R$, such that $r_1 \equiv_C r_2$, there is $r \in R$ such that $r_1 \equiv_{A,C} r \equiv_{B,C} r_2$.*
3. $\mathcal{P} \vDash \varphi \to \psi$ *if and only if* $\mathcal{P} \nvDash \varphi$ *or* $\mathcal{P} \vDash \psi$.

3. GRAPH NOTATIONS

In this paper we deal with graphs that might have directed as well as undirected edges. An example G of such graph is depicted in Figure 1. We use word "path" for any sequences of adjacent vertices without taking into account the directions of edges. For example, sequence of vertices v_1, v_2, v_3 is a path in graph G. The graphs that we consider are "labeled". By that we mean that each edge of the graph is labeled with a *set* of secret variables. If vertices u and w of the graph are connected by a path such that each edge of the path is labeled with a set containing label x, then we write $u \sim_x w$. For example, $v_1 \sim_a v_3$ in graph G. We allow paths that consist of just a single vertex. This assumption implies that relation $u \sim_x w$ is an equivalence relation on graphs for any fixed label x.

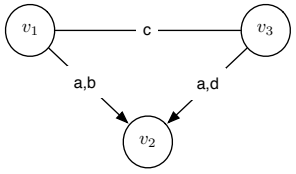
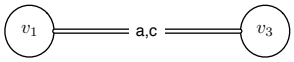


Figure 1: Graph G

For any *set* of labels X, we write $u \sim_X w$ if $u \sim_x w$ for each $x \in X$. For example, $v_1 \sim_{a,c} v_3$ in graph G. Note that a-path and c-path from v_1 to v_3 are not the same. Relation $u \sim_X w$ is also an equivalence relation on vertices for any fixed set of secret variables X. Sometimes we draw only a fragment of a graph. To show that vertices u and w are in relation $u \sim_X w$ on the whole graph, we connect vertices u and w in our partial drawing by a *double line* labeled with set X. For example,



is a partial drawing of the graph G from Figure 1.

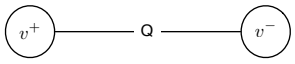
4. DIAGRAMS

The description of the axiomatic system for conditional independence proposed in this paper is using the notion of a diagram. Informal drawing similar to our diagrams have been used before to visualize arguments about *specific* properties of conditional independence. See, for example, illustrations in Parker and Parsaye-Ghomi [16]. In this work, however, we give such drawings a precise mathematical definition and show, through the proof of completeness theorem, that *all* properties of conditional independence can be observed by analyzing the diagrams.

A diagram is a labeled graph with a special structure. For each diagram Δ there is a set of formulas $[\Delta]$ that, informally, is used to "construct" the diagram. Formally, the diagrams and the corresponding sets of formulas are defined below.

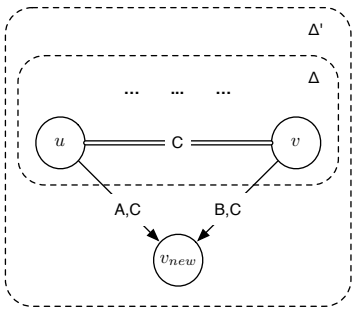
DEFINITION 4. *For any set of secret variables Q, the set of diagram $Diag(Q)$ is the minimal set such that*

1. *it contains the "basic" diagram Δ_0 consisting of two vertices, called v^+ and v^-, and an undirected edge between v^+ and v^- labeled with Q:*



   By definition, set $[\Delta_0]$ is empty.

2. *For any pair-wise disjoint sets A, B, and C, and any two vertices u and v of a diagram $\Delta \in Diag(Q)$, such that $u \sim_C v$, there is a diagram $\Delta' \in Diag(Q)$:*



   such that

   (a) *Diagram Δ', in addition to all vertices of the diagram Δ, contains a new vertex v_{new},*

   (b) *Diagram Δ', in addition to all edges of the diagram Δ, contains two new directed edges (u, v_{new}) and (v, v_{new}) labeled by sets $A \cup C$ and $B \cup C$ respectively.*

(c) $[\Delta'] = [\Delta] \cup \{A \parallel_C B\}$.

If diagrams Δ and Δ' are related as described above, then we say that diagram Δ' is an extension of the diagram Δ. The same diagram Δ has multiple extensions. The unique vertices v^+ and v^- from which construction of a diagram Δ was started will be referred to as v^+_Δ and v^-_Δ. Note that if $\Delta \in Diag(Q)$, then $v^+_\Delta \sim_Q v^-_\Delta$.

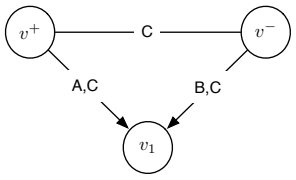


Figure 2: Diagram $\Delta_1 \in Diag(C)$

For example, diagram Δ_1 in Figure 2 is obtained from the basic diagram through a single extension using sets A, B, and C. Thus, $[\Delta_1] = \{A \parallel_C B\}$.

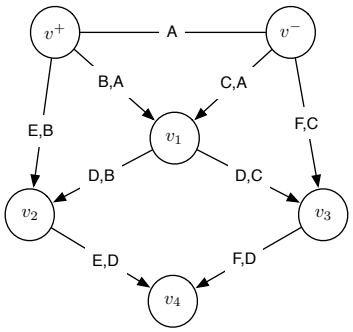


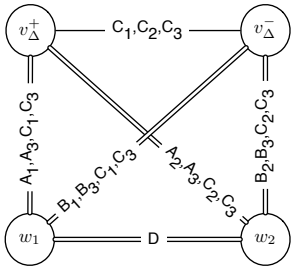
Figure 3: Diagram $\Delta_2 \in Diag(A)$

On the other hand, diagram Δ_2 in Figure 3 can be constructed from the basic diagram by first adding vertex v_1, next v_2, next v_3, and finally v_4. Alternatively, the order can be v_1, v_3, v_2, and v_4. In either case, $[\Delta_2] = \{(B \parallel_A C), (E \parallel_B D), (D \parallel_C F), (E \parallel_D F)\}$. Note that vertex v_4 was added in spite of the lack of a direct edge from vertex v_2 to vertex v_3. For the diagram to extand to v_4 we only require $v_2 \sim_D v_3$.

Definition 5. *Let*

$$t = (A_1, A_2, A_3; B_1, B_2, B_3; C_1, C_2, C_3; D)$$

be a tuple of disjoint sets of labels. We say that diagram $\Delta \in Diag(C_1 \cup C_2 \cup C_3)$ renders tuple t if diagram Δ contain vertices w_1 and w_2 such that

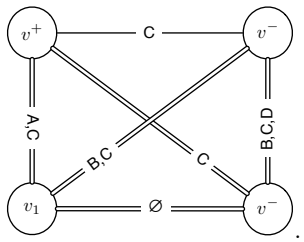


or, in other words, $w_1 \sim_{A_1,A_3,C_1,C_3} v^+_\Delta$; $w_1 \sim_{B_1,B_3,C_1,C_3} v^-_\Delta$; $w_2 \sim_{A_2,A_3,C_2,C_3} v^+_\Delta$; $w_2 \sim_{B_2,B_3,C_2,C_3} v^-_\Delta$; $w_1 \sim_D w_2$.

For example, Diagram Δ_1, depicted in Figure 2, renders (with $w_1 = v_1$ and $w_2 = v^+$) tuple

$$(A, \varnothing, \varnothing; \varnothing, D, B; \varnothing, \varnothing, C; \varnothing)$$

for an arbitrary set of secrets D, because

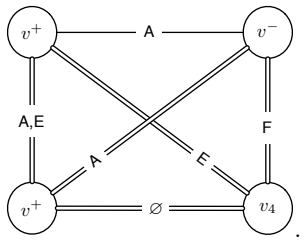


Here, of course, we use the fact that $v^- \sim_{B,C,D} v^-$ for each set of secrets D.

As another example, Diagram Δ_2, depicted in Figure 3, renders (with $w_1 = v^+$ and $w_2 = v_4$) tuple

$$(\varnothing, \varnothing, E; \varnothing, F, \varnothing; A, \varnothing, \varnothing; \varnothing),$$

because



5. AXIOMS

In this section we introduce a logical system describing properties of conditional independence. The axioms of the system are:

1. Symmetry: $A \parallel_C B \to B \parallel_C A$,
2. Monotonicity: $A \parallel_C B, D \to A \parallel_C B$,
3. Diagram:

$$\wedge[\Delta] \quad \to \quad (A_1, B_1, C_1 \parallel_{A_3,B_3,C_3,D} A_2, B_2, C_2 \to A_1, A_2, A_3 \parallel_{C_1,C_2,C_3} B_1, B_2, B_3),$$

if diagram Δ renders tuple

$$(A_1, A_2, A_3; B_1, B_2, B_3; C_1, C_2, C_3; D)$$

and $\wedge[\Delta]$ stands for conjunction of all formulas in $[\Delta]$.

We write $\vdash \varphi$ if formula $\varphi \in \Phi$ is provable from the above axioms and propositional tautologies in the language Φ using Modes Ponens inference rule. We write $X \vdash \varphi$ if formula φ is provable in our logical system using an additional set of axioms X.

Theorem 1. *The set of axioms of this logical system is recursively enumerable.*

Proof. The statement of the theorem follows from recursive enumerability of diagrams, recursive enumerability of tuples, and decidability of "diagram renders tuple" relation. □

6. EXAMPLES

In this section we give several examples of formal proofs in our logical system. The soundness of the axioms will be shown in Section 7. We start with the three non-trivial properties of the conditional independence mentioned in the introduction.

PROPOSITION 1. $\vdash A \parallel_C B \wedge A \parallel_{B,C} D \rightarrow A \parallel_C B, D$.

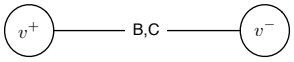
PROOF. Consider diagram Δ_1 depicted in Figure 2. As we have shown in Section 4, this diagram renders tuple $(A, \varnothing, \varnothing; \varnothing, D, B; \varnothing, \varnothing, C; \varnothing)$. Thus, by the Diagram axiom,

$$\vdash [\Delta_1] \rightarrow (A \parallel_{B,C} D \rightarrow A \parallel_C B, D).$$

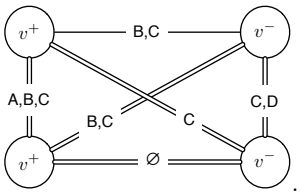
Recall from Section 4 that $[\Delta_1] = \{A \parallel_C B\}$. Therefore, $\vdash A \parallel_C B \rightarrow (A \parallel_{B,C} D \rightarrow A \parallel_C B, D)$. □

PROPOSITION 2. $\vdash A, B \parallel_C D \rightarrow A \parallel_{B,C} D$.

PROOF. Consider basic diagram Δ_3:



This diagram renders (with $w_1 = v^+$ and $w_2 = v^-$) tuple $(A, \varnothing, \varnothing; \varnothing, D, \varnothing; B, \varnothing, C; \varnothing)$ because



Hence, by the Diagram axiom,

$$\vdash \wedge[\Delta_3] \rightarrow (A, B \parallel_C D \rightarrow A \parallel_{B,C} D).$$

Recall that Δ_3 is a basic diagram. Thus, by Definition 4, set $[\Delta_3]$ is empty. Therefore, $\vdash A, B \parallel_C D \rightarrow A \parallel_{B,C} D$. □

PROPOSITION 3.

$$\vdash B \parallel_A C \wedge E \parallel_B D \wedge D \parallel_C F \wedge E \parallel_D F \wedge A \parallel_E F \rightarrow E \parallel_A F.$$

PROOF. Consider diagram Δ_2 depicted in Figure 3. As we have shown in Section 4, this diagram renders tuple

$$(\varnothing, \varnothing, E; \varnothing, F, \varnothing; A, \varnothing, \varnothing; \varnothing).$$

Thus, by the Diagram axiom,

$$\vdash [\Delta_2] \rightarrow (A \parallel_E F \rightarrow E \parallel_A F).$$

Recall from Section 4 that

$$[\Delta_2] = \{(B \parallel_A C), (E \parallel_B D), (D \parallel_C F), (E \parallel_D F)\}.$$

Therefore,

$$\vdash B \parallel_A C \wedge E \parallel_B D \wedge D \parallel_C F \wedge E \parallel_D F \wedge A \parallel_E F \rightarrow E \parallel_A F.$$

□

As our final example, we prove the Exchange axiom mentioned in the introduction. Although it is a property of non-conditional independence, it can be rephrased in the language of the conditional independence.

PROPOSITION 4.

$$\vdash A, B \parallel_\varnothing C \rightarrow (A \parallel_\varnothing B \rightarrow A \parallel_\varnothing B, C).$$

PROOF. Suppose that $A, B \parallel_\varnothing C$. Thus, $A \parallel_B C$ by Proposition 2. Therefore, by Proposition 1 and due to the assumption $A \parallel_\varnothing B$, we can conclude that $A \parallel_\varnothing B, C$. □

7. SOUNDNESS

We prove soundness of each axiom as a separate lemma.

LEMMA 1 (SYMMETRY). *For any protocol $\mathcal{P} = (V, R)$, if $\mathcal{P} \vDash A \parallel_C B$, then $\mathcal{P} \vDash B \parallel_C A$.*

PROOF. Assume that $r_1 \equiv_C r_2$ for some runs $r_1, r_2 \in R$. Thus, $r_2 \equiv_C r_1$. Hence, by the assumption of the lemma, there is $r \in R$ such that $r_2 \equiv_{A,C} r \equiv_{B,C} r_1$. Therefore, $r_1 \equiv_{B,C} r \equiv_{A,C} r_2$. □

LEMMA 2 (MONOTONICITY). *For any $\mathcal{P} = (V, R)$, if $\mathcal{P} \vDash A \parallel_C B, D$, then $\mathcal{P} \vDash A \parallel_C B$.*

PROOF. Assume that $r_1 \equiv_C r_2$ for some runs $r_1, r_2 \in R$. Hence, by the assumption of the lemma, there is $r \in R$ such that $r_1 \equiv_{A,C} r \equiv_{B,D,C} r_2$. Therefore, $r_1 \equiv_{A,C} r \equiv_{B,C} r_2$. □

Next, we establish a technical lemma that is used in the proof of soundness of the Diagram axiom.

LEMMA 3. *For any diagram $\Delta \in Diag(Q)$ and any protocol $\mathcal{P} = (V, R)$ such that $\mathcal{P} \vDash \delta$ for each $\delta \in [\Delta]$, if $r^+, r^- \in R$ and $r^+ \equiv_Q r^-$, then there is a function ρ that maps vertices of the diagram Δ into runs in R that satisfies the following conditions:*

1. *$\rho(v^+_\Delta) = r^+$ and $\rho(v^-_\Delta) = r^-$,*
2. *if $v_1 \sim_S v_2$, then $\rho(v_1) \equiv_S \rho(v_2)$.*

PROOF. Induction on the number of vertices in diagram Δ. If Δ is a basic diagram, then define ρ to be such that $\rho(v^+_\Delta) = r^+$ and $\rho(v^-_\Delta) = r^-$. Condition 2 is satisfied because of the assumption $r^+ \equiv_Q r^-$.

Suppose now that diagram Δ' is obtained from diagram Δ by adding a new vertex v_{new}, connected to vertices u and v by edges labeled with sets $A \cup C$ and $B \cup C$ respectively, such that $u \sim_C v$. By the induction hypothesis, there is a function ρ on the vertices of the diagram Δ that satisfies conditions 1. and 2. of this lemma. In particular, $\rho(u) \equiv_C \rho(v)$. We will show how function ρ could be extended to the vertex v_{new} preserving conditions 1. and 2.

Note that $A \parallel_C B \in [\Delta']$, by Definition 4. Hence, by the assumption of this lemma, $\mathcal{P} \vDash A \parallel_C B$. Therefore, there is a run $r \in R$ such that $\rho(u) \equiv_{A,C} r \equiv_{B,C} \rho(v)$. Define $\rho(v_{new}) = r$.

To finish the proof of the lemma, we need to show that if $v_{new} \sim_S w$, where $w \neq v_{new}$ is a vertex in the diagram Δ', then $\rho(v_{new}) \equiv_S \rho(w)$. Note that vertex w is also a vertex in the diagram Δ, because $w \neq v_{new}$. Thus, Set S could be partitioned into sets S_1 and S_2 such that: $S_1 \subset A \cup C$, $S_2 \subset B \cup C$, $u \sim_{S_1} w$ and $v \sim_{S_2} w$. Hence, by the induction hypothesis, $\rho(u) \equiv_{S_1} \rho(w)$ and $\rho(v) \equiv_{S_2} \rho(w)$. Thus,

$$\rho(v_{new}) = r \equiv_{S_1} \rho(u) \equiv_{S_1} \rho(w),$$

$$\rho(v_{new}) = r \equiv_{S_2} \rho(v) \equiv_{S_2} \rho(w).$$

Therefore, $\rho(v_{new}) \equiv_S \rho(w)$. □

LEMMA 4 (DIAGRAM). *For any protocol $\mathcal{P} = (V, R)$, if*

1. *diagram Δ renders tuple*

$$(A_1, A_2, A_3; B_1, B_2, B_3; C_1, C_2, C_3; D),$$

2. $\mathcal{P} \vDash \delta$ *for each* $\delta \in [\Delta]$,

3. $\mathcal{P} \vDash A_1, B_1, C_1 \parallel_{A_3,B_3,C_3,D} A_2, B_2, C_2$

then $\mathcal{P} \vDash A_1, A_2, A_3 \parallel_{C_1,C_2,C_3} B_1, B_2, B_3$.

PROOF. Let $r^+, r^- \in R$ be such that $r^+ \equiv_{C_1,C_2,C_3} r^-$. We will prove the existence of a run $r \in R$ such that

$$r^+ \equiv_{A_1,A_2,A_3,C_1,C_2,C_3} r,$$

$$r^- \equiv_{B_1,B_2,B_3,C_1,C_2,C_3} r.$$

By Lemma 3, there is a function ρ that maps vertices of the diagram into runs of the protocol $\mathcal{P}$ that satisfies conditions 1. and 2. of Lemma 3.

By Definition 5, there are vertices w_1 and w_2 in the diagram Δ that satisfy conditions 1.-5. of that definition. In particular, $w_1 \sim_{A_3,C_3} v^+_\Delta$ and $w_2 \sim_{A_3,C_3} v^+_\Delta$. Thus, $w_1 \sim_{A_3,C_3} w_2$. Similarly, $w_1 \sim_{B_3,C_3} w_2$. By condition 5. of Definition 5, $w_1 \sim_D w_2$. Therefore, $w_1 \sim_{A_3,B_3,C_3,D} w_2$. Thus, by the assumption 3. of this lemma, there is a run $r \in R$ such that

$$\rho(w_1) \equiv_{A_1,B_1,C_1,A_3,B_3,C_3,D} r \quad (1)$$

$$\rho(w_2) \equiv_{A_2,B_2,C_2,A_3,B_3,C_3,D} r \quad (2)$$

By Definition 5,

$$w_1 \sim_{A_1,A_3,C_1,C_3} v^+_\Delta$$

$$w_2 \sim_{A_2,A_3,C_2,C_3} v^+_\Delta.$$

Hence, by condition 2. of Lemma 3,

$$\rho(w_1) \equiv_{A_1,A_3,C_1,C_3} r^+$$

$$\rho(w_2) \equiv_{A_2,A_3,C_2,C_3} r^+.$$

Finally, taking into account equations (1) and (2),

$$r^+ \equiv_{A_1,A_2,A_3,C_1,C_2,C_3} r.$$

Similarly, $r^- \equiv_{B_1,B_2,B_3,C_1,C_2,C_3} r$. □

8. COMPLETENESS

In the rest of the paper we establish completeness of our logical system.

THEOREM 2 (COMPLETENESS). *For any* $\varphi \in \Phi$, *if* $\nvdash \varphi$, *then there is a protocol* $\mathcal{P}$ *such that* $\mathcal{P} \nvDash \varphi$.

Suppose that $\nvdash \varphi$. Let X be any maximal consistent subset of Φ containing formula $\neg\varphi$.

8.1 Chains of Diagrams

The chains of diagrams is a technical construction that we use to prove of the completeness theorem.

DEFINITION 6. *A* Q*-chain is an infinite sequence of diagrams* $\Delta_0, \Delta_1, \ldots, \Delta_n, \ldots$ *from* $Diag(Q)$ *such that* Δ_0 *is the basic diagram and diagram* Δ_{i+1} *is an extension of the diagram* Δ_i *for each* $i \geq 0$.

LEMMA 5. *For any* Q*-chain* $\Delta_0, \Delta_1, \ldots, \Delta_n, \ldots$, *any label* p, *any* n, *and any* $N \geq n$, *if* x *and* y *are vertices on a diagram* Δ_n *and* $x \sim_p y$ *on diagram* Δ_N, *then* $x \sim_p y$ *on diagram* Δ_n.

PROOF. Suppose that there is $n \leq k < N$ such that $x \sim_p y$ on diagram Δ_{k+1}, but not on diagram Δ_k. By Definition 4, diagram Δ_{k+1} is obtained from diagram Δ_k by adding vertex w connected to vertices u and v by edges labeled with sets $A \cup C$ and $B \cup C$, such that $u \sim_C v$ on diagram Δ_k.

Since $x \sim_p y$ on diagram Δ_{k+1}, but not on diagram Δ_k, there must be a path labeled by p between vertices x and y on diagram Δ_{k+1} that goes through both added edges: (u, w) and (w, v). Hence, $p \in (A \cup C) \cap (B \cup C)$. By Definition 1, sets A, B, and C are disjoint. Thus, $p \in C$. Recall, however, that $u \sim_C v$ on diagram Δ_k. Therefore, $x \sim_p y$ on diagram Δ_k, which is a contradiction with the choice of k. □

DEFINITION 7. *A* Q*-chain* $\Delta_0, \Delta_1, \Delta_2, \ldots$ *is called sound if* $[\Delta_n] \subseteq X$ *for each* $n \geq 0$.

DEFINITION 8. *A* Q*-chain* $\Delta_0, \Delta_1, \Delta_2, \ldots$ *is complete if for any* $A \parallel_C B \in X$, *for any* $n \geq 0$ *and any two vertices* u, v *of the diagram* Δ_n *such that* $u \sim_C v$, *there is* $N \geq n$ *and a vertex* w *in the diagram* Δ_N *such that relations* $u \sim_{A,C} w$ *and* $w \sim_{B,C} v$ *hold in diagram* Δ_N.

LEMMA 6. *For any set of secrets* Q, *there is a* Q*-chain which is complete and sound with respect to the set* X.

PROOF. The statement of the lemma follows from the Definition 4 and the fact that set X is countable. □

8.2 Chain Protocol

We now show how a chain of diagrams can be converted into a protocol with certain desirable properties. Later, several such protocols will be combined into one in order to finish the proof of the completeness theorem.

LEMMA 7. *For each finite set of secrets* Q *there is a protocol* $\mathcal{P}$ *such that*

1. *protocol* $\mathcal{P}$ *has at least one run,*
2. $\mathcal{P} \vDash A \parallel_C B$ *for each sets of secret variables* A, B, *and* C *such that* $A \parallel_C B \in X$,
3. $\mathcal{P} \nvDash P \parallel_Q R$ *for each sets of secret variables* P *and* R *such that* $P \parallel_Q R \notin X$.

PROOF. By Lemma 6, there is Q-chain of diagrams

$$\Delta_0, \Delta_1, \Delta_2, \ldots,$$

which is complete and sound with respect to the set X. Let $V_0 \subset V_1 \subset V_2 \subset \ldots$ be the sets of vertices of these diagrams.

For any label a and any two vertices $u, v \in \bigcup_i V_i$, we say that vertices u and v are a-equivalent if there is k such that $u \sim_a v$ in diagram Δ_k. Let $Val(a)$ be the set of equivalence classes on $\bigcup_i V_i$ with respect to this equivalence relation.

For any $v \in \bigcup_i V_i$ and any label a, define function $r_v(a)$ to be equal to the a-equivalence class of v:

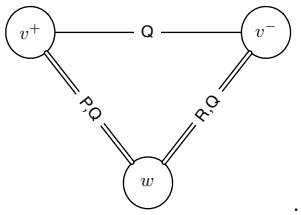
$$r_v(a) = [v]_a.$$

Let $\mathcal{R} = \{r_v \mid v \in \bigcup_i V_i\}$. This concludes the definition of the protocol $\mathcal{P} = (Val, \mathcal{R})$. We will now show that this protocol satisfies conditions 1., 2., and 3. of the lemma.

To prove the first condition, notice that set $\bigcup_i V_i$ is not empty, because it contains vertices v^+ and v^- from the basic diagram Δ_0. Thus, set $\{r_v \mid v \in \bigcup_i V_i\}$ is also not empty.

To prove the second condition, consider any $r_u, r_v \in \mathcal{R}$ such that $r_u \equiv_C r_v$. We will show that there is $r_w \in \mathcal{R}$ such that $r_u \equiv_{A,C} r_w \equiv_{B,C} r_v$. Indeed, $r_u \equiv_C r_v$ implies that $[u]_c = [v]_c$ for each $c \in C$. Thus, for each $c \in C$, vertices u and v are c-equivalent. Hence, there must exists $n \geq 0$ such that $u \sim_C v$ in Δ_n. By Definition 8, there is $N \geq n$ and a vertex w in the diagram Δ_N such that relations $u \sim_{A,C} w$ and $w \sim_{B,C} v$ hold in diagram Δ_N. Thus, $[u]_x = [w]_x$ for each $x \in A \cup C$ and $[w]_y = [v]_y$ for each $y \in B \cup C$. Therefore, $r_u \equiv_{A,C} r_w \equiv_{B,C} r_v$.

To prove the third condition, assume the opposite: $\mathcal{P} \vDash P \parallel_Q R$. Consider vertices v^+ and v^- of the based diagram Δ_0. By Definition 4, $v^+ \sim_Q v^-$ on diagram Δ_0. Thus, $[v^+]_q = [v^-]_q$ for each $q \in Q$. Hence, $r_{v^+} \equiv_Q r_{v^-}$. Then, by the assumption $\mathcal{P} \vDash P \parallel_Q R$, there must be a run r_w such that $r_{v^+} \equiv_{P,Q} r_w \equiv_{R,Q} r_{v^-}$. Hence, $[v^+]_t = [w]_t$ for each $t \in P \cup Q$ and $[w]_t = [v^-]_t$ for each $t \in R \cup Q$. Thus,



.

Let n be the smallest integer such that Δ_n contains vertex w. By Definition 4, there are vertices u and v in diagram Δ_n such that

1. vertex w is only connected in diagram Δ_n to u and v,
2. edge (u, w) is labeled with a set A,
3. edge (w, v) is labeled with set B,
4. $u \sim_{A \cap B} v$ in diagram Δ_{n-1}, and
5. $A \setminus B \parallel_{A \cap B} B \setminus A \in [\Delta_n]$.

Since chain $\Delta_0, \Delta_1, \ldots$ is sound with respect to set X, the last condition above implies that

$$A \setminus B \parallel_{A \cap B} B \setminus A \in X.$$

By Monotonicity axiom,

$$X \vdash A \setminus B \parallel_{A \cap B} P \cap (B \setminus A), R \cap (B \setminus A), Q \cap (B \setminus A).$$

By Symmetry axiom,

$$X \vdash P \cap (B \setminus A), R \cap (B \setminus A), Q \cap (B \setminus A) \parallel_{A \cap B} A \setminus B.$$

By Monotonicity axiom,

$$\begin{aligned} X \vdash \quad & P \cap (B \setminus A), R \cap (B \setminus A), Q \cap (B \setminus A) \parallel_{A \cap B} \\ & P \cap (A \setminus B), R \cap (A \setminus B), Q \cap (A \setminus B). \end{aligned}$$

Again by Symmetry axiom,

$$\begin{aligned} X \vdash \quad & P \cap (A \setminus B), R \cap (A \setminus B), Q \cap (A \setminus B) \parallel_{A \cap B} \\ & P \cap (B \setminus A), R \cap (B \setminus A), Q \cap (B \setminus A). \end{aligned}$$

In other words,

$$\begin{aligned} X \vdash \quad & P \cap (A \setminus B), R \cap (A \setminus B), Q \cap (A \setminus B) \\ & \parallel_{P \cap (A \cap B), R \cap (A \cap B), Q \cap (A \cap B), (A \cap B) \setminus (P \cup Q \cup R)} \\ & P \cap (B \setminus A), R \cap (B \setminus A), Q \cap (B \setminus A). \end{aligned}$$

We now apply the Diagram axiom (see Figure 4) with

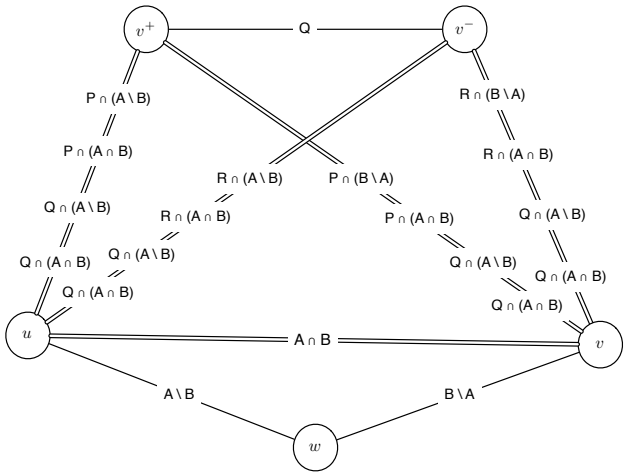


Figure 4: Diagram Δ_n.

$$A_1 = P \cap (A \setminus B) \qquad A_2 = P \cap (B \setminus A)$$

$$B_1 = R \cap (A \setminus B) \qquad B_2 = R \cap (B \setminus A)$$

$$C_1 = Q \cap (A \setminus B) \qquad C_2 = Q \cap (B \setminus A)$$

$$A_3 = P \cap (A \cap B)$$

$$B_3 = R \cap (A \cap B)$$

$$C_3 = Q \cap (A \cap B)$$

$$D = (A \cap B) \setminus (P \cup Q \cup R)$$

to conclude that

$$\begin{aligned} X \vdash \quad & P \cap (A \setminus B), P \cap (B \setminus A), P \cap (A \cap B) \\ & \parallel_{Q \cap (A \setminus B), Q \cap (B \setminus A), Q \cap (A \cap B)} \\ & R \cap (A \setminus B), R \cap (B \setminus A), R \cap (A \cap B). \end{aligned} \tag{3}$$

Recall that $w \sim_{P \cup Q} v^+$ and $w \sim_{R \cup Q} v^-$. At the same time, vertex w is only connected in diagram Δ_n to u and v, edge (u, w) is labeled with a set A, and edge (w, v) is labeled with set B. Hence, $P \cup Q \subseteq A \cup B$. Thus, statement (3) implies that $X \vdash P \parallel_Q R$, which is a contradiction with the assumption. □

8.3 Protocol Composition

In this section we introduce a way to combine several different protocols over (S, G) into a single protocol.

Definition 9. *For any protocols $\mathcal{P}_1 = (V_1, R_1), \ldots, \mathcal{P}_n = (V_n, R_n)$, let $\mathcal{P}_1 \times \cdots \times \mathcal{P}_n$ be a protocol (V, R) such that*

1. *$V(a) = V_1(a) \times \cdots \times V_n(a)$, for each $a \in S$,*
2. *R is a set of all functions $r(x) = \langle r_1(x), \ldots, r_n(x) \rangle$ for all $r_1 \in R_1, \ldots, r_n \in R_n$.*

Lemma 8. *Let $\mathcal{P}_1 = (V_1, R_1), \ldots, \mathcal{P}_n = (V_n, R_n)$ be protocols such that set R_k is not empty for each $k \leq n$. Then $\mathcal{P}_1 \times \cdots \times \mathcal{P}_n \vDash A \parallel_C B$ if and only if $\mathcal{P}_k \vDash A \parallel_C B$ for each $k \leq n$.*

PROOF. ($\Rightarrow$) : Suppose that $r_k^1, r_k^2 \in R_k$ are such that $r_k^1 \equiv_C r_k^2$. We will show that there is a run $r_k \in R_k$ such that $r_k^1 \equiv_{A,C} r_k \equiv_{B,C} r_k^2$.

Let (V, R) be protocol $\mathcal{P}_1 \times \cdots \times \mathcal{P}_n$. Consider any runs

$$r_1 \in R_1, \ldots, r_{k-1} \in R_{k-1}, r_{k+1} \in R_{k+1}, \ldots, r_n \in R_n.$$

Such runs exists due to the assumption of the lemma. Let $r^1, r^2 \in R$ be such that for each secret variable x,

$$r^1(x) = \langle r_1(x), \ldots, r_{k-1}(x), r_k^1(x), r_{k+1}(x), \ldots, r_n(x)\rangle,$$

$$r^2(x) = \langle r_1(x), \ldots, r_{k-1}(x), r_k^2(x), r_{k+1}(x), \ldots, r_n(x)\rangle.$$

Note that $r_k^1 \equiv_C r_k^2$ implies that $r^1(c) \equiv_C r^2(c)$. Hence, by the assumption of the lemma, there is a run $r \in R$ such that $r^1 \equiv_{A,C} r \equiv_{B,C} r^2$. Let $r_k(x)$ be defined to be the k-th component of $r(x)$ for each secret variable x. Thus, by Definition 9, $r_k \in R_k$. Finally, $r^1 \equiv_{A,C} r \equiv_{B,C} r^2$ implies that $r_k^1 \equiv_{A,C} r_k \equiv_{B,C} r_k^2$.

($\Leftarrow$) : Suppose that $r^1, r^2 \in R$ are such that $r^1 \equiv_C r^2$. We will show that there is $r \in R$ such that $r^1 \equiv_{A,C} r \equiv_{B,C} r^2$. Assume that $r^1(x) = \langle r_1^1(x), \ldots, r_n^1(x)\rangle$, and $r^2(x) = \langle r_1^2(x), \ldots, r_n^2(x)\rangle$. Assumption $r^1 \equiv_C r^2$ implies that $r_k^1 \equiv_C r_k^2$ for each $k \leq n$. Thus, by the assumption of the lemma, there are runs $r_1 \in R_1, \ldots, r_n \in R_n$ such that $r_k^1 \equiv_{A,C} r_k \equiv_{B,C} r_k^2$ for each $k \leq n$. Define $r(x) = \langle r_1(x), \ldots, r_n(x)\rangle$, Therefore, $r^1 \equiv_{A,C} r \equiv_{B,C} r^2$. $\square$

8.4 Completeness: final steps

We are now ready to finish the proof of the completeness theorem. Let S be the finite set of all variables that appear in the formula φ. Let $Q_1, \ldots, Q_n$ be all subsets of S. By Lemma 7, there are protocols $\mathcal{P}_1, \ldots, \mathcal{P}_n$ such that

1. $\mathcal{P}_k \vDash A \perp_C B$ for each sets of secret variables A, B, and C such that $A \perp_C B \in X$,

2. $\mathcal{P}_k \nvDash P \perp_{Q_k} R$ for each sets of secret variables P and R such that $P \perp_{Q_k} R \notin X$.

Let $\mathcal{P} = \mathcal{P}_1 \times \cdots \times \mathcal{P}_n$.

LEMMA 9. *For each $\psi \in \Phi$ that only uses secret variables from set S, $\mathcal{P} \vDash \psi$ if and only if $\psi \in X$.*

PROOF. Induction on the structural complexity of formula ψ. Case ψ being $\bot$ follows from the assumption of consistency of X and Definition 3. The induction case $\psi \equiv \psi_1 \to \psi_2$ follows from the maximality and consistence of set X in the standard way. We are only left to consider the case when ψ is an atomic formula $P \perp_Q R$ for some $P, Q, R \subseteq S$. Assume that $Q = Q_{k_0}$.

($\Rightarrow$) : Suppose that $X \nvdash P \perp_Q R$. Thus, $\mathcal{P}_{k_0} \nvDash P \perp_Q R$ due to the choice of the protocol $\mathcal{P}_{k_0}$. Note that each of the protocols $\mathcal{P}_1, \ldots, \mathcal{P}_n$ has at least one run due to Lemma 7. Thus, by Lemma 8, $\mathcal{P} \nvDash P \perp_Q R$.

($\Leftarrow$) : If $X \vdash P \perp_Q R$, then, $\mathcal{P}_k \nvDash P \perp_Q R$ for each $k \leq n$ due to the choice of the protocols $\mathcal{P}_1, \ldots, \mathcal{P}_n$. Note again that each of the protocols $\mathcal{P}_1, \ldots, \mathcal{P}_n$ has at least one run due to Lemma 7. Thus, by Lemma 8, $\mathcal{P} \vDash P \perp_Q R$. $\square$

Recall now that $\neg\varphi \in X$. Hence, $\varphi \notin X$ due to consistency of X. Therefore, $\mathcal{P} \nvDash \varphi$ by Lemma 9. This concludes the proof of Theorem 2. $\square$

9. CONCLUSION

In this paper we gave a recursively enumerable axiomatization of propositional properties of relation $A \perp_C B$, assuming that sets A, B, and C are pair-wise disjoint. Although Definition 3 is meaningful if the sets are not disjoint, our completeness proof will not work (see Lemma 5). At the same time, it is interesting to point out that due to Definition 3, statement $B \perp_A B$ means that *any two runs that agree on A also agree on B*. Thus, $B \perp_A B$ represents *functional dependency* relation between values of A and B. Functional dependency alone was axiomatized by Armstrong [1]. It appears that allowing sets A, B, and C to be non-disjoint leads to a significantly more powerful language. Complete axiomatization of all properties expressible in such language remains an open question.

10. REFERENCES

[1] W. W. Armstrong. Dependency structures of data base relationships. In *Information processing 74 (Proc. IFIP Congress, Stockholm, 1974)*, pages 580–583. North-Holland, Amsterdam, 1974.

[2] Michael S. Donders, Sara Miner More, and Pavel Naumov. Information flow on directed acyclic graphs. In Lev D. Beklemishev and Ruy de Queiroz, editors, *WoLLIC*, volume 6642 of *Lecture Notes in Computer Science*, pages 95–109. Springer, 2011.

[3] Dan Geiger, Azaria Paz, and Judea Pearl. Axioms and algorithms for inferences involving probabilistic independence. *Inform. and Comput.*, 91(1):128–141, 1991.

[4] Erich Grädel and Jouko Väänänen. Dependence and Independence. To appear in Studia Logica.

[5] Erich Grädel and Jouko Väänänen. Dependence, Independence, and Incomplete Information. In *Proceedings of 15th International Conference on Database Theory, ICDT 2012*, 2012.

[6] Joseph Y. Halpern and Kevin R. O'Neill. Secrecy in multiagent systems. *ACM Trans. Inf. Syst. Secur.*, 12(1):1–47, 2008.

[7] Christian Herrmann. On the undecidability of implications between embedded multivalued database dependencies. *Inf. Comput.*, 122(2):221–235, 1995.

[8] Christian Herrmann. Corrigendum to "on the undecidability of implications between embedded multivalued database dependencies" [inform. and comput. 122(1995) 221-235]. *Inf. Comput.*, 204(12):1847–1851, 2006.

[9] Jérôme Lang, Paolo Liberatore, and Pierre Marquis. Conditional independence in propositional logic. *Artif. Intell.*, 141(1/2):79–121, 2002.

[10] Sara Miner More and Pavel Naumov. An independence relation for sets of secrets. In H. Ono, M. Kanazawa, and R. de Queiroz, editors, *Proceedings of 16th Workshop on Logic, Language, Information and Computation (Tokyo, 2009), LNAI 5514*, pages 296–304. Springer, 2009.

[11] Sara Miner More and Pavel Naumov. Hypergraphs of multiparty secrets. In *11th International Workshop on Computational Logic in Multi-Agent Systems CLIMA XI (Lisbon, Portugal), LNAI 6245*, pages 15–32. Springer, 2010.

[12] Sara Miner More and Pavel Naumov. Logic of secrets in collaboration networks. *Ann. Pure Appl. Logic*, 162(12):959–969, 2011.

[13] Sara Miner More, Pavel Naumov, Brittany Nicholls, and Andrew Yang. A ternary knowledge relation on secrets. In Krzysztof R. Apt, editor, *Proceedings of the 13th Conference on Theoretical Aspects of Rationality and Knowledge (TARK-2011), Groningen, The Netherlands, July 12-14, 2011*, pages 46–54. ACM, 2011.

[14] Sara Miner More, Pavel Naumov, and Benjamin Sapp. Concurrency semantics for the Geiger-Paz-Pearl axioms of independence. In Marc Bezem, editor, *20th Annual Conference on Computer Science Logic, , CSL 2011, September 12-15, 2011, Bergen, Norway, Proceedings*, volume 12 of *LIPIcs*, pages 443–457. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2011.

[15] Pavel Naumov and Brittany Nicholls. Game semantics for the Geiger-Paz-Pearl axioms of independence. In *The Third International Workshop on Logic, Rationality and Interaction (LORI-III), LNAI 6953*, pages 220–232. Springer, 2011.

[16] D. Stott Parker, Jr. and Kamran Parsaye-Ghomi. Inferences involving embedded multivalued dependencies and transitive dependencies. In *Proceedings of the 1980 ACM SIGMOD international conference on Management of data*, SIGMOD '80, pages 52–57, New York, NY, USA, 1980. ACM.

[17] Milan Studený. Conditional independence relations have no finite complete characterization. In *Information Theory, Statistical Decision Functions and Random Processes. Transactions of the 11th Prague Conference vol. B*, pages 377–396. Kluwer, 1990.

[18] David Sutherland. A model of information. In *Proceedings of Ninth National Computer Security Conference*, pages 175–183, 1986.

When is an example a counterexample?

[Extended Abstract]

Eric Pacuit
University of Maryland
TiLPS, Tilburg University
e.j.pacuit@uvt.nl

Arthur Paul Pedersen
Department of Philosophy
Carnegie Mellon University
ppederse@andrew.cmu.edu

Jan-Willem Romeijn
Faculty of Philosophy
Groningen University
j.w.romeijn@rug.nl

ABSTRACT

In this extended abstract, we carefully examine a purported counterexample to a postulate of iterated belief revision. We suggest that the example is better seen as a failure to apply the theory of belief revision in sufficient detail. The main contribution is conceptual aiming at the literature on the philosophical foundations of the AGM theory of belief revision [1]. Our discussion is centered around the observation that it is often unclear whether a specific example is a "genuine" counterexample to an abstract theory or a misapplication of that theory to a concrete case.

1. INTRODUCTION

Starting with the seminal paper [1], the so-called AGM theory of belief revision has been extensively studied by logicians, computer scientists, and philosophers. The general setup is well-known, and we review it here to fix ideas and notation.

Let K be a *belief set*, a set of propositional formulae closed under classical consequence representing an agent's initial collection of beliefs. Given a belief φ that the agent has acquired, the set $K * \varphi$ represents the agent's collection of beliefs upon acquiring φ. A central project in the theory of belief revision is to study constraints on functions $*$ mapping a belief set K and a propositional formula φ to a new belief set $K * \varphi$. For reference, the key AGM postulates are listed in the Appendix (Section A). This simple framework has been analyzed, extended, and itself revised in various ways (see [2] for a survey of this literature), and much has been written about the status of its philosophical foundations (cf. [10, 21, 20]).

The basic AGM theory does not explicitly address the question of how to respond to a sequence of belief changes. The only salient constraint on iterated revision implied by the eight AGM postulates is the requirement that $(K * \varphi) * \psi \subseteq K * (\varphi \wedge \psi)$ provided $\neg\psi \notin K * \varphi$. [1] However, if $\neg\psi \in K * \varphi$, there is no constraint on $(K * \varphi) * \psi$. Various authors have attempted to rectify this situation, proposing additional rationality constraints on belief revision given a sequence of input beliefs [8, 9, 5, 15, 16, 18, 21, 6]. Two postulates which have been extensively discussed in the literature are the following constraints:

I1 If $\psi \in \mathrm{Cn}(\{\varphi\})$ then $(K * \psi) * \varphi = K * \varphi$

I2 If $\neg\psi \in \mathrm{Cn}(\{\varphi\})$ then $(K * \varphi) * \psi = K * \psi$

Each of these postulates have some intuitive appeal. Postulate **I1** demands if $\varphi \rightarrow \psi$ is a theorem (with respect to the background theory), then first learning ψ followed by the more specific information φ is equivalent to directly learning the more specific information φ. Postulate **I2** demands that first learning φ followed by learning a piece of information ψ incompatible with φ is the same as simply learning ψ outright. So, for example, first learning φ and then $\neg\varphi$ should result in the same belief state as directly learning $\neg\varphi$. [2]

Many recent developments in this area have been offered on the basis of analyses of *concrete examples*. These range from toy examples—such as the infamous muddy children puzzle, the Monty Hall problem, and the Judy Benjamin problem—to everyday examples of social interaction. Different frameworks are then judged, in part, on how well they conform to the analyst's intuitions about the perceived relevant set of examples. This raises an important issue: Implicit assumptions about what the agents know and believe about the situation being modeled often guide the analyst's intuitions. In many cases, it is crucial to make these underlying assumptions explicit.

The following simple example illustrates the type of implicit assumption that we have in mind. There are two opaque boxes, labeled 1 and 2, each containing a coin. The believer is interested in the status of the coins in each box. Suppose that Ann is an expert on the status (heads up or tails up) of the coin in box 1 and that Bob is an expert on the status (heads up or tails up) of the coin in box 2. Currently the believer under consideration does not have an opinion about whether the coins are lying heads up or tails up in the boxes; more specifically, the believer thinks that all four possibilities are equally plausible. Suppose that both Ann and Bob report that their respective coins are lying tails

[1] By AGM 7 ($K * (\varphi \wedge \psi) \subseteq \mathrm{Cn}(K * \varphi \cup \{\psi\})$) and AGM 8 ($\neg\psi \notin K * \varphi$ then $\mathrm{Cn}(K * \varphi \cup \{\psi\}) \subseteq K * (\varphi \wedge \psi)$), we have $K * (\varphi \wedge \psi) = \mathrm{Cn}(K * \varphi \cup \{\psi\})$ provided that $\neg\psi \notin K * \varphi$, whence by an application of AGM 3 ($(K * \varphi) * \psi \subseteq \mathrm{Cn}((K * \varphi) \cup \{\psi\})$), it follows that $(K * \varphi) * \psi \subseteq \mathrm{Cn}(K * \varphi \cup \{\psi\})$ if $\neg\psi \notin K * \varphi$.

[2] Of course, one might object to this on the basis of the observation that if the believer is in a situation in which she is receiving inconsistent evidence, then she should recognize this and accordingly adopt beliefs about the source(s) of information. This issue of *higher order evidence* is interesting (cf. [7]), but we set it aside in this paper. We are interested in situations in which the believer never loses her trust in the process generating evidence. AGM theory may be the only theory applicable in such situations.

TARK 2013, Chennai, India. 2013, Chennai, India

up. Since both experts are trusted, this is what the believer believes. Now further suppose that there is a third expert, Charles, who is considered more reliable than both Ann and Bob. What should the believer think about the coin in box 2 after receiving a report from Charles that the coin in box 1 is lying heads up?

Of course, an answer to this question depends in part on the believer's initial opinions about the relationship between the coins in the two boxes. If the believer initially thinks that the status of the coins are independent and that the reports from Ann and Bob are independent, then she should believe that the coin in box 2 is lying tails up. However, if she has reason to think that the coins, or reports about the coins, are somehow correlated, upon learning that the coin in box 1 is lying heads up, she may be justified in changing her belief about the status of the coin in box 2.

Robert Stalnaker [21] has discussed the potential role that such *meta-information*, as illustrated in the above example, plays in the evaluation of proposed counterexamples to the AGM postulates. The general message is that once salient meta-information has been made explicit, many of the purported counterexamples to the AGM theory of belief revision do not demonstrate a failure of the theory itself, but rather a failure to *apply* the theory correctly and include all the relevant components in the model. [3] After an illuminating discussion of a number of well-known counterexamples to the AGM postulates, Stalnaker proposes two "genuine" counterexamples to postulates **I1** and **I2** for the theory of iterated belief revision. The conclusion Stalnaker draws in his discussion is that "... little of substance can be said about constraints on iterated belief revision at a level of abstraction that lacks the resources for explicit representation of meta-information" (pg. 189).

In this extended abstract, we carefully examine one of Stalnaker's purported counterexamples (Section 4), provide a model for it that complies with the AGM postulates, suggesting that it is again better seen as a failure to apply the theory of belief revision in sufficient detail. We end with a critical discussion of the opposition between genuine counterexamples and misapplications of the theory (Section 6).

2. STALNAKER'S EXAMPLE

An indicated in the introduction, Stalnaker [21] proposes counterexamples to both postulates **I1** and **I2**. In this extended abstract, we only have space to discuss one of the examples (the full paper has an extensive discussion of both examples). We discuss an example which is "clearer and a more decisive problem" for **I2**.

Example. Suppose that two fair coins are flipped and placed in two boxes. Two independent and reliable observers deliver reports about the status (heads up or tails up) of the coins in the opaque boxes. On the one hand, Alice reports that the coin in box 1 is lying heads up, and on the other hand, Bert reports that the coin in box 2 is lying heads up.

Two new independent witnesses, whose reliability trumps that of Alice's and Bert's, provide additional reports on the status of the coins. Carla reports that the coin in box 1 is lying tails up, and Dora reports that the coin in box 2 is lying tails up. Finally, Elmer, a third witness considered the most reliable overall, reports that the coin in box 1 is lying heads up.

Let H_i be the proposition expressing the statement that the coin in box i is lying heads up ($i = 1, 2$). Similarly, for $i = 1, 2$, let T_i be the proposition expressing the statement that the coin in box i is lying tails up. After the first belief revision, the belief set is $K' = K * (H_1 \wedge H_2)$, where K is the agent's original set of beliefs. After receiving the reports, the belief set is $K' * (T_1 \wedge T_2) * H_1$. As Stalnaker suggests, since Elmer's report is irrelevant to the status of the coin in box 2, it seems natural to assume that $H_1 \wedge T_2 \in K' * (T_1 \wedge T_2) * H_1$.

Now to the hitch. Since $(T_1 \wedge T_2) \rightarrow \neg H_1$ is a theorem (given the background theory), by **I2** it follows that $K' * (T_1 \wedge T_2) * H_1 = K' * H_1$. Yet since $H_1 \wedge H_2 \in K'$ and H_1 is consistent with H_2, we must have $H_1 \wedge H_2 \in K' * H_1$, which yields a conflict with the assumption that $H_1 \wedge T_2 \in K' * (T_1 \wedge T_2) * H_1$.

Stalnaker diagnoses the situation as follows:

> ...[Postulate $I2$] directs us to take back the totality of any information that is overturned. Specifically, if we first receive information α, and then receive information that conflicts with α, we should return to the belief state we were previously in, before learning α. But this directive is too strong. Even if the new information conflicts with the information just received, it need not necessarily cast doubt on *all* of that information.
>
> (pg. 207–208)

It seems that, for lack of independent guidelines of how we must identify the component of the evidence that needs overturning to accommodate the new information, the epistemic advice provided by AGM conflicts with the intuitively correct answer.

But what are we to do with this apparent conflict? In what follows we attempt to model Stalnaker's puzzling example. This is a conceptual paper aiming to contribute to the literature on the philosophical foundations of the theory of belief revision (cf. [10, 21, 20]). Accordingly, it is not our main goal to extend this theory, resolve the problems, and be done with it. Our focus lies rather on the fact that it is unclear how to appropriately respond to a purported counterexample to a postulate of iterated belief revision. To illustrate, the foregoing example may be regarded as demonstrating either:

1. There is no suitable way to formalize the scenario in such a way that the AGM postulates (possibly including postulates of iterated belief revision) can be saved;

[3] This is not to say that there are no genuine conceptual problems with the AGM theory of belief revision. The point raised here is that it is often unclear what exactly a specific counterexample to an AGM postulate demonstrates about the abstract theory of belief revision. This is nicely explained by Stalnaker in his analysis of Hans Rott's well-known counterexample to various AGM postulates (see [20]):

> ... Rott seems to take the point about meta-information to explain why the example conflicts with the theoretical principles, whereas I want to conclude that it shows why the example does not conflict with the theoretical principles, since I take the relevance of the meta-information to show that the conditions for applying the principles in question are not met by the example.
>
> (pg. 204)

2. The AGM framework can be made to agree with the scenario but does not furnish a systematic way to formalize the relevant meta-information; or

3. There is a suitable and systematic way to make the meta-information explicit, but this is something that the AGM framework cannot properly accommodate.

The first response is very drastic and, indeed, the models presented in the next section may be taken to show that the meta-information driving the belief change can be made suitably explicit. The second response to the example is already well-appreciated in the literature on belief revision (cf. the discussion of sources of evidence in [6] and "ontology" in [10]). Of interest to us for this paper is the third response, which is concerned with the absence of guidelines for *applying* the theory of belief revision.

In other words, we suggest that there is a problem with the AGM theory, and that this problem arises because a clear distinction between counterexample and misapplication has yet to be drawn. It is not clear from the theory of belief revision how its applications are supposed to be organized, and hence it is not clear whether the examples reveal shortcomings in the theory or rather in its application. Stalnaker suggests that his purported counterexamples turn on *independence*:

> There are different kinds of independence—conceptual, causal and epistemic—that interact, and one might be able to say more about constraints on rational belief revision if one had a model theory in which causal-counterfactual and epistemic information could both be represented. There are familiar problems, both technical and philosophical, that arise when one tries to make meta-information explicit, since it is self-locating (and auto-epistemic) information, and information about changing states of the world. (pg. 208)

Part of our response is to show how models from AGM belief revision can accommodate such considerations. In addition, we offer a different perspective on the third response to the counterexample.

This shift in perspective has a positive part and a negative part. On the one hand, we argue that probabilistic models can facilitate an explicit incorporation of meta-information underlying rational belief changes for many examples. Furthermore, these models can offer principled ways to distinguish genuine counterexamples from misapplications of the AGM theory of belief revision. In the example at hand, the salient categories are the event and report structure, and the belief states that range over them.

On the other hand, the connection between AGM theory and probabilistic models of belief revision offers an opportunity to exploit various insights from critical discussions concerning probabilistic models of belief dynamics. At the end of our paper, we critically discuss two such insights. The first insight draws attention to the absence of a genuine belief dynamics in probabilistic models: Bayesian models and their extensions are completely static. The second insight draw attention to a relationship with a result due to [13] identifying situations in which conditioning in so-called "naive" spaces matches conditioning in so-called "sophisticated" spaces (in which the relevant meta-information is made explicit).

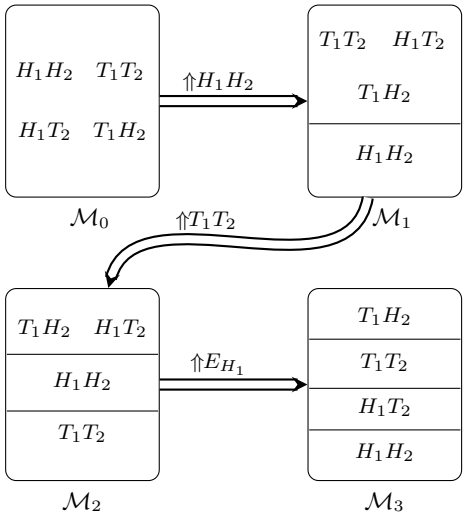
3. A HEURISTIC TREATMENT

We begin with a heuristic treatment of Stalnaker's counterexample to postulate **I2**, serving to explain the role that the Bayesian model of Section 4 plays to respond to Stalnaker's challenge to an AGM theory of iterated belief revision.

The heuristic treatment is cast in the semantic model of AGM belief revision introduced in Grove's seminal paper [12]. The key idea is to describe the belief state of an agent as a set of possible worlds and a *plausibility relation* on this set of states (formally, a plausibility ordering is a reflexive, transitive and well-founded relation). To illustrate, in the example there are four possible worlds corresponding to the configurations of the coins in the two boxes. Initially, the believer considers all the configurations of the coins equally plausible. The agent *believes* any proposition implied by the set of most plausible worlds. A *belief revision policy* describes how to modify a plausibility ordering given a nonempty subset of the set of states (intuitively, this subset represents a belief that the agent has acquired).

A number of different belief revision policies have been identified and explored in the literature (cf. [20, 3, 22]). For our discussion of Stalnaker's counterexample, we focus on the so-called **radical upgrade** belief revision policy: If φ is a set of worlds, the radical upgrade with φ, denoted $\Uparrow\varphi$, defines a new plausibility relation as follows: all the states in φ become strictly more plausibility than all the states not in φ, while the ordering for states within φ and outside of φ remains the same.

Starting from an initial model in which the believer considers all positions of the coins equally plausible, the belief changes in Stalnaker's counterexample can be represented as follows:

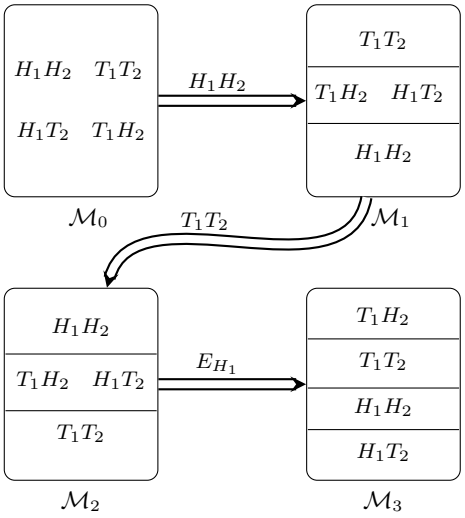


Interpret this diagram as follows: Each state is labeled by the position of the coins in the different boxs. The ordering is represented by the straight lines, with the states at the bottom the most plausible overall. For example, in model $\mathcal{M}_2$, since the state T_1T_2 is the most plausible overall, the agent believes that the coins in both boxs are lying tails up. Each transition corresponds to a radical upgrade with the identified set (we write $\Uparrow w$ instead of $\Uparrow\{w\}$, and the last transition is with the event $E_{H_1} = \{H_1H_2, H_1T_2\}$).

The above formalization highlights the crucial issue raised by Stalnaker's example: A side effect of first learning that

both coins are lying heads up followed by learning that both coins are lying tails up is that the agent comes to believe that the coins in the two boxs are correlated. Note that in the third model $\mathcal{M}_2$, the state H_1H_2 is ranked more plausible than both H_1T_2 and T_1H_2. This is not necessarily problematic provided the agent's initial beliefs about the learning situation warrant such a conclusion. However, such meta-information is not made explicit in the description of the example. This leaves open the possibility of a counterintuitive reading of the example in which it is not rational for the believer to come to the conclusion that the coins are correlated.

Our suggestion is *not* that it is impossible to define a belief revision policy that incorporates the assumption that the believer takes it for granted that the coins are independent. Indeed, the following sequence represents such a belief revision policy:



In the above formalization, each time the agent learns something about the position of the coins, the initial belief that the position of the coins are independent is retained. This leaves open the question of whether one can find a defensible belief revision policy generating such a sequence of belief changes. The models from sections 4 and 5 demonstrate that this question has an affirmative answer. The models provide a systematic way to explicitly describe the meta-information in the background underlying an application of the AGM theory of belief revision. However, as we argue in Section 6, this does not entirely resolve the issue that Stalnaker's raises.

4. BAYESIAN MODELS

In what follows, we sketch a Bayesian model formalizing salient meta-information in the example from Section 2. The model demonstrates that such information can be suitably captured in terms of a coherent set of revision rules. Of course, the model may be unsatisfactory to someone seeking to extend AGM belief revision theory with rules for iteration. Many Bayesian modeling choices, most notably concerning the representation of belief, are at odds with AGM theory. However, as we show in Section 5, the Bayesian model can be refined to cover belief revision policies in the style of AGM and, for example, Darwiche and Pearl [9] while retaining the formalization of salient intuitions. To be sure, we make no claim to a general model covering *all* cases and *all* potentially relevant meta-information. But in the example under discussion we think that insufficient detail has been offered to warrant any such claim in the first place.

The Basic Formalization

To fix ideas, a *Bayesian model* of Example 1 consists of an algebra over a set of states including all relevant propositions and a probability function expressing the agent's beliefs about these propositions as held at consecutive stages of her epistemic development. We lay down this basic structure, subsequently presenting three related probability functions accommodating meta-information.

The hypotheses at stake in the example concern the results of coin tosses in the two boxes, denoted X^i_j with $i \in \{0,1\}$ for tails up 0 and heads up 1, respectively, and $j \in \{1,2\}$ for boxes 1 and 2 respectively (here, for convenience, we use numerals rather than letters for the boxes). Furthermore, there are five reports, denoted R^i_{jt}, each with $i \in \{0,1\}$ for a report of tails up or heads up and $j \in \{1,2\}$ for boxes 1 and 2, and $t \in \{0,1,2,3\}$ for the four update stages in the epistemic development of the agent. Letting $\mathcal{X}_j = \{X^0_j, X^1_j\}$ and $\mathcal{R}_{jt} = \{R^0_{jt}, R^1_{jt}\}$, we can write the state space Ω as

$$\Omega \;=\; \mathcal{X}_1 \times \mathcal{X}_2 \times (\prod_{t=1,2,3} \mathcal{R}_{1t} \times \mathcal{R}_{2t})$$

Thus a state $\omega \in \Omega$ is of the form

$$\omega = (X_1^{i_1}, X_2^{i_2}, R_{11}^{i_3}, R_{21}^{i_4}, R_{12}^{i_5}, R_{22}^{i_6}, R_{13}^{i_7}, R_{23}^{i_8}),$$

where $i_k \in \{0,1\}$ for each $k = 1, \ldots, 8$. We take the algebra $\mathcal{F}$ to be the power set of Ω. Because reports and coins are mostly considered in pairs, we will use the abbreviations $X^{ik} = X^i_1 \cap X^k_2$ and $R^{uv}_t = R^u_{1t} \cap R^v_{2t}$.

The beliefs of the agent are represented as probability functions over this algebra, $P_t : \mathcal{F} \to [0,1]$. Summarizing the set of reports received up and until stage t by the event S_t, and taking X as the proposition of interest, the agent belief's are determined by Bayesian conditioning:

$$P_t(X) \;=\; P_0(X|S_t) = P_0(X)\frac{P_0(S_t|X)}{P_0(S_t)}.$$

In terms of the example, if we are interested in whether the coin in box 2 landed heads, X^1_2, the agent's belief state is a function of the probability conditional upon the reports of Alice and Bob, $P_1(X^1_2) = P_0(X^1_2|R^{11}_1)$.

Since the two coins are fair and independent, the priors are $P_0(X^{ik}) = \frac{1}{4}$ for all $i, k = 0, 1$. We can now fill in the probability assignments to express the specific meta-information at stake in the example. The crucial point is that we can set the initial likelihoods in accordance with different intuitions about the meta-information in the example.

The reports are independent

In this case, after receiving the reports about the coins, the agent assigns high probability to the coin in box 1 lying heads up and the coin in box 2 lying tails up. According to the example, the the content of the reports are very probable, while the content of subsequent reports are even more probable, thereby cancelling out the impact of preceding reports. We can express this in the likelihoods of the hypotheses X^{ik}. For each combination of j and t, let Q be the event

$X^i_j \cap X^k_{3-j} \cap S_{t-1} \cap R^v_{(3-j)t}$. We have:

$$P_0(R^u_{jt} \mid Q) = \frac{1}{1+\gamma^t} \times \begin{cases} 1 & \text{if } u = i, \\ \gamma^t & \text{if } u \neq i. \end{cases} \quad (1)$$

The above expression fixes the probability of all report combinations given any state of the coins. Note that the likelihoods are independent of the reports S_{t-1} of the preceding stage t, and that the reports R^u_{jt} and $R^v_{(3-j)t}$ at stage t are independent of each other too. Moreover, notice that γ is the same for each report, expressing that reports at the same stage are equally reliable. Finally, the value γ is close to zero, since the content of the reports are probable.[4]

These priors and likelihoods determine a full probability function P_0 over $\mathcal{F}$. By Bayes' rule, each probabilistic judgment at a later update stage is thereby fixed as well. We obtain the following posteriors:

Time t	0	1	2	3
After learning	$\top$	R^{11}_1	R^{00}_2	R^1_{13}
Odds for X^{11}	1	1	γ^2	γ
Odds for X^{10}	1	γ	γ	1
Odds for X^{01}	1	γ	γ	γ^3
Odds for X^{00}	1	γ^2	1	γ^2
Prob. Evidence	1	$\frac{1}{4}$	$\frac{1}{4}\gamma^2$	$\frac{1}{4}\gamma^3$

After the first update stage with R^{11}_1, when the agent has received reports on the coins in both boxes, she is highly confident that both coins have landed heads. After the second pair of reports R^{00}_2, the agent is confident that both coins have landed tails. Finally, after Elmer's report R^1_{13}, the agent has revised her opinion about the coin in box 1 while leaving her opinion about the coin in box 2 unchanged.

Natural assumptions about the relationship amongst the reports, however, lead the agent to assign high probability to the event that both coins are lying heads up, as predicted by postulate **I2**.

The reports are dependent

The meta-information in the example may be such that Elmer's report also encourages the agent to change her mind about the coin in the second box. We can organize the Bayesian model in such a way that Elmer's report indeed has these consequences. Specifically, for $t < 3$ we may choose

$$P_0(R^{uv}_t | X^{ik} \cap S_{t-1}) = \frac{1}{1+2\gamma^{1+t}+\gamma^{2+t}} \times \begin{cases} 1 & \text{if } u = i \text{ and } v = k, \\ \gamma^{1+t} & \text{if either } u \neq i, v = k \\ & \text{ or } u = i, v \neq k, \\ \gamma^{2+t} & \text{if both } u \neq i, v \neq k. \end{cases}$$

and use the likelihood of Equation (1) for $t = 3$. This indicates that to some extent, the reports stand or fall together: if one of the reports at a particular stage is false, the other one is less reliable as well. In this case, deteriorating reliability is a factor γ, but we may organize the likelihoods differently to obtain different dependencies.

With the likelihood functions set up as above, we obtain the following posterior probability assignments:

Time t	0	1	2	3
After learning	$\top$	R^{11}_1	R^{00}_2	R^1_{13}
Odds for X^{11}	1	1	γ	1
Odds for X^{10}	1	γ^2	γ^2	γ
Odds for X^{01}	1	γ^2	γ^2	γ^4
Odds for X^{00}	1	γ^3	1	γ^2
Prob. Evidence	1	$\frac{1}{4}$	$\frac{1}{4}\gamma^3$	$\frac{1}{4}\gamma^4$

In words, the first two belief changes of the agent are as before: for small γ the beliefs shift from X^{11} to X^{00} with the pairs of reports. But after the final report about the coin in box 1, the agent also revises her opinion about the coin in box 2. Importantly, this arises not because the final report about the coin in box 1 has a direct bearing on our beliefs concerning the coin in box 2, but rather because in shifting the probability mass back towards X^1_1, the dominating factor in the probability for X^1_2 becomes $P_3(X^1_2|X^1_1)$. The belief dynamics is in this sense similar to the dynamics of so-called analogical predictions (cf. [19]).

It might be suggested that the foregoing analysis somehow fails to unfold what Stalnaker has in mind:

> Because my sources were independent, my belief revision policies, at one stage, will give priority to the [possibilities of one report being false] over the [possibility of both reports being false]. (Were I to learn that [one report] was wrong, I would continue to believe [the other report] and vice versa.). (p. 207)

In a footnote, Stalnaker adds, "nothing [in] the theory as it stands provides any constraints on what counts as a single input, or any resources for representing the independence of sources."

We agree with the assertion that the AGM theory does not itself furnish such resources and so in this sense the theory is lacking. Indeed, the assertion has likeminded friends, who when read air similar platitudes about other axiomatic theories offering minimal rationality principles, theories which also abstain from imposing substantial constraints on admissible states of belief. But the assertion does not also serve as a compelling excuse to advertise a poorly posed example as a counterexample. A good counterexample is packaged for self-assembly, equipped with details obviously relevant to its evaluation and relevant to its challenge in a meaningful debate about its significance.

In the present case, Stalnaker neglects to elaborate upon the form of independence relevant to the example, and he has not articulated the example in a way univocally suggesting a particular form of independence. Even for a familiar form of independence, the incomplete example may be supplemented with details which conform to a reading according to which the truth of either report is vastly more probable, independently of the truth of the other report. Yet the deficient example may also seek assistance with details which

[4] In order for later reports to overrule earlier ones, it suffices to assume that for $t > 1$, the likelihoods are all $\frac{1}{1+\gamma^2}$. The present likelihoods indicate that reports also become increasingly reliable.

conform to another reading in which independence finds expression in terms of correlated reliability of the two reports, a reading consistent with independently varying reports.

Thus, a reply suggesting that our analysis somehow fails to unfold what Stalnaker has in mind simultaneously undertakes an obligation to articulate an argument supporting the claim that a relevantly different reading warrants recognition as an image of Stalnaker's thoughts—or at least recognition over our proposed readings.

Elmer and Carla's reports are correlated

With some imagination, we can also provide a model in which the pairs of reports are independent in the strictest sense, and in which Elmer's report is fully responsible for the belief change regarding both coins. To achieve this we employ the likelihoods of Equation (1) for the first two stages, but for the report of Elmer we use a rather gerrymandered set of likelihoods:

$$P_0(R^u_{13}|X^{ik} \cap R^{vw}_2 \cap S_1) = \frac{1}{1+\gamma^2+\gamma^3+\gamma^5} \times \begin{cases} 1 & \text{if } u = i \neq v \text{ and } w \neq k, \\ \gamma^2 & \text{if } u = i \neq v \text{ and } w = k, \\ \gamma^3 & \text{if } u \neq i = v \text{ and } w \neq k, \\ \gamma^5 & \text{if } u \neq i = v \text{ and } w = k. \end{cases}$$

Notice that the conditions on the right cover all combinations of indexes i, k, and v, but only half of their combinations with u. The likelihood for the opposite values of u follow, because the probability of R^u_{13} and R^{1-u}_{13} must add up to 1.

Of course we may vary the exact conditions under which Elmer's report overturns the reports of both Carla and Dora. Moreover, as before, the specific numerical values chosen for the likelihoods only matter up to order of magnitude. The likelihoods given here make sure that until stage 2 the posteriors are as determined by Equation (1), and we have:

Time t	2	3
After learning	$R^{11}_1 \wedge R^{00}_2$	R^1_{13}
Odds for X^{11}	γ^2	1
Odds for X^{10}	γ	γ
Odds for X^{01}	γ	γ^2
Odds for X^{00}	1	γ^3
Prob. Evidence	$\frac{1}{4}\gamma^2$	$\frac{1}{4}\gamma^4$

Importantly, the likelihoods used to arrive at these posteriors square with the example provided by Stalnaker: Elmer's report is most probably reliable and indeed overturns the report by Carla. But the likelihoods are organized in such a way that they also overturn Dora's report under particular circumstances.

The full story of the agent might be that Carla and Dora use the same method to determine the state of their respective coins. Elmer almost always defers to Carla, unless he suspects something is amiss with her method, in which case he resorts to his own superior judgment. But he will only suspect something if in actual fact both Carla and Dora report falsely. Accordingly, conditional on both Carla's and Dora's reports being false, the agent expects Elmer's report to be true and hence at odds with Carla's. Similarly, on the condition that Carla's and Dora's report are both true, the agent considers it extremely probable that Elmer's report is true and in agreement with Carla's. Finally, if either Carla's or Dora's report is false, the agent considers Elmer's report to be most probably in line with Carla's, although less probably so if Carla's report is actually the false one. The agent imagines that Elmer tends to agree with Carla because he does not suspect anything is wrong with her method, and hence most likely defers to her.

Taking a step back, we admit that there will be many more ways of filling in the priors and likelihoods so as to represent particular aspects of the meta-information. However, the details of the full solution space need not concern us here. At this point, we simply note that the puzzle allows for Bayesian models that accommodate a range of intuitions.

5. NONSTANDARD PROBABILITY

As we have already noted, the Bayesian model in the previous section does not, by itself, offer a response to Stalnaker's challenge to the AGM-based theory of iterated belief revision. In this section, we explain precisely how the Bayesian model does in fact suggest a solution to Stalnaker's challenge which is in line with the standard assumption of the AGM theory of belief revision. The key step is to forge a connection between the AGM theory of belief revision and nonstandard probability measures. This connection between AGM and nonstandard probability measures is not surprising given the results in Appendix B of [17] relating nonmonotonic logics with nonstandard probabilities.[5]

The key observation is that our discussion of the Bayesian model in the previous section and the conclusions we draw regarding Stalnaker's example do not depend on the specific values of the likelihoods used to calculate the agents' posterior beliefs. What is important are the order of magnitudes. Indeed, we can assume that the likelihoods are *arbitrarily small* and still derive the same qualitative consequences about belief change from the model. So if we represent the agents' full belief states by *nonstandard probability measures* and reinterpret the Bayesian model in those terms, we obtain a model that complies to the AGM postulates, and that nevertheless captures the role of meta-information in the desired way.

In the remainder of this section, we formally connect the nonstandard probability measures and the AGM theory of belief revision.

Definition 1. Let $\mathscr{A}$ be an algebra over a set of states Ω, and let ${}^*\mathbb{R}$ be a nonstandard model of the reals. A ${}^*\mathbb{R}$-*valued probability function* on $\mathscr{A}$ is a mapping $\mu : \mathscr{A} \to {}^*\mathbb{R}$ satisfying the following properties:

(i) $\mu(A) \geq 0$ for every $A \in \mathscr{A}$;

(ii) $\mu(\Omega) = 1$;

(iii) For all disjoint $A, B \in \mathscr{A}$: $\mu(A \cup B) = \mu(A) + \mu(B)$.

We say that μ is *regular* if $\mu(A) > 0$ for every $A \in \mathscr{A}^\circ$ (where $\mathscr{A}^\circ$ is $\mathscr{A}$ without the emptyset).

[5] In what follows, we assume the reader is familiar with the basic concepts of nonstandard analysis. See [11] for a discussion.

For a limited hyperreal[6] $r \in {}^*\mathbb{R}$, let $\mathrm{st}(r)$ be the unique real number infinitely close to r. Given a ${}^*\mathbb{R}$-valued probability function μ on an algebra $\mathscr{A}$, a collection $\mathscr{B} \subseteq \mathscr{A}$, an event $E \in \mathscr{A}$, and $r \in {}^*[0,1]$, let:

$$\mathrm{st}_r(\mu(\mathscr{B}|E)) \quad := \quad \begin{cases} \{A \in \mathscr{B} : \mathrm{st}(\mu(A|E)) \geq r\} & \text{if } \mu(E) > 0; \\ \{A \in \mathscr{B} : \mathrm{st}(\mu(A)) \geq r\} & \text{otherwise.} \end{cases}$$

When $\mathscr{A}$ is finite, we associate a set $K_\mu \in \mathscr{A}$ by setting:

$$K_\mu \quad := \quad \bigcap \mathrm{st}_1(\mu(\mathscr{A}|\Omega)).$$

Observe that K_μ is consistent, since whenever $\mathrm{st}(\mu(A)) = 1$ and $\mathrm{st}(\mu(B)) = 1$ for some $A, B \in \mathscr{A}$, $\mathrm{st}(\mu(A) + \mu(B) - \mu(A \cup B))) = \mathrm{st}(\mu(A)) + \mathrm{st}(\mu(B)) - \mathrm{st}(\mu(A \cup B)))$ and so $\mathrm{st}(\mu(A \cap B)) = 1$. Deifne an operator $*_\mu$ by setting for every $E \in \mathscr{A}$:

$$K_\mu *_\mu E \quad := \quad \begin{cases} \bigcap \mathrm{st}_1(\mu(\mathscr{A}|E)) & \text{if } E \in \mathscr{A}^\circ; \\ \emptyset & \text{otherwise.} \end{cases}$$

As before, we omit subscripts when there is no danger of confusion. The precise connection between nonstandard probability measures and the AGM theory of belief revision is given by the following Proposition:

PROPOSITION 1. *Let $\mathscr{A}$ be a finite algebra over Ω, and let $K \in \mathscr{A}^\circ$. Then $*$ is a belief revision operator for K if and only if there is a regular ${}^*\mathbb{R}$-valued probability function μ on $\mathscr{A}$ such that $K = K_\mu$ and $* = *_\mu$.*

REMARK 1. *There is also an important connection with lexicographic probability systems in the sense of of [4] (cf. [14] for a full discussion). Given a finite algebra $\mathscr{A}$, there is an obvious one-to-one correspondence between conditional probability functions and lexicographic probability systems with disjoint supports. However, even on a finite algebra, there is no nontrivial one-to-one correspondence between lexicographic probability systems with disjoint supports and ${}^*\mathbb{R}$-valued probability functions. In addition, it is clear from the connection between lexicographic probability systems and conditional probability functions that in general while it may be that $*_P = *_{P'}$ and $K_P = K_{P'}$ it does not follow that $P = P'$ and indeed $*_\mu = *_{\mu'}$ and $K_\mu = K_\mu$ does not entail that $\mu = \mu'$.*

REMARK 2. *If one wishes to admit zero probabilities in the nonstandard setting, one may introduce the concept of a ${}^*\mathbb{R}$-valued (full) conditional probability function, thereupon defining a revision operator as for ${}^*\mathbb{R}$-valued probability functions, without the implicit requirement of regularity.*

With this connection between the AGM postulates and Bayesian models using nonstandard probability measures in place, let us return to the example of Stalnaker. In virtue of the connection, we can now reinterpret the Bayesian models of the example to obtain models for the dynamics of full belief that comply to the AGM postulates, while accommodating the role of meta-information in the right way. Specifically, we can make γ, the central parameter in the definition of the likelihoods in Section 4, arbitrarily small. In the tables detailing the probabilistic belief states from the example, we thereby set all entries to the extremal values 0 or 1. Depending on how the meta-information on the dependence of reports is spelled out, the belief dynamics thus retains the desired qualitative features.

[6]A hyperreal $r \in {}^*\mathbb{R}$ is said to be *limited* if there is a (standard) natural number n such that $|r| \leq n$.

6. COUNTEREXAMPLES VS MISAPPLICATIONS

The immediate upshot of the analysis in the previous section is that the puzzle from [21] does not present insurmountable problems for a theory of iterated belief revision. After all, the nonstandard probabilistic models present us with a formally worked out revision policy. In what follows we present an evaluation of what this analysis achieves and a more nuanced view on the status of the counterexample.

We would like to flag that it is not clear from his paper that Stalnaker thinks the example reveals fundamental limitations for the AGM theory of iterated belief revision. Therefore, rather than thinking that the models prove him wrong, we think of their potential virtue as being more positive: they indicate how a belief revision policy can incorporate particular kinds of meta-information. The nonstandard Bayesian models allow us to systematically accommodate information that, in the words of Stalnaker, pertains to the conceptual, causal, and epistemic relations among factual information items. We chose to focus on information that concerns the reliability of the reports provided, and the relations that obtain between those reliabilities. The claim is certainly not that we have thereby exhausted the meta-information that may be relevant in the puzzle. But at least we have illustrated how such meta-information may come into play in a Bayesian model complying to the basic AGM postulates.

Our illustration allows us to draw some general lessons about the balance between counterexamples and misapplications in the context of modeling belief dynamics. The models bring out how tentative counterexamples can be overcome by carefully explicating various aspects from the problematic example case. Several categories of analysis deserve further analysis. First, a proper conceptualization of the event and report structure is crucial: we need a sufficiently rich structure of events, messages, epistemic states and the like to express all the meta-information. Note, however, that such a conceptualization is never part and parcel of the theory about the belief dynamics itself. A theory needs to be able to accommodate the conceptualization, but other than that it hardly counts in favor of a theory that the modeler gets this conceptualization right. Secondly it stands out that we must allow ourselves all the requisite tools for representing beliefs. In the puzzles at hand, the language must allow us to separate reports by different agents from the content of the reports. And most importantly, the expressions of belief must allow for some notion of graded disbelief or, as one may also put it, memory.

It may be thought that we think any purported counterexample can in the end be accommodated by a nonstandard Bayesian model or similar structure, and that any type of meta-information is amenable to the kind of treatment just illustrated. Are there any genuine counterexamples to be had, or do we want to reduce everything to misapplication? Here we get to the negative part of our perspective on the discussion on counterexamples to the theory of belief revision. We do believe that the theory of AGM belief revision and its probabilistic counterpart may have fundamental lim-

itations. In the remainder of this section, we first consider one specific aspect in which the probabilistic models we have provided miss the mark, suggesting that the counterexamples still stand unresolved. Secondly, we sketch some results from [13] on how far Bayesian models can come in capturing meta-information, and thereby provide a prospect for the construction of counterexamples.

Researchers coming from the literature on iterated belief revision and current dynamic logics of belief revision, may be unsatisfied with the Bayesian models presented here as a solution to Stalnaker's counterexample. The Bayesian models represent belief change by conditioning (or one of its generalization, such as Jeffrey or Adam's conditioning). It can be argued that this is not a truly *dynamic* model of belief change. If the challenge from Stalnaker's example is to capture the belief changes while maintaining the dynamic character, then the Bayesian models presented here do not present a proper rebuttal. In turn, we suggest that purported counterexample must place more emphasis on the dynamic aspect of the problem.

This raises an interesting question for future research. There seems to be a trade-off between a rich set of states and event structure, and a rich theory of "doxastic actions" (eg., as found in the literature on dynamic logics of belief revision [22, 3]). How should we resolve this trade-off when analyzing counterexamples to postulates that are intended to apply to belief changes over time. More generally, what is it about a dynamic model of belief revision that makes it truly dynamic?

We now turn to another prospect of genuine counterexamples to the theory of belief revision. For readers familiar with the flexibility of Bayesian models, it is not surprising that they allow us to formalize the relevant aspects of the meta-information in Stalnaker's example. The challenge seems rather to find out under what conditions we can ignore the meta-information, which is often not specified in the description of an example. Halpern and Grünwald [13] identify such a condition for Bayesian models, called *coarsening at random* or CAR for short. They study situations in which conditioning on a "naive" space gives the same results as conditioning on a "sophisticated" space'. Generally speaking, a "sophisticated" space is one that includes an explicit description of the relevant meta-information (eg., the reports from the sources and how they may be correlated). In the full paper we show how to apply this condition to the AGM framework. Or more precisely, we generalize the condition from [13] to nonstandard probability measures and then use the general link between AGM and nonstandard probability measures to apply the condition to AGM. In this extended abstract, however, we only have space to sketch the main idea of our result.

As indicated, CAR tells us how probabilities in naive and sophisticated spaces need to relate in order for updating by conditionalisation to be a correct inference rule in the naive space. But recall that in the generalization to nonstandard probability models, such updates follow the AGM postulates. The direct link to the examples given above is that whenever we run into a tentative counterexample, we can blame the failures of the update rule on a failure of CAR and start the repairs by building a more sophisticated state space. In cases like that, the culprit is arguably the application of the theory of belief revision: in a more refined space the update will again comply to AGM.

The same line of reasoning can now be used to clarify when misapplication turns into counterexample. In particular, we might argue that genuine counterexamples to AGM are cases in which we cannot blame failures of CAR. We see at least two ways in which this might happen. First, we might simply have no formalization of the problem case that allows for a representation of the update as a conditioning operation. Perhaps we cannot construct a sophisticated space, because the report or event structure does not allow for the definition of a partition of possible learning events. And second, it may so happen that AGM outputs an unintuitive epistemic state, even though we have employed an independently motivated formalization of the problem case. Attempts to redo the construction of a sophisticated space, just in order to remedy the failure of CAR, will be contrived. Instead, it may seem fair to blame the theory of belief revision itself.

In sum, we submit that the condition CAR may help us to formulate a principled distinction between misapplications of, and genuine counterexamples against a theory for belief dynamics. The appropriate response to the former is to refine the model and run the belief dynamics on the more refined space. Genuine counterexamples of the theory, on the other hand, are such that refinements of the model are impossible or contrived.

7. CONCLUSION

Our contribution in this paper is conceptual. First we have made explicit the meta-information implicit in one of Stalnaker's counterexample to a postulate of iterated belief revision. We have done so by identifying the salient meta-information in a heuristic model using plausibility orderings, by formalizing this information in a Bayesian model, and finally by generalizing this Bayesian model towards nonstandard probability models and showing that such models comply to the AGM postulates. This link between AGM and nonstandard probabilities allows us to use the characterization of the CAR condition to classify when a more refined state space can be used to explain the counterexample. Genuine counterexamples to AGM and iterated belief revision are cases when we cannot blame the structure of the state space. Our eventual goal is to develop a framework in which this intuition can be used to classify purported counterexamples.

8. REFERENCES

[1] C. E. Alchourrón, P. Gärdenfors, and D. Makinson. On the logic of theory change: Partial meet contraction and revision functions. *Journal of Symbolic Logic*, 50:510 – 530, 1985.

[2] H. Arló-Costa and A. P. Pedersen. Belief revision. In L. Horsten and R. Pettigrew, editors, *Continuum Companion to Philosophical Logic*. Continuum Press, 2011.

[3] A. Baltag and S. Smets. Conditional doxastic models: A qualitative approach to dynamic belief revision. In G. Mints and R. de Queiroz, editors, *Proceedings of WOLLIC 2006, Electronic Notes in Theoretical Computer Science*, volume 165, pages 5 – 21, 2006.

[4] L. Blume, A. Brandenburger, and E. Dekel. Lexicographic probabilities and choice under uncertainty. *Econometrica*, 59(1):61 – 79, 1991.

[5] C. Boutilier. Iterated revision and minimal revision of conditional beliefs. *Journal of Philosophical Logic*, 25:262 – 304, 1996.

[6] S. Chopra, A. Ghose, T. Meyer, and K.-S. Wong. Iterated belief change and the recovery axiom. *Journal of Philosophical Logic*, 37:501– 520, 2008.

[7] D. Christensen. Higher-order evidence. *Philosophy and Phenomenological Research*, 81(1):185 – 215, 2010.

[8] A. Darwiche and J. Pearl. On the logic of iterated belief revision. In *Proceedings of the 5th conference on Theoretical aspects of reasoning about knowledge*, pages 5–23, 1994.

[9] A. Darwiche and J. Pearl. On the logic of iterated belief revision. *Artificial Intelligence*, 89(1–2):1–29, 1997.

[10] N. Friedman and J. Y. Halpern. Belief revision: A critique. In L. C. Aiello, J. Doyle, and S. Shapiro, editors, *Proceedings of the Fifth International Conference on Principles of Knowledge Representation and Reasoning, KR'96*, pages 421–431. Morgan Kaufmann, Cambridge, Mass., November 1996.

[11] R. Goldblatt. *Lectures on Hyperreals: An Introduction to Nonstandard Analysis*. Springer, 1998.

[12] A. Grove. Two modellings for theory change. *Journal of Philosophical Logic*, 17(2):157–170, 1988.

[13] P. Grünwald and J. Y. Halpern. Updating probabilities. *Journal of Artificial Intelligence Research*, 19:243 – 278, 2003.

[14] J. Y. Halpern. Lexicographic probability, conditional probability, and nonstandard probability. *Games and Economic Behavior*, 68(1):155 – 179, 2010.

[15] H. Katsuno and A. O. Mendelzon. Propositional knowledge base revision and minimal change. *Artifcial Intelligence*, 52:263 – 294, 1991.

[16] D. Lehman. Belief revision, revised. In *Fourteenth International Joint Conference on Artificial Intelligence*, pages 1534–1541, 1995.

[17] D. Lehman and M. Magidor. What does a conditional knowledge base entail? *Artificial Intelligence*, 55(1):1 – 60, 1992.

[18] A. Nayak, M. Pagnucco, and P. Peppas. Dynamic belief revision operators. *Artificial Intelligence*, 146:193 – 228, 2003.

[19] J.-W. Romeijn. Analogical predictions for explicit similarity. *Erkentnis*, 64:253 – 280, 2006.

[20] H. Rott. *Change, Choice and Inference: A Study of Belief Revision and Nonmonotonic Reasoning*. Oxford University Press, 2001.

[21] R. Stalnaker. Iterated belief revision. *Erkentnis*, 70:189 – 209, 2009.

[22] J. van Benthem. Dynamic logic for belief revision. *Journal of Applied Non-classical Logics*, 17(2):129–155, 2007.

APPENDIX

A. THE AGM POSTULATES

In what follows, K is a deductively closed and consistent set of propositional formulas and φ, ψ are propositional formulas. Furthermore, $\mathrm{Cn}(X)$ denotes the propositional consequences of a set X of formulas.

The following are the *basic* revision postulates of AGM belief revision:

AGM 1	(Closure)	$K * \varphi = \mathrm{Cn}(K * \varphi)$.
AGM 2	(Success)	$\varphi \in K * \varphi$.
AGM 3	(Inclusion)	$K * \varphi \subseteq \mathrm{Cn}(K \cup \{\varphi\})$.
AGM 4	(Vacuity)	If $\neg\varphi \notin K$, then $\mathrm{Cn}(K \cup \{\varphi\}) \subseteq K * \varphi$.
AGM 5	(Consistency)	If $\mathrm{Cn}(\{\varphi\}) \neq \mathrm{For}(\mathcal{L})$, then $K * \varphi \neq \mathrm{For}(\mathcal{L})$.
AGM 6	(Extensionality)	If $\mathrm{Cn}(\{\varphi\}) = \mathrm{Cn}(\{\psi\})$, then $K * \varphi = K * \psi$.

These six basic postulates are elementary requirements of belief revision and taken by themselves are much too permissive. Additional postulates are required to rein in this permissiveness and to reflect a conception of *relational* belief revision.

AGM 7	$K * (\varphi \wedge \psi) \subseteq \mathrm{Cn}((K * \varphi) \cup \{\psi\})$.
AGM 8	If $\neg\psi \notin K * \varphi$, then $\mathrm{Cn}(K * \varphi \cup \{\psi\}) \subseteq K * (\varphi \wedge \psi)$.

In the context of a propositional model (where K is now a set of states and E, F are also sets of states), all eight postulates may be reduced to four:

Success ($*1$)	$K * E \subseteq E$.
Conditionalization ($*2$)	If $K \cap E \neq \emptyset$, then $K * E = K \cap E$.
Consistency ($*3$)	If $E \neq \emptyset$, then $K * E \neq \emptyset$.
(Arrow) ($*4$)	If $(K * E) \cap F \neq \emptyset$, then $(K * E) \cap F = K * (E \cap F)$.

We say that $*$ is a *belief revision operator* for K if it satisfies postulates ($*1$) – ($*4$). See [20] for an extended discussion.

A.1 AGM and conditional probability

In order to facilitate the relationship between the AGM theory of belief revision and nonstandard probability measures, we point out the relationship between AGM and *conditional probability measures*.

Definition 2. Let $\mathscr{A}$ be an algebra over a set of states Ω. A (*full*) *conditional probability function* on $\mathscr{A}$ is a mapping $P : \mathscr{A} \times \mathscr{A} \to \mathbb{R}$ satisfying the following properties:

(i) $P(\cdot|E)$ is a finitely additive probability function for every $E \in \mathscr{A}^\circ$;

(ii) $P(A|E) = 1$ for every $A, E \in \mathscr{A}$ such that $E \subseteq A$;

(iii) For all $A, B, E \in \mathscr{A}$ such that $A \subseteq B \subseteq E$:

$$P(A|E) \;=\; P(A|B)P(B|E).$$

Here $\mathscr{A}^\circ$ is $\mathscr{A}$ without the null event $\emptyset$. Observe that $P(\cdot|\emptyset) \equiv \mathbf{1}$.

Given a conditional probability function P on a finite algebra $\mathscr{A}$, we associate a set $K_P \in \mathscr{A}$ by setting:

$$K_P \;:=\; \mathrm{supp}\; P(\cdot|\Omega),$$

where as usual supp $P(\cdot|E)$ denotes the probabilistic support of $P(\cdot|E)$, i.e., the smallest set in $\mathscr{A}$ receiving probability one on the condition that E obtains. Define a belief revision operator $*_P$ by setting for every $E \in \mathscr{A}$:

$$K_P *_P E \quad := \quad \text{supp } P(\cdot|E).$$

We drop subscripts when the context is clear. The following is easily verified.

LEMMA 1. *Let P be a conditional probability function on a finite algebra $\mathscr{A}$ over Ω. Then $*_P$ is a belief revision operator for K_P.*

We also have the converse for consistent K, resulting in the following proposition.

PROPOSITION 2. *Let $\mathscr{A}$ be a finite algebra over Ω, and let $K \in \mathscr{A}^\circ$. Then $*$ is a belief revision operator for K if and only if there is a conditional probability function P on $\mathscr{A}$ such that $K = K_P$ and $* = *_P$.*

This concludes our exposition of AGM in relation to conditional probability functions.

A.2 Nonstandard probability

We now show how a similar link can be forged between AGM and nonstandard probability functions, admitting the possibility of arbitrarily small probability values.

Recall that given a ${}^*\mathbb{R}$-valued probability function μ on an algebra $\mathscr{A}$, a collection $\mathscr{B} \subseteq \mathscr{A}$, an event $E \in \mathscr{A}$, and $r \in {}^*[0,1]$:

$$\text{st}_r(\mu(\mathscr{B}|E)) \quad := \quad \begin{cases} \{A \in \mathscr{B} : \text{st}(\mu(A|E)) \geq r\} & \text{if } \mu(E) > 0; \\ \{A \in \mathscr{B} : \text{st}(\mu(A)) \geq r\} & \text{otherwise.} \end{cases}$$

Where $\mathscr{A}$ is finite, we associate a set $K_\mu \in \mathscr{A}$:

$$K_\mu \quad := \quad \bigcap \text{st}_1(\mu(\mathscr{A}|\Omega)).$$

Define an operator $*_\mu$ by setting for every $E \in \mathscr{A}$:

$$K_\mu *_\mu E \quad := \quad \begin{cases} \bigcap \text{st}_1(\mu(\mathscr{A}|E)) & \text{if } E \in \mathscr{A}^\circ; \\ \emptyset & \text{otherwise.} \end{cases}$$

As before, we omit subscripts when there is no danger of confusion.

LEMMA 2. *Let μ be a regular ${}^*\mathbb{R}$-valued probability function on a finite algebra $\mathscr{A}$ over Ω. Then $*_\mu$ is a belief revision operator for K_μ.*

PROOF. Clearly postulates ($*1$) and ($*3$) are satisfied. While routine, for the sake of completeness we verify postulates ($*2$) and ($*4$) in turn.

($*2$) Suppose that $K \cap E \neq \emptyset$. Let $A \in \mathscr{A}$ be such that $K * E \subseteq A$. Then $\text{st}(\mu(A|E)) = 1$. Observe that:

$$\begin{aligned} \text{st}(\mu(A \cup E^c)) &= \text{st}(\mu(E)\mu(A|E) + \mu(E^c)) \\ &= \text{st}(\mu(E))\text{st}(\mu(A|E)) + \text{st}(\mu(E^c)) \\ &= \text{st}(\mu(E) + \mu(E^c)) \\ &= 1. \end{aligned}$$

It follows that $K \cap E \subseteq A$, establishing that $K \cap E \subseteq K * E$. Now let $A \in \mathscr{A}$ be such that $K \cap E \subseteq A$. Then $K \subseteq A \cup E^c$ and so $\text{st}(\mu(A \cup E^c)) = 1$, whence:

$$\begin{aligned} \text{st}(\mu(E)) &= \text{st}(\mu(A^c \cap E)) + \text{st}(\mu(A \cap E)) \\ &= \text{st}(\mu(A \cap E)) \\ &= \text{st}(\mu(A|E))\text{st}(\mu(E)). \end{aligned}$$

Thus, since $K \cap E \neq \emptyset$, it follows that $\text{st}(\mu(E)) > 0$ and therefore $\text{st}(\mu(A|E)) = 1$, whereby $K * E \subseteq A$. Hence, $K * E \subseteq K \cap E$.

($*4$) Suppose that $(K * E) \cap F \neq \emptyset$. Let $A \in \mathscr{A}$ be such that $K * (E \cap F) \subseteq A$. Then $\text{st}(\mu(A|E \cap F)) = 1$, so:

$$\begin{aligned} \text{st}(\mu(A \cup F^c|E)) &= 1 - \text{st}(\mu((A^c \cap F)|E)) \\ &= 1 - \text{st}(\mu(A^c|F \cap E))\text{st}(\mu(F|E)) \\ &= 1. \end{aligned}$$

Hence, $(K*E) \cap F \subseteq A$, showing that $(K*E) \cap F \subseteq K*(E \cap F)$. Now let $A \in \mathscr{A}$ be such that $(K*E) \cap F \subseteq A$. Then $\text{st}(\mu(A \cup F^c|E)) = 1$, and since $(K * E) \cap F \neq \emptyset$, it follows that $\text{st}(\mu(F|E)) \neq 0$, so:

$$\begin{aligned} \text{st}(\mu(A|E \cap F))) &= 1 - \text{st}(\mu(A^c|E \cap F))) \\ &= 1 - \text{st}(\frac{\mu((A^c \cap F)|E)}{\mu(F|E)}) \\ &= 1 - \frac{\text{st}(\mu((A^c \cap F)|E))}{\text{st}(\mu(F|E))} \\ &= 1. \end{aligned}$$

Therefore, $K*(E \cap F) \subseteq A$, so $K*(E \cap F) \subseteq (K*E) \cap F$, as desired. □

PROPOSITION 3. *Let $\mathscr{A}$ be a finite algebra over Ω, and let $K \in \mathscr{A}^\circ$. Then $*$ is a belief revision operator for K if and only if there is a regular ${}^*\mathbb{R}$-valued probability function μ on $\mathscr{A}$ such that $K = K_\mu$ and $* = *_\mu$.*

PROOF. The 'if' part has been established in Lemma 2. We turn to the 'only if' part. Let P be a conditional probability function on $\mathscr{A}$ such that $K = K_P$ and $* = *_P$, as given by Proposition 2. Then there is a partition $(\pi_m)_{m<n}$ of Ω in $\mathscr{A}$ and a sequence $(\mu_m)_{m<n}$ of real-valued probability functions on $\mathscr{A}$ such that:

(a) $\pi_m = \text{supp } \mu_m$ for each $m < n$;
(b) $P(\cdot|E) = \mu_{\min\{m : E \cap \pi_m \neq \emptyset\}}(\cdot|E)$ for every $E \in \mathscr{A}^\circ$.

Define a regular ${}^*\mathbb{R}$-valued probability function μ on $\mathscr{A}$ by setting for every $A \in \mathscr{A}$:

$$\mu(A) \quad := \quad \mu_0(A) + \sum_{0<m<n} (\mu_m(A) - \mu_0(A))\epsilon^m,$$

where ϵ is a positive infinitesimal. We claim that (i) $K_P = K_\mu$ and that (ii) $*_P = *_\mu$. Clearly $K *_P \emptyset = K *_\mu \emptyset$. Now let $E \in \mathscr{A}^\circ$. Set $m_0 := \min\{m : E \cap \pi_m \neq \emptyset\}$, and for each $m < n$, let $\nu_m := \mathbf{0}$ if $m = 0$ and μ_m otherwise. Then for every $A \in \mathscr{A}$:

$$\begin{aligned} &\text{st}(\mu(A|E)) \\ &= \text{st}(\tfrac{\mu_0(A\cap E)+\sum_{0<m<n}(\mu_m(A\cap E)-\mu_0(A\cap E))\epsilon^m}{\mu_0(E)+\sum_{0<m<n}(\mu_m(E)-\mu_0(E))\epsilon^m}) \\ &= \text{st}(\frac{\mu_{m_0}(A|E)+\sum_{m_0<m<n}\frac{\mu_m(A\cap E)-\nu_m(A\cap E)}{\mu_{m_0}(E)}\epsilon^{m-m_0}}{1+\sum_{m_0<m<n}\frac{\mu_m(E)-\nu_m(E)}{\mu_{m_0}(E)}\epsilon^{m-m_0}}) \\ &= \frac{\text{st}(\mu_{m_0}(A|E)+\sum_{m_0<m<n}\frac{\mu_m(A\cap E)-\nu_m(A\cap E)}{\mu_{m_0}(E)}\epsilon^{m-m_0})}{\text{st}(1+\sum_{m_0<m<n}\frac{\mu_m(E)-\nu_m(E)}{\mu_{m_0}(E)}\epsilon^{m-m_0})} \\ &= \mu_{m_0}(A|E). \end{aligned}$$

Then by property (b), claims (i) and (ii) follow, thereby establishing the desired conclusion. □

REMARK 3. *The sequence $(\mu_m)_{m<n}$ of real-valued probability functions in the proof of Proposition 3 is a lexicographic probability system as discussed in Remark 1.*

Agreeing on decisions: an analysis with counterfactuals

Bassel Tarbush
Department of Economics, University of Oxford
bassel.tarbush@economics.ox.ac.uk

ABSTRACT

Moses & Nachum ([7]) identify conceptual flaws in Bacharach's generalization ([3]) of Aumann's seminal "agreeing to disagree" result ([1]). Essentially, Bacharach's framework requires agents' decision functions to be defined over events that are informationally meaningless for the agents. In this paper, we argue that the analysis of the agreement theorem should be carried out in information structures that can accommodate for counterfactual states. We therefore develop a method for constructing such "counterfactual structures" (starting from partitional structures), and prove a new agreement theorem within such structures. Furthermore, we show that our approach also resolves the conceptual flaws in the sense that, within our framework, decision functions are always only defined over events that are informationally meaningful for the agents.

Categories and Subject Descriptors

J.4 [**Social and behavioral sciences**]: Economics; I.2.4 [**Knowledge Representation Formalisms and Methods**]: Frames and scripts

General Terms

Theory

Keywords

Agreeing to disagree, knowledge, belief, counterfactuals

1. INTRODUCTION

In [3], Bacharach generalized Aumann's seminal "agreeing to disagree" result ([1]) to the non-probabilistic case. Essentially, he isolated the relevant properties that hold of conditional probabilities, and of the common prior assumption - which drive the original result - and imposed them as independent conditions on general decision functions in partitional information structures. As such, he was able to isolate and interpret the underlying assumptions of the original result as (i) an assumption of "like-mindedness", which requires agents to take the same action given the same information, and (ii) an assumption that he claimed is analogous to requiring the agents' decision functions to satisfy Savage's Sure-Thing Principle ([9]). This principle is intended to capture the intuition that "if an agent takes the same action in every case when she is more informed, she takes the same action in the case when she is more ignorant".

However, in [7], Moses & Nachum found conceptual flaws in Bacharach's analysis, showing that his interpretations of "like-mindedness" and of the Sure-Thing Principle are problematic. Indeed, given that Bacharach is operating within partitional information structures, the information of agents is modeled as partitions of the state space. Furthermore, decision functions are defined over sets of states in a manner that is supposed to be consistent with the information that each agent has - in this way, decisions can be interpreted as being functions of agents' information. In Bacharach's set-up, like-mindedness requires the decision function of an agent i to be defined over elements of the partitions of other agents j. But, except for the trivial case in which agent i's partition element corresponds exactly to that of agent j, there is no sense in requiring i's function to be defined over j's partition element since that element is informationally meaningless to agent i. The Sure-Thing Principle is also problematic. An agent's decision function is said to satisfy the Sure-Thing Principle if whenever the decision over each element of a set of disjoint events is x, the decision over the union of all those events is also x. Notably, this implies that an agent's decision function must be defined over the union of her partition elements, but again, this is informationally meaningless for that agent since there is no partition element of that agent that corresponds to a union of her partition elements. More generally, Moses & Nachum show that Bacharach's set-up is such that the domains of the agents' decision functions contain elements that are informationally meaningless for the agents.

The basic premise of this paper is that the Sure-Thing Principle ought to be understood as an inherently counterfactual notion, and so any analysis that involves this principle but is carried out in an information structure that does not explicitly model the counterfactuals must be lacking in some way. Indeed, one could reformulate the intuition that the Sure-Thing Principle is intended to capture as: "If the agent takes the same action in every case when she is more informed, she *would* take the same action *if she were* more ignorant".

This distinction is important, but cannot be captured within Bacharach's framework because his analysis in [3] is carried out in partitional structures, and all information in those structures must be factual (in the sense that any belief

TARK 2013, Chennai, India.

that an agent holds must be true). In this paper, we therefore develop a method of transforming any given partitional structure into an information structure that explicitly includes the relevant counterfactual states. We interpret these "counterfactual structures" as being more complete pictures of the situation that is being modeled in the original partitional structure. The new set-up allows us to provide new formal definitions of the Sure-Thing Principle and of like-mindedness, that sit well with intuition, and we prove a new agreement theorem within these counterfactual structures.

Ultimately we show that our set-up resolves the conceptual issues raised by [7], in the sense that within counterfactual structures, decision functions are always only defined over events that are informationally meaningful for the agents.

In section 2 we present the formal definitions required to analyze information structures, and in section 3 we set up the framework of Bacharach, prove his version of the agreement theorem and provide Moses & Nachum's argument regarding the conceptual flaws. In section 4 we develop a method for constructing counterfactual structures, provide new definitions for the Sure-Thing Principle and for like-mindedness, and prove a new agreement theorem within such structures. Furthermore, we show that our approach resolves the conceptual flaws. Finally, in section 5 we relate our approach to other results and proposed solutions to the conceptual flaws found in the "agreeing to disagree" literature, and section 6 concludes. All proofs are in the appendix.

2. INFORMATION STRUCTURES

This section introduces the formal apparatus that will be used to derive the agreement theorem. In large part, the formal definitions given are completely standard.

2.1 General information structures

Let Ω denote a finite set of *states* and N a finite set of agents. A subset $e \subseteq \Omega$ is called an *event*. For every agent $i \in N$, define a binary relation $R_i \subseteq \Omega \times \Omega$, called a *reachability* relation. So, we say that the state $\omega \in \Omega$ *reaches* the state $\omega' \in \Omega$ if $\omega R_i \omega'$. It terms of interpretation, if $\omega R_i \omega'$, then at ω, agent i considers the state ω' *possible*. An *information structure* $\mathcal{S} = (\Omega, N, \{R_i\}_{i \in N})$ is entirely determined by the state space, the set of agents, and the reachability relations.

The reachability relations $\{R_i\}_{i \in N}$ are said to be:

1. Serial if $\forall i \in N, \forall \omega \in \Omega, \exists \omega' \in \Omega, \omega R_i \omega'$.
2. Reflexive if $\forall i \in N, \forall \omega \in \Omega, \omega R_i \omega$.
3. Transitive if $\forall i \in N, \forall \omega, \omega', \omega''' \in \Omega$, if $\omega R_i \omega' \,\&\, \omega' R_i \omega''$, then $\omega R_i \omega''$.
4. Euclidean if $\forall i \in N, \forall \omega, \omega', \omega''' \in \Omega$, if $\omega R_i \omega' \,\&\, \omega R_i \omega''$, then $\omega' R_i \omega''$.

We have not yet imposed any particular restrictions on the reachability relations. We will therefore provide the definitions below in a general setting, with the understanding that they will only be applied in (i) **S5**, (ii) **KD45** and (iii) a special class of **KD4** structures. Respectively, this is when the reachability relations are (i) equivalence relations (reflexive and Euclidean), (ii) serial, transitive and Euclidean, and (iii) serial and transitive.

A *possibility set* at state ω for agent $i \in N$ is defined by

$$b_i(\omega) = \{\omega' \in \Omega | \omega R_i \omega'\} \tag{1}$$

A possibility set $b_i(\omega)$ is therefore, simply the set of all states that i considers possible at ω. In terms of notation, let us have $\mathcal{B}_i = \{b_i(\omega) | \omega \in \Omega\}$. For any $e \subseteq \Omega$, a *belief operator* is given by

$$B_i(e) = \{\omega \in \Omega | b_i(\omega) \subseteq e\} \tag{2}$$

Also, for any $e \subseteq \Omega$, and any $G \subseteq N$, a *mutual belief operator* is given by

$$M_G(e) = \cap_{i \in G} B_i(e) \tag{3}$$

This operator can be iterated by letting $M_G^1(e) = M_G(e)$ and $M_G^{m+1}(e) = M_G(M_G^m(e))$ for $m \geq 1$. For any $e \subseteq \Omega$, and any $G \subseteq N$, we can thus define a *common belief operator*,

$$C_G(e) = \cap_{m=1}^{\infty} M_G(e) \tag{4}$$

Finally, we say that a state $\omega' \in \Omega$ is *reachable* among the agents in G from a state $\omega \in \Omega$ if there exists $\omega \equiv \omega_0, \omega_1, \omega_2, ..., \omega_n \equiv \omega'$ such that for each $k \in \{0, 1, ..., n-1\}$, there exists an agent $i \in G$ such that $\omega_k R_i \omega_{k+1}$. The *component* $T_G(\omega)$ (among the agents in G) of the state ω is the set of all states that are reachable among the agents in G from ω. Common belief can now be given an alternative characterization,

$$C_G(e) = \{\omega \in \Omega | T_G(\omega) \subseteq e\} \tag{5}$$

This is standard, and for example follows [6, p. 12].

2.2 Partitional structures

Consider an information structure $\mathcal{S} = (\Omega, N, \{R_i\}_{i \in N})$ and suppose that the reachability relations $\{R_i\}_{i \in N}$ are equivalence relations. Then, we say that $\mathcal{S}$ is a *partitional structure*. Indeed, the remark below shows that in this case, the information structure $\mathcal{S}$ becomes a standard partitional, or **S5**, or "knowledge" structure (see for example, [1]).

REMARK 1. *Suppose $\mathcal{S} = (\Omega, N, \{R_i\}_{i \in N})$ is a partitional structure. For any agent $i \in N$, $\omega \in b_i(\omega)$, and any $b_i(\omega)$ and $b_i(\omega')$ are either identical or disjoint; and, $\mathcal{B}_i$ is a partition of the state space.*

Note that in a partitional structure, at any state ω, an agent i considers any of the states in $b_i(\omega)$ (including ω itself) possible. The belief operator becomes the standard "knowledge" operator, and satisfies the following properties, which are well-known in the literature:

K $B_i(\neg e \cup f) \cap B_i(e) \subseteq B_i(f)$	*Kripke*
D $B_i(e) \subseteq \neg B_i(\neg e)$	*Consistency*
T $B_i(e) \subseteq e$	*Truth*
4 $B_i(e) \subseteq B_i(B_i(e))$	*Positive Introspection*
5 $\neg B_i(e) \subseteq B_i(\neg B_i(e))$	*Negative Introspection*

Note that in a partitional structure, the operator C_G has the familiar interpretation of being the "common knowledge" operator. Furthermore, since this reduces to a completely standard framework, we easily obtain familiar results, such as the proposition below.

PROPOSITION 1. *Suppose $\mathcal{S} = (\Omega, N, \{R_i\}_{i\in N})$ is a partitional structure. Then, for any $\omega \in \Omega$ and any $i \in G$, $\cup_{\omega' \in T_G(\omega)} b_i(\omega') = T_G(\omega)$.*

2.3 Belief structures

Suppose now that the reachability relations $\{R_i\}_{i\in N}$ in an information structure $\mathcal{S} = (\Omega, N, \{R_i\}_{i\in N})$ are serial, transitive and Euclidean. Then, say we that $\mathcal{S}$ is a *belief structure*. Indeed, the information structure $\mathcal{S}$ becomes a standard **KD45** structure, for example, as presented in [6].

REMARK 2. *Suppose $\mathcal{S} = (\Omega, N, \{R_i\}_{i\in N})$ is a belief structure. For any agent $i \in N$, and any $\omega \in \Omega$, $b_i(\omega) \neq \emptyset$, and if $\omega \in b_i(\omega')$, then $b_i(\omega) = b_i(\omega')$.*

It is important to note that although every possibility set must be non-empty, it can be the case that $\omega \notin b_i(\omega)$. This means that at state ω, agent i considers states *other than* ω to be possible, and not ω itself. The agent is therefore "deluded". (In fact, this terminology is directly borrowed from [6, p. 5]). Unsurprisingly, the belief operator now no longer satisfies the truth property **T**, but it does satisfy **K**, **D**, **4**, and **5**.

The salient point here is that the set-up presented has very close analogues in the literature, and allows us to drop - among other things - the property **T** of the belief operator, as compared with partitional structures. This will be important when including counterfactual states since by their very nature, these will be used to model information that can be false.

3. AGREEING ON DECISIONS

In this section, we present the original set-up of [3], derive his version of the agreement theorem, and then outline its inherent conceptual flaws which were originally raised in [7].

3.1 The original result

The original result was derived in a partitional information structure. The set-up in this entire section therefore assumes that we are working with a partitional structure $\mathcal{S} = (\Omega, N, \{R_i\}_{i\in N})$. Notably, this means that $\mathcal{B}_i$ is taken to be a partition of the state space for every agent $i \in N$ (see Remark 1).

For every agent $i \in N$, an *action function* $\delta_i : \Omega \to \mathcal{A}$, which maps from states to actions, specifies agent i's action at any given state as a function of i's possibility set at that state (which is intended to represent i's "information" at that state); so the value of the action function will fully depend on the partition $\mathcal{B}_i$. A *decision function* D_i for agent i, maps from a field $\mathcal{F}$ of subsets of Ω into a set $\mathcal{A}$ of actions. That is,

$$D_i : \mathcal{F} \to \mathcal{A} \tag{6}$$

Following the terminology of [7], we will say that the agent i using the action function δ_i *follows* the decision function D_i if for all states $\omega \in \Omega$, $\delta_i(\omega) = D_i(b_i(\omega))$.

Bacharach imposes two main restrictions in order to derive his result, namely, the *Sure-Thing Principle* and *like-mindedness*. The definitions of these terms are given below.

Definition 1. The decision function D_i of agent i satisfies the *Sure-Thing Principle* if whenever for all $e \in \mathcal{E}$, $D_i(e) = x$ then $D_i(\cup_{e\in\mathcal{E}} e) = x$, where $\mathcal{E} \subseteq \mathcal{F}$ is a non-empty set of disjoint events.

In terms of interpretation, we can think of an event as representing some information and a decision over that event as determining the action that is taken as a function of that information. The union of events is intended to capture some form of "coarsening" of the information. So, following [7], the Sure-Thing Principle is intended to capture the intuition that "If the agent takes the same action in every case when she is more informed, she takes the same action in the case when she is more ignorant". Regarding like-mindedness, we have the following definition.

Definition 2. Agents are said to be *like-minded* if they have the same decision function.

That is, over the same subsets of states, the agents take the same action if they are like-minded. This is intended to capture the intuition that given the same information, the agents would take the same action.

THEOREM 1. *Let $\mathcal{S} = (\Omega, N, \{R_i\}_{i\in N})$ be a partitional structure. Then within $\mathcal{S}$, if the agents $i \in N$ are like-minded (as defined in Definition 2) and follow the decision functions $\{D_i\}_{i\in N}$ (as defined in (6)) that satisfy the Sure-Thing Principle (as defined in Definition 1), then for any $G \subseteq N$, if $C_G(\cap_{i\in G}\{\omega' \in \Omega | \delta_i(\omega') = x_i\}) \neq \emptyset$ then $x_i = x_j$ for all $i, j \in G$.*

This theorem states that if the actions taken by each member of a group of like-minded agents, who follow decision functions that satisfy the Sure-Thing Principle, are common knowledge among that group, then the members of the group must all take the same action. That is, the agents cannot "agree to disagree" about what action to take.

3.2 Conceptual flaws

[7] find conceptual flaws in the set-up of [3] outlined above. In broad terms, they find that the requirements that Bacharach imposes on the decision functions forces them to be defined over sets of states, the interpretation of which is meaningless within the information structure he is operating in. Formally, consider the following definition.

Definition 3. Let $\mathcal{S} = (\Omega, N, \{R_i\}_{i\in N})$ be some arbitrary information structure. We say that an event e is a *possible belief* for agent i in $\mathcal{S}$ if there exists a state $\omega \in \Omega$ such that $e = b_i(\omega)$.

When $\mathcal{S}$ is a partitional structure, this definition corresponds exactly to e being a "possible state of knowledge" as defined in [7]. In [7], it is shown that

1. The Sure-Thing Principle forces decisions to be defined over unions of possibility sets, but no union of possibility sets can be a possible belief for any agent (see [7, Lemma 3.2]).
2. The assumption of like-mindedness forces the decision function of an agent i to be defined over the possibility sets of agents $j \neq i$, but - other than the case when they correspond trivially - these are not possible beliefs for agent i (see [7, Lemma 3.3]).

In other words, Bacharach's framework requires the decision functions to be defined over events that are not possible beliefs for the agents (within the information structure).

4. COUNTERFACTUAL STRUCTURES

The basic premise of this paper is that the Sure-Thing Principle ought to be understood as an inherently counterfactual notion, and so any analysis that involves this principle but is carried out in an information structure that does not explicitly model the counterfactuals must be lacking in some way. Indeed, one could reformulate the intuition that the Sure-Thing Principle is intended to capture as: "If the agent takes the same action in every case when she is more informed, she *would* take the same action *if she were* more ignorant" (where "more ignorant" has a well-defined meaning). This is counterfactual in the sense that there is no requirement for the agent to actually be more ignorant. Rather, the requirement is that the agent would take the same action in the situation where she imagines herself, counterfactually, to be more ignorant.

This distinction is important, but cannot be captured within Bacharach's framework. Indeed, the analysis in [3] is carried out in partitional structures. However, since the truth property **T** holds in such structures, every conceivable belief must be factual, and so by definition, counterfactual situations cannot be considered.[1] In this section, we therefore develop a method of transforming any given partitional structure into an information structure that explicitly includes the relevant counterfactual states. We interpret such "counterfactual structures" as being more complete pictures of the situation being modeled in the original partitional structure. We then provide new formal definitions for the Sure-Thing Principle and for like-mindedness and derive a new agreement theorem within these new structures. Ultimately this will resolve the conceptual issues raised by [7], in the sense that within counterfactual structures, decision functions are defined only over events that are possible beliefs for the agents.

4.1 Set-up with counterfactual states

In this section we define a method of transforming any given partitional structure into an information structure that explicitly includes the relevant counterfactual states.

It will be useful to introduce some new definitions. Suppose $\mathcal{S} = (\Omega, N, \{R_i\}_{i\in N})$ is a partitional structure. For every agent $i \in N$, define $I_i(\omega) = \{\omega' \in \Omega | \omega R_i \omega'\}$. Trivially, $I_i(\omega)$ is the equivalence class of the state ω, and for each $i \in N$, $\mathcal{I}_i = \{I_i(\omega)|\omega \in \Omega\}$ is a partition of the state space (by Remark 1). Finally, let us define,

$$\Gamma_i = \{\cup_{e\in\mathcal{E}} e | \mathcal{E} \subseteq \mathcal{I}_i, \mathcal{E} \neq \emptyset\} \quad (7)$$

Clearly, Γ_i consists of all the partition elements of i, and of all the possible unions across those partitions elements.

Construction of counterfactuals. Let $\mathcal{S} = (\Omega, N, \{R_i\}_{i\in N})$ be a partitional structure. We can immediately define $I_i(\omega) = \{\omega' \in \Omega | \omega R_i \omega'\}$, the partition $\mathcal{I}_i = \{I_i(\omega)|\omega \in \Omega\}$, and the set Γ_i (described above) for every $i \in N$. From $\mathcal{S}$, we can create a new structure $\mathcal{S}' = (\Omega', N, \{R'_i\}_{i\in N})$, which we call the *counterfactual structure* of $\mathcal{S}$, where $\Omega' = \Omega \cup \Lambda$, Λ is a set of states distinct from Ω, and $R'_i \subseteq \Omega' \times \Omega$ is a reachability relation for every $i \in N$. The construction of the set Λ and of the reachability relations $\{R'_i\}_{i\in N}$ is described below.

- For every $i \in N$, and for every $e \in \Gamma_i$, create a set Λ^e_i of *new* states, which contains exactly one *duplicate* $\lambda^e_{i,\omega}$ of the state ω for every $\omega \in \Omega$ (so $|\Lambda^e_i| = |\Omega|$). We say that the *counterfactual state* $\lambda^e_{i,\omega}$ is the *counterfactual* of ω for agent i with respect to the event e. The set of states Λ is simply the set of all counterfactual states. Namely, $\Lambda = \cup_{i\in N} \cup_{e\in\Gamma_i} \Lambda^e_i$.[2]

- We now describe the process to construct the reachability relations $\{R'_i\}_{i\in N}$. For every agent $i \in N$, start with $R'_i = R_i$. We will add new elements to R'_i according to the following method: For every $\lambda \in \Lambda$, if $\lambda = \lambda^e_{i,\omega}$ for some $\omega \in \Omega$ and $e \in \Gamma_i$, then (i) if $\omega \in e$ (that is, if $\lambda^e_{i,\omega}$ is the duplicate of a state in e), then for every $\omega' \in e$, add $(\lambda^e_{i,\omega}, \omega')$ as an element to R'_i, and (ii) if $\omega \notin e$, then for every $\omega' \in I_i(\omega)$, add $(\lambda^e_{i,\omega}, \omega')$ as an element to R'_i. Finally, if $\lambda = \lambda^e_{j,\omega}$ for some $\omega \in \Omega$, and $e \in \Gamma_j$ where $j \in N\backslash\{i\}$, then for every $\omega' \in I_i(\omega)$, add $(\lambda^e_{j,\omega}, \omega')$ as an element to R'_i. Nothing else is an element of R'_i.

This is best explained by means of an example. Consider a partitional structure $\mathcal{S}$ with $\Omega = \{\omega_0, \omega_1, \omega_2, \omega_3, \omega_4\}$, $N = \{a, b\}$, and partitions $\mathcal{I}_a$ and $\mathcal{I}_b$ as represented in Figure 1. In Figures 2-4, we represent a selection of substructures of the counterfactual structure $\mathcal{S}'$ of $\mathcal{S}$.[3] Figure 2 shows the set of counterfactual states $\Lambda^{\{\omega_3,\omega_4\}}_a$, as well as Ω, and the reachability relations, $R'_i \subseteq \Lambda^{\{\omega_3,\omega_4\}}_a \times \Omega$, of both agents across these two sets. The reachability relations $R'_i \subseteq \Omega \times \Omega$ are left out, but they are unchanged (relative to $\mathcal{S}$) and therefore identical to what is shown in Figure 1. Note that each state in $\Lambda^{\{\omega_3,\omega_4\}}_a$ is simply a duplicate of a corresponding state in Ω. For agent b, every state $\lambda^{\{\omega_3,\omega_4\}}_{a,\omega}$ simply points to all the states $\omega' \in I_b(\omega)$ (and nothing else). For agent a, every state $\lambda^{\{\omega_3,\omega_4\}}_{a,\omega}$ such that $\omega \in \{\omega_0, \omega_1, \omega_2\}$ simply points to all the states $\omega' \in I_a(\omega)$ (and nothing else). However, for a state $\omega \in \{\omega_3, \omega_4\}$, every state $\lambda^{\{\omega_3,\omega_4\}}_{a,\omega}$ points to both ω_3 *and* ω_4 (and nothing else), even though $I_i(\omega_3) \cap I_i(\omega_4) = \emptyset$. A similar patterns holds in Figures 3 and 4 which are there as additional examples for the reader. For practical reasons, we do not represent the full sets Λ and $R'_i \subseteq \Omega' \times \Omega$ in a single diagram; and, note that even when taken together Figures 1-4 do not offer a complete picture of $\mathcal{S}'$.

The counterfactual structure of a partitional structure has several interesting properties, which we derive below.

PROPOSITION 2. *Suppose that $\mathcal{S}' = (\Omega', N, \{R'_i\}_{i\in N})$ is the counterfactual structure of a partitional structure $\mathcal{S} =$*

[1] An agent i's belief in an event E if factual if $B_i(E) \subseteq E$.

[2] Note that the indexing of the sets Λ^e_i by both e and i is crucial. Indeed, one must note that for any $i \in N$, and for any $e, e' \in \Gamma_i$ such that $e \neq e'$, $\Lambda^e_i \cap \Lambda^{e'}_i = \emptyset$. Furthermore, for any $i, j \in N$ such that $i \neq j$, if $e \in \Gamma_i$ and $e' \in \Gamma_j$, $\Lambda^e_i \cap \Lambda^{e'}_j = \emptyset$ (even if $e = e'$).

[3] Consider any two information structures $\mathcal{S}^+ = (\Omega^+, N, \{R^+_i\}_{i\in N})$ and $\mathcal{S}^- = (\Omega^-, N, \{R^-_i\}_{i\in N})$. We say that $\mathcal{S}^-$ is a *substructure* of $\mathcal{S}^+$ if $\Omega^- \subseteq \Omega^+$ and $R^-_i \subseteq R^+_i$ for every $i \in N$.

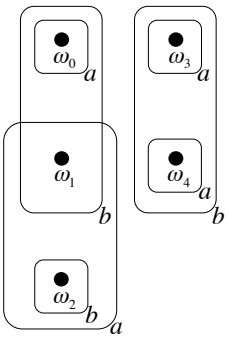


Figure 1: Ω and the partitions $\mathcal{I}_a$ and $\mathcal{I}_b$

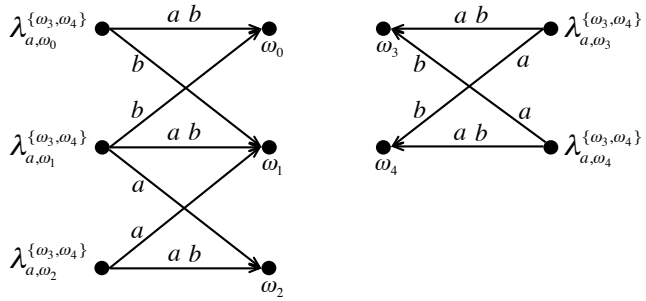


Figure 2: $\Lambda_a^{\{\omega_3,\omega_4\}} \cup \Omega$ and $R_i' \subseteq \Lambda_a^{\{\omega_3,\omega_4\}} \times \Omega$ for $i \in \{a, b\}$

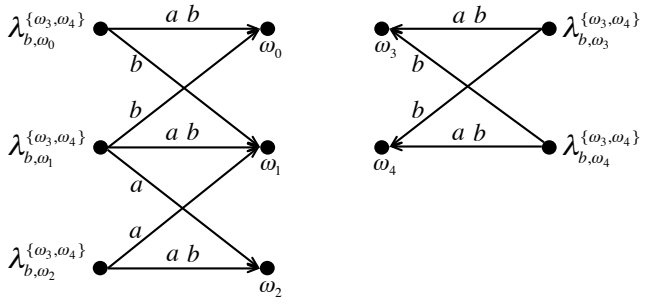


Figure 3: $\Lambda_b^{\{\omega_3,\omega_4\}} \cup \Omega$ and $R_i' \subseteq \Lambda_b^{\{\omega_3,\omega_4\}} \times \Omega$ for $i \in \{a, b\}$

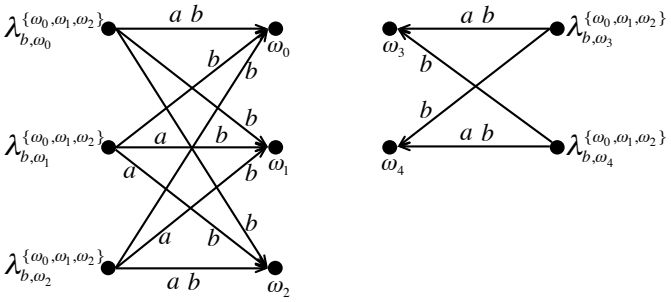


Figure 4: $\Lambda_b^{\{\omega_0,\omega_1,\omega_2\}} \cup \Omega$ and $R_i' \subseteq \Lambda_b^{\{\omega_0,\omega_1,\omega_2\}} \times \Omega$ for $i \in \{a, b\}$

$(\Omega, N, \{R_i\}_{i\in N})$. *Then the reachability relations* $\{R_i'\}_{i\in N}$ *are serial and transitive.*

PROPOSITION 3. *Suppose that* $\mathcal{S}' = (\Omega', N, \{R_i'\}_{i\in N})$ *is the counterfactual structure of a partitional structure* $\mathcal{S} = (\Omega, N, \{R_i\}_{i\in N})$. *Then for any agent* $i \in N$, *(i) for any* $\omega \in \Omega'$, $b_i(\omega) \neq \emptyset$, *and if* $\omega \in b_i(\omega')$, $b_i(\omega) \subseteq b_i(\omega')$, *and (ii) for any* $\omega \in \Omega$, $b_i(\omega) = I_i(\omega)$.

From the above, we have that counterfactual structures of partitional structures belong to the class of **KD4** structures. In particular, the belief operator now only satisfies properties **K**, **D**, and **4**; so "negative introspection" no longer holds, relative to belief structures. (See section 5.2 for further discussion of this point). Note however that within the counterfactual structure $\mathcal{S}' = (\Omega', N, \{R_i'\}_{i\in N})$ of a partitional structure $\mathcal{S} = (\Omega, N, \{R_i\}_{i\in N})$, the substructure $(\Omega, N, \{R_i\}_{i\in N})$ of $\mathcal{S}'$ corresponds exactly to the original structure $\mathcal{S}$ and is therefore partitional. A further result will be useful.

PROPOSITION 4. *Suppose that* $\mathcal{S}' = (\Omega', N, \{R_i'\}_{i\in N})$ *is the counterfactual structure of a partitional structure* $\mathcal{S} = (\Omega, N, \{R_i\}_{i\in N})$. *Then for any* $\omega \in \Omega'$ *and any* $G \subseteq N$, *(i) if* $\omega' \in T_G(\omega)$, *then* $\omega' \in \Omega$, *and (ii) for any* $i \in G$, $\cup_{\omega' \in T_G(\omega)} b_i(\omega') = T_G(\omega)$.

4.2 The agreement theorem

We will now adapt the main definitions required to derive the agreement theorem within the counterfactual structure of a partitional structure.

Throughout this section, we consider a partitional structure $\mathcal{S} = (\Omega, N, \{R_i\}_{i\in N})$, and the counterfactual structure $\mathcal{S}' = (\Omega', N, \{R_i'\}_{i\in N})$ of $\mathcal{S}$. As before, we can define $I_i(\omega) = \{\omega' \in \Omega | \omega R_i \omega'\}$, the partition $\mathcal{I}_i = \{I_i(\omega) | \omega \in \Omega\}$, and the set Γ_i for every $i \in N$.

A decision function D_i for an agent $i \in N$ maps from Γ_i to a set of actions. That is,

$$D_i : \Gamma_i \to \mathcal{A} \tag{8}$$

We now say that an action function $\delta_i : \Omega' \to \mathcal{A}$ *follows* decision function D_i if for all states $\omega \in \Omega'$, $\delta_i(\omega) = D_i(b_i(\omega))$. The following proposition guarantees that this is well-defined.

PROPOSITION 5. *Suppose that* $\mathcal{S}' = (\Omega', N, \{R_i'\}_{i\in N})$ *is the counterfactual structure of a partitional structure* $\mathcal{S} = (\Omega, N, \{R_i\}_{i\in N})$. *Then for any* $\omega \in \Omega'$, $b_i(\omega) \in \Gamma_i$.

Below, we provide definitions for the Sure-Thing Principle and like-mindedness that are analogous to the ones proposed by Bacharach. We elaborate on their interpretations in section 4.4.

Definition 4. The decision function D_i of agent i satisfies the *Sure-Thing Principle* if for any non-empty subset $\mathcal{E}$ of $\mathcal{I}_i$, whenever for all $e \in \mathcal{E}$, $D_i(e) = x$ then $D_i(\cup_{e\in\mathcal{E}} e) = x$.

The domain Γ_i includes all possible unions of elements of the partition $\mathcal{I}_i$, so this is well-defined. Furthermore, note that $\mathcal{E}$ must be a set of disjoint events.[4]

Definition 5. Agents i and j are said to be *like-minded* if for any $e \in \Gamma_i$ and any $e' \in \Gamma_j$, if $e = e'$ then $D_i(e) =$

[4]This contrasts with [7] who, in their solution, propose adopting a version of the Sure-Thing Principle that does not require the disjointness of events.

$D_j(e')$.[5]

THEOREM 2. *Let $\mathcal{S}' = (\Omega', N, \{R'_i\}_{i\in N})$ be the counterfactual structure of a partitional structure $\mathcal{S} = (\Omega, N, \{R_i\}_{i\in N})$. Then, within $\mathcal{S}'$, if the agents $i \in N$ are like-minded (as defined in Definition 5) and follow the decision functions $\{D_i\}_{i\in N}$ (as defined in (8)) that satisfy the Sure-Thing Principle (as defined in Definition 4), then for any $G \subseteq N$, if $C_G(\cap_{i\in G}\{\omega' \in \Omega' | \delta_i(\omega') = x_i\}) \neq \emptyset$ then $x_i = x_j$ for all $i, j \in G$.*

Although this agreement theorem might appear to have many similarities with the previous one, it is conceptually entirely distinct. In particular, we show below (in section 4.3) that we were able to obtain the result while avoiding the conceptual flaws that were discussed in section 3.2. We also provide an interpretation of Theorem 2 and of counterfactual structures of partitional structures more generally in section 4.4.

4.3 Solution to the conceptual flaws

As discussed in section 3.2, Bacharach's framework requires the decision functions to be defined over events that are not possible beliefs for the agents. The proposition below shows that this is not the case in our set-up.

PROPOSITION 6. *Suppose that $\mathcal{S}' = (\Omega', N, \{R'_i\}_{i\in N})$ is the counterfactual structure of a partitional structure $\mathcal{S} = (\Omega, N, \{R_i\}_{i\in N})$. Then for any $e \in \Gamma_i$, there exists an $\omega \in \Omega'$ such that $b_i(\omega) = e$. (In fact, there exists a state $\lambda^e_{i,\omega} \in \Lambda$ for some $\omega \in e$ such that $b_i(\lambda^e_{i,\omega}) = e$).*

This proposition, in conjunction with Proposition 5, shows that in our set-up, the domain of the decision function of every agent is exactly the set of all possible beliefs for that agent. Indeed, our decision functions are defined over unions of partition elements, but these are possible beliefs for the agents because for every such union, there exists a counterfactual state at which the possibility set is precisely that union. We therefore avoid the first point in the conceptual flaws raised by [7]. Regarding the second point, the decision function D_i of agent i is now only defined over events in Γ_i. There is therefore no requirement for the function to determine the agent's action in the case where the event corresponds to a partition element of another agent.

4.4 Interpretation

In this section, we provide an interpretation of our assumptions, showing that the formal definitions of the Sure-Thing Principle and of like-mindedness given in our set-up match well with intuition. We also provide an interpretation of the agreement theorem in counterfactual structures, and of those structures more generally.

Our notion of like-mindedness is straightforward: Over the same information, like-minded agents take the same action. However, our definition has an advantage over Bacharach's which is that an agent i is not required to consider what action to take over the partition elements of another agent j.

[5]In contrast with the previous definition, we do not say that agents are like-minded if they have the "same" decision functions since the domains of the decision functions will now typically be different for different agents.

With regards to the Sure-Thing Principle, the proposition below, in particular part (ii), allows us to interpret our version of the principle as capturing the intuition that: "If the agent takes the same action in every case when she is more informed, she *would* take the same action *if she were* (*secretly*) more ignorant".

PROPOSITION 7. *Suppose that $\mathcal{S}' = (\Omega', N, \{R'_i\}_{i\in N})$ is the counterfactual structure of a partitional structure $\mathcal{S} = (\Omega, N, \{R_i\}_{i\in N})$. Then, (i) for any $e \subseteq \Omega'$, and $\omega, \omega' \in \Omega'$, $b_i(\omega) \subseteq e$ and $b_i(\omega') \subseteq e$ if and only if $b_i(\lambda^{b_i(\omega)\cup b_i(\omega')}_{i,\omega''}) \subseteq e$ (for some $\omega'' \in \Omega$). (ii) For any $e \subseteq \Omega'$, and $\omega, \omega' \in \Omega$, $b_i(\omega) \subseteq e$ and $b_i(\omega') \subseteq e$ if and only if $b_i(\lambda^{b_i(\omega)\cup b_i(\omega')}_{i,\omega}) \subseteq e$.*

Indeed, suppose $\mathcal{S}' = (\Omega', N, \{R'_i\}_{i\in N})$ is the counterfactual structure of a partitional structure $\mathcal{S} = (\Omega, N, \{R_i\}_{i\in N})$. Now consider an agent i, and two partition elements $I_i(\omega)$, $I_i(\omega') \in \mathcal{I}_i$ (where $\omega, \omega' \in \Omega$), and suppose that her decision function is such that $D_i(I_i(\omega)) = D_i(I_i(\omega')) = x$. The Sure-Thing Principle requires that $D_i(I_i(\omega) \cup I_i(\omega')) = x$. Propositions 6 shows that the possibility set that corresponds to $I_i(\omega) \cup I_i(\omega')$ is $b_i(\lambda^{I_i(\omega)\cup I_i(\omega')}_{i,\omega})$. Proposition 7 part (ii) shows that for any event e, i believes e at the counterfactual state $\lambda^{I_i(\omega)\cup I_i(\omega')}_{i,\omega}$ if and only if i also believes e at the states within *each* of those partition elements. Informally, if we can call a belief in an event "information", then the information that i has at the counterfactual state preserves only the information that is the same across both the partition elements. In this sense, the information that i has at the counterfactual state is the information that i would have if she were *just* more ignorant than at a state in either of the partition elements.[6] Furthermore, by construction of counterfactual structures, there is no state $\omega''' \in \Omega'$ and no $j \in N$ such that $(\omega''', \lambda^{I_i(\omega)\cup I_i(\omega')}_{i,\omega}) \in R'_j$; and, for any $j \neq i$, $(\lambda^{I_i(\omega)\cup I_i(\omega')}_{i,\omega}, \omega''') \in R'_j$ for every $\omega''' \in I_j(\omega)$ *only*. In words, this means that at this counterfactual state, i may have become "more ignorant", but the information of all other agents is unchanged. The information at this state therefore truly captures the fact that i is imagining herself *secretly* to be more ignorant. The situation is counterfactual since all other agents still believe that i has the information that she does in the partition $\mathcal{I}_i$.

We believe that this interpretation of the Sure-Thing Principle matches well with intuition. In particular, given that the principle finds its origins in single-agent decision theory (see [9]), it makes sense that the requirement on the decisions in cases where the agents are more ignorant is imposed only when ignorance is secret - in the sense that the information of all other agents is unchanged.

More generally, our interpretation of the counterfactual structure $\mathcal{S}'$ of a partitional structure $\mathcal{S}$ is therefore that it is simply a more complete picture of the situation that is being modeled by the structure $\mathcal{S}$ since it also includes states in which the agents imagine themselves (secretly, and counterfactually) to be more ignorant. The inclusion of these states turns out to be relevant in deriving appropriate formal definitions of the Sure-Thing Principle and of like-mindedness, and in resolving the conceptual flaws regarding

[6]In fact, it corresponds to being *just* "less informed", in a sense similar to that given in [8].

the domain of the decision functions. Indeed, we can think of the substructure $\mathcal{S}$ of $\mathcal{S}'$ as representing the "actual" situation, and the counterfactual states Λ are essentially "fake" in the sense that they do not actually occur. However, they are connected to the "actual" states in Ω in a manner that captures every possible way in which every agent could be secretly more ignorant relative to the "actual" situation; and although the "fake" states do not occur, the decision functions are essentially defined at such states (More precisely, they are defined over possibility sets that are defined as such states).[7] This turns out to be crucial: Theorem 2 is derived by showing that when the actions of agents are commonly known, the Sure-Thing Principle and like-mindedness imply that the actions must be the same precisely in the case when the decision functions are based on the information at some counterfactual (or "fake") states. The equality at the counterfactual states then carries over to the decisions over the information in the "actual" situation, and therefore agents cannot agree to disagree.

5. RELATION TO THE LITERATURE

We now discuss our approach in relation to other solutions that were proposed regarding the conceptual flaws. We then also compare our construction of the counterfactual states to other models that carry out a related exercise.

5.1 Other solutions

[7] propose a solution to the conceptual flaws that they found in the result of [3]. Essentially, they define a "relevance projection", which maps from sets of states to the "relevant information" at that set of states (see [7, p. 158]). They then impose conditions on this projection and on the decision functions to derive a new agreement theorem. However, it is not always obvious how a projection satisfying their conditions ought to be found. In contrast, the approach presented here offers a *constructive method* of obtaining a structure in which the analysis can be carried out.[8] Furthermore, our Sure-Thing Principle does require the disjointness of events, which their version does not.

[2] also propose a solution using a purely syntactic approach. The approach presented here is completely set-theoretic. Furthermore, they impose the condition that higher-order information must be irrelevant to the agents' decision, which we do not impose here.

Finally, [8] presented a very interesting solution to the conceptual flaws by redefining the Sure-Thing Principle entirely. Roughly, Samet's "Interpersonal Sure-Thing Principle" states that if agent i knows that agent j is more informed than he is, and knows that j's action is x, then i takes action x. Combining this with the assumption of the existence of an "epistemic dummy" - an agent who is less informed than every other agent - [8] proves a new agreement theorem in paritional structures. The large differences in the assumptions make a formal comparison between the approach here and in [8] difficult.

[7]Notice that this shows that our counterfactual structures are particular "impossible-world" structures (e.g. see [12]). We return to this point in section 5.

[8]Also, the resulting counterfactual structure does satisfy properties that resemble, in spirit, the conditions imposed on the relevance projection.

5.2 Action models

Loosely speaking, it was shown that the information at the counterfactual states in a counterfactual structure corresponds to secretly "losing" information. It turns out that secretly "gaining" information is well-studied in the dynamic epistemic logic literature (e.g. [4]). *Action models* formalize how the underlying structure (both the state space and the reachability relations) must be modified to model various protocols by which agents may gain some new information.

It was shown, [11, Theorem 17], that in the case of secretly gaining new information, a partitional structure would have to be transformed into a belief structure. In this paper, we have defined a method of modeling secret loss of information by transforming a partitional structure into a (counterfactual) structure that belongs to the **KD4** class. In particular, this means that "negative introspection" is dropped as a property of the belief operator. We have not shown that it is necessary to drop negative introspection in order to model secret loss of information, so in principle, it remains an open question as to whether it is possible to define a purely semantic transformation of a partitional structure (i.e. only involving the states and the reachability relations) that can model secret loss of information where the resulting structure is a belief structure in which the primitives of the original model (i.e. the original state space and partitions over them) are unchanged.[9]

5.3 Counterfactuals

General set-theoretic information structures have been proposed to model counterfactuals (e.g. see [5]), especially in relation to the literature on backwards induction. In extensive form games, to implement the backwards induction solution, agents must consider what they would do at histories of the game that might never be reached. They must therefore be able to define what they would do in situations that never occur. This therefore bears some resemblance to our set-up in which agents are required to have decisions that are defined over information at counterfactual (or "fake") states that never actually occur, but there are important differences which we briefly outline below.

There is a multitude of ways in which counterfactuals can be modeled, and we cannot hope to survey the literature here. However, it will suffice to say that a general approach to modeling counterfactuals proceeds in roughly the following manner: One defines a "closeness" relation on states and then says that a state ω belongs to the event "If f were the case, then e would be true" if e is true in all the closest states to ω where f is true. It is possible to then augment this approach with epistemic operators and decisions, but the salient point is simply that the standard approach to counterfactuals aims to be quite general, in capturing all possible hypothetical situations f.

In contrast, we only model counterfactuals for a very particular set of hypothetical situations, namely, every possible

[9][10] analyzes counterfactuals in **KD45** structures. However, his *initial* structures are **KD45**, whereas the point made here is regarding a method that would transform a *partitional* structure into a **KD45** structure while building the relevant counterfactual states and leaving the primitives of the original model unchanged.

situation (relative to the "actual" situation) in which every agent considers herself to be secretly more ignorant. This is not done by imposing a closeness relation, but by creating a new set of "fake" counterfactual states and carefully re-wiring them to the "actual" states. (Note however, that the resulting information at the counterfactual states was shown to be interpretable as being secretly "just" more ignorant than in the "actual" situation being considered, so in this sense, the counterfactual state can be seen as being "close" to the actual situation). As a result, it is not obvious to see how the method developed here can be applied to studying backwards induction, which requires considering a richer set of hypothetical situations, but the method is well-adapted for the analysis of agreement theorems carried out in this paper.

Note that there is another approach to modeling counterfactuals that is related to ours. What is known as the "impossible-worlds" approach (e.g. [12]) augments information structures with a new set of states and with modified reachability relations. The set of states in the original structure are then referred to as "possible", or "normal", worlds, while the ones in the new set are referred to as "impossible", or "non-normal". In our framework, these actually correspond to our "actual" states Ω, and our "fake" states Λ (and the reachability relations are modified from R_i to R_i' for every i). The counterfactual structures presented here can therefore be seen as specific "impossible-worlds" structures. However, we are not aware of any paper that use impossible-worlds structures as a tool for modeling counterfactuals in the manner presented here.

6. CONCLUSION

We provided a *constructive method* for creating an information structure that includes the relevant counterfactual states (starting from a partitional structure). This new *counterfactual structure* is interpreted as providing a more complete picture of the situation that is being modeled by the original partitional structure. As such, our analysis of the agreement theorem is carried out in such structures.

Having provided new formal definitions for the Sure-Thing Principle and for like-mindedness, we prove an agreement theorem within such structures, and show that we can interpret our version of the Sure-Thing principle as capturing the intuition that: "If the agent takes the same action in every case when she is more informed, she *would* take the same action *if she were* (*secretly*) more ignorant". We also show that our version of like-mindedness has more desirable properties than Bacharach's. Furthermore, we show that our approach resolves the conceptual issues raised by [7], in the sense that within counterfactual structures, decision functions are defined only over events that are possible beliefs for the agents.

Therefore, in providing a constructive method for creating counterfactual structures, our approach achieves the goal of maintaining an interpretation of the underlying assumptions of the agreement theorem that fits well with intuition, while simultaneously resolving the conceptual issues (identified in [7]) regarding the domain of the decision functions.

7. ACKNOWLEDGMENTS

I would like to thank John Quah for his invaluable help, Francis Dennig and Alex Teytelboym for very useful discussions, as well as Harvey Lederman and three anonymous referees for their comments. I would also like to thank Dov Samet for conversations on earlier versions of the ideas presented here, and Burkhard Schipper for his helpful comments.

8. REFERENCES

[1] R. Aumann. Agreeing to disagree. *The annals of statistics*, 4(6):1236–1239, 1976.

[2] R. Aumann and S. Hart. Agreeing on decisions. Technical report, mimeo, 2006.

[3] M. Bacharach. Some extensions of a claim of aumann in an axiomatic model of knowledge. *Journal of Economic Theory*, 37(1):167–190, 1985.

[4] A. Baltag and L. Moss. Logics for epistemic programs. *Information, Interaction and Agency*, pages 1–60, 2005.

[5] J. Halpern. Hypothetical knowledge and counterfactual reasoning. *International Journal of Game Theory*, 28(3):315–330, 1999.

[6] Z. Hellman. Deludedly agreeing to agree. Technical report, 2012.

[7] Y. Moses and G. Nachum. Agreeing to disagree after all. In *Proceedings of the 3rd conference on Theoretical aspects of reasoning about knowledge*, pages 151–168. Morgan Kaufmann Publishers Inc., 1990.

[8] D. Samet. Agreeing to disagree: The non-probabilistic case. *Games and Economic Behavior*, 69(1):169–174, 2010.

[9] L. Savage. *The foundations of statistics*. Dover Pubns, 1972.

[10] R. Stalnaker. Knowledge, belief and counterfactual reasoning in games. *Economics and philosophy*, 12:133–164, 1996.

[11] J. Van Eijck. Advances in dynamic epistemic logic. 2008.

[12] H. Wansing. A general possible worlds framework for reasoning about knowledge and belief. *Studia Logica*, 49(4):523–539, 1990.

Poster Presentations

Model checking an Epistemic μ-calculus with Synchronous and Perfect Recall Semantics*

Rodica Bozianu
École Normale Supérieure de Cachan,
61, avenue du Pdt. Wilson,
94235 Cachan cedex

Cătălin Dima
LACL,
Université Paris Est-Créteil,
61 av. du G-ral de Gaulle,
94010 Créteil, France

Constantin Enea
LIAFA, CNRS UMR 7089,
Université Paris Diderot - Paris 7,
Case 7014, 75205 Paris Cedex 13,
France

ABSTRACT

We identify a subproblem of the model-checking problem for the epistemic μ-calculus which is decidable. Formulas in the instances of this subproblem allow free variables within the scope of epistemic modalities in a restricted form that avoids embodying any form of common knowledge. Our subproblem subsumes known decidable fragments of epistemic CTL/LTL, may express winning strategies in two-player games with one player having imperfect information and non-observable objectives, and, with a suitable encoding, decidable instances of the model-checking problem for ATL_{iR}.

1. INTRODUCTION

The epistemic μ-calculus is an enrichment of the μ-calculus on trees with individual epistemic modalities K_a (and its dual, denoted P_a). It is designed with the aim that, like the classical modal μ-calculus, it would subsume most combinations of temporal and epistemic logics. The epistemic μ-calculus is more expressive than linear or branching temporal epistemic logics [15, 24], propositional dynamic epistemic logics [25], or the alternating epistemic μ-calculus [6]. On the other hand, some gaps in its expressive power seem to exist, as witnessed by recent observations in [6] showing that formulas like $\langle\!\langle a\rangle\!\rangle p_1 \mathcal{U} p_2$ are not expressible in the *Alternating Epistemic μ-calculus*. This expressivity gap can be reproduced in the epistemic μ-calculus, though the epistemic μ-calculus is richer than the alternating μ-calculus.

The model-checking problem for epistemic μ-calculus is undecidable in the presence of a semantics with perfect recall, as it is more expressive than combinations of temporal epistemic logics that include the common knowledge operator. A rather straightforward fragment of the epistemic μ-calculus which has a decidable model-checking problem is the one in which knowledge modalities apply only to closed formulas, that is, formulas in which all second-order variables are bound by some fixpoint operator. The decidability of this fragment follows from recent results on the decidability of the emptiness problem for two player games [7].

However more expressive fragments having a decidable model-checking problem seem to exist. For example, winning strategies in two-player games in which one player has imperfect information and non-observable winning conditions can be encoded as fixpoint formulas in the epistemic μ-calculus, but not in the above-mentioned restricted fragment. The same holds for some formulas in ATL with imperfect information and perfect recall (ATL_{iR}) [23, 5]: the ATL formula $\langle\!\langle a\rangle\!\rangle \Box p$ can be expressed in a modal μ-calculus of knowledge as

$$\nu Z. \bigvee_{\alpha \in Act_a} K_a\big(p \wedge \bigwedge_{\beta \in Act_{Ag\setminus\{a\}}} [\alpha, \beta] Z\big)$$

And there are variants of ATL_{iR} for which the model-checking problem is decidable [10]. Note that a translation of each instance of the model-checking problem for ATL into instances of the model-checking problems for the epistemic μ-calculus is also possible but requires the modification of the models.

Our aim in this paper is to identify a larger and decidable class of instances of the model-checking problem for the epistemic μ-calculus. The fragment we propose here allows an epistemic modality K_a to be applied to a non-closed μ-calculus formula ϕ, but in such a way that avoids expressing properties that construct any variant of common knowledge for two or more agents. Roughly, the technical restriction is the following: two epistemic operators, referring to the knowledge of two different agents a and b, can be applied to non-closed parts of a formula only if the two agents have *compatible* observations in the system M in the sense that the observability relation of one of the agents is a refinement of the observability relation of the other. Similar restrictions have been proposed for various combinations of temporal epistemic logics [12], or for the synthesis problem in distributed environments [18, 27, 13]. The variant presented here relies on a *concrete* semantics, in the sense of [9], with the observability relation for each agent a being identified, in the given system M, by a subset Π_a of atomic propositions. We require this in order to syntactically define our decidable subproblem: the compatibility of two observability relations $\sim_a$ and $\sim_b$ is specified by imposing that either $\Pi_a \subseteq \Pi_b$ or vice-versa.

The epistemic μ-calculus with perfect recall has a history-based semantics: for each finite transition system T, the formulas of the epistemic μ-calculus must be interpreted over the *tree unfolding* of T. This makes it closer with the tree interpretations of the μ-calculus from [11]. For the classical μ-calculus, there are two ways of proving that the satisfiability and the model-checking problem for the tree interpretation of the logic are decidable: either by providing translations to parity games, or by means of a Finite Model Theorem which ensures that a formula has a tree interpretation iff it has a *state-based* interpretation over a finite transition system (this is known to be equivalent with memoryless determinacy for parity games, see e.g. [4]).

The generalization of the automata approach does not seem to be possible for epistemic μ-calculus, mainly due to the absence of an appropriate generalization of tree automata equivalent with the epistemic μ-calculus. So we take the approach of providing a generalization of the Finite Model Theorem for our fragment of

*Work partially supported by the ANR research project "EQINOCS" no. ANR-11-BS02-0004

TARK 2013, Chennai, India.

the epistemic μ-calculus. This result says roughly that the tree interpretation of a formula over the tree unfolding of a given finite transition system T which contains the epistemic operators K_a or P_a is exactly the "tree unfolding" of the finitary interpretation of the formula in a second transition system T', which is obtained by determinizing the projection of T onto the observations of agent a, a construction that is common for decidable fragments of temporal epistemic logics. Our contribution consists in showing that this construction can be applied for all instances in our model-checking subproblem. The proof is given in terms of commutative diagramms between boolean algebraic operators that are the interpretations of non-closed formulas.

The model checking subproblem is non-elementary hard due to the non-elementary hardness of the model-checking problem for the linear temporal logic of knowledge [26]. In the full version of this paper [3], we provide a self-contained proof of this result, by a reduction of the emptiness problem for star-free regular expressions.

The rest of the paper is divided as follows: in the next section we recall the semantics of the μ-calculus and adapt it to our epistemic extension, both for the tree interpretation and the finitary interpretation. We then give, in the third section, our weak variant of the Finite Model Theorem for the classical μ-calculus. The fourth section serves for introducing our fragment of the epistemic μ-calculus and for proving the decidability of its model-checking problem. We end with a section with conclusions and comments.

An extended version of this paper with proofs is available as [3].

2. PRELIMINARIES

We start by fixing a series of notions and notations used in the rest of the paper.

A^* denotes the set of words over A. The length of $\alpha \in A^*$, is denoted $|\alpha|$ and the prefix of α up to position i is denoted $\alpha[1..i]$. Hence, $\alpha[1..0] = \varepsilon$ is the empty word. The prefix ordering on A^* is denoted $\preceq$ ($\prec$ for the strict prefix ordering).

Given a set A and an integer $n \in \mathbb{N}$, an *A-tree of outdegree $\leq n$* is a partial function $t : [1 \dots n]^* \rightharpoonup A$ whose support, denoted $\mathsf{supp}(t)$, is a prefix-closed subset of the finite sequences of integers in $[1 \dots n]$. A *node* of t is an element of its support. A *path* in t is a pair (x, ρ) consisting of a node x and the sequence of t-labels of all the nodes which are prefixes of x, $\rho = \big(t(x[1 \dots i])\big)_{0 \leq i \leq |x|}$.

Boolean operators: Given a set A, a *boolean A-operator* is a mapping $f : \big(2^A\big)^n \to 2^A$.

For an A-operator $f : \big(2^A\big)^n \to 2^A$, a tuple of sets $B_1, \dots, B_n \subseteq A$ and some $k \leq n$ we denote $f_k(B_1, \dots, B_{k-1}, \bullet, B_{k+1}, \dots, B_n) : 2^A \to 2^A$ the A-operator with

$$f_k(B_1, \dots, B_{k-1}, \bullet, B_{k+1}, \dots, B_n)(B) = f(B_1, \dots, B_{k-1}, B, B_{k+1}, \dots, B_n)$$

Note that when f is monotone, $f_k(B_1, \dots, B_{k-1}, \bullet, B_{k+1}, \dots, B_n)$ is monotone too.

Following the Knaster-Tarski theorem, any monotone A-operator $f : 2^A \to 2^A$ has a unique least and greatest fixpoint, denoted lfp_f, resp. gfp_f. We may then define two A-operators, $\mathsf{lfp}_f^k : (2^A)^n \to 2^A$ and $\mathsf{gfp}_f^k : (2^A)^n \to 2^A$, respectively as:

$$\mathsf{lfp}_f^k(B_1, \dots, B_n) = \mathsf{lfp}_{f_k(B_1, \dots, B_{k-1}, \bullet, B_{k+1}, \dots, B_n)}$$
$$\mathsf{gfp}_f^k(B_1, \dots, B_n) = \mathsf{gfp}_{f_k(B_1, \dots, B_{k-1}, \bullet, B_{k+1}, \dots, B_n)}$$

Note that both these A-operators are constant in their k-th argument. It is well-known that both operators are monotone if f is monotone.

3. THE μ-CALCULUS OF KNOWLEDGE

Syntax: The syntax of the **epistemic μ-calculus** is based on the following sets of symbols: a finite set of *agents* Ag, a family of sets of *atomic propositions* $(\Pi_a)_{a \in Ag}$ for which we denote $\Pi = \bigcup_{a \in Ag} \Pi_a$ and a set of *fixpoint variables* $\mathcal{Z} = \{Z_1, Z_2, \dots\}$.

The grammar for the formulas of the epistemic μ-calculus is:

$$\varphi ::= p \mid \varphi \wedge \varphi \mid \neg\varphi \mid AX\varphi \mid K_a\phi \mid \mu Z.\varphi$$

where $p \in \Pi$, $a \in Ag$ and $Z \in \mathcal{Z}$, and with the usual restriction that an operator μZ may be applied on formulas in which the variable Z has only positive occurrences.

Formulas of the type $K_a\phi$ are read as *agent a knows that ϕ holds*. μZ is the *least fixpoint* operator, while AX is the usual *nexttime* operator from CTL, universally quantified over the successors of the current state.

Several derived operators can be defined as usual:

1. The dual of AX is denoted EX and defined as $EX\phi \equiv \neg AX\neg\phi$.
2. The dual of K_a is denoted P_a and defined as $P_a\phi \equiv \neg K_a\neg\phi$. $P_a\phi$ reads as *agent a considers that ϕ is possible*.
3. The greatest fixpoint operator is denoted νZ and defined as $\nu Z.\phi \equiv \neg\mu Z.\neg\phi[Z/\neg Z]$, where $\phi[Z/\neg Z]$ is the result of the syntactic substitution of each occurrence of Z with $\neg Z$ in ϕ.

As usual, for a subset of agents $A \subseteq Ag$ we may denote E_A the "everybody knows" operator, $E_A\phi = \bigwedge_{a \in A} K_a\phi$.

Since our model checking construction relies heavily on formulas being interpreted as monotone mappings and, on the other side, set complementation (which is the usual interpretation of negation) is not a monotone operator we will prefer the following syntax *in positive form* for the epistemic μ-calculus:

$$\varphi ::= p \mid \neg p \mid Z \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid AX\varphi \mid EX\varphi \mid K_a\phi \mid P_a\phi \mid \mu Z.\varphi \mid \nu Z.\varphi$$

It is easy to see that each formula of the epistemic μ-calculus can be transformed into a formula in positive form, by pushing negations through the operators and using the definitions of the dual operators.

The fragment of the epistemic μ-calculus which does not involve the knowledge operator K_a (or its dual P_a) is called here the *plain μ-calculus*, or simply the μ-calculus, when there's no risk of confusion. As usual, we say that a formula ϕ is *closed* if each variable Z in ϕ occurs in the scope of a fixpoint operator for Z.

We will also briefly consider in this paper the *modal epistemic μ-calculus*, for the sake of comparison with other combinations of temporal and epistemic logics. The language of this variant of the epistemic μ-calculus is based on a family of sets $(Act_a)_{a \in Ag}$, meant to represent actions available to each agent at a given state. Its grammar is the following:

$$\varphi ::= p \mid \varphi \wedge \varphi \mid \neg\varphi \mid \langle\alpha\rangle\varphi \mid K_a\phi \mid \mu Z.\varphi$$

where $p \in \Pi$, $a \in Ag$, $\alpha \in \times_{a \in Ag} Act_a$ and $Z \in \mathcal{Z}$, and bearing the same restriction on the utilization of the fixpoint operators. Formulas of the type $\langle\alpha\rangle\varphi$ read as *there exists an α-successor of the current state in which φ holds*. The dual of the $\langle\alpha\rangle$ operator is denoted $[\alpha]$.

3.1 Semantics

The tree semantics of the epistemic μ-calculus is given in terms of $2^{\Pi \cup \mathcal{Z}}$-trees endowed with a family of relations $(\sim_a)_{a \in Ag}$ with

$\sim_a \subseteq \mathsf{supp}(t) \times \mathsf{supp}(t)$. The nodes of the tree represent instant descriptions of the system state, while the relation $\sim_a$ models the *indistinguishability* relation which disallows agent a to tell apart two behaviors of the system.

Formaly, given a tree t and the family of relations $(\sim_a)_{a\in Ag}$, each formula ϕ which contains variables $Z_1, \ldots, Z_n$ is associated with a $\mathsf{supp}(t)$-operator $\|\phi\| : \left(2^{\mathsf{supp}(t)}\right)^n \to 2^{\mathsf{supp}(t)}$ defined by structural induction, as follows:

- The atom p is interpreted as the constant $\mathsf{supp}(t)$-operator $\|p\| : \left(2^{\mathsf{supp}(t)}\right)^n \to 2^{\mathsf{supp}(t)}$ defined as follows:

$$\|p\|(S_1, \ldots, S_n) = \{x \in \mathsf{supp}(t) \mid p \in \pi(t(x))\}$$

- The semantics of the boolean operators is classical:

$$\|\neg\phi\| = \mathsf{supp}(t) \setminus \|\phi\|$$
$$\|\phi_1 \wedge \phi_2\| = \|\phi_1\| \cap \|\phi_2\|$$

- Each variable $Z_i \in \mathcal{Z}$ is interpreted as the i-th projection on $\left(2^{\mathsf{supp}(t)}\right)^n$, that is, as the operator $\|Z_i\| : \left(2^{\mathsf{supp}(t)}\right)^n \to 2^{\mathsf{supp}(t)}$ with

$$\|Z_i\|(S_1, \ldots, S_n) = S_i, \forall S_1, \ldots, S_n \subseteq \mathsf{supp}(t)$$

- The nexttime operator AX is mapped to a $\mathsf{supp}(t)$-operator, denoted $AX : 2^{\mathsf{supp}(t)} \to 2^{\mathsf{supp}(t)}$, such that for each $S \subseteq \mathsf{supp}(t)$,

  $AX(S) = \{x \in \mathsf{supp}(t) \mid \forall i \in \mathbb{N} \text{ if } xi \in \mathsf{supp}(t) \text{ then } xi \in S\}$

  Then the semantics of formulas of the type $AX\phi$ is defined as:

$$\|AX\phi\| = AX \circ \|\phi\|$$

- Each epistemic operator K_a is mapped to a $\mathsf{supp}(t)$-operator denoted $K_a : 2^{\mathsf{supp}(t)} \to 2^{\mathsf{supp}(t)}$, such that for each $S \subseteq \mathsf{supp}(t)$,

  $K_a(S) = \{x \in \mathsf{supp}(t) \mid \forall y \in \mathsf{supp}(t), \text{ if } x \sim_a y \text{ then } y \in S\}$

  Then the semantics of formulas of the type $K_a\phi$ is defined as:

$$\|K_a\phi\| = K_a \circ \|\phi\|$$

- The fixpoint operators are interpreted as usual:

$$\|\mu Z_i.\phi\| = \mathsf{lfp}^i_{\|\phi\|}$$

We denote $t \vDash \phi$ iff $\varepsilon \in \|\phi\|$.

The semantics of the epistemic μ-calculus can be also described without set complementation, by keeping the definition of negation only for atomic formulas, and appending the following definitions:

$$\|\neg p\|(S_1, \ldots, S_n) = \{x \in \mathsf{supp}(t) \mid p \notin \pi(t(x))\}$$
$$EX(S) = \{x \in \mathsf{supp}(t) \mid \exists i \in \mathbb{N} \text{ with } xi \in \mathsf{supp}(t) \text{ and } xi \in S\}$$
$$\|EX\phi\| = EX \circ \|\phi\|$$
$$P_a(S) = \{x \in \mathsf{supp}(t) \mid \exists y \in \mathsf{supp}(t) \text{ with } x \sim_a y \text{ and } y \in S\}$$
$$\|P_a\phi\| = P_a \circ \|\phi\|$$
$$\|\nu Z_i.\phi\| = \mathsf{gfp}^i_{\|\phi\|}$$

Note that, this way, all operators are interpreted as monotone $\mathsf{supp}(t)$-operators, which is more convenient for manipulating fixpoints.

As we are interested in the model-checking problem, we will only work with finitely-generated trees as models for the epistemic μ-calculus. These finitely-generated models occur as unfoldings of *multi-agent systems*, whose definition is recalled here.

A **multi-agent system** is a tuple $M = \left(Q, Ag, \delta, q_0, \Pi, (\Pi_a)_{a\in Ag}, \pi\right)$ with Ag being the set of agents, Q the set of states, q_0 the initial state of the system, $\delta \subseteq Q \times Q$, Π the set of *atomic propositions*, $\pi : Q \to 2^\Pi$ is the *state labeling* and for all $a \in Ag$, $\Pi_a \subseteq \Pi$ is the set of atoms *observable by* agent a. A run in the system M is an infinite sequence of states $\rho = q_0q_1q_2\ldots$ such that $(q_i, q_{i+1}) \in \delta$ for all $i \geq 0$. The set of finite runs in M is denoted $\mathsf{Runs}(M)$. Throughout this paper we consider only finite systems, with $Q = \{1, \ldots, n\}$ and $q_0 = 1$, and we assume that Q contains only reachable states.

The Q-tree representing the *unfolding* of a multi-agent system M is denoted t_M and defined by

$$\mathsf{supp}(t_M) = \{x \in \mathbb{N}^* \mid 1x \in \mathsf{Runs}(M)\} \text{ and } t_M(x) = x[|x|]$$

The actual tree that can be used as a model of the epistemic μ-calculus is $\pi(t_M) = \pi \circ t_M : \mathsf{supp}(t_M) \rightharpoonup 2^\Pi$. We denote this tree as πt_M.

The family of indistinguishability relations $(\sim_a)_{a\in Ag}$ that we consider in this paper are defined as follows: for any two positions $x, y \in \mathsf{supp}(t_M)$ with $|x| = |y|$, we denote $x \sim_a y$ if for any $n \leq |x|$ we have that

$$\pi(t(x[1..n])) \cap \Pi_a = \pi(t(y[1..n])) \cap \Pi_a$$

This way, the indistinguishability relation $\sim_a$ models the fact that agent a has perfect knowledge of the absolute time and remembers all his past observations – that is, $\sim_a$ is a *synchronous and perfect recall* indistinguishability.

Definition 1. The **model-checking problem** for the epistemic μ-calculus is the problem of deciding, given a multi-agent system M and a closed formula ϕ, whether $\pi t_M \vDash \phi$.

The undecidability of the model-checking problem for combinations of temporal and epistemic logics based on a synchronous and perfect recall semantics and containing the common knowledge operator [26, 25], together with the connections between the epistemic μ-calculus and such temporal epistemic logics that are explored in the next section, imply the following result:

THEOREM 1. *The model-checking problem for the epistemic μ-calculus is undecidable.*

The semantics of the **modal epistemic μ-calculus** is a slight variation of the above semantics, in that we utilize a different type of trees, as mappings $t : \mathbb{N} \rightharpoonup 2^{\Pi\cup\mathcal{Z}} \times \bigtimes_{a\in Ag} Act_a$. We decompose such a tree as $t = (t^{node}, t^{edge})$: the tree of *nodes* is $t^{node}(x) = t(x)\Big|_{2^{\Pi\cup\mathcal{Z}}}$, while the tree of *edges* is $t^{edge}(x)t(x)\Big|_{\bigtimes_{a\in Ag} Act_a}$. The only item that changes in the above list of semantic rules for operators is that we replace the definition of the nexttime operator with the following definition of the a boolean operator $\langle\alpha\rangle : 2^{\mathsf{supp}(t)} \to 2^{\mathsf{supp}(t)}$: for each $S \subseteq \mathsf{supp}(t)$,

$$\langle\alpha\rangle(S) = \{x \in \mathsf{supp}(t) \mid \exists i \in \mathbb{N} \text{ with } xi \in \mathsf{supp}(t) \text{ and } xi \in S\}$$

A family of indistinguishability relations in such a tree model for the modal epistemic μ-calculus is, like in the non-modal case, a family of relations $(\sim_a)_{a\in Ag}$ with $\sim_a \subseteq \mathsf{supp}(t) \times \mathsf{supp}(t)$.

Then, finite presentations of tree models for the modal epistemic μ-calculus are *multi-agent systems with transition labels*, which are tuples $M = \left(Q, Ag, (Act_a)_{a\in Ag}, \delta, q_0, \Pi, (\Pi_a)_{a\in Ag}, \pi\right)$ with $\delta \subseteq Q \times \bigtimes_{a\in Ag} Act_a \times Q$ and all the other components bearing the same name and definition as in (plain) multi-agent systems.

The tree representing the *unfolding* of M, denoted t_M also, is defined inductively as follows:

- $\varepsilon \in \mathsf{supp}(t_M)$ and $t^{node}(\varepsilon) = q_0$; $t^{edge}(\varepsilon)$ is left unconstrained.
- If $x \in \mathsf{supp}(t_M)$ and $t^{node}(x) = q$, then for each state r and tuple of actions $\alpha \in \times_{a\in Ag} Act_a$, for which $q \xrightarrow{\alpha} r \in \delta$, there exists a succesor of x denoted $xi_{r,\alpha}$, and $t(xi_{r,\alpha}) = \big(r, \alpha\big)$.
- All successors of a node x are obtained by the previous rule.

The family of indistinguishability relations is defined in a slightly different way for unfoldings of transition-labeled multi-agent systems, as agents may know their own past actions. Formally, for two nodes $x, y \in \mathsf{supp}(t_M)$ and an agent $a \in Ag$ we put $x \sim_a y$ if for any $n \leq |x|$ we have that

$$\pi(t^{node}(x[1..n])) \cap \Pi_a = \pi(t^{node}(y[1..n])) \cap \Pi_a$$
$$t^{edge}(x[1..n])\Big|_a = t^{edge}(y[1..n])\Big|_a$$

The modal epistemic μ-calculus can be translated into the (non-modal) epistemic μ-calculus by converting each action name $\alpha_a \in Act_a$ into an atomic proposition, so the main results of this paper generalize easily to this calculus.

3.2 Comparison with other temporal epistemic frameworks

In this subsection we discuss the relationship between the epistemic μ-calculus and other temporal epistemic logics or game models with imperfect information and perfect recall.

First, it is easy to see that the epistemic μ-calculus is more expressive than linear or branching temporal epistemic logics with common knowledge operators [15]. This was already noted e.g. in [24], since the following fixpoint formula defines the common knowledge operator for two agents: $C_{a,b}\phi = \nu Z.(\phi \wedge K_a Z \wedge K_b Z)$.

Secondly, the (modal variant of the) epistemic μ-calculus is more expressive than the alternating epistemic μ-calculus of [6], due to the possibility to insert knowledge operators "in between" the quantifiers that occur in the semantics of the coalition operators. More precisely, for any instance of the model-checking problem for the alternating epistemic μ-calculus, let Act_A, denote, for each set of agents $A \subseteq Ag$, the cartesian product of the set of action symbols for each agent in A, $Act_A = \times_{a\in A} Act_a$. Then:

$$\langle\!\langle A\rangle\!\rangle X\phi = \bigvee_{\alpha\in Act_A} \big(K_a \bigwedge_{\beta\in Act_{Ag\setminus A}} [\alpha,\beta]\phi\big)$$
$$[\![A]\!] X\phi = \bigwedge_{\alpha\in Act_A} \big(P_a \bigvee_{\beta\in Act_{Ag\setminus A}} [\alpha,\beta]\phi\big)$$

Recall briefly that the *strategy operator* $\langle\!\langle A\rangle\!\rangle\phi$ says that the agents in the group (coalition) A have a *strategy* ensuring that, whatever the other agents do, the objective ϕ is achieved on each resulting run. Also the strategy must be based on the observability of each agent of the system state. See [5] for a recent account on alternating temporal logics.

The relationship with ATL_{iR} is more involved, as we detail in the sequel. Formulas of the type $\langle\!\langle A\rangle\!\rangle \Box p$ can be expressed as the fixpoint formula $\nu Z. \bigvee_{\alpha\in Act_a} K_a\big(p \wedge \bigwedge_{\beta\in Act_{Ag\setminus\{a\}}} [\alpha,\beta]Z\big)$.

On the other hand, formulas containing the until operator cannot be translated into the epistemic μ-calculus. The reason is explained in [6]: in formulas of the type $\langle\!\langle a\rangle\!\rangle \Diamond p$ the objective p might not be observable by the agent a, who might only be able to know, in the future of some given time instant, that sometimes in the past of that future moment (but after the reference instant), the objective was achieved on all identically observable traces.

However, for the decidable case of coalitions based on distributed knowledge [10], a translation exists for each instance of the model-checking problem. We provide here this translation for simple reachability formulas: given an ATL_{iR} formula $\phi = \langle\!\langle a\rangle\!\rangle p_1 \mathcal{U} p_2$, a multi-agent system M and a finite run ρ in M, the instance of the model-checking problem $M, \rho \vDash \phi$ can be translated to an instance of the model-checking problem in the epistemic modal μ-calculus of the following formula:

$$\tilde{\phi} = \mu Z. \bigvee_{\alpha\in Act_a} K_a\Big(p_2 \vee past_{p_2} \vee \big(p_1 \wedge \bigwedge_{\beta\in Act_{Ag\setminus\{a\}}} [\alpha,\beta]Z\big)\Big) \quad (1)$$

and the *modified* system M', in which the new atomic proposition $past_{p_2}$ labels all the states occurring *after* a state carrying a p_2 and lying on runs which extend ρ. This mechanism is similar with the "bookkeeping" employed in the two-player games utilized in [10] for checking whether the same formula ϕ holds at a state of a multi-agent system.

Formally, given a multi-agent system $M = (Q, Ag, \delta, q_0, \Pi, (\Pi_a)_{a\in Ag}, \pi, (Act_a)_{a\in Ag})$, we build the system $M' = (Q', Ag, \delta', q_0', \Pi, (\Pi_a)_{a\in Ag}, \pi', (Act_a')_{a\in Ag})$ in which:

- $Q' = Q \times \{0,1\}$ and $q_0' = (q_0, 0)$.
- $\pi'(q,0) = \pi(q)$, $\pi'(q,1) = \pi(q) \cup \{past_{p_2}\}$.
- $Act'_{a_0} = Act_{a_0} \times \{0,1\}$ and $Act'_b = Act_b$ for all $b \neq a_0$.
- For any transition $q \xrightarrow{(\alpha,\beta)} r$ with $\alpha \in Act_{a_0}$ and $\beta = (\beta_b)_{b\neq a_0}$, we put in δ' the following transitions:
  - $(q,0) \xrightarrow{((\alpha,0),\beta)} (r,0)$
  - $(q,1) \xrightarrow{((\alpha,x),\beta)} (r,1)$, $x \in \{0,1\}$
  - $(q,0) \xrightarrow{((\alpha,1),\beta)} (r,1)$ if $p_2 \in \pi(q)$
  - $(q,0) \xrightarrow{((\alpha,1),\beta)} (r,0)$ if $p_2 \notin \pi(q)$

Note that, given a node $x \in \mathsf{supp}(t_{M'})$, if we replace, on the path from the root to x, all actions of the type $(\alpha,0)$ with α, we get a run in t_M corresponding with a note of t_M. We denote this corresponding node as $x\big|_M$. Furthermore, for each node $x \in \mathsf{supp}(t_M)$, we denote $x\uparrow^{M'}$ the node in $\mathsf{supp}(t_{M'})$ with $(x\uparrow^{M'})\big|_M = x$ and having the property that on the path from the root of $t_{M'}$ to $x\uparrow^{M'}$, a's actions are only of the type $(\alpha,1)$.

The following proposition gives the connection between the instances of the model-checking problem in M and M':

PROPOSITION 2. *For each node x in the tree t_M, $x \vDash \phi = \langle\!\langle a_0\rangle\!\rangle p_1 \mathcal{U} p_2$ if and only if $x\uparrow^{M'} \vDash \tilde{\phi}$, with $\tilde{\phi}$ defined as follows:*

$$\tilde{\phi} = \mu Z. \bigvee_{\alpha\in Act_{a_0}} K_{a_0}\Big(p_2 \vee past_{p_2} \vee \big(p_1 \wedge \bigwedge_{\beta\in Act_{Ag\setminus\{a_0\}}} [\alpha,\beta]Z\big)\Big)$$

The same property holds for $\phi = [\![a_0]\!] p_1 \mathcal{U} p_2$ (which reads "agent a_0 cannot avoid $p_1 \mathcal{U} p_2$") and

$$\tilde{\phi} = \mu Z. \bigwedge_{\alpha\in Act_{a_0}} P_{a_0}\Big(p_2 \vee past_{p_2} \vee \big(p_1 \wedge \bigvee_{\beta\in Act_{Ag\setminus\{a_0\}}} \langle\alpha,\beta\rangle Z\big)\Big)$$

The problem of *solving multi-player games with imperfect information* can also be translated into the epistemic μ-calculus. Recall that a (synchronous) two-player game is a tuple

$$G = \Big(Q, Act_0, Act_1, \delta, Q_0, Obs_0, Obs_1, o_0, o_1, par\Big)$$

with Q denoting the set of states, Act_0 (resp Act_1) denoting the set of actions available to player 0 (resp. player 1), $\delta \subseteq Q \times Act_0 \times Act_1 \times Q$ denoting the transition relation, Obs_0, resp. Obs_1 denoting finite sets of observations available to agent 0 (resp. agent 1), $o_0 : Q \to Obs_0$, resp. $o_1 : Q \to Obs_1$ denoting the observability relation for each player and $par : Q \to \mathbb{N}$ defining the *priority* of each state.

A player i ($i \in \{0,1\}$) plays by choosing a *feasible strategy*, which is a mapping $\sigma : (Obs_i)^* \to Act_i$. A strategy for i is *winning* when each runs that is compatible with that strategy satisfies the property that the maximal priority of a state which occurs infinitely often in the run is even. The winning condition might be non-observable to player i, as there might exist states $q_1, q_2 \in Q$ that are identically observable by player i, i.e. $o_i(q_1) = o_i(q_2)$, might have different priorities.

The set of winning strategies for a player in a multi-player game with imperfect information is then expressible within the epistemic μ-calculus, similarly to the encoding of the set of winning strategies in a parity game into the μ-calculus from e.g. [11, 22]. Assuming that the largest priority in Q is even and the atomic proposition p_k holds exactly in all states with priority k, the following epistemic modal μ-calculus formula encodes the winning strategies for player i:

$$\nu Z_n \mu Z_{n-1} \ldots \mu Z_1. \bigvee_{\alpha \in Act_i} K_a \bigvee_{k \leq n} \left(p_k \wedge \bigwedge_{\beta \in Act_{1-i}} [\alpha, \beta] Z_k\right)$$

provided that player i's indistinguishability in the multi-agent system constructed from G is based on Obs_i.

3.3 Revisiting the decidability of the model checking problem for the tree semantics of the plain μ-calculus

In this subsection we provide a variant of the Finite Model Theorem for the μ-calculus, which will serve as a basis for our search of a decidable subproblem of the model-checking problem for the epistemic μ-calculus.

Given a multi-agent system $M = (Q, Ag, \delta, q_0, \Pi, (\Pi_a)_{a \in Ag}, \pi)$, and an agent $a \in Ag$, we define the relation $\Gamma_a^M \subseteq Q \times Q$ as follows: $(q,r) \in \Gamma_a^M$ if for any run ρ in M ending in q (i.e. $\rho[|\rho|] = q$) there exists a run ρ' ending in r with $\rho \sim_a \rho'$.

We now define a second semantics for the epistemic μ-calculus, which works on the *set of states* of a multi-agent system M, necessary for the decision problem. This semantics is the extension of the state-based semantics for the μ-calculus [21] by defining a state-based semantics for the epistemic operators.

Formally, each formula ϕ which contains variables $Z_1, \ldots, Z_n$ is associated with a Q-operator $\lceil \phi \rceil_M : \left(2^Q\right)^n \to 2^Q$, again by structural induction (we provide here the semantics for the epistemic μ-calculus in positive form):

- $\lceil p \rceil_M$ resp. $\lceil \neg p \rceil_M$ are the constant Q-operators
$$\lceil p \rceil_M(S_1, \ldots, S_n) = \{q \in Q \mid p \in \pi(q)\}$$
$$\lceil \neg p \rceil_M(S_1, \ldots, S_n) = \{q \in Q \mid p \notin \pi(q)\}$$
- $\lceil Z_i \rceil_M : \left(2^Q\right)^n \to 2^Q$ is the i-th projection Q-operator, i.e. given $S_1, \ldots, S_n \subseteq Q$, $\lceil Z_i \rceil_M(S_1, \ldots, S_n) = S_i$.
- $\lceil \phi_1 \vee \phi_2 \rceil_M = \lceil \phi_1 \rceil_M \cup \lceil \phi_2 \rceil_M$, and $\lceil \phi_1 \wedge \phi_2 \rceil_M = \lceil \phi_1 \rceil_M \cap \lceil \phi_2 \rceil_M$.
- Both nexttime modalities are associated with Q-operators $AX^f, EX^f : 2^Q \to 2^Q$ such that:
$$\lceil AX\phi \rceil_M = AX^f \circ \lceil \phi \rceil, \qquad \lceil EX\phi \rceil_M = EX^f \circ \lceil \phi \rceil$$
where:
$$AX^f(S) = \{q \in Q \mid \forall r \in Q \text{ if } (q,r) \in \delta \text{ then } r \in S\}$$
$$EX^f(S) = \{q \in Q \mid \exists r \in Q \text{ with } (q,r) \in \delta \text{ and } r \in S\}$$
- Each pair of epistemic operators K_a / P_a is associated with a pair of Q-operators $K_a^f, P_a^f : 2^Q \to 2^Q$ such that:
$$\lceil P_a \phi \rceil_M = P_a^f \circ \lceil \phi \rceil$$
$$\lceil K_a \phi \rceil_M = K_a^f \circ \lceil \phi \rceil$$
where:
$$K_a^f(S) = \overline{\Gamma_a(\overline{S})} = \{q \in Q \mid \forall s \in Q, \text{ if } (s,q) \in \Gamma_a \text{ then } s \in S\}$$
$$P_a^f(S) = \Gamma_a(S) = \{q \in Q \mid \exists s \in S \text{ s.t. } (s,q) \in \Gamma_a\}$$
- $\lceil \mu Z_i.\phi \rceil_M = \mathsf{lfp}^i_{\lceil \phi \rceil_M}$ and $\lceil \nu Z_i.\phi \rceil_M = \mathsf{gfp}^i_{\lceil \phi \rceil_M}$.

In the sequel, when the multi-agent system M is fixed, we will utilize the notation $\lceil \varphi \rceil$ instead of $\lceil \varphi \rceil_M$.
The following result, giving the connection between the tree semantics and the state-based semantics for the μ-calculus, contains the essence of the Finite Model Theorem for μ-calculus. The result is proved by structural induction on the formula ϕ in [3]:

THEOREM 3. *Given a multi-agent system $M = (Q, Ag, \delta, q_0, \Pi, (\Pi_a)_{a \in Ag}, \pi)$ in which $Q = \{1, \ldots, n\}$ and $q_0 = 1$, and a (plain) μ-calculus formula ϕ, the following diagram[1] commutes:*

$$\begin{array}{ccc} \left(2^Q\right)^n & \xrightarrow{\lceil \phi \rceil} & 2^Q \\ \downarrow{\scriptstyle (t_M^{-1})^n} & & \downarrow{\scriptstyle t_M^{-1}} \\ \left(2^{\mathsf{supp}(t_M)}\right)^n & \xrightarrow{\|\phi\|} & 2^{\mathsf{supp}(t_M)} \end{array} \tag{2}$$

We also say that the diagram 2 holds (or commutes) for the formula ϕ in the system M.

The commutativity of diagram 2 is based on some commutativity properties for the tree operators and the state operators associated with all the logical operators of the μ-calculus. For instance, the AX operator satisfies the following commutativity property:

$$\begin{array}{ccc} 2^Q & \xrightarrow{AX^f} & 2^Q \\ \downarrow{\scriptstyle (t_M^{-1})^n} & & \downarrow{\scriptstyle t_M^{-1}} \\ 2^{\mathsf{supp}(t_M)} & \xrightarrow{AX} & 2^{\mathsf{supp}(t_M)} \end{array} \tag{3}$$

Our search will be directed towards finding particular instances of the model-checking problem where similar commutative diagrams can be provided for the epistemic operators involved in the given epistemic μ-calculus formula.

4. A FRAGMENT OF THE EPISTEMIC μ-CALCULUS WITH A DECIDABLE MODEL CHECKING PROBLEM

In this section, we first introduce some additional notations and notions. Given a multi-agent system M and two agents $a_1, a_2 \in Ag$, we say that the two agents **have compatible observability** if either $\Pi_{a_1} \subseteq \Pi_{a_2}$ or $\Pi_{a_1} \supseteq \Pi_{a_2}$.

[1]The category in which this diagram holds is Set, the category of sets.

Given a formula ϕ, let T_ϕ denote the syntactic tree of ϕ. The following fixes the definition of T_ϕ by structural induction, as it will be needed in the rest of the proof. Note that, in our definition of T_ϕ, each node labeled with a *variable* also has a *successor*, labeled with $\top$. This convention brings the property that each node in T_ϕ whose formula is a variable has a closed subformula (which is $\top$):

- $\mathsf{supp}(T_p) = \{\epsilon\}, T_p(\epsilon) = p$,
- $\mathsf{supp}(T_{\neg p}) = \{\epsilon\}, T_{\neg p}(\epsilon) = \neg p$,
- $\mathsf{supp}(T_Z) = \{\epsilon, 1\}, T_Z(\epsilon) = Z, T_Z(1) = \top$,
- $\mathsf{supp}(T_{Op\phi_1}) = \{\epsilon\} \cup \{1x \mid x \in \mathsf{supp}(T_{\phi_1})\}, T_{Op\phi_1}(\epsilon) = Op, T_{Op\phi_1}(1x) = T_{\phi_1}(x)$, where $Op \in \{AX, EX, K_a, P_a, \mu Z, \nu Z\}$
- $\mathsf{supp}(T_{\phi_1 Op \phi_2}) = \{\epsilon\} \cup \{1x \mid x \in \mathsf{supp}(T_{\phi_1})\} \cup \{2x \mid x \in supp(T_{\phi_2})\}, T_{\phi_1 Op\phi_2}(\epsilon) = Op, T_{\phi_1 Op\phi_2}(1x) = T_{\phi_1}(x), T_{\phi_1 Op\phi_2}(2x) = T_{\phi_2}(x), Op \in \{\wedge, \vee\}$.

We then denote $form(x)$ the subformula of ϕ whose syntactic tree is $T_\phi\big|_x$, i.e. the subtree of T_ϕ rooted at x, and say that x is **closed** if $form(x)$ is closed.

We then say that an epistemic operator $Op \in \{K_a, P_a \mid a \in Ag\}$ is **non-closed** at a node x in a formula ϕ if $form(x)$ is not closed, Op labels a node $y \geq x$ and for all the nodes y' lying on the path between x and y we have that $form(y')$ is not closed.

For each node $x \in \mathsf{supp}(T_\phi)$, we also define $AgNCl_\phi(x)$ as the set of agents a for which K_a or P_a is not closed at x. In addition, given two distinct nodes $x_1 \prec x_2$ with x_2 being closed, we say that x_2 is a *nearest closed successor* of x_1 if no other closed node lies on the path from x_1 to x_2.

Definition 2. A formula ϕ is said to **mix observations of agents** a **and** b (or also: agents a, b have **mixed observations** in ϕ) if the following property holds

> For some epistemic operators $Op_a \in \{K_a, P_a\}, Op_b \in \{K_b, P_b\}$ there exists a node x of T_ϕ such that both Op_a and Op_b are not closed at x.

The **non-mixing model-checking problem** for the epistemic μ-calculus is the problem of deciding whether $t_M \vDash \phi$ for a given multi-agent system M and a closed formula ϕ bearing the restriction that any two agents a, b which have mixed observations in ϕ have compatible observability in M.

All instances of the model-checking problem for KB_n [15, 16], that is, CTL with individual knowledge operators, are formulas of the μ-calculus of non-mixing epistemic fixpoints. Other instances of this model-checking problem consist of the following formulas

$$\mu Z_1.(p \vee K_a(EX.Z_1) \wedge \nu Z_2.(q \wedge Z_1 \wedge K_a(EXZ_2)))$$
$$\mu Z_1.(p \vee K_a(EX.Z_1) \wedge \nu Z_2.(q \wedge K_b(EXZ_2)))$$

in pair with systems M in which $\Pi_a \subseteq \Pi_b$. Also any instance of the model-checking problem for the following common knowledge formula:

$$C_{a,b}\phi = \nu Z.(\phi \wedge K_a Z \vee K_b Z)$$

and with systems M in which a and b do not have compatible observability, is not an instance of the non-mixing model-checking problem.

THEOREM 4. *The non-mixing model-checking problem for the epistemic μ-calculus is decidable.*

The crux of the proof relies on a commutativity property relating t_M^{-1} with the operators K_a/K_a^f, resp. P_a/P_a^f, similar with the properties relating t_M^{-1} with AX/AX^f in diagram 3. Unfortunately, such a commutativity property does not hold for K_a in any multi-agent system M, as is shown in the following example.

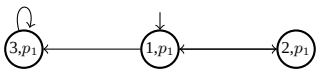


Figure 1: A one-agent system with $\Pi_a = \{p_1\}$.

EXAMPLE 5. *For the one-agent system in Fig. 1 we have that $t_M^{-1}(K_a^f(\{1,3\})) \neq K_a(t_M^{-1}(\{1,3\}))$, as*

$$t_M^{-1}(K_a^f(\{1,3\})) = \{x \in \mathsf{supp}(t_M) \mid x[|x|] = 1\}$$
$$K_a(t_M^{-1}(\{1,3\})) = \{x \in \mathsf{supp}(t_M) \mid x[|x|] = 1 \vee (x[|x|] = 3 \wedge |x| \text{ is odd})\}.$$

Definition 3. Given two multi-agent systems $M_i = (Q_i, Ag, \delta_i, q_0^i, \Pi, (\Pi_a)_{a \in Ag}, \pi_i)$ $(i = 1, 2)$ over the same set of atomic propositions, we say that M_1 is an **in-splitting** of M_2 if there exists a surjective mapping with $\chi : Q_1 \to Q_2$, satisfying the following properties:

1. For each $q, r \in Q_1$, if $(q, r) \in \delta_1$ then $(\chi(q), \chi(r)) \in \delta_2$. Moreover, for any $(q', r') \in \delta_2$ there exist $(q, r) \in \delta_1$ such that $\chi(q) = q', \chi(r) = r'$.
2. For each $q \in Q_1$, $\pi_2(\chi(q)) = \pi_1(q)$.
3. For each $q \in Q_1$, $\mathsf{outdeg}(\chi(q)) = \mathsf{outdeg}(q)$, where $\mathsf{outdeg}(q)$ is the number of transitions leaving q.
4. $\chi(q_0^1) = q_0^2$.

The in-splitting is an **isomorphism** whenever χ is a bijection.

We will call the mapping χ as an *in-splitting mapping*. Also, we write $\chi : M_1 \xrightarrow{Ins} M_2$ to denote the fact that χ is a witness for M_1 being an in-splitting of M_2.

Note that an in-splitting mapping (term borrowed from symbolic dynamics [19]) represents a surjective functional bisimulation between two transition systems. The following proposition can be seen as a generalization of this remark (the proof is given in [3]):

PROPOSITION 6. *Consider two multi-agent systems $M_i = (Q_i, Ag, \delta_i, q_0^i, \Pi, (\Pi_a)_{a \in Ag}, \pi_i)$ $(i = 1, 2)$ over the same set of atoms, connected by an in-splitting mapping $\chi : M_1 \xrightarrow{Ins} M_2$. Then for any plain μ-calculus formula ϕ the following diagram commutes:*

$$\begin{array}{ccc} \left(2^{Q_1}\right)^n & \xrightarrow{\lceil\phi\rceil_{M_1}} & 2^{Q_1} \\ {\scriptstyle (\chi^{-1})^n}\uparrow & & \uparrow{\scriptstyle \chi^{-1}} \\ \left(2^{Q_2}\right)^n & \xrightarrow{\lceil\phi\rceil_{M_2}} & 2^{Q_2} \end{array} \quad (4)$$

REMARK 7. *Proposition 6 does not hold for any epistemic μ-calculus formula. To see this, consider the system depicted in Fig. 2, which is an in-splitting of the system from Fig. 1, obtained by splitting state 3 in Fig. 1 in two states, denoted 3 and 4 in Fig. 2, (i.e. $\chi(1) = 1, \chi(2) = 2, \chi(3) = \chi(4) = 3$) with transitions $(3, 4) \in \delta$ and $(4, 4) \in \delta$.*

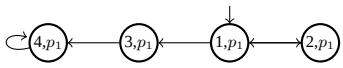


Figure 2: An in-splitting of the system from Fig. 1.

Note that we have

$$\lceil K_a^f X \rceil_{M_2}\big(\{1,2,3\}\big) = \{1,3\}$$
$$\lceil K_a^f X \rceil_{M_1}\big(\{1,2,3\}\big) = \{1,2,3\}$$

and hence $\lceil K_a^f X \rceil_{M_1} \circ \chi^{-1} \neq \chi^{-1} \circ \lceil K_a^f X \rceil_{M_2}$.

The following notion corresponds with the "subset construction" used for model-checking LTLK/CTLK [26, 8] or solving 2-player parity games with one player having incomplete information [7]:

Definition 4. Given a multi-agent system $M = (Q, Ag, \delta, q_0, \Pi, (\Pi_a)_{a\in Ag}, \pi)$, we define the multi-agent system

$$\Delta_a^{pre}(M) = (\tilde{Q}^{pre}, Ag, \tilde{\delta}, \tilde{q}_0, \Pi, (\Pi_a)_{a\in Ag}, \tilde{\pi})$$

as follows:

- $\tilde{Q}^{pre} = \{(s,S) \mid s \in Q, S \subseteq \{q \in Q \mid \pi_a(q) = \pi_a(s)\}\}$ and $\tilde{q}_0 = (q_0, \{q_0\})$.
- $\tilde{\delta}$ is composed of all tuples of the form $((s,S),(r,R))$ where $(s,r) \in \delta$ and $R = \{r' \in Q \mid \pi_a(r') = \pi_a(r)$ and $\exists s' \in S$ with $(s',r') \in \delta\}$.
- $\tilde{\pi}(s,S) = \pi(S) = \pi(s)$.

The a-**distinction** of M, denoted $\Delta_a(M)$, is the restriction of $\Delta_a^{pre}(M)$ to reachable states, i.e.,

$$\Delta_a(M) = (\tilde{Q}, Ag, \tilde{\delta}\mid_{\tilde{Q}}, \tilde{q}_0, \Pi, (\Pi_a)_{a\in Ag}, \tilde{\pi}\mid_{\tilde{Q}})$$

where $\tilde{Q} = \{\tilde{s} \in \tilde{Q}^{pre} \mid \tilde{s}$ is reachable from $\tilde{q}_0\}$.

Given a multi-agent system $M = (Q, Ag, \delta, q_0, \Pi, (\Pi_a)_{a\in Ag}, \pi)$, and an agent $a \in Ag$, we say that M is a-**distinguished** if Γ_a^M (relation defined on page 5) is a **congruence relation**, that is, an equivalence relation with the following property:

$$\text{for any } q,r \in Q, \text{ if } (q,r) \in \Gamma_a^M, (q,q') \in \delta, (r,r') \in \delta \text{ and}$$
$$\pi_a(q') = \pi_a(r'), \text{ then } (q',r') \in \Gamma_a^M. \tag{5}$$

We utilize from now on the notation Γ_a whenever the system M is understood from the context.

PROPOSITION 8. *1. For any multi-agent system M, $\Delta_a(M)$ is an in-splitting of M. We denote this in-splitting as $\Delta_{a,M}^{-1} : \Delta_a(M) \to M$. Whenever the system M is clear from the context, we use the notation Δ_a^{-1} instead of $\Delta_{a,M}^{-1}$.*

2. For any agent $a \in Ag$ we have that $\Delta_a(M)$ is a-distinguished.

PROPOSITION 9. *For any multi-agent system M and two agents $a,b \in Ag$ with $\Pi_a \subseteq \Pi_b$, if M is b-distinguished, then $\Delta_a(M)$ is b-distinguished too.*

PROPOSITION 10. *For any multi-agent system M, the following diagram commutes iff M is a-distinguished:*

$$\begin{array}{ccc} 2^Q & \xrightarrow{K_a^f} & 2^Q \\ {\scriptstyle t_M^{-1}}\downarrow & & \downarrow{\scriptstyle t_M{}^{-1}} \\ 2^{\mathsf{supp}(t_M)} & \xrightarrow{K_a} & 2^{\mathsf{supp}(t_M)} \end{array} \tag{6}$$

The same holds if the pair K_a/K_a^f is replaced with P_a/P_a^f.

Definition 5. We say that the pair of epistemic operators K_a/K_a^f, resp. P_a/P_a^f, **commutes for** M if the diagram 6 is commutative for the respective pair.

Proposition 10 gives the first restricted form which ensures the commutativity of diagram 2 for formulas of the epistemic μ-calculus. The second restricted form in which the pair K_a/K_a^f (resp. P_a/P_a^f) commutes for a system is stated as point 2 in the next proposition:

PROPOSITION 11. *Consider two multi-agent systems $M_i = (Q_i, Ag, \delta_i, q_0^i, \Pi, (\Pi_a)_{a\in Ag}, \pi_i)$ with $Q_i = \{1,\dots,n_i\}$, $(i = 1,2)$, related by an in-splitting $\chi : M_1 \xrightarrow{Ins} M_2$, and define the tree mapping $\hat{\chi} : \mathsf{supp}(t_{M_1}) \xrightarrow{Ins} \mathsf{supp}(t_{M_2})$, where $\hat{\chi}(\varepsilon) = \varepsilon$ and $\hat{\chi}(xi) = \hat{\chi}(x)\cdot\chi(i)$, for any $x \in \mathsf{supp}(t_{M_1})$ and $i \in Q_1$. Then the following properties hold:*

1. *$\hat{\chi}$ is a tree isomorphism between t_{M_1} and t_{M_2} and $t_{M_2} \circ \hat{\chi} = \chi \circ t_{M_1}$.*
2. *For any closed formula ϕ of the epistemic μ-calculus for which the diagram 2 commutes in the system M_2, the following property holds:*

$$\|\phi\|_{M_1} = t_{M_1}^{-1}(\chi^{-1}(\lceil\phi\rceil_{M_2}))$$

REMARK 12. *The previous proposition tells us that, for* closed *formulas of the epistemic μ-calculus for which diagram 2 commutes in M_2, in the eventuality that the system M_2 needs to be replaced with a "larger" system M_1 (for reasons related with the "subset construction" that ensures the first type of commutativity of K_a/P_a), the validity of ϕ on the tree t_{M_1} can be recovered from the set of states $\chi^{-1}\big(\lceil\phi\rceil_{M_2}\big)$, through the inverse tree mapping $t_{M_1}^{-1}$.*

We have now the essential ingredients that ensure the decidability of the model-checking problem for the μ-calculus of nonmixing epistemic fixpoints. The algorithm runs as follows: we proceed by constructing the Q-operator interpretations of the subformulas of ϕ on the given system M, in a bottom-up traversal of the syntactic tree T_ϕ. As long as we only treat subformulas not containing any epistemic operator, Theorem 3 ensures that these boolean operators are correct finitary abstractions of the tree semantics of our subformulas.

The first time we encounter in T_ϕ an epistemic operator, say, K_a, s.t. the subformula in the current node is $K_a\phi'$, we need to replace M with its *a-distinction*, $\Delta_a(M)$, in order for the appropriate diagram to commute. This replacement is easier when ϕ' is a closed plain μ-calculus formula. By combining Propositions 11 and 10, the tree semantics of the formula $K_a\phi'$ can be computed using the boolean operator $K_a^f\big(\Delta_a^{-1}\big(\lceil\phi'\rceil_M\big)\big)$ in $\Delta_a(M)$, where $\Delta_a^{-1}\big(\lceil\phi'\rceil_M\big)$ represents the set of states in $\Delta_a(M)$ on which ϕ' holds.

The procedure is different when ϕ' is not closed. In this situation, we cannot determinize M, as observed in the remark 7. Therefore we need to descend along the syntactic tree to *all* the "nearest" nodes whose formulas are closed, and only there apply the a-distinction construction, as required by Proposition 11.

Suppose even further that ϕ' itself contains other knowledge operators, and some other knowledge operator K_b is encountered during this descent. The "nonmixing" assumption on our formula implies that this other agent b has compatible observability with our a (K_a and K_b are not closed at the node associated with K_a). Therefore, the a-distinction of the models applied at lower levels commutes with K_b, fact which is ensured by Proposition 10 when the two agents have compatible observability.

This whole process ends when we arrive in the root of the syntactic tree, with an in-splitting M' of the initial system M and a (constant) boolean operator σ, which gives the finitary abstraction of the set of nodes of the tree t_M where ϕ holds. The following paragraphs formalize this process.

PROOF OF THEOREM 4. Given a formula ϕ in the μ-calculus of non-mixing epistemic fixpoints and a multi-agent system M, we associate with each node x of T_ϕ an in-splitting mapping, denoted $T_\phi^{Ins}(x)$, such that the following properties hold:

1. For the root ϵ we have $T_\phi^{Ins}(\epsilon) = id_M$. Also for any non-closed node x in $\mathsf{supp}(T_\phi)$, we have that $T_\phi^{Ins}(x) = id_{M'}$, where M' is an in-splitting of M.

2. For any $x, xi \in \mathsf{supp}(T_\phi), i \in \{1,2\}$, $codom(T_\phi^{Ins}(x)) = dom(T_\phi^{Ins}(xi))$,

3. For any nodes $x_1, x_2 \in \mathsf{supp}(T_\phi)$ with $x_1 \preceq x_2$, define first the *in-splitting mapping* between x_1 and x_2 as:

$$T_\phi^{Ins}(x_1...x_2) = T_\phi^{Ins}(x_1) \circ ... \circ T_\phi^{Ins}(x_2)$$

Then, for any leaves x_1, x_2 in T_ϕ we have that $T_\phi^{Ins}(\epsilon...x_1) = T_\phi^{Ins}(\epsilon...x_2)$, where ϵ is the root of T_ϕ.

4. For any node x_1 which is a nearest closed successor of the root ϵ, if $AgNCl_\phi(\epsilon) = \{a_1, \ldots, a_k\}$ and $\Pi_{a_1} \subseteq \ldots \subseteq \Pi_{a_k}$, then $T_\phi^{Ins}(x_1)$ has the form:

$$T_\phi^{Ins}(x_1) = \Delta_{a_1}^{-1} \circ \ldots \circ \Delta_{a_k}^{-1} \circ \chi, \text{ for some } \chi,$$

Assuming that T_ϕ^{Ins} is constructed with all the properties above, we denote $InS(T_\phi^{Ins}) = T_\phi^{Ins}(\epsilon...x)$ where x is any leaf in T_ϕ. In the sequel, whenever we want to emphasize a property of the root of the syntactic tree T_ϕ, we denote it ϵ^ϕ.

The construction of T_ϕ^{Ins} proceeds by structural induction on ϕ. For the base case $\phi = p$ or $\phi = \neg p$, we put $T_p^{Ins}(\epsilon) = T_{\neg p}^{Ins}(\epsilon) = id_M$, for any $p \in \Pi$. Also for $\phi = Z$, $Z \in \mathcal{Z}$, note that, by construction, the root of T_Z has a leaf successor which is the only child node. Then, $T_Z^{Ins}(\epsilon) = T_Z^{Ins}(1) = id_M$.

For the induction case, take a formula $\phi = Op.\phi'$ where $Op \in \{AX, EX, \mu Z, \nu Z\}$, and assume $T_{\phi'}^{Ins}(x)$ is defined. Then we put $T_\phi^{Ins}(1x) = T_{\phi'}^{Ins}(x)$ for any node x of $\mathsf{supp}(T_{\phi'})$, and $T_\phi^{Ins}(\epsilon^\phi) = id_{M'}$, where $M' = dom(T_{\phi'}^{Ins}(\epsilon^{\phi'}))$.

Suppose $\phi = K_a\phi'$ or $\phi = P_a\phi'$. Note that for each node $1x$ which is not closed in T_ϕ, the node x is not closed in $T_{\phi'}$ either. Then we put $T_\phi^{Ins}(1x) = T_{\phi'}^{Ins}(x) = id_{M'}$, with M' the appropriate multi-agent system. We also put $T_\phi^{Ins}(\epsilon^\phi) = id_{M_0}$ for the appropriate M_0. Furthermore, for each closed node $1x_1 \in \mathsf{supp}(T_\phi)$ which is *not* a nearest closed successor of ϵ^ϕ, we put $T_\phi^{Ins}(1x_1) = T_{\phi'}^{Ins}(x_1)$.

Take further a node $1x_1$ which is a nearest closed successor of the root ϵ^ϕ and assume $AgNCL(\epsilon^\phi) = \{a_1, ..., a_k\}$. By the above property 4 in the induction hypothesis, the in-splitting mapping in x_1 is $T_{\phi'}^{Ins}(x_1) = \Delta_{a_1}^{-1} \circ \ldots \circ \Delta_{a_k}^{-1} \circ \chi$ with $\Pi_{a_1} \subseteq \ldots \subseteq \Pi_{a_k}$. On the other hand, by the assumption that ϕ is a nonmixing formula, a must have compatible observability with all the agents $a_1, \ldots, a_k$. Therefore, there must exist some $i \leq k$ such that $\Pi_{a_1} \subseteq \ldots \subseteq \Pi_{a_i} \subseteq \Pi_a \subseteq \Pi_{a_{i+1}} \subseteq \ldots \subseteq \Pi_{a_k}$. We then define

$$T_\phi^{Ins}(1x_1) = \Delta_{a_1}^{-1} \circ \ldots \circ \Delta_{a_i}^{-1} \circ \Delta_a^{-1} \circ \Delta_{a_{i+1}}^{-1} \circ \ldots \circ \Delta_{a_k}^{-1} \circ \chi$$

Note that the domain and the codomain of each $\Delta_{a_j}^{-1}$, $(j \leq i)$ are different in T_ϕ^{Ins} from those in $T_{\phi'}^{Ins}$, due to the insertion of Δ_a^{-1}.

According to the above constructions for $\phi = K_a\phi'$ of $\phi = P_a\phi'$, all the four properties are satisfied by T_ϕ^{Ins}, the fourth one resulting from the construction of the in-splitting mapping for the nearest closed successors of the root.

Finally, take $\phi = \phi_1 Op\, \phi_2$ ($Op \in \{\wedge, \vee\}$). If $T_{\phi_1}^{Ins} = T_{\phi_2}^{Ins}$, put $T_\phi^{Ins}(1x) = T_{\phi_1}^{Ins}(x)$ for all nodes $x \in \mathsf{supp}(T_{\phi_1})$, $T_\phi^{Ins}(2x) = T_{\phi_2}^{Ins}(x)$ for all $x \in \mathsf{supp}(T_{\phi_2})$ and $T_\phi^{Ins}(\epsilon) = id_M$.

Suppose now $T_{\phi_1}^{Ins} \neq T_{\phi_2}^{Ins}$. Consider $AgNCl(1) = \{a_1, \ldots, a_k\}$ and $AgNCl(2) = \{b_1, \ldots, b_l\}$ with $\Pi_{a_1} \subseteq \ldots \subseteq \Pi_{a_k}$ and $\Pi_{b_1} \subseteq \ldots \subseteq \Pi_{b_l}$. Take then a node x_1 which is a nearest closed successor of the root of T_{ϕ_1}, ϵ^{ϕ_1}, and a node x_2 which is a nearest closed successor of ϵ^{ϕ_2}. By the induction hypothesis we have:

$$T_{\phi_1}^{Ins}(x_1) = \Delta_{a_1}^{-1} \circ \ldots \circ \Delta_{a_k}^{-1} \circ \chi_1 \quad InS(T_{\phi_1}^{Ins}) = T_{\phi_1}^{Ins}(x_1) \circ \chi_1'$$
$$T_{\phi_2}^{Ins}(x_2) = \Delta_{b_1}^{-1} \circ \ldots \circ \Delta_{b_l}^{-1} \circ \chi_2 \quad InS(T_{\phi_2}^{Ins}) = T_{\phi_2}^{Ins}(x_2) \circ \chi_2'$$

with appropriate in-splittings $\chi_1, \chi_1', \chi_2, \chi_2'$.

On the other hand, by the assumption on ϕ being nonmixing, for any $i \leq k, j \leq l$, the two agents a_i and b_j must have compatible observability. It therefore follows that there exists a reordering of the union $\{a_1, \ldots, a_k\} \cup \{b_1, \ldots, b_l\}$ as $\{c_1, \ldots, c_m\}$ such that $\Pi_{c_i} \subseteq \Pi_{c_{i+1}}$ for all $i \leq m-1$. Denote then:

$$\chi_0 = \Delta_{c_1}^{-1} \circ \ldots \circ \Delta_{c_m}^{-1}$$

By Proposition 9, χ_0 is a c-distinction for any $c \in \{a_1, \ldots, a_k\} \cup \{b_1, \ldots, b_l\}$. Also, by property 2 of the induction hypothesis, χ_0 is independent of the choice of the nodes x_1, x_2.

The same property from the induction hypothesis also ensures that, for any nearest closed successor $\overline{x}_2$ of ε^{ϕ_2}, there exist in-splittings $\overline{\chi}_2^{\phi_2,\overline{x}_2}, \tilde{\chi}_2^{\phi_2,\overline{x}_2}$ such that:

$$T_{\phi_2}^{Ins}(\overline{x}_2) = \Delta_{b_1}^{-1} \circ \ldots \circ \Delta_{b_l}^{-1} \circ \overline{\chi}_2^{\phi_2,\overline{x}_2} \tag{7}$$
$$InS(T_{\phi_2}^{Ins}) = T_\phi^{Ins}(\overline{x}_2) \circ \tilde{\chi}_2^{\phi_2,\overline{x}_2} \tag{8}$$

We will then construct $T_\phi^{Ins}(\cdot)$ as follows:

1. For each closed node x which is a leaf in T_{ϕ_1} but not a nearest closed successor of ϵ^{ϕ_1}, we put $T_\phi^{Ins}(1x) = T_{\phi_1}^{Ins}(x) \circ \chi_2 \circ \chi_2'$.

2. For each non-leaf, closed node x in T_{ϕ_1} which is not a nearest closed successor of ϵ^{ϕ_1} we copy $T_\phi^{Ins}(1x) = T_{\phi_1}^{Ins}(x)$.

3. For each nearest closed successor x of ϵ^{ϕ_1} which is not a leaf in T_{ϕ_1} we put $T_\phi^{Ins}(1x) = \chi_0 \circ \chi_1$.

4. For each closed node x which is a leaf in T_{ϕ_1} and a nearest closed successor of ϵ^{ϕ_1}, we put $T_\phi^{Ins}(1x) = \chi_0 \circ \chi_1 \circ \chi_1' \circ \chi_2 \circ \chi_2'$.

5. For each closed node x which is not a nearest closed successor of ϵ^{ϕ_2} we copy $T_\phi^{Ins}(2x) = T_{\phi_2}^{Ins}(x)$.

6. For each closed node x which is a nearest closed successor of ϵ^{ϕ_2} we put $T_\phi^{Ins}(2x) = \chi_0 \circ \chi_1 \circ \chi_1' \circ \overline{\chi}_2^{\phi_2,x}$, where $\overline{\chi}_2^{\phi_2,x}$ is the in-splitting mapping associated with the node x as in Identity 8 above.

7. For the root ϵ and the non-closed nodes x of T_ϕ, $T_\phi^{Ins}(\epsilon) = id_{M'}$ and $T_\phi^{Ins}(x) = id_{M''}$, with M' and M'' appropriate multi-agent systems.

It's not difficult to see that the resulting mapping $T_{\phi_2}^{Ins}(\cdot)$ satisfies the five desired properties. More specifically, property 2 amounts to the following identity:

$$InS(T_\phi^{Ins}) = \chi_0 \circ \chi_1 \circ \chi_1' \circ \chi_2 \circ \chi_2'$$

Now we may show how T_ϕ^{Ins} can be used to build our algorithm. Let M_x denote the multi-agent system which is the *domain* of the in-splitting $T_\phi^{Ins}(x)$, and denote Q_x its state-space. Also, for convenience, we denote $\overline{M}_x$ the multi-agent system which represents the *codomain* of $T_\phi^{Ins}(x)$, and $\overline{Q}_x$ its state-space. Note that when $x, x1 \in \mathsf{supp}(T_\phi)$, $\overline{M}_x = M_{x1}$, and similarly $\overline{M}_x = M_{x2}$ when $x2 \in \mathsf{supp}(T_\phi)$.

Once we built the tree T_ϕ^{Ins}, we associate with each node x in T_ϕ a $\overline{Q}_x$-operator that will give all the information on the satisfiability of $form(x)$ in the given model. Formally, we build the tree T_ϕ^{str} whose domain is $\mathsf{supp}(T_\phi) \setminus \{x \mid T_\phi(x) = \top\}$ and which associates with each node x a $\overline{Q}_x$-operator $T_\phi^{str}(x) : (2^{Q_x})^n \to 2^{Q_x}$. The construction will be achieved such that

$$\|form(x)\| \circ \left(t_{M_x}^{-1}\right)^n = t_{M_x}^{-1} \circ T_\phi^{str}(x) \tag{9}$$

for each node x with $form(x) \neq \top$.

The construction proceeds bottom-up on $\mathsf{supp}(T_\phi)$. We actually build *two* trees, T_ϕ^{str} and $\overline{T}_\phi^{str}$, such that $\overline{T}_\phi^{str}(x) : (2^{\overline{Q}_x})^n \to 2^{\overline{Q}_x}$ and $T_\phi^{str}(x) = \overline{T}_\phi^{str}(x) \circ \left[\left(T_\phi^{Ins}(x)\right)^{-1}\right]^n$, that is,

$$T_\phi^{str}(x)(S_1, \ldots, S_n) = \overline{T}_\phi^{str}(x)\left(\left(T_\phi^{Ins}(x)\right)^{-1}(S_1, \ldots, S_n)\right) \tag{10}$$

Note that, once we build $\overline{T}_\phi^{str}(x)$ for a node x, $T_\phi^{str}(x)$ is defined by Identity 10, so we only explain the construction for $\overline{T}_\phi^{str}(x)$.

For nodes x that are leaves in T_ϕ with $T_\phi(x) = p \in \Pi$, we put $\overline{T}_\phi^{str}(x) = \lceil p \rceil_M$, the constant $\overline{Q}_x$-operator. Recall that we do not define $T_\phi^{str}(x)$ for $\overline{T}_\phi(x) = \top$.

For $T_\phi(x) = Z_i \in \mathcal{Z}$ we put $\overline{T}_\phi^{str}(x)(S_1, \ldots, S_n) = S_i$, the i-th projection on $(2^{\overline{Q}_x})^n$.

For nodes x with $T_\phi(x) = Op \in \{AX, EX, K_a, P_a \mid a \in Ag\}$ we put

$$\overline{T}_\phi^{str}(x)(S_1, \ldots, S_n) = Op^f\left(T_\phi^{str}(x1)(S_1, \ldots, S_n)\right)$$

For $T_\phi(x) = \wedge$ we put

$$\overline{T}_\phi^{str}(x)(S_1, \ldots, S_n) = \left(T_\phi^{str}(x1)(S_1, \ldots, S_n)\right) \cap \left(T_\phi^{str}(x2)(S_1, \ldots, S_n)\right)$$

and similarly for $T_\phi(x) = \vee$, with $\cap$ replaced with $\cup$ in the above formula defining $\overline{T}_\phi^{str}(x)(S_1, \ldots, S_n)$.

For $T_\phi(x) = \mu Z_i$ with $1 \leq i \leq n$ we put

$$\overline{T}_\phi^{str}(x) = \mathsf{lfp}^i_{\lceil T_\phi^{str}(x1) \rceil}$$

and, similarly, for $T_\phi(x) = \nu Z_i$ we define

$$\overline{T}_\phi^{str}(x) = \mathsf{gfp}^i_{\lceil T_\phi^{str}(x1) \rceil}$$

The validity of Identity 9 follows then from Propositions 10 and 11.

The final step consists in checking whether $q_0^\varepsilon \in T_\phi^{str}(\varepsilon)$, where q_0^ε is the initial state in the multi-agent system M_ε associated with the root of T_ϕ. The result of this check gives the answer to the problem whether $\varepsilon \vDash \phi$ in t_M.

□

The following result follows from a similar result for LTLK from [26]. A self-contained proof can be found in [3]:

THEOREM 13. *The model checking problem for the μ-calculus of non-mixing epistemic fixpoints is hard for non-elementary time.*

5. CONCLUSIONS AND COMMENTS

We have presented a fragment of the epistemic μ-calculus having a decidable model-checking problem. We argued in the introduction that the decidability result does not seem to be achievable using tree automata or multi-player games. Two-player games with one player having incomplete information and with non-observable winning conditions from [7] do not seem to be appropriate for the whole calculus as they are only equivalent with a restricted type of combinations of knowledge operators and fixpoints. We conjecture that the formula $\nu Z\left(p \vee AX.P_a Z\right)$ is not equivalent with any (tree automaton presentation of a) two-player game with path winning conditions. Translating this formula to a generalized tree automaton seems to require specifying a winning condition on concatenations of finite paths in the tree with "jumps" between two identically-observable positions in the tree. This conjecture extends the non-expressivity results from [6] relating ATL and $\mu - ATL$.

The second reason for which the above-mentioned generalization would not work comes from results in [9] showing that the satisfiability problem for CTL or LTL is undecidable with the concrete observability relation presented here. It is then expectable that if a class of generalized tree automata is equivalent with the μ-calculus of non-mixing epistemic fixpoints, then that class would have an undecidable emptiness problem and only its "testing problem" would be decidable. Therefore, the classical determinacy argument for two-player games would not be translatable to such a class of automata.

Acknowledgments

We are grateful to D. Guelev for his careful reading of earlier versions of this paper.

6. REFERENCES

[1] Th. Ågotnes. Action and knowledge in alternating-time temporal logic. *Synthese*, 149(2):375–407, 2006.

[2] A. Arnold and D. Niwiński. *Rudiments of μ-calculus*, volume 146 of *Studies in Logic and the Foundations of Mathematics*. North-Holland, 2001.

[3] R. Bozianu, C. Dima, and C. Enea. Model-checking an epistemic μ-calculus with synchronous and perfect recall semantics. Available at: `http://arxiv.org/abs/1204.2087`, 2012.

[4] J. Bradfield and C. Stirling. Modal mu-calculi. In J. van Benthem P. Blackburn and F. Wolter, editors, *The Handbook of Modal Logic*, pages 721–756. Elsevier, 2006.

[5] N. Bulling, J. Dix, and W. Jamroga. Model checking logics of strategic ability: Complexity. In M. Dastani, K. V. Hindriks, and J.-J. C. Meyer, editors, *Specification and Verification of Multi-Agent Systems*, pages 125–160. Springer, 2010.

[6] N. Bulling and W. Jamroga. Alternating epistemic mu-calculus. In *Proceedings of IJCAI'2011*, pages 109–114. IJCAI/AAAI, 2011.

[7] K. Chatterjee and L. Doyen. The complexity of partial-observation parity games. In *Proceedings of LPAR-17*, volume 6397 of *Lecture Notes in Computer Science*, pages 1–14. Springer, 2010.

[8] C. Dima. Revisiting satisfiability and model-checking for CTLK with synchrony and perfect recall. In *Proceedings of CLIMA IX*, volume 5405 of *LNAI*, pages 117–131, 2008.

[9] C. Dima. Non-axiomatizability for linear temporal logic of knowledge with concrete observability. *Journal of Logic and Computation*, pages 939–958, 2011.

[10] C. Dima, C. Enea, and D. Guelev. Model-checking an alternating-time temporal logic with knowledge, imperfect information, perfect recall and communicating coalitions. *Electronic Proceedings in Theoretical Computer Science*, 25:103–117, 2010.

[11] E. A. Emerson and C. S. Jutla. Tree automata, mu-calculus and determinacy (extended abstract). In *Proceedings of FOCS'91*, pages 368–377. IEEE Computer Society, 1991.

[12] K. Engelhardt, R. van der Meyden, and K. Su. Modal logics with a linear hierarchy of local propositional quantifiers. In *Proceedings of AiML'02*, pages 9–30. King's College Publications, 2003.

[13] B. Finkbeiner and S. Schewe. Uniform distributed synthesis. In *Proceedings of LICS'05*, pages 321–330. IEEE Computer Society, 2005.

[14] V. Goranko and G. van Drimmelen. Complete axiomatization and decidability of alternating-time temporal logic. *TCS*, 353(1-3):93–117, 2006.

[15] J. Halpern and M. Vardi. The complexity of reasoning about knowledge and time: Extended abstract. In *Proceedings of STOC'86*, pages 304–315, 1986.

[16] J. Halpern and M. Vardi. The complexity of reasoning about knowledge and time. I. Lower bounds. *Journal of Computer System Sciences*, 38(1):195–237, 1989.

[17] M. Kacprzak and W. Penczek. Fully symbolic unbounded model checking for alternating-time temporal logic. *Autonomous Agents and Multi-Agent Systems*, 11(1):69–89, 2005.

[18] O. Kupferman and M.Y. Vardi. Synthesizing distributed systems. In *Proceedings of LICS 2001*, pages 389–398. IEEE Computer Society, 2001.

[19] D. Lind and B. Marcus. *An Introduction to Symbolic Dynamics and Coding*. Cambridge University Press, 1995.

[20] A. Lomuscio and Fr. Raimondi. Mcmas: A model checker for multi-agent systems. In *Proceedings of TACAS'2006*, volume 3920 of *LNCS*, pages 450–454, 2006.

[21] D. Niwiński and I. Walukiewicz. Games for the mu-calculus. *Theor. Comput. Sci.*, 163(1&2):99–116, 1996.

[22] J. Obdržálek. *Algorithmic Analysis of Parity Games*. PhD thesis, University of Edinburgh, 2006.

[23] P.-Y. Schobbens. Alternating-time logic with imperfect recall. *Electronic Notes in Theoretical Computer Science*, 85(2):82–93, 2004.

[24] N. Shilov and N. Garanina. Model checking knowledge and fixpoints. In *Proceedings of FICS'02*, pages 25–39, 2002.

[25] J. van Benthem and E. Pacuit. The tree of knowledge in action: Towards a common perspective. In *Proceedings of AiML'06*, pages 87–106. College Publications, 2006.

[26] R. van der Meyden and N. Shilov. Model checking knowledge and time in systems with perfect recall (extended abstract). In *Proceedings of FSTTCS'99*, volume 1738 of *LNCS*, pages 432–445, 1999.

[27] R. van der Meyden and Th. Wilke. Synthesis of distributed systems from knowledge-based specifications. In *Proceedings of CONCUR'05*, LNCS, pages 562–576. Springer Verlag, 2005.

Hybrid-Logical Reasoning in False-Belief Tasks

Torben Braüner
Programming, Logic and Intelligent Systems Research Group
Roskilde University
P.O. Box 260
DK-4000 Roskilde, Denmark
torben@ruc.dk

ABSTRACT

The main aim of the present paper is to use a proof system for hybrid modal logic to formalize what are called false-belief tasks in cognitive psychology, thereby investigating the interplay between cognition and logical reasoning about belief. We consider two different versions of the Smarties task, involving respectively a shift of perspective to another person and to another time. Our formalizations disclose that despite this difference, the two versions of the Smarties task have exactly the same underlying logical structure. We also consider the Sally-Anne task, having a somewhat more complicated logical structure, presupposing a "principle of inertia" saying that a belief is preserved over time, unless there is belief to the contrary.

1. INTRODUCTION

In the area of cognitive psychology there is a reasoning task called the *Smarties task*. The following is one version of the Smarties task.

> A child is shown a Smarties tube where unbeknownst to the child the Smarties have been replaced by pencils. The child is asked: "What do you think is inside the tube?" The child answers "Smarties!" The tube is then shown to contain pencils only. The child is then asked: "If your mother comes into the room and we show this tube to her, what will she think is inside?"

It is well-known from experiments that most childred above the age of four correctly say "Smarties" (thereby attributing a false belief to the mother) whereas younger children say "Pencils" (what they know is inside the tube). For autistic[1] children the cutoff age is higher than four years, which is one reason to the interest in the Smarties task.

The Smarties task is one out of a family of reasoning tasks called *false-belief tasks* showing the same pattern, that most children above four answer correctly, but autistic children have to be older. This was first observed in the paper [4] in connection with another false-belief task called the *Sally-Anne task*. Starting with the authors of that paper, many researchers in cognitive psychology have argued that there is a link between autism and a lack of what is called *theory of mind*, which is a capacity to imagine other people's mental states, for example their beliefs. For a very general formulation of the theory of mind deficit hypothesis of autism, see the book [3].

Giving a correct answer to the Smarties task involves a shift of perspective to another person, namely the mother. You have to put yourself in another person's shoes, so to speak. Since the capacity to take another perspective is a precondition for figuring out the correct answer to the Smarties task and other false-belief tasks, the fact that autistic children have a higher cutoff age is taken to support the claim that autists have a limited or delayed theory of mind. For a critical overview of these arguments, see the book [23] by Keith Stenning and Michiel van Lambalgen. The books [23] and [3] not only consider theory of mind at a cognitive level, such as in connection with false-belief tasks, but they also discuss it from a biological point of view.

In a range of works van Lambalgen and co-authors have given a detailed logical analysis (but not a full formalization) of the reasoning taking place in the Smarties task and other false-belief tasks in terms of closed-world reasoning as used in non-monotonic logics, see in particular [23]. The analysis of the Smarties task of [23] (in Subsection 9.4.4) makes use of a modality B for belief satisfying two standard modal principles.[2] The first principle is $B(\phi \to \psi) \to (B\phi \to B\psi)$ (principle (9.5) at page 251 in [23]). The second principle is the rule called necessitation, that is, from ϕ derive $B\phi$ (this principle is not mentioned explicitly in [23], but is implicit in the analysis given at the bottom of page 256). These two principles together imply that belief is closed under logical consequence, that is, $B\psi$ can be derived from $\phi \to \psi$ together with $B\phi$, which at least for human agents is implausible (when the modal operator stands for knowledge, this is called logical omniscience).

In the present paper we give a logical analysis of the perspective shift required to give a correct answer to the Smarties and Sally-Anne tasks, and we demonstrate that these tasks can be fully formalized in a hybrid-logical proof system not assuming principles implying logical omniscience, namely the natural deduction system described in Chapter

[1]Autism is a psychiatric disorder with the following three diagnostic criteria: 1. Impairment in social interaction. 2. Impairment in communication. 3. Restricted repetitive and stereotyped patterns of behavior, interests, and activities. For details, see *Diagnostic and Statistical Manual of Mental Disorders, 4th Edition (DSM-IV)*, published by the American Psychiatric Association.

[2]Strictly speaking, the modality B in [23] is not formalized in terms of modal logic, but in terms of what is called event calculus, where B is a predicate that can take formulas as arguments.

TARK 2013, Chennai, India.

4 of the book [8], and the paper [7] as well. Beside not suffering from logical omniscience, why is a *natural deduction* system for *hybrid modal logic* appropriate to this end?

- The subject of proof-theory is the notion of proof and formal, that is, symbolic, systems for representing proofs. Formal proofs built according to the rules of proof systems can be used to represent—describe the structure of—mathematical arguments as well as arguments in everyday human practice. Beside giving a way to distinguish logically correct arguments from incorrect ones, proof systems also give a number of ways to characterize the structure of arguments. Natural deduction style proofs are meant to formalize the way human beings actually reason, so natural deduction is an obvious candidate when looking for a proof system to formalize the Smarties task in.
- In the standard Kripke semantics for modal logic, the truth-value of a formula is relative to points in a set, that is, a formula is evaluated "locally" at a point, where points usually are taken to represent possible worlds, times, locations, epistemic states, persons, states in a computer, or something else. Hybrid logics are extended modal logics where it is possible to directly refer to such points in the logical object language, whereby locality can be handled explicitly, for example, when reasoning about time one can formulate a series of statements about what happens at specific times, which is not possible in ordinary modal logic. Thus, when points in the Kripke semantics represent local perspectives, hybrid-logical machinery can handle explicitly the different perspectives in the Smarties task.

For the above reasons, we have been able to turn our informal logical analysis of the Smarties and Sally-Anne tasks into formal hybrid-logical natural deduction proofs closely reflecting the shift between different perspectives.

The natural deduction system we use for our formalizations is a modified version of a natural deduction system for a logic of situations similar to hybrid logic, originally introduced in the paper [19] by Jerry Seligman. The modified system was introduced in the paper [7], and later on considered in Chapter 4 of the book [8], both by the present author. In what follows we shall simply refer to the modified system as Seligman's system.

Now, Seligman's system allows any formula to occur in it, which is different from the most common proof systems for hybrid logic that only allow formulas of a certain form called satisfaction statements. This is related to a different way of reasoning in Seligman's system, which captures particularly well the reasoning in the Smarties and Sally-Anne tasks. We prove a completeness result which also says that Seligman's system is analytic, that is, we prove that any valid formula has a derivation satisfying the subformula property. Analyticity guarentees that any valid argument can be formalized using only subformulas of the premises and the conclusion. The notion of analyticity goes back to G.W. Leibniz (1646–1716) who called a proof analytic if and only if the proof is based on concepts contained in the proven statement, the main aim being to be able to construct a proof by an analysis of the result, cf. [2].

The present paper is structured as follows. In the second section we recapitulate the basics of hybrid logic, readers well-versed in hybrid logic can safely skip this section. In the third section we introduce Seligman's natural deduction system for hybrid logic. In the fourth and fifth sections we formalize two versions of the Smarties task using this system, and in the sixth section we formalize the Sally-Anne task. A discussion can be found in the seventh section, in the eightth section there are some brief remarks on other work, and in the final section some remarks on further work. In the appendix we prove the above mentioned completeness result, which also demonstrates analyticity.

2. HYBRID LOGIC

The term "hybrid logic" covers a number of logics obtained by adding further expressive power to ordinary modal logic. The history of what now is known as hybrid logic goes back to the philosopher Arthur Prior's work in the 1960s. See the handbook chapter [1] for a detailed overview of hybrid logic. See the book [8] on hybrid logic and its proof-theory.

The most basic hybrid logic is obtained by extending ordinary modal logic with *nominals*, which are propositional symbols of a new sort. In the Kripke semantics a nominal is interpreted in a restricted way such that it is true at exactly one point. If the points are given a temporal reading, this enables the formalization of natural language statements that are true at exactly one time, for example

> it is five o'clock May 10th 2007

which is true at the time five o'clock May 10th 2007, but false at all other times. Such statements cannot be formalized in ordinary modal logic, the reason being that there is only one sort of propositional symbol available, namely ordinary propositional symbols, which are not restricted to being true at exactly one point.

Most hybrid logics involve further additional machinery than nominals. There is a number of options for adding further machinery; here we shall consider a kind of operator called *satisfaction operators*. The motivation for adding satisfaction operators is to be able to formalize a statement being true at a particular time, possible world, or something else. For example, we want to be able to formalize that the statement "it is raining" is true at the time five o'clock May 10th 2007, that is, that

> at five o'clock May 10th 2007 it is raining.

This is formalized by the formula $@_a r$ where the nominal a stands for "it is five o'clock May 10th 2007" as above and where r is an ordinary propositional symbol that stands for "it is raining". It is the part $@_a$ of the formula $@_a r$ that is called a satisfaction operator. In general, if a is a nominal and ϕ is an arbitrary formula, then a new formula $@_a \phi$ can be built (in some literature the notation $a : \phi$ is used instead of $@_a \phi$). A formula of this form is called a *satisfaction statement*. The formula $@_a \phi$ expresses that the formula ϕ is true at one particular point, namely the point to which the nominal a refers. Nominals and satisfaction operators are the most common pieces of hybrid-logical machinery, and are what we need for the purpose of the present paper.

In what follows we give the formal syntax and semantics of hybrid logic. It is assumed that a set of ordinary propositional symbols and a countably infinite set of nominals are given. The sets are assumed to be disjoint. The metavariables p, q, r, ... range over ordinary propositional symbols

and a, b, c, ... range over nominals. Formulas are defined by the following grammar.

$$S ::= p \mid a \mid S \wedge S \mid S \to S \mid \bot \mid \Box S \mid @_a S$$

The metavariables ϕ, ψ, θ, ... range over formulas. Negation is defined by the convention that $\neg\phi$ is an abbreviation for $\phi \to \bot$. Similarly, $\Diamond\phi$ is an abbreviation for $\neg\Box\neg\phi$.

DEFINITION 2.1. *A* model *for hybrid logic is a tuple*

$$(W, R, \{V_w\}_{w \in W})$$

where

1. *W is a non-empty set;*
2. *R is a binary relation on W; and*
3. *for each w, V_w is a function that to each ordinary propositional symbol assigns an element of $\{0, 1\}$.*

The pair (W, R) is called a *frame*. Note that a model for hybrid logic is the same as a model for ordinary modal logic. Given a model $\mathfrak{M} = (W, R, \{V_w\}_{w \in W})$, an *assignment* is a function g that to each nominal assigns an element of W. The relation $\mathfrak{M}, g, w \models \phi$ is defined by induction, where g is an assignment, w is an element of W, and ϕ is a formula.

$$\begin{array}{rll}
\mathfrak{M}, g, w \models p & \text{iff} & V_w(p) = 1 \\
\mathfrak{M}, g, w \models a & \text{iff} & w = g(a) \\
\mathfrak{M}, g, w \models \phi \wedge \psi & \text{iff} & \mathfrak{M}, g, w \models \phi \text{ and } \mathfrak{M}, g, w \models \psi \\
\mathfrak{M}, g, w \models \phi \to \psi & \text{iff} & \mathfrak{M}, g, w \models \phi \text{ implies } \mathfrak{M}, g, w \models \psi \\
\mathfrak{M}, g, w \models \bot & \text{iff} & \text{falsum} \\
\mathfrak{M}, g, w \models \Box\phi & \text{iff} & \text{for any } v \in W \text{ such that } wRv, \\
 & & \mathfrak{M}, g, v \models \phi \\
\mathfrak{M}, g, w \models @_a\phi & \text{iff} & \mathfrak{M}, g, g(a) \models \phi
\end{array}$$

By convention $\mathfrak{M}, g \models \phi$ means $\mathfrak{M}, g, w \models \phi$ for every element w of W and $\mathfrak{M} \models \phi$ means $\mathfrak{M}, g \models \phi$ for every assignment g. A formula ϕ is *valid* if and only if $\mathfrak{M} \models \phi$ for any model $\mathfrak{M}$.

3. SELIGMAN'S SYSTEM

In this section we introduce Seligman's natural deduction systems for hybrid logic. Before defining the system, we shall sketch the basics of natural deduction. Natural deduction style derivation rules for ordinary classical first-order logic were originally introduced by Gerhard Gentzen in [11] and later on developed much further by Dag Prawitz in [16, 17]. See [24] for a general introduction to natural deduction systems. With reference to Gentzen's work, Prawitz made the following remarks on the significance of natural deduction.

> ... the essential logical content of intuitive logical operations that can be formulated in the languages considered can be understood as composed of the atomic inferences isolated by Gentzen. It is in this sense that we may understand the terminology *natural* deduction.
>
> Nevertheless, Gentzen's systems are also natural in the more superficial sense of corresponding rather well to informal practices; in other words, the structure of informal proofs are often preserved rather well when formalized within the systems of natural deduction. ([17], p. 245)

Similar views on natural deduction are expressed many places, for example in a textbook by Warren Goldfarb.

> What we shall present is a system for *deductions*, sometimes called a system of *natural deduction*, because to a certain extent it mimics certain natural ways we reason informally. In particular, at any stage in a deduction we may introduce a new premise (that is, a new supposition); we may then infer things from this premise and eventually eliminate the premise (*discharge* it). ([13], p. 181)

Basically, what is said by the second part of the quotation by Prawitz, and the quotation by Goldfarb as well, is that the structure of informal human arguments can be described by natural deduction derivations.

Of course, the observation that natural deduction derivations often can formalize, or mimic, informal reasoning does not itself prove that natural deduction is the mechanism underlying human deductive reasoning, that is, that formal rules in natural deduction style are somehow built into the human cognitive architecture. However, this view is held by a number of psychologists, for example Lance Rips in the book [18], where he provides experimental support for the claim.

> ... a person faced with a task involving deduction attempts to carry it out through a series of steps that takes him or her from an initial description of the problem to its solution. These intermediate steps are licensed by mental inference rules, such as modus ponens, whose output people find intuitively obvious. ([18], p. x)

This is the main claim of the "mental logic" school in the psychology of reasoning (whose major competitor is the "mental models" school, claiming that the mechanism underlying human reasoning is the construction of models, rather than the application of topic-neutral formal rules).

We have now given a brief motivation for natural deduction and proceed to a formal definition. A *derivation* in a natural deduction system has the form of a finite tree where the nodes are labelled with formulas such that for any formula occurrence ϕ in the derivation, either ϕ is a leaf of the derivation or the immediate successors of ϕ in the derivation are the premises of a rule-instance which has ϕ as the conclusion. In what follows, the metavariables π, τ, ... range over derivations. A formula occurrence that is a leaf is called an *assumption* of the derivation. The root of a derivation is called the *end-formula* of the derivation. All assumptions are annotated with numbers. An assumption is either *undischarged* or *discharged*. If an assumption is discharged, then it is discharged at one particular rule-instance and this is indicated by annotating the assumption and the rule-instance with identical numbers. We shall often omit this information when no confusion can occur. A rule-instance annotated with some number discharges all undischarged assumptions that are above it and are annotated with the number in question, and moreover, are occurrences of a formula determined by the rule-instance.

Two assumptions in a derivation belong to the same *parcel* if they are annotated with the same number and are occurrences of the same formula, and moreover, either are both

Figure 1: Rules for connectives

$$\frac{\phi \quad \psi}{\phi \wedge \psi}\,(\wedge I) \qquad \frac{\phi \wedge \psi}{\phi}\,(\wedge E1) \qquad \frac{\phi \wedge \psi}{\psi}\,(\wedge E2)$$

$$\frac{\begin{array}{c}[\phi]\\ \vdots\\ \psi\end{array}}{\phi \to \psi}\,(\to I) \qquad \frac{\phi \to \psi \quad \phi}{\psi}\,(\to E)$$

$$\frac{\begin{array}{c}[\neg\phi]\\ \vdots\\ \bot\end{array}}{\phi}\,(\bot)^{*}$$

$$\frac{a \quad \phi}{@_a\phi}\,(@I) \qquad \frac{a \quad @_a\phi}{\phi}\,(@E)$$

$$\frac{\begin{array}{c}[\Diamond c]\\ \vdots\\ @_c\phi\end{array}}{\Box\phi}\,(\Box I)^{\dagger} \qquad \frac{\Box\phi \quad \Diamond e}{@_e\phi}\,(\Box E)$$

$*$ ϕ is a propositional letter.
† c does not occur free in $\Box\phi$ or in any undischarged assumptions other than the specified occurrences of $\Diamond c$.

Figure 2: Rules for nominals

$$\frac{\begin{array}{ccc}[\phi_1]\ldots[\phi_n][a] & & [a]\\ \vdots & & \vdots\\ \phi_1 \quad \ldots \quad \phi_n & \psi & \psi\end{array}}{\psi}\,(Term)^{*} \qquad \frac{}{\psi}\,(Name)^{\dagger}$$

$*$ $\phi_1, \ldots, \phi_n$, and ψ are all satisfaction statements and there are no undischarged assumptions in the derivation of ψ besides the specified occurrences of $\phi_1, \ldots, \phi_n$, and a.
† a does not occur in ψ or in any undischarged assumptions other than the specified occurrences of a.

undischarged or have both been discharged at the same rule-instance. Thus, in this terminology rules discharge parcels. We shall make use of the standard notation

$$\begin{array}{c}[\phi^r]\\ \vdots\,\pi\\ \psi\end{array}$$

which means a derivation π where ψ is the end-formula and $[\phi^r]$ is the parcel consisting of all undischarged assumptions that have the form ϕ^r.

We shall make use of the following conventions. The metavariables Γ, Δ, ... range over sets of formulas. A derivation π is called a *derivation of* ϕ if the end-formula of π is an occurrence of ϕ, and moreover, π is called a *derivation from* Γ if each undischarged assumption in π is an occurrence of a formula in Γ (note that numbers annotating undischarged assumptions are ignored). If there exists a derivation of ϕ from $\emptyset$, then we shall simply say that ϕ is *derivable*.

A typical feature of natural deduction is that there are two different kinds of rules for each connective; there are rules called introduction rules which introduce a connective (that is, the connective occurs in the conclusion of the rule, but not in the premises) and there are rules called elimination rules which eliminate a connective (the connective occurs in a premiss of the rule, but not in the conclusion). Introduction rules have names in the form $(\ldots I \ldots)$, and similarly, elimination rules have names in the form $(\ldots E \ldots)$.

Now, Seligman's natural deduction system is obtained from the rules given in Figure 1 and Figure 2. We let $\mathbf{N'}_{\mathcal{H}}$ denote the system thus obtained. The system $\mathbf{N'}_{\mathcal{H}}$ is taken from [7] and Chapter 4 of [8] where it is shown to be sound and complete wrt. the formal semantics given in the previous section. As mentioned earlier, this system is a modified version of a system originally introduced in [19]. The system of [19] was modified in [7] and [8] with the aim of obtaining a desirable property called closure under substitution, see Subsection 4.1.1 of [8] for further explanation.

4. A FIRST EXAMPLE

The way of reasoning in Seligman's system is different from the way of reasoning in most other proof systems for hybrid logic[3]. In this section we give the first example of reasoning using the (*Term*) rule (displayed in Figure 2).

Beside the (*Term*) rule, the key rules in the example are the rules $(@I)$ and $(@E)$ (displayed in Figure 1), which are the introduction and elimination rules for the satisfaction operator. The rule $(@I)$ formalizes the following informal argument.

> It is Christmas Eve 2011; it is snowing, so at Christmas Eve 2011 it is snowing.

And the rule $(@E)$ formalizes the following.

> It is Christmas Eve 2011; at Christmas Eve 2011 it is snowing, so it is snowing.

The (*Term*) rule enables hypothetical reasoning where reasoning is about what is the case at a specific time, possibly different from the actual time. Consider the following informal argument.

> At May 10th 2007 it is raining; if it is raining it is wet, so at May 10th 2007 it is wet.

The reasoning in this example argument is about what is the case at May 10th 2007. If this argument is made at a specific actual time, the time of evaluation is first shifted from the actual time to a hypothetical time, namely May 10th 2007, then some reasoning is performed involving the premise "if it is raining it is wet", and finally the time of evaluation is shifted back to the actual time. The reader is invited to verify this shift of time by checking that the argument is correct, and note that the reader himself (or herself) imagines being at the time May 10th 2007. Note that the premise "if it is raining it is wet" represents a causal relation holding at all times.

[3]We here have in mind natural deduction, Gentzen, and tableau systems for hybrid logic, not axiom systems. Proof systems of the first three types are suitable for actual reasoning, carried out by a human, a computer, or in some other medium. Axiom systems are usually not meant for actual reasoning, but are of a more foundational interest.

Now, in a temporal setting, the side-condition on the rule (*Term*) requiring that all the formulas $\phi_1, \ldots, \phi_n, \psi$ are satisfaction statements (see Figure 2) ensures that these formulas are temporally definite, that is, they have the same truth-value at all times, so the truth-value of these formulas are not affected by a shift of temporal perspective. The rule would not be sound if the formulas were not temporally definite.

We now proceed to the formalization of the above argument about what is the case at May 10th 2007. We make use of the following symbolizations

p It is raining
q It is wet
a May 10th 2007

and we take the formula $p \to q$ as an axiom since it represents a causal relation between p and q holding at all times (note that we use an axiom since the relation $p \to q$ holds between the particular propositions p and q, we do not use an axiom schema since the relation obviously does not hold between any pair of propositions).[4] Then the argument can be formalized as the right-hand-side derivation in Figure 3.

It is instructive to see how the right-hand-side derivation in Figure 3 is built, so at the left-hand-side of the figure we have displayed the derivation to which the (*Term*) rule is applied, whereby the right-hand-side derivation is obtained. Thus, the application of the (*Term*) rule in the right-hand-side derivation delimits a piece of reasoning taking place at a certain hypothetical time, which is the left-hand-side derivation.

The above example argument is similar to an example given in the paper [19]. The following is a slightly reformulated version.

> In Abu Dabi alcohol is forbidden; if alcohol is forbidden Sake is forbidden, so in Abu Dabi Sake is forbidden.

Thus, the example of [19] involves spatial locations rather than times, and the shift is to a hypothetical place, namely the city of Abu Dabi.

Formally, the shift to a hypothetical point of evaluation effected by the rule (*Term*) can be seen by inspecting the proof that the rule (*Term*) is sound: The world of evaluation is shifted from the actual world to the hypothetical world where the nominal a is true (see Figure 2), then some reasoning is performed involving the delimited subderivation which by induction is assumed to be sound, and finally the world of evaluation is shifted back to the actual world. Soundness of the system $\mathbf{N}'_{\mathcal{H}}$, including soundness of the rule (*Term*), is proved in Theorem 4.1 in Section 4.3 of [8].

[4]One of the anonymous reviewers asked why the premise "if it is raining it is wet" is formalized as $p \to q$ using classical implication, rather than a form of non-monotonic implication. Like in many cases when classical logic is used to formalize natural language statements, there is an idealization in our choice of classical implication. We think this idealization is justified since our main goal is to formalize the perspective shift involved in the example argument, which we presume is orthogonal to the issue of non-monotonicity. We note in passing that our premise "if it is raining it is wet" corresponds to the premise "if alcohol is forbidden Sake is forbidden" in Seligman's example argument briefly described below, and Seligman also uses classical implication, or to be precise, machinery equivalent to classical implication, [19]. See also Footnote 6.

The rule (*Term*) is very different from other rules in proof systems for hybrid logic, roughly, this rule replaces rules for equational reasoning in other systems, see for example the rules in the natural deduction system given in Section 2.2 of the book [8].

In passing we mention that the way in which the (*Term*) rule delimits a subderivation is similar to the way subderivations are delimited by so-called boxes in linear logic, and more specifically, the way a subderivation is delimited by the introduction rule for the modal operator $\Box$ in the natural deduction system for S4 given in [5], making use of explicit substitutions in derivations.

5. THE SMARTIES TASK (TEMPORAL SHIFT VERSION)

In this section we will give a formalization which has exactly the same structure as the formalization in the previous section, but which in other respects is quite different. It turns out that a temporal shift like the one just described in the previous section also takes place in the following version of the Smarties task, where instead of a shift of perspective to another person, there is a shift of perspective to another time.[5]

> A child is shown a Smarties tube where unbeknownst to the child the Smarties have been replaced by pencils. The child is asked: "What do you think is inside the tube?" The child answers "Smarties!" The tube is then shown to contain pencils only. The child is then asked: "Before this tube was opened, what did you think was inside?"

See [14] for more on the temporal version of the Smarties task. Below we shall formalize each step in the logical reasoning taking place when giving a correct answer to the task, but before that, we give an informal analysis. Let us call the child Peter. Let a be the time where Peter answers the first question, and let t be the time where he answers the second one. To answer the second question, Peter imagines himself being at the earlier time a where he was asked the first question. At that time he deduced that there were Smarties inside the tube from the fact that it is a Smarties tube. Imagining being at the time a, Peter reasons that since he at that time deduced that there were Smarties inside, he must also have come to believe that there were Smarties inside. Therefore, at the time t he concludes that at the earlier time a he believed that there were Smarties inside.

We now proceed to the full formalization. We first extend the language of hybrid logic with two modal operators, D and B. We make use of the following symbolizations

D Peter deduces that ...
B Peter believes that ...
p There are Smarties inside the tube
a The time where the first question is asked

and we take the principle $D\phi \to B\phi$ as an axiom schema (it holds whatever proposition is substituted for the metavariable ϕ, hence an axiom schema). This is principle (9.4) in

[5]The author thanks Michiel van Lambalgen for mentioning the Smarties task in an email exchange where the author suggested that the shift of perspective in the hybrid-logical rule (*Term*) could be of relevance in connection with the theory of mind view of autism.

Figure 3: First example formalization (before and after application of the (*Term*) rule)

$$\dfrac{\dfrac{a \qquad @_a p}{p}\,(@E) \qquad \dfrac{}{p \to q}\,(Axiom)}{\dfrac{a \qquad\qquad q}{@_a q}\,(@I)}\,(\to E)$$

$$\dfrac{\dfrac{[a] \qquad [@_a p]}{p}\,(@E) \qquad \dfrac{}{p \to q}\,(Axiom)}{\dfrac{[a] \qquad\qquad q}{@_a q}\,(@I)}\,(\to E)$$

$$\dfrac{@_a p \qquad\qquad @_a q}{@_a q}\,(Term)$$

Figure 4: Formalization of the child's reasoning in the Smarties task (both temporal and person version)

$$\dfrac{\dfrac{[a] \qquad [@_a Dp]}{Dp}\,(@E) \qquad \dfrac{}{Dp \to Bp}\,(Axiom\ schema)}{\dfrac{[a] \qquad\qquad Bp}{@_a Bp}\,(@I)}\,(\to E)$$

$$\dfrac{@_a Dp \qquad\qquad @_a Bp}{@_a Bp}\,(Term)$$

[23].[6] Then the shift of temporal perspective in the Smarties task can be formalized very directly in Seligman's system as the derivation in Figure 4. Recall that the derivation is meant to formalize each step in Peters's reasoning at the time t where the second question is answered. The premise $@_a Dp$ in the derivation says that Peter at the earlier time a deduced that there were Smarties inside the tube, which he remembers at t.

Note that the formalization in Figure 4 does not involve the $\Box$ operator, so this operator could have been omitted together with the associated rules $(\Box I)$ and $(\Box E)$ in Figure 1. Since this proof system is complete, the $\Box$ operator satisfies logical omniscience. The operators D and B are only taken to satisfy the principle $D\phi \to B\phi$, as mentioned above.

Compare the derivation in Figure 4 to the right-hand-side derivation in Figure 3 in the previous section and note that the structure is exactly the same. Note that what we have done is that we have formalized the logical reasoning taking place when giving the correct answer "Smarties". Note also that the actual content of the tube, namely pencils, is not even mentioned in the formalization, so it is clear from the formalization that the actual content of the tube is not relevant to figure out the correct answer. Accordingly, our formalization does not tell what goes wrong when a child incorrectly answers "Pencils".

6. THE SMARTIES TASK (PERSON SHIFT VERSION)

As a stepping stone between the temporal version of the Smarties task we considered in the previous section, and the Sally-Anne task we shall consider in the next section, we in the present section take a look again at the version of the Smarties task described in the introduction. The only difference between the version in the introduction and the version in the previous section is the second question where

> "Before this tube was opened, what did you think was inside?"

obviously gives rise to a temporal shift of perspective, whereas

> "If your mother comes into the room and we show this tube to her, what will she think is inside?"

gives rise to a shift of perspective to another person, namely the imagined mother.

To give a correct answer to the latter of these two questions, the child Peter imagines being the mother coming into the room. Imagining being the mother, Peter reasons that the mother must deduce that there are Smarties inside the tube from the fact that it is a Smarties tube, and from that, she must also come to believe that there are Smarties inside. Therefore, Peter concludes that the mother would believe that there are Smarties inside.

The derivation formalizing this argument is exactly the same as in the temporal case dealt with in previous section, Figure 4, but the symbols are interpreted differently, namely as

D	Deduces that ...
B	Believes that ...
p	There are Smarties inside the tube
a	The imagined mother

So now nominals refer to persons rather than times. Accordingly, the modal operator B now symbolize the belief of the person represented by the point of evaluation, rather than Peter's belief at the time of evaluation, etc. Thus, the premise $@_a Dp$ in the derivation in Figure 4 says that the imagined mother deduces that there are Smarties inside the tube, which the child doing the reasoning takes to be the case since the mother is imagined to be present in the room.

[6] Analogous to the question in Footnote 4, it can be asked why we use classical implication in $D\phi \to B\phi$, rather than a form of non-monotonic implication. Again, the answer is that this is an idealization, but we presume that the perspective shift involved in the Smarties task is orthogonal to the issue of non-monotonicity, at least from a logical point of view. In this connection we remark that principle (9.4) in [23] also uses classical implication (the non-monotonicity in the logical analysis of the Smarties task of [23] does not concern principle (9.4), but other principles).

Incidentally, letting points in the Kripke model represent persons is exactly what is done in Arthur Prior's *egocentric logic*, see Section 1.3 in the book [8], in particular pp. 15–16. In egocentric logic the accessibility relation represents the taller-than relation, but this relation is obviously not relevant here.

7. THE SALLY-ANNE TASK

In this section we will give a formalization of a somewhat more complicated reasoning task called the Sally-Anne task. The following is one version.

> A child is shown a scene with two doll protagonists, Sally and Anne, having respectively a basket and a box. Sally first places a marble into her basket. Then Sally leaves the scene, and in her absence, the marble is transferred by Anne and hidden in her box. Then Sally returns, and the child is asked: "Where will Sally look for her marble?"

Most children above the age of four correctly responds where Sally must falsely believe the marble to be (in the basket) whereas younger children respond where they know the marble to be (in the box). Again, for autists, the cutoff is higher.

Below we shall formalize the correct response to the task, but before that, we give an informal analysis. Let us call the child Peter again. Let t_1 be the time where he answers the question. To answer the question, Peter imagines himself being Sally at an earlier time t_0 before she leaves the scene, but after she places the marble in her basket. Imagining being Sally, he reasons as follows: At the time t_0 Sally believes that the marble is in the box since she can see it. At the time t_1, after she has returned, she deduces that the marble is still in the box as she has no belief to the contrary, and since Sally deduces that the marble is in the box, she must also come to believe it. Therefore, Peter concludes that Sally believes that the marble is in the box.

In our formalization we make use of a tiny fragment of first-order hybrid logic, involving the unary predicate $P(t)$, the binary predicate $t < u$, and the modal operators S, D and B, but no quantifiers. We make use of the following symbolizations

$p(t)$	The marble is in the basket at the time t
$t < u$	The time t is before the time u
S	Sees that ...
D	Deduces that ...
B	Believes that ...
a	The person Sally

We also make use of the following three principles

$$S\phi \to B\phi$$
$$D\phi \to B\phi$$
$$B\phi(t) \wedge t < u \wedge \neg B\neg\phi(u) \to D\phi(u)$$

The first two are versions of principles (9.2) and (9.4) in the book [23] and the third is similar to principle (9.11) in that book. In order to make the formalization more compact, and also more in the spirit of natural deduction style, we do not take the principles as axiom schemas, but instead we turn them into the following proof-rules.

$$\frac{S\phi}{B\phi}\,(R1) \qquad \frac{D\phi}{B\phi}\,(R2) \qquad \frac{B\phi(t) \qquad t < u \qquad \neg B\neg\phi(u)}{D\phi(u)}\,(R3)$$

The second and third proof-rule together formalizes a "principle of inertia" saying that a belief is preserved over time, unless there is belief to the contrary.

We liberalize the side-condition on the (Term) rule such that the formulas $\phi_1, \ldots, \phi_n$, and ψ may include formulas on the form $t < u$, since we assume that the truth-values of such formulas are not changed by the perspective shift effected by the rule.

With this machinery in place, the shift of person perspective in the Sally-Anne task can be formalized as the derivation in Figure 5. Recall that this derivation is meant to formalize the child's reasoning at the time t_1 where the question is answered. The first premise $@_a Sp(t_0)$ in the derivation says that Sally (the reference the nominal a) at the earlier time t_0 saw that the marble was in the basket, which the child remembers. The third premise $@_a \neg B\neg p(t_1)$ says that Sally at the time t_1 does not believe that the marble is not in the basket, which the child realizes as Sally was absent when the marble was transferred to the box.

Note that the actual position of the marble at the time t_1 is irrelevant to figure out the correct response. Note that in the Sally-Anne task there is a shift of person perspective which we deal with in a modal-logical fashion letting points of evaluation stand for persons, like in the person version of the Smarties task in the previous section, but there is also a temporal shift in the Sally-Anne task, from the time t_0 to the time t_1, which we deal with using first-order machinery.

8. DISCUSSION

In the introduction of the present paper we remarked that reasoning in Seligman's system is different from reasoning in the most common proof systems for hybrid logic, and that reasoning in Seligman's system captures well the reasoning in the Smarties and Sally-Anne tasks, in particular the involved shift between different local perspectives.

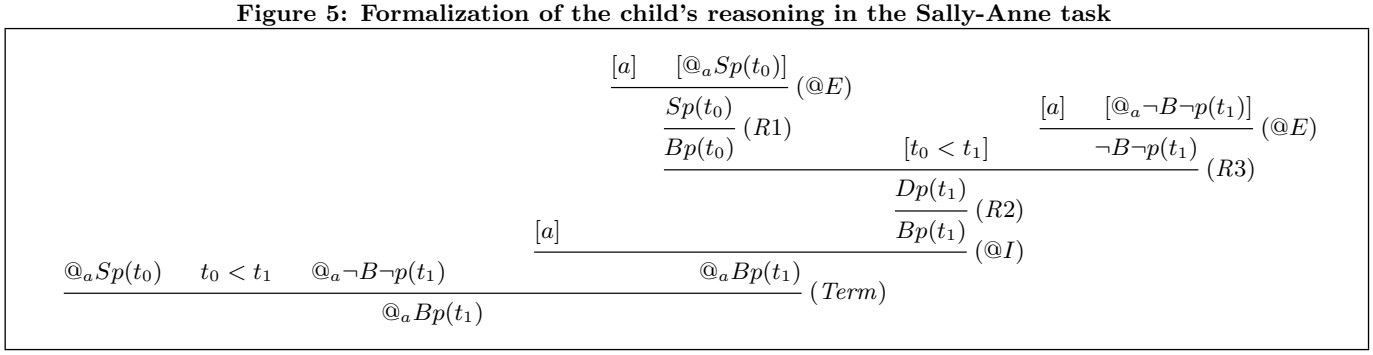
More can be said about this difference between the proof systems and how local perspectives are (or are not) represented. A truth-bearer is an entity that is either true or false. According to Peter Simons' paper [22], there have historically been two fundamentally opposed views of how truth-bearers have their truth-values.

> One view takes truth to be absolute: a truth-bearer's truth-value (whether truth or falsity) is something it has *simpliciter*, without variation according to place, time, by whom and to whom it is said. The other view allows a truth-bearer's truth-value to vary according to circumstances: typically time or place, but also other factors may be relevant. ([22], p. 443)

Peter Simons calls the first view the *absolute* view and the second the *centred* view. It is well-known that Arthur Prior often expressed sympathy for what is here called the centred view, most outspoken with respect to time, one reason being that he wanted to allow statements to change truth-value from one time to another. What a truth-bearer's truth-value varies according to, is by Simons called a *location*.

> I understand 'location' broadly to include not just spatial location but also temporal location, spatiotemporal location, modal location, and more broadly still location in any relational structure. I consider that the concept of an object being

Figure 5: Formalization of the child's reasoning in the Sally-Anne task



> located at a position among other positions is a formal concept, applicable topic-neutrally in any field of discourse. This means that logical considerations about location are not limited in extent or parochial in interest. ([22], p. 444)

The proposition expressed in the quotation above is defended in Simons' paper [21]. See also the paper [20]. Obviously, a frame for modal and hybrid logic is a mathematically precise formulation of Simons' concept of a location, see Definition 2.1.

What does all this have to do with proof systems for hybrid logic? The distinction between the absolute view and the centred view is useful for describing proof systems and the formulas that occur in them. The basic building blocks of the most common proof systems for hybrid logic are satisfaction statements, and satisfaction statements have constant truth-values, so the basic building blocks of such systems are absolute, although it is arguable that such systems have both absolute and centred features since arbitrary subformulas of satisfaction statements do have varying truth-values, and therefore have to be evaluated for truth at some location. On the other hand, the basic building blocks of Seligman's system are arbitrary formulas, and arbitrary formulas have varying truth-values, so this system is centred, involving local perspectives in the reasoning.

9. SOME REMARKS ON OTHER WORK

Beside analysing the reasoning taking place when giving a correct answer to a reasoning task, the works by van Lambalgen and co-authors also analyse what goes wrong when an incorrect answer is given. We note that Stenning and van Lambalgen in [23] warn against simply characterizing autism as a lack of theory of mind. Rather than being an explanation of autism, Stenning and van Lambalgen see the theory of mind deficit hypothesis as "an important label for a problem that needs a label", cf. [23], p. 243. Based on their logical analysis, they argue that another psychological theory of autism is more fundamental, namely what is called the *executive function deficit theory*. Very briefly, executive function is an ability to plan and control a sequence of actions with the aim of obtaining a goal in different circumstances.

The paper [15] reports empirical investigations of closed-world reasoning in adults with autism. Incidentally, according to the opening sentence of that paper, published in 2009, "While autism is one of the most intensively researched psychiatric disorders, little is known about reasoning skills of people with autism."

With motivations from the theory of mind literature, the paper [25] models examples of beliefs that agents may have about other agents' beliefs (one example is an autistic agent that always believes that other agents have the same beliefs as the agent's own). This is modelled by different agents preference relations between states, where an agent prefers one state over another if the agent considers it more likely. The beliefs in question turn out to be frame-characterizable by formulas of epistemic logic.

The paper [10] reports empirical investigations of what is called *second-order* theory of mind, which is a person's capacity to imagine other people's beliefs about the person's own beliefs (where *first-order* theory of mind is what we previously in the present paper just have called theory of mind). The investigations in [10] make use of a second-order false-belief task, as well as other tasks.

The paper [12] does not deal with false-belief tasks or theory of mind, but it is nevertheless relevant to mention since it uses formal proofs to compare the cognitive difficulty of deductive tasks. To be more precise, the paper associates the difficulty of a deductive task in a version of the Mastermind game with the minimal size of a corresponding tableau tree, and it uses this measure of difficulty to predict the empirical difficulty of game-plays, for example the number of steps actually needed for solving a task.

The method of reasoning in tableau systems can be seen as attempts to construct a model of a formula: A tableau tree is built step by step using rules, whereby more and more information about models for the formula is obtained, and either at some stage a model can be read off from the tableau tree, or it can be concluded that there cannot be such a model (in fact, in the case of [12], any formula under consideration has exactly one model, so in that case it is a matter of building a tableau tree that generates this model). Hence, if the building of tableau trees is taken to be the underlying mechanism when a human is solving Mastermind tasks, then the investigations in [12] can be seen to be in line with the mental models school (see the third section of the present paper).

A remark from a more formal point of view: The tableau system described in [12] does not include the cut-rule[7]. Much has been written on the size of proofs in cut-free proof systems, in particular, the paper [6] gives examples of first-

[7]The cut-rule says that the end of any branch in a tableau tree can extended with two branches with ϕ on the one branch and $\neg\phi$ on the other (expressing the bivalence of classical logic).

order formulas whose derivations in cut-free systems are much larger than their derivations in natural deduction systems, which implicitly allow unrestricted cuts (in one case more than 10^{38} characters compared to less than 3280 characters). Similarly, the paper [9] points out that ordinary cut-free tableau systems have a number of anomalies, one of them being that for some classes of propositional formulas, decision procedures based on cut-free systems are much slower than the truth-table method (in the technical sense that there is no polynomial time computable function that maps truth-table proofs of such formulas to proofs of the same formulas in cut-free tableau systems). Instead of prohibiting cuts completely, the paper [9] advocates allowing a restricted version of the cut-rule, called the analytic cut-rule.

10. FUTURE WORK

We would like to extend the work of the present paper to further false-belief tasks, perhaps using different hybrid-logical machinery (and moreover, to see if we can also use hybrid-logical proof-theory to analyse what goes wrong when incorrect answers are given). Not only will formalization of further reasoning tasks be of interest on their own, but we also expect that such investigations can be feed back into logical research, either as corroboration of the applicability of existing logical constructs, or in the form of new logical constructs, for example new proof-rules or new ways to add expressive power to a logic.

We are also interested in further investigations in when two seemingly dissimilar reasoning tasks have the same underlying logical structure, like we in the present paper have disclosed that two different versions of the Smarties task have exactly the same underlying logical structure. Such investigations might be assisted by a notion of identity on proofs (exploiting the longstanding effort in proof-theory to give a notion of identity between proofs, that is, a way to determine if two arguments have common logical structure, despite superficial dissimilarity).

More speculatively, we expect that our formalizations can contribute to the ongoing debate between two dominating views on theory of mind, denoted *theory-theory* and *simulation-theory*. According to theory-theory, theory of mind should be viewed as an explicit theory of the mental realm of another person, like the theories of the physical world usually going under the heading "naive physics", whereas according to simulation-theory, theory of mind should be viewed as a capacity to put yourself in another person's shoes, and simulate the person's mental states.

11. ACKNOWLEDGEMENTS

The author thanks Thomas Bolander for comments on an early version of this paper. Also thanks to Jerry Seligman for a discussion of the paper. The author acknowledges the financial support received from The Danish Natural Science Research Council as funding for the project HYLO-CORE (Hybrid Logic, Computation, and Reasoning Methods, 2009–2013).

12. REFERENCES

[1] C. Areces and B. ten Cate. Hybrid logics. In P. Blackburn, J. van Benthem, and F. Wolter, editors, *Handbook of Modal Logic*, pages 821–868. Elsevier, 2007.

[2] M. Baaz and A. Leitsch. *Methods of Cut-Elimination*, volume 34 of *Trends in Logic Series*. Springer, 2011.

[3] S. Baron-Cohen. *Mindblindness: An Essay on Autism and Theory of Mind*. MIT Press, 1995.

[4] S. Baron-Cohen, A. Leslie, and U. Frith. Does the autistic child have a 'theory of mind'? *Cognition*, 21:37–46, 1985.

[5] G. Bierman and V. de Paiva. On an intuitionistic modal logic. *Studia Logica*, 65:383–416, 2000.

[6] G. Boolos. Don't eliminate cut. *Journal of Philosophical Logic*, 13:373–378, 1984.

[7] T. Braüner. Two natural deduction systems for hybrid logic: A comparison. *Journal of Logic, Language and Information*, 13:1–23, 2004.

[8] T. Braüner. *Hybrid Logic and its Proof-Theory*, volume 37 of *Applied Logic Series*. Springer, 2011.

[9] M. D'Agostino and M. Mondadori. The taming of the cut. Classical refutations with analytical cut. *Journal of Logic and Computation*, 4:285–319, 1994.

[10] L. Flobbe, R. Verbrugge, P. Hendriks, and I. Krämer. Children's application of theory of mind in reasoning and language. *Journal of Logic, Language and Information*, 17:417–442, 2008.

[11] G. Gentzen. Investigations into logical deduction. In M. Szabo, editor, *The Collected Papers of Gerhard Gentzen*, pages 68–131. North-Holland Publishing Company, 1969.

[12] N. Gierasimczuk, H. van der Maas, and M. Raijmakers. Logical and psychological analysis of Deductive Mastermind. In *Proceedings of the Logic & Cognition Workshop at ESSLLI 2012, Opole, Poland, 13–17 August, 2012*, volume 883 of *CEUR Workshop Proceedings*, pages 21–39. CEUR-WS.org, 2012.

[13] W. Goldfarb. *Deductive Logic*. Hackett Pub. Co., 2003.

[14] A. Gopnik and J. Astington. Children's understanding of representational change and its relation to the understanding of false belief and the appearance-reality distinction. *Child Development*, 59:26–37, 1988.

[15] J. Pijnacker, B. Geurts, M. van Lambalgen, C. Kan, J. Buitelaar, and P. Hagoort. Defeasible reasoning in high-functioning adults with autism: Evidence for impaired exception-handling. *Neuropsychologia*, 47:644–651, 2009.

[16] D. Prawitz. *Natural Deduction. A Proof-Theoretical Study*. Almqvist and Wiksell, Stockholm, 1965.

[17] D. Prawitz. Ideas and results in proof theory. In J. E. Fenstad, editor, *Proceedings of the Second Scandinavian Logic Symposium*, volume 63 of *Studies in Logic and The Foundations of Mathematics*, pages 235–307. North-Holland, 1971.

[18] L. Rips. *The Psychology of Proof: Deductive Reasoning in Human Thinking*. MIT Press, 1994.

[19] J. Seligman. The logic of correct description. In M. de Rijke, editor, *Advances in Intensional Logic*, volume 7 of *Applied Logic Series*, pages 107 – 135. Kluwer, 1997.

[20] P. Simons. Absolute truth in a changing world. In *Philosophy and Logic. In Search of the Polish Tradition. Essays in Honour of Jan Wolenski on the Occasion of his 60th Birthday*, volume 323 of *Synthese*

Library, pages 37–54. Kluwer, 2003.
[21] P. Simons. Location. *Dialectica*, 58:341–347, 2004.
[22] P. Simons. The logic of location. *Synthese*, 150:443–458, 2006. Special issue edited by T. Braüner, P. Hasle, and P. Øhrstrøm.
[23] K. Stenning and M. van Lambalgen. *Human Reasoning and Cognitive Science*. MIT Press, 2008.
[24] A. Troelstra and H. Schwichtenberg. *Basic Proof Theory*, volume 43 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, 1996.
[25] H. van Ditmarsch and W. Labuschagne. My beliefs about your beliefs – a case study in theory of mind and epistemic logic. *Synthese*, 155:191–209, 2007.

APPENDIX

A. PROOF OF ANALYTICITY

Usually, when considering a natural deduction system, one wants to equip it with a normalizing set of reduction rules such that normal derivations satisfy the subformula property. Normalization says that any derivation by repeated applications of reduction rules can be rewritten to a derivation which is normal, that is, no reduction rules apply. From this it follows that the system under consideration is analytic.

Now, the works [7] and Section 4.3 by the present author devise a set of reduction rules for $\mathbf{N}'_{\mathcal{H}}$ obtained by translation of a set of reduction rules for a more common natural deduction system for hybrid logic. This more common system, which we denote $\mathbf{N}_{\mathcal{H}}$, can be found in [7] and in [8], Section 2.2. All formulas in the system $\mathbf{N}_{\mathcal{H}}$ are satisfaction statements. Despite other desirable features, it is not known whether the reduction rules for $\mathbf{N}'_{\mathcal{H}}$ are normalizing, and normal derivations do not always satisfy the subformula property. In fact, Chapter 4 of the book [8] ends somewhat pessimistically by exhibiting a normal derivation without the subformula property. It is remarked that a remedy would be to find a more complete set of reduction rules, but the counter-example does not give a clue how such a set of reduction rules should look.

In what follows we shall take another route. We prove a completeness result saying that any valid formula has a derivation in $\mathbf{N}'_{\mathcal{H}}$ satisfying a version of the subformula property. This is a sharpened version of a completeness result for $\mathbf{N}'_{\mathcal{H}}$ originally given in [7] and in Section 4.3 of [8] (Theorem 4.1 in [8]). Thus, we prove that $\mathbf{N}'_{\mathcal{H}}$ is analytic without going via a normalization result. So the proof of the completeness result does not involve reduction rules. The result is mathematically weaker than normalization together with the subformula property for normal derivations, but it nevertheless demonstrates analyticity. Analyticity is a major success criteria in proof-theory, one reason being that analytic provability is a step towards automated theorem proving (which obviously is related to Leibniz' aim mentioned in the intoduction of the present paper).

In the proof below we shall refer to $\mathbf{N}_{\mathcal{H}}$ as well as a translation $(\cdot)^\circ$ from $\mathbf{N}_{\mathcal{H}}$ to $\mathbf{N}'_{\mathcal{H}}$ given in [7] and Section 4.3 of [8]. This translates a derivation π in $\mathbf{N}_{\mathcal{H}}$ to a derivation π° in $\mathbf{N}'_{\mathcal{H}}$ having the same end-formula and parcels of undischarged assumptions. The reader wanting to follow the details of our proof is advised to obtain a copy of the paper [7] or the book [8]. The translation $(\cdot)^\circ$ satisfies the following.

LEMMA A.1. *Let π be a derivation in $\mathbf{N}_{\mathcal{H}}$. Any formula θ occuring in π° has at least one of the following properties.*

1. *θ occurs in π.*
2. *$@_a\theta$ occurs in π for some satisfaction operator $@_a$.*
3. *θ is a nominal a such that some formula $@_a\psi$ occurs in π.*

PROOF. Induction on the structure of the derivation of π. Each case in the translation $(\cdot)^\circ$ is checked. □

Note that in item 1 of the lemma above, the formula θ must be a satisfaction statement since only satisfaction statements occur in π. In what follows $@_d\Gamma$ denotes the set of formulas $\{@_d\xi \mid \xi \in \Gamma\}$.

THEOREM A.2. *Let π be a normal derivation of $@_d\phi$ from $@_d\Gamma$ in $\mathbf{N}_{\mathcal{H}}$. Any formula θ occuring in π° has at least one of the following properties.*

1. *θ is of the form $@_a\psi$ such that ψ is a subformula of ϕ, some formula in Γ, or some formula of the form c or $\Diamond c$.*
2. *θ is a subformula of ϕ, some formula in Γ, or some formula of the form c or $\Diamond c$.*
3. *θ is a nominal.*
4. *θ is of the form $@_a(p \to \bot)$ or $p \to \bot$ where p is a subformula of ϕ or some formula in Γ.*
5. *θ is of the form $@_a\bot$ or $\bot$.*

PROOF. Follows from Lemma A.1 above together with Theorem 2.4 (called the quasi-subformula property) in Subsection 2.2.5 of [8]. □

We are now ready to give our main result, which is a sharpened version of the completeness result given in Theorem 4.1 in Section 4.3 of [8].

THEOREM A.3. *The first statement below implies the second statement. Let ϕ be a formula and Γ a set of formulas.*

1. *For any model $\mathcal{M}$, any world w, and any assignment g, if, for any formula $\xi \in \Gamma$, $\mathcal{M}, g, w \models \xi$, then $\mathcal{M}, g, w \models \phi$.*
2. *There exists of derivation of ϕ from Γ in $\mathbf{N}'_{\mathcal{H}}$ such that any formula θ occuring in the derivation has at least one of the five properties listed in Theorem A.2.*

PROOF. Let d be a new nominal. It follows that for any model $\mathcal{M}$ and any assignment g, if, for any formula $@_d\xi \in @_d\Gamma$, $\mathcal{M}, g \models @_d\xi$, then $\mathcal{M}, g \models @_d\phi$. By completeness of the system $\mathbf{N}_{\mathcal{H}}$, Theorem 2.2 in Subsection 2.2.3 of the book [7], there exists a derivation π of $@_d\phi$ from $@_d\Gamma$ in $\mathbf{N}_{\mathcal{H}}$. By normalization, Theorem 2.3 in Subsection 2.2.5 of the book, we can assume that π is normal. We now apply the rules $(@I)$, $(@E)$, and $(Name)$ to π° obtaining a derivation of ϕ from Γ in $\mathbf{N}'_{\mathcal{H}}$ satisfying at least one of the properties mentioned in Theorem A.2. □

Remark: If the formula occurrence θ mentioned in the theorem above is not of one of the forms covered by item 4 in Theorem A.2, and does not have one of a finite number of very simple forms not involving propositional symbols, then either θ is a subformula of ϕ or some formula in Γ, or θ is of the form $@_a\psi$ such that ψ is a subformula of ϕ or some formula in Γ. This is the version of the subformula property we intended to prove.

Strategic voting and the logic of knowledge

Hans van Ditmarsch
LORIA, Université de Lorraine
hvd@us.es

Jérôme Lang
LAMSADE, Université Paris Dauphine
lang@irit.fr

Abdallah Saffidine
LAMSADE, Université Paris Dauphine
abdallah.saffidine@gmail.com

ABSTRACT

We propose a general framework for strategic voting when a voter may lack knowledge about other votes or about other voters' knowledge about her own vote. In this setting we define notions of manipulation and equilibrium. We also model action changing knowledge about votes, such as a voter revealing its preference or as a central authority performing a voting poll. Some forms of manipulation are preserved under such updates and others not. Another form of knowledge dynamics is the effect of a voter declaring its vote. We envisage Stackelberg games for uncertain profiles. The purpose of this investigation is to provide the epistemic background for the analysis and design of voting rules that incorporate uncertainty.

Keywords

social choice, voting, epistemic logic, dynamics

1. INTRODUCTION

A well-known fact in social choice theory is that strategic voting, also known as manipulation, becomes harder when voters know less about the preferences of other voters. Standard approaches to manipulation in social choice theory [13, 24] as well as in computational social choice [5] assume that the manipulating voter knows perfectly how the other voters will vote. Some approaches [11, 4] assume that voters have a probabilistic prior belief on the outcome of the vote, which encompasses the case where each voter has a probability distribution over the set of profiles. A recent paper [9] extends coalitional manipulation to incomplete knowledge, by distinguishing manipulating from non-manipulating voters and by considering that the manipulating coalition has, for each voter outside the coalition, a set of possible votes encoded in the form of a partial order over candidates. Still, we think that the study of strategic voting under complex belief states has received little attention so far, especially when voters are uncertain about the uncertainties of other voters, i.e., when we model higher-order beliefs of voters.

An extreme case of uncertainty is when a voter is completely ignorant about other votes. In that case, if a manipulation under incomplete knowledge is defined in a pessimistic way, i.e., if it is said to be successful if it succeeds for all possible votes of other voters, voting rules may well be non-manipulable. For the special case where all other voters are non-strategic this is shown for most common voting rules in [9].

In the first place we model how uncertainty about the preferences of other voters may determine a strategic vote, and how a reduction in this uncertainty may change a strategic vote. We restrict ourselves to the case where uncertainty is over a number of well-described alternatives, including the true state of affairs, between which the voter is unable to distinguish.

We also investigate the dynamics of uncertainty. The uncertainty reduction may be due to receiving information on voting intentions in polls or to voters directly telling you their preference. For simplicity we assume that received information is correct, or rather, we only model the consequences of incorporating new information after the decision to consider the information reliable. Such informative actions can then be modelled as truthful public announcements [23].

Another form of dynamics is the dynamics of declaring votes. Declaring votes can be modeled as assignments (ontic / factual change). Just as there may be uncertainty about truthful votes, there may also be uncertainty about declared votes. Consider the following. Half of the votes are declared. It is not known whether candidate x or y has taken the lead, but z has clearly lost. You still have to vote. Does this influence your strategy? Another example is that of *safe manipulation* [25], where the manipulating voter announces her vote to a (presumably large) set of voters sharing her preferences but is unsure of how many will follow her. Finally, consider Stackelberg voting games, wherein voters declare their votes in sequence, following a fixed, exogeneously defined order. Our framework applies to Stackelberg voting games with uncertainty about profiles.

There are several ways of expressing incomplete knowledge about the linear order of a voter. The literature on possible and necessary winners assumes that it is expressed by a collection of partial strict orders (one for each voter), while Hazon *et al.* [15] consider it to consist of a collection of probability distributions, or a collection of sets of linear orders (one for each voter). Whereas the latter is more expressive (some sets of linear orders do not correspond to the set of extensions of a partial order), the former is more succinct. Ours is a more expressive modelling than both modes of representation, because an uncertain profile can be any set of profiles. A set of profiles such as $\{(a \succ_1 b \succ_1 c, a \succ_2 b \succ_2 c), (b \succ_1 a \succ_1 c, b \succ_2 a \succ_2 c)\}$ expresses uncertainty (ignorance) which candidate voters 1 and 2 rank first, but knowledge (certainty) that voters 1 and

TARK 2013, Chennai, India.

2 have identical preferences — which is not possible in [15], and *a fortiori* also not in [17] and subsequent works on the possible winner problem. Of course, this mode of representation is also the less succinct of all. However, succinctness and complexity issues will play no role yet in this paper, where we focus on modelling and expressivity.

Somewhat surprisingly, there are yet more complex scenarios that cannot be seen as uncertainty between a number of given profiles: it may be that a voter cannot distinguish between two situations with identical profiles, because in the first case yet another voter has some uncertainty about the profile, but in the other case not.

Our investigation is restricted in various ways: (i) we model uncertainty and manipulability of individuals but not of coalitions, (ii) we model knowledge but not belief, and, in the dynamics, truthful announcements but not lying, (iii) we model incomplete knowledge (uncertainty) but not other forms of incompleteness, and (iv) as already said, we have not investigated complexity and succinctness. The reason for these restrictions is our desire to, first, present this complete logical framework for voters uncertain about profiles. Later we wish to broaden our scope. Let us briefly comment on these issues here.

Epistemic and voting notions for coalitions are treated in Section 8 in some detail.

There are many scenarios wherein voters may have incorrect beliefs about preferences, or where information changing actions are intended to deceive. I may incorrectly believe that you prefer a over b, whereas you really prefer b over a. I may tell you that I prefer a over b, but I may be lying. Such scenarios can also be modelled in epistemic logic, with the same tools and techniques as presented in this paper, but we have restricted ourselves to knowledge: reliable beliefs. This is already a far and high enough jump from the typical social choice theory perspective of reliable common knowledge of preferences, and we think that the variety of phenomena described within the restriction of knowledge and reliable information already sufficiently demonstrate the expressive power of the extension of voting with uncertainty.

The study of uncertain votes is different from the study of other forms of incompleteness, e.g., when the number of voters or candidates may be unknown — the only form of incompleteness that we model is incomplete knowledge in the form of inability to determine which of a number of well-defined alternatives is the case. Here, we also restrict ourselves.

Complexity issues will be occassionally referred to in running text and in the concluding Section 9.

A link between epistemic logic and voting has first been given, as far as we know, in [8]—they use knowledge graphs to indicate that a voter is uncertain about the preference of another voter. A more recent approach, within the area known as social software, is [21]. The recent [9] walks a middle way namely where equivalence classes are called information sets, as in treatments of knowledge and uncertainty in economics, but where the uncertain voter does not take the uncertainty of other voters into account.

2. VOTING

This section recalls standard voting terminology.

Assume a finite set $\mathcal{N} = \{1, \ldots, n\}$ of n *voters* (or *agents*), and a finite set $\mathcal{C} = \{a, b, c, \ldots\}$ of m *candidates* (or *alternatives*). Voter variables are i and j, and candidate variables are x and y (and $x_1, x_2, ...$).

Definition 1 (Vote) *For each voter i a* vote $\succ_i \subseteq \mathcal{C} \times \mathcal{C}$ *is a linear order on $\mathcal{C}$.*

If voter i prefers candidate a to candidate b in vote $\succ_i$, we write $a \succ_i b$. Vote variables are $\succ_i$, $\succ'_i$, etc. Instead of $x_1 \succ_i \cdots \succ_i x_n$ we also write $i : x_1 \ldots x_n$, or depict it vertically in a table.

Definition 2 (Profile) *A* profile P *is a collection* $\{\succ_1, \ldots, \succ_n\}$ *of n votes.*

Let $O(\mathcal{C})$ be the set of linear orders of $\mathcal{C}$. Then $O(\mathcal{C})^n$ is the set of all profiles for $\mathcal{N}$. Profile variables are $P, P', ...$ If $P \in O(\mathcal{C})^n$, $\succ_i \in P$, and $\succ'_i \in O(\mathcal{C})$, then $P[\succ_i/\succ'_i]$ is the profile wherein $\succ_i$ is substituted by $\succ'_i$ in P.

Definition 3 (Voting rule) *A* voting rule *is a function $F : O(\mathcal{C})^n \to \mathcal{C}$ from the set of profiles to the set of candidates.*

The voting rule determines which candidate wins the election — $F(P)$ is the *winner*. A *voting correspondence* $C : O(\mathcal{C})^n \to 2^{\mathcal{C}} \setminus \{\emptyset\}$ maps a profile to a nonempty set of *tied cowinners*. To obtain a voting rule from a voting correspondence (to obtain a unique winner from a non-empty set of cowinners) we assume an exogeneously specified *tie-breaking mechanism*, that is a total order $\succ$ over candidates.

Voters cannot be assumed to vote according to their preferences. Relative to a given profile P, a vote $\succ_i \in P$ can be called the *truthful vote* or *preference*. A voter may change her truthful vote if this improves the outcome of the voting. This is called a *manipulation* or *strategic vote*.

Definition 4 (Manipulation) *Let $i \in \mathcal{N}$, $P \in O(\mathcal{C})^n$ and $\succ_i \in P$, and let $\succ'_i \in O(\mathcal{C})$. If $F(P[\succ_i/\succ'_i]) \succ_i F(P)$, then $\succ'_i$ is a* successful manipulation *by voter i.*

Of course some votes that are not truthful still do not improve the outcome — relative to the truthful vote $\succ_i \in P$, any $\succ'_i \in O(\mathcal{C})$ can be called a *possible vote*. Finally, there is the case of the *declared vote*, after which a voter can no longer change her vote. Information on declared votes may be available to other voters (such as in Stackelberg games), and that may change their subsequent strategic votes. This is an overview of different votes.

- truthful vote / preference
- strategic vote / successful manipulation
- possible vote
- declared vote

We now define stable outcomes of the voting rule. The combination of a profile P and a voting rule F defines a strategic game: a player is a voter, an individual strategy for a player is a vote (an individual strategy for a player in the game theoretical sense may not be a strategic vote in the social choice theoretical sense), a strategy profile (of players) is therefore a profile in our defined sense (of voters), and the preference of a player among the outcomes is according to his preferred vote: given voter i with truthful vote $\succ_i \in P$, and profiles P', P'', i prefers outcome $F(P')$ over outcome $F(P'')$ in the game theoretical sense iff $F(P') \succ_i F(P'')$. The relevant equilibrium notion is:

Definition 5 (Equilibrium profile) *Given a profile P, a profile P' is an* equilibrium profile *iff no agent has a successful manipulation.*

In the view of a voting process as a game, an equilibrium profile corresponds to a Nash equilibrium.

Manipulation and equilibrium for coalitions will be addressed in Section 8, later.

3. KNOWLEDGE PROFILES

We model uncertainty about voting in the sense of incomplete knowledge about votes. The terminology to describe such uncertainty that we introduce in this section is fairly standard in modal logic [12], but its application to social choice theory is novel. The novelty consists in taking models with *profiles* instead of *valuations of propositional variables*. An expression like $b \succ_i a$ is a proposition 'voter i prefers candidate b over candidate a', which is true or false for any given profile; and from that perspective, a profile is nothing but a collection where for all voters all such variables are given a value true or false: a valuation.

Definition 6 (Knowledge profile) *Given is the set $O(\mathcal{C})^n$ of all profiles for a set $\mathcal{N} = \{1, \ldots, n\}$ of n voters. A* profile model *is a structure $\mathcal{P} = (S, \{\sim_1, \ldots, \sim_n\}, \pi)$, where S is a* domain *of abstract objects called states; where for $i = 1, \ldots, n$, $\sim_i$ is an* indistinguishability relation *that is an equivalence relation; and where* valuation $\pi : S \to O(\mathcal{C})^n$ *assigns a profile to each state. A* knowledge profile *is pointed structure $\mathcal{P}_s$ where $\mathcal{P}$ is a profile model and s is a state in the domain of $\mathcal{P}$.*

If $s \sim_i s'$, $\pi(s) = P$, and $\pi(s') = P'$, then voter i is uncertain if the profile is P or P'; e.g. if $j : bca$ in P and $j : cba$ in P', then voter i is uncertain if voter j prefers b over c or c over b. Instead of 'voter i is uncertain if' we also say 'voter i does not know that'. We can do this formally in a logical language interpreted on knowledge profiles.

Definition 7 (Logical language) *The language $\mathcal{L}$ over the set of voters $\mathcal{N} = \{1, \ldots, n\}$ and the set of preferences is defined as follows, where i is an agent and $a, b \in \mathcal{C}$:*

$$\varphi ::= a \succ_i b \mid \neg\varphi \mid \varphi \wedge \varphi \mid K_i\varphi$$

A profile P is defined in $\mathcal{L}$ by abbreviation as the description of the valuation (the conjunction of all its terms $a \succ_i b$ and all its excluded terms $\neg(a \succ_i b)$). Similarly, a vote $\succ_i$ is defined in $\mathcal{L}$ by abbreviation as the i-part of that.

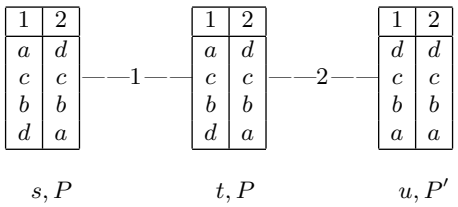
An element of the language is called a formula, φ is a formula variable. Formula $K_i\varphi$ stands for 'voter i knows that φ'. We have allowed ourselves to overload the meaning of $a \succ_i b$, as it is really the name for the atomic proposition uniquely interpreted (below) as the truth of $a \succ_i b$.

Definition 8 (Semantics) *The interpretation of formulas in a knowledge profile is defined as follows:*

$$\begin{array}{lll} \mathcal{P}_s \models a \succ_i b & \text{iff} & a \succ_i b, \text{ where } \succ_i \in \pi(s) \\ \mathcal{P}_s \models \neg\varphi & \text{iff} & \mathcal{P}_s \not\models \varphi \\ \mathcal{P}_s \models \varphi \wedge \psi & \text{iff} & \mathcal{P}_s \models \varphi \text{ and } \mathcal{P}_s \models \psi \\ \mathcal{P}_s \models K_i\varphi & \text{iff} & \text{for every } t \text{ such that } s \sim_i t, \mathcal{P}_t \models \varphi \end{array}$$

Given a knowledge profile $\mathcal{P}_s$ and a proposition φ, *agent i knows that φ* if and only if φ holds for all states in $\mathcal{P}$ indistinguishable for i from s (i.e., for all $s' \in \mathcal{P}$ such that $s \sim_i s'$). The expression $\mathcal{P}_s \not\models \varphi$ stands for 'It is not the case that $\mathcal{P}_s \models \varphi$'. If $\mathcal{P}_s \models \varphi$ for all $s \in S$, we write $\mathcal{P} \models \varphi$ (φ is valid on $\mathcal{P}$) and if this is the case for all $\mathcal{P}$, we say that φ is valid, and we write $\models \varphi$. Propositions like 'voter i knows the profile' now have a precise description.

Example 1 *Consider the following $\mathcal{P}$ consisting of three states s, t, u and for two voters 1 and 2. State s is assigned to profile P, wherein $a \succ_1 c \succ_1 b \succ_1 d$ and $d \succ_2 c \succ_2 b \succ_2 a$, etc. States that are indistinguishable for a voter i are linked with an i-labelled edge. The partition for 1 on the domain is therefore $\{\{s, t\}, \{u\}\}$, and the partition for 2 on the domain is $\{\{s\}, \{t, u\}\}$.*



States s and t have been assigned the same profile P but have different epistemic properties. In s, 2 knows that 1 prefers a over d, whereas in t 2 does not know that. We list some such relevant formulas:

- $\mathcal{P}_s \models K_2\ a \succ_1 d$
- $\mathcal{P}_t \not\models K_2\ a \succ_1 d$
- $\mathcal{P} \models (\succ_1 \to K_1 \succ_1) \wedge (\succ_2 \to K_2 \succ_2)$
  (Both voters know there preference.)

The example demonstrates than we cannot do away with states. Sometime, different states are being assigned the same profile. But in many typical scenarios different states are assigned different profiles, and then we can truly say that the uncertainty of a voter is about a collection of profiles.

We now define the notion of 'voter i changes her vote' in $\mathcal{L}$.

Definition 9 (Changing a vote) *We define $P \leftrightarrow_i P'$ as*

$$P \to \succ_i \wedge P' \to \succ_i' \wedge \bigvee_{j \neq i, a, b \in \mathcal{C}} (a \succ_j b \leftrightarrow a \succ_j' b)$$

Given the abbreviations defined, $P \to \succ_i$ stands for $\succ_i \in P$. Formula $P \to \succ_i$ says that there is a vote $\succ_i'$ such that $P' = P[\succ_i / \succ_i']$.

Surprisingly, our logic of knowledge and voter preferences, that we extend with dynamics in the next sections, is not in fact a dynamic logic of preference [19]. Given that, the following perspective may be of interest. In our models, the preferences are modelled as propositional variables. These induce preferences between states by enriching the model with total orders expressing that: one state is more preferred than another one, if the outcome of the truthful vote for the profile of the first state is more preferred than the outcome of the vote for the profile of the second state.

Definition 10 (Models for knowledge and preference) *Given a knowledge profile $\mathcal{P}_s$ with $\mathcal{P} = (S, \{\sim_1, \ldots, \sim_n\}, \pi)$ the induced* preference knowledge profile *$\mathcal{P}_s^{\succ}$ is defined as $\mathcal{P}^{\succ} = (S, \{\sim_1, \ldots, \sim_n\}, \{\succ_1, \ldots, \succ_n\}, \pi)$ where $\succ_i$ is defined as: for all $s, t \in S$, $s \succ_i t$ iff $F(\pi(s)) \succ_i F(\pi(t))$.*

Thus we reclaim the epistemic plausibility models of [3] (and therefore, indirectly, approaches as [19]), although not in the meaning of 'agent i considers state s more plausible than state t', but in the sense of 'voter i prefer the outcome of voting of the profile in s to the outcome of voting of the profile in t'. As there, one has a choice between global preferences or 'local' preferences (intersection of global preferences with equivalence classes). This embedding seems important enough to mention as a result:

Proposition 1 *The epistemic logic of votes can be embedded into epistemic plausibility logic.*

PROOF. We refer to the embedding of Definition 10.

4. MANIPULATION AND KNOWLEDGE

In a knowledge profile it may be that a voter can manipulate the vote but does not know that, because she considers it possible that another profile is the case in which she cannot manipulate the vote. Such situations call for more refined notions of manipulation that also involve knowledge. They can be borrowed from the knowledge and action literature [26, 16].

Given is a knowledge profile $\mathcal{P}_s$ where $\pi(s) = P$. If voter i can manipulate P, then voter i also can manipulate $\mathcal{P}_s$. The uncertainty is about what the profile is. But this does not affect that P is the actual profile.

In our modelling, if the voter can manipulate P, she always considers it possible that she can manipulate P. This is a consequence of modelling uncertain knowledge instead of uncertain belief. However, there are situations wherein she considers it possible that she can manipulate, but where in fact she cannot manipulate, namely if she considers a state possible with a profile that is not the profile in the actual state.

A curious situation is the one wherein in all states that the voter considers possible there is a successful manipulation, but where, unfortunately, this is not the same strategic vote in all such states! So she knows that she has a successful manipulation, but she does not know what the manipulation is. This is called *de dicto knowledge* of manipulation.

A stronger form of knowing is when there is a vote that is strategic in the profile for any state that the voter considers possible. This is called *de re knowledge* of manipulation.

A further situation of interest for voting theory is when (a) in any profile that the voter considers possible she can vote such that the outcome is either the same or better than when she had voted sincerely, and when (b) for at least one possible profile the outcome is better. This can be called *weakly successful manipulation.* (It is somewhat unclear if the qualification weak should apply to the manipulation or to the knowledge, as it is a property of a set of profiles.)

Definition 11 (Knowledge of manipulation) *Given a knowledge profile $\mathcal{P}_s$.*

- *Voter i can successfully manipulate $\mathcal{P}_s$ if she can successfully manipulate the profile $\pi(s)$.*
- *Voter i considers possible that she can successfully manipulate $\mathcal{P}_s$ if there is a t such that $s \sim_i t$ and she can successfully manipulate $\pi(t)$.*
- *Voter i knows 'de dicto' that she can successfully manipulate $\mathcal{P}_s$, if for all t such that $s \sim_i t$ she can successfully manipulate $\pi(t)$.*
- *Voter i knows 'de re' that she can successfully manipulate $\mathcal{P}_s$ if there is a vote $\succ_i'$ such that for all t such that $s \sim_i t$, $\succ_i'$ is a successful manipulation for profile $\pi(t)$.*
- *Voter i knows 'de re' that she can weakly successfully manipulate $\mathcal{P}_s$ if: (a) there is a vote $\succ_i'$ such that for all t such that $s \sim_i t$, either $\succ_i'$ is a successful manipulation for profile $\pi(t)$ or the outcome of that vote in $\pi(t)$ does not change, and (b) there is a t such that $s \sim_i t$ and $\succ_i'$ is a successful manipulation for profile $\pi(t)$.*

There is also a weakly successful version of 'de dicto' knowledge of manipulation.

These notions of knowledge of manipulation do not assume that voters know their own vote, although to apply them under these circumstances could lead to counterintuitive results.

If voter i knows 'de re' that she can manipulate the election, she has the ability to manipulate, namely by strategically voting $\succ_i'$. On the other hand, 'de dicto' manipulations do not have any practical interest, since the voter does not seem to have the ability to manipulate the election. It is akin to 'game of chicken' type equilibria in game theory [20]. Therein, for each strategy of a player there is a complementary strategy of the other player such that the pair is an equilibrium. This cannot be guaranteed without coordination. Example 2, below, illustrates 'de dicto' manipulability.

Example 2 *We consider manipulation with voting according to the Borda voting rule. Consider three agents, four candidates, and two profiles P and P' that are indistinguishable for agent 1, but that agents 2 and 3 can tell apart; as follows.*

1	2	3
c	d	b
b	a	d
a	c	c
d	b	a

P

1

1	2	3
c	d	b
b	a	a
a	c	c
d	b	d

P'

There is also a tie-breaking preference $b \succ c \succ d \succ a$. The difference between the profiles P and P' is that 3 prefers d over a in P but a over d in P'. We prove that 1 can manipulate the election if the profile is P, and that 1 can manipulate the election if the profile is P', but that the manipulation for P gives a worse outcome for P', and that the manipulation for P' gives a worse outcome for P. Therefore she is not effectively able to manipulate the outcome of the election.

In Borda, the ranks for each candidate in each vote are added, and the candidate with the highest sum wins, modulo the tie-breaking preference. The preferred candidate gets 3 points, the 2nd choice 2 points, etc. First, the outcome when all three agents give their truthful vote. We write $xyzw$ when there are x points for a, y for b, z for c, w for d.

profile	*count*	*observation*	*outcome*
P	3555	b, c, d *are tied*	b
P'	5553	a, b, c *are tied*	b

Voter 1 can manipulate P or P' by downgrading b. But this is tricky, because it comes at the price of making a or d, or both, more preferred. This price is indeed too high:

In P, 1 can achieve a better outcome by $\succ_1'$ defined as $1 : cabd$. Let $Q = P[\succ_1/\succ_1']$, and $Q' = P[\succ_1/\succ_1']$. Although 1 prefers the winner in Q over the winner in P, the winner in Q' is less preferred by her than the winner in P':

profile	*count*	*observation*	*outcome*
Q	4455	c, d *are tied*	c
Q'	6453		a

In P', 1 can achieve a better outcome by $\succ_1''$ defined as $1 : cdba$. Let $R = P[\succ_1/\succ_1'']$, and $R' = P[\succ_1/\succ_1'']$. Now, 1 prefers the winner in R' over the winner in P', but the

winner in R is less preferred by her than the winner in P:

profile	*count*	*observation*	*outcome*
R	2457	*1's worst dream*	d
R'	4455	c, d *are tied*	c

For the record, the winners for all different votes for voter 1 where c is most preferred.

$1 : cbad$	$1 : cabd$	$1 : cdba$	$1 : cadb$	$1 : cdab$	$1 : cbda$
$b(3555)$	$c(4455)$	$d(2457)$	$d(4356)$	$d(3357)$	$d(2556)$
$b(5553)$	$a(6453)$	$c(4455)$	$a(6354)$	$c(5355)$	$b(4554)$

In the language $\mathcal{L}$ we cannot say that the outcome of the election in P is preferred by a voter to the outcome of the election in P'. For that, we need to add primitives $P \succ_i P'$ to the language. These act as background knowledge. They encode the voting function so that its results are available in all states and in all profile models.

Definition 12 (Language $\mathcal{L}^+$) *We expand the set of propositional variables with $P \succ_i P'$ for any $P, P' \in O(\mathcal{C})^n$, and we add the following clause to the semantics:*

$$\mathcal{P}_s \models P \succ_i P' \quad \textit{iff} \quad F(P) \succ_i F(P')$$

The variables $P \succ_i P'$ mean that voter i prefers the candidate chosen by the votes in P over the candidate chosen by the votes in P'. This is a(n) (inefficient) way to encode the voting function. We observe that the semantics is indeed independent from state s and profile model P. These are model validities $\models P \succ_i P'$.

All notions of manipulation in Definition 11 are definable in the extended language $\mathcal{L}^+$.

Definition 13 *Let $\mathcal{P}_s$ be a knowledge profile with profile P.*

- *Voter i has a successful manipulation:*

$$P \wedge (P \rightarrow \succ_i) \wedge \bigvee_{P'} (P' \succ_i P \wedge (P' \leftrightarrow_i P))$$

- *Voter i has a successful manipulation $\succ_i'$:*

$$P \wedge (P \rightarrow \succ_i) \wedge (P' \rightarrow \succ_i') \wedge (P' \leftrightarrow_i P) \wedge P' \succ_i P$$

- *Voter i knows de dicto that she has a successful manipulation:*

$$P \wedge (P \rightarrow \succ_i) \wedge K_i \bigvee_{P'} ((P' \leftrightarrow_i P)) \wedge P' \succ_i P)$$

- *Voter i knows de re that she has a successful manipulation:*

$$P \wedge (P \rightarrow \succ_i) \wedge \bigvee_{\succ_i'} [((P' \leftrightarrow_i P) \wedge P' \succ_i P \wedge P' \rightarrow \succ_i')) \wedge K_i(P'' \rightarrow ((P' \leftrightarrow_i P'') \wedge P' \succ_i P''))$$

De re knowledge of weak manipulation is similarly defined.

Proposition 2 *Knowledge of manipulation is definable in $\mathcal{L}^+$.*

Proof. As evidenced in Definition 13.

5. EQUILIBRIUM AND KNOWLEDGE

Determining equilibria under incomplete knowledge comes down to decision taking under incomplete knowledge. Therefore we have to choose a decision criterion. Expected utility makes no sense here, because we didn't start with probabilities over profiles in the first place, nor with utilities. In the absence of prior probabilities, the following three criteria make sense. (*i*) The *insufficient reason* (or *Laplace*) criterion considers all possible states in a given situation as equiprobable. This criterion was used in [1] to determine equilibria of certain (Bayesian) games of imperfect information. (*ii*) The *maximum regret* criterion selects the decision minimizing the maximum utility loss, taken over all possible states, compared to the best decision, had the voter known the true state. (*iii*) The *pessimistic* (or *Wald*, or *maximin*) criterion compares decisions according to their worst possible consequences. The latter criterion, that we also call *risk averse*, is one that fits well our probability-free and utility-free model; this was also the criterion chosen in [9]. The only assumption here is that the probability distribution is positive in all states. We now fix this criterion for the rest of the paper. (Pessimistic, optimistic, and yet other criteria only assuming positive probability are applied to social choice settings in the recent [21]. We think their interesting results can be modelled as games using our setting.)

In the presence of knowledge, the definition of an equilibrium extends naturally. The trick is that for each agent, the combination of an agent i and an equivalence class $[s]_{\sim_i}$ for that agent (for some state s in the knowledge profile) defines a so-called virtual agent (we model these imperfect information games as Bayesian games [14]). Thus, agent i is multiplied in as many virtual agents as there are equivalences classes for $\sim_i$ in the model.

In our setting we can almost think of these equivalence classes as sets of indistinguishable profiles. Almost but not quite: we recall that states with different properties in a given equivalence class, or states in different equivalence classes, may be assigned the same profile.

An equilibrium is then a combination of votes such that none of the virtual agents has an interest to deviate. A intuitively more appealing solution than virtual agents, also applied in [1], is to stick to the agents we already have, but change the set of votes into a larger set of *conditional votes* — where the conditions are the equivalence classes for the agents. This we will now follow in the definition below. For risk-averse voters we can effectively determine if a conditional profile is an equilibrium without taking probability distributions into account, unlike in the more general setting of Bayesian games that it originates with.

Definition 14 (Conditional equilibrium) *Given is a knowledge profile model $\mathcal{P}$ such that every voter knows her preference (truthful vote). For each agent i, a* conditional vote *is a function $[\succ]_i : S/\sim_i \rightarrow O(\mathcal{C})$, i.e., a function that assigns a vote to each equivalence class for that agent. A* conditional profile *is a collection of n conditional votes, one for each agent. A* conditional voting game *is then a (standard) strategic game where voters declare conditional votes. A conditional profile is an* equilibrium *iff no agent has a successful manipulation in any of its equivalence classes.*

The outcome of a conditional profile consisting of conditional votes is a n-tuple of vectors $(x_1, \ldots, x_m)$ where voter i has m equivalence classes. The definition of equilibrium for the conditional voting game is derived from the Bayesian game form. It is not the standard form of strategic games! Consider a case for two equivalence classes for a voter 1 where two outcome vectors for 1 are (a, d) and (d, a), and $a \succ_i d$. We cannot say which of these two are preferred: therefore, the outcomes for 1 are not ordered, and therefore, it does

not define a standard strategic game. However, if we only vary 1's vote in the first argument (equivalence class) or in the second argument, the outcomes are ordered. This is the Bayesian game computation of equilibrium, where we determine manipulability for each virtual agent. Therefore, in the definition we did not write 'A conditional profile is an equilibrium iff no agent has a successful manipulation' but '(...) iff no agent has a successful manipulation *in any of its equivalence classes*.'

The requirement in Def. 14 that voters need to know their preference (truthful vote), is because the value they associate with that class is the worst outcome. This might otherwise be undefined.

Example 3 *We recall Example 1. There are two voters 1, 2, and four candidates a, b, c, d. Consider a plurality vote with a tie-breaking rule $b \succ a \succ c \succ d$.*

First consider the profile P defined as

1	2
a	d
c	c
b	b
d	a

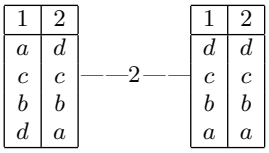
If 1 votes for her preference a and 2 votes for his preference d, then the tie prefers a, 2's least preferred candidate. If instead 2 votes c, a will still win. But if 2 votes b, b wins. We observe that (a, b) and (b, b) are equilibria pairs of votes, and that for 1 voting a is dominant.

This is also apparent from the voting matrix (wherein equilibria are boxed), and even more so when we express the payoffs for both voters by their ranking for the winner, as on the right.

1\2	a	b	c	d
a	a	$\boxed{b}$	a	a
b	b	$\boxed{b}$	b	b
c	a	b	c	d
d	a	b	d	d

1\2	a	b	c	d
a	30	$\boxed{11}$	30	30
b	11	$\boxed{11}$	11	11
c	30	11	22	03
d	30	11	03	03

Example 4 *We now add uncertainty to the setting of Example 3. Consider another profile P', that is as P, but where 1's vote is $1 : dcba$. Now consider a knowledge profile as follows. It remains the case that the actual profile is P; voter 2 is uncertain which of P and P' is the case; whereas voter 1 knows that. (It is tempting to add: voter 1* of course *knows that, as he knows his own vote; but our framework equally applies to situations where he does not, e.g., because he has not yet made up his mind.) And, as one should always add: 1 and 2 know that this is the uncertainty about the profile. This knowledge profile $\mathcal{P}_P$ consists of states t and u.*



t, P $\qquad$ u, P'

What are the conditional equilibria of $\mathcal{P}$? Votes (a, b) and (b, b) still lead to elect b and are the equilibria in state t with profile P. The only equilibrium vote for for state u with profile P' is (d, d)—the preferences are identical for 1 and 2, and d is their top candidate.

We argue our way towards the equilibria of this conditional voting game. There are two. Of course, alternatively to this argument one can directly determine these are equilibria by applying Definition 14 in a 16×4 matrix (below). Recall that we assumed that voters are risk-averse.

First, consider voter 1. For each equivalence class of 1, we have to determine her optimal vote. If the profile is P, 1's vote for a is dominant, so no matter what strategic considerations 2 may have due to the additional uncertainty about the profile, does not make a difference. Voter 1 votes a. If the profile is P', d is dominant for 1.

Next, consider voter 2. Because 2 is risk-averse he will vote b. Because if 2 votes d and the profile is P, a wins because 1 votes a, as this is dominant for 1 (or b wins because 1 votes b); whereas if the profile is P' and 2 votes d, then d wins because 1 votes d, which is dominant there. The worst outcome of these two is a (or b). Whereas if 2 votes b, the worst outcome is b. (The votes c and a can be eliminated from consideration as well.)

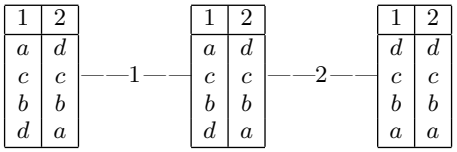
The two equilibria that we can associate with this knowledge profile are below. The conditional vote for 1 in the first equilibrium actually is actually defined as: $[\succ]_1(\{t\}) = \succ_1$ and $[\succ]_1(\{u\}) = \succ'_1$; and the vote for 2 is conditional to one equivalence class — in other words, it is unconditional. The equivalent verbose formulation is more intelligable.

- *(if 1 prefers a then a and if 1 prefers d then d, b),*
- *(if 1 prefers a then b and if 1 prefers d then d, b).*

In particular, 2 does not know that d is his equilibrium vote in P', because he considers it possible that the profile is P, where, if 2 votes d, 1 votes a (or 1 can improve her outcome by voting a), in which case 2 is worse off than d.

We can represent the game by a 16×4 matrix (Table 1). A conditional vote ab for 1 means: in t she votes a and in u she votes b. The outcome triples xyz represent: (worst and only) outcome for 1 in equivalence class of t, (worst and only) outcome for 1 in equivalence class of u; (worst) outcome for 2 in equivalence class of $\{t, u\}$. The table contains much symmetry. We omitted the table in terms of ranked outcomes. A triple like aaa corresponds to ranked outcome 144*: the equal winners a for voter 1 are ranked according to different profiles, a is preferred in state t / in profile P, hence 1, but a is least preferred in state u / in profile P', hence 4. In Table 1, the third of a triple xyz is necessarily equal to the least preferred of x and y, but this is an artifact of the example (namely, that the two equivalence classes for 1 together comprise the equivalence class for 2).*

Example 5 *We can add further uncertainty to Example 5.*



s, P $\qquad$ t, P $\qquad$ u, P'

Consider a third state that has the same profile P as the actual state, but that has different epistemic properties: 2 is not uncertain about the profile there, but 1 cannot distinguish this from the other state for P wherein 2 is uncertain about the profile. This is the profile model from Example 1.

Will 1 vote differently in s and t? In fact, she will not, nor will 2, and the conditional equilibria votes remain the same;

1\2	a	b	c	d
aa	aaa	bbb	aaa	aaa
ab	aba	bbb	aaa	aaa
ac	aaa	bbb	aca	aca
ad	aaa	$\boxed{bbb}$	aaa	ada
ba	baa	bbb	baa	baa
bb	bbb	bbb	bbb	bbb
bc	baa	bbb	bcb	bcb
bd	baa	$\boxed{bbb}$	bbb	bdb
ca	aaa	bbb	caa	caa
cb	aaa	bbb	cbb	cbb
cc	aaa	bbb	ccc	ccc
cd	aaa	bbb	ccc	cdc
da	aaa	bbb	caa	daa
db	aaa	bbb	cbb	dbb
dc	aaa	bbb	ccc	dcc
dd	aaa	bbb	ccc	ddd

Table 1: Conditional equilibria

strictly, 2's vote should depend on his equivalence class, but as 2's choice is the same either way, namely b, his vote is more succinctly described as an unconditional: b.

We did not yet attempt to characterize conditional equilibria in the logic of the previous sections, as we did for manipulation and knowledge of manipulation (Def. 9 and 13). This might be interesting for epistemic game theory [2, 22], but even so we only deal with the special case of voting games.

6. DYNAMICS: REVEALING PREFERENCE

We can extend the modal logical setting for voting and knowledge of the previous sections with logical operations that are dynamic in character. In the context of voting, two obvious choices here are *public announcement of a proposition* (such as an agent revealing her true preference), and *declaring a vote*. Such actions can be modelled as semantic operations $\mathcal{P}_s \mapsto \mathcal{P}_s|\varphi$ (for propositions φ, e.g., respectively, $\varphi = \succ_i$ for revealing her preference) and $\mathcal{P}_s \mapsto \mathcal{P}_s^{\gg_i := \top}$ (for voter i declaring vote $\gg_i$). In this section we deal with public announcement, in the next section, with public assignment.

A well-known dynamic feature of epistemic logics is *truthful public announcement* [23]. Given a knowledge profile $\mathcal{P}_s$, the requirement for execution of public announcement of φ is that φ is true in $\mathcal{P}_s$, and the way to execute it is to restrict the model $\mathcal{P}$ to all the states where φ is true. We can then investigate the truth of propositions in that model restriction: we can evaluate formulas of form $[\varphi]\psi$, for 'After announcement of φ, ψ (is true)', such as: 'After 1 reveals her preference (truthful vote) to 2, 2 knows that he has a successful manipulation'. We need to add a clause to the logical language for these announcements and define their semantics. The model restriction to the φ-states is denoted as $\mathcal{P}_s|\varphi$.

Definition 15 (Public announcement) *We add an inductive clause $[\varphi]\varphi$ to the logical language $\mathcal{L}$ (i.e., a dynamic modal operator with an argument of type formula followed by a postcondition also of type formula). Its semantics is:*

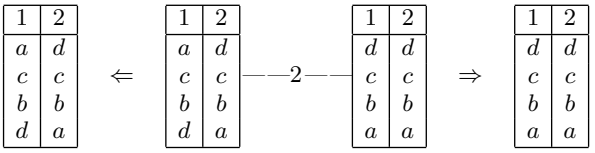
$$\mathcal{P}_s \models [\varphi]\psi \text{ iff } \mathcal{P}_s \models \varphi \text{ implies } \mathcal{P}_s|\varphi \models \psi,$$

where $\mathcal{P}_s|\varphi = (S', \sim_1', \ldots, \sim_n', \pi')$ such that $S' = \{t \in S : \mathcal{P}_t \models \varphi\}$, $\sim_i' = \sim_i \cap (S' \times S')$, and $\pi'(a \succ_i b) = \pi(a \succ_i b) \cap S'$.

Example 6 *Consider again Examples 1 and 4, with plurality voting. In state t (for profile P), after voter 1 informs voter 2 of her true preference (a public announcement), the uncertainty in the model disappears and 1 and 2 commonly know that the profile is P. The equilibrium vote remains (b, b). So this seems not a big deal.*

On the other hand, in state u voter 1 has an incentive to make her preference known to 2: after that, 2's equilibrium vote changes from b to d, and the equilibrium profile is now (d, d). And that is a big deal.

The transitions can be depicted as follows:



We can now formalize statements as

$$\mathcal{P}_t \models \neg K_2\ a \succ_1 c \wedge [a \succ_1 c] K_2\ a \succ_1 c.$$

There are two obvious ways to interpret such public announcements in voting theory: (i) when voters make announcements about their own preferences (and such that these announcements are trusted by other voters), and, more properly from the viewpoint of public announcement logic, (ii) when external observers, such as a central authority, reveal preferences to voters. The last can be interpreted as holding a voting poll. Successive voting polls reduce the uncertainty for the individual voter of the preferences (truthful vote) of *other* voters. And this may determine the strategic vote.

Two obvious results are that:

Proposition 3 *Knowledge of weakly successful manipulation is not preserved after update.*

PROOF. We recall Definition 11. For the weak form of manipulation there were two requirements: (a) the profile of at least one state in a given equivalence class for voter i needs to have a manipulation, and (b) the profiles of all states in that equivalence class must have either equal or better outcome. The state with a manipulation need not be the actual state, therefore, after model restriction the existential requirement (a) may no longer hold. This holds for 'de re' as well as 'de dicto' knowledge.

Proposition 4 *Knowledge of successful manipulation is preserved after update.*

PROOF. The profiles of all states have a manipulation, a universal property that is preserved after update.

7. DYNAMICS OF DECLARING VOTES

A voter i declaring a vote $\succ_i$ can be modelled in dynamic epistemic terms as an *assignment* (a.k.a. ontic change, in

contrast to an informative change like an announcement and coalition deliberation). A succinct way to model this is to expand the knowledge profiles with a *duplicate set of propositional variables* expressing voter preference, initially all set to false. To distinguish the preference (truthful vote) from the declared vote we keep writing $\succ_i$ for the former whereas we write $\gg_i$ for the latter. So, the set of variables $a \succ_i b$ encode the preferences of the voters, whereas variables $a \gg_i b$ encode their declared votes.

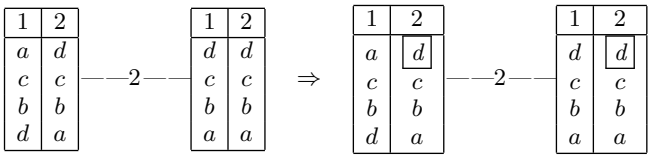
The action of declaring a vote $\gg_i$, defined by preferences $a \gg_i b$, sets the value of the propositions encoding $\gg_i$ in the model to true: these are the assignments $a \gg_i b := \top$ executed for all $a \gg_i b$ in $\gg_i$. If we assume that the declared vote is public, then this assignment can be executed in all states of the knowledge profile. The dynamic epistemic logic equivalent to achieve that is a public assignment [29, 27].

Definition 16 (Public assignment) *We add an inductive clause $[a \gg_i b := \top]\varphi$ to the logical language. For the semantics, given a knowledge profile $\mathcal{P}_s$, $\mathcal{P}_s \models [a \gg_i b := \top]\varphi$ iff $(\mathcal{P}^{a \gg_i b})_s \models \varphi$, where $\mathcal{P}^{a \gg_i b}$ is as $\mathcal{P}$ except that $\pi(a \gg_i b) = \mathcal{D}(\mathcal{P})$. By abbreviation we define $\gg_i := \top$ as the sequential execution of all assignments $a \gg_i b := T$ for all terms $a \gg_i b$ in $\gg_i$.*

Assignments need not be to 'true' ($\top$) but can be to any formula. Such an assignment $a \gg_i b := \psi$ has semantics $\pi(a \gg_i b) = \{t \in \mathcal{D}(\mathcal{P}) \mid \mathcal{P}_t \models \psi$. Declaring one's preference, the truthful vote, can then be seen as the assignment $\gg_i := \succ_i$.

Example 7 *Consider $a \gg_1 b \gg_1 c$. The assignment declaring this vote is the sequence of three assignments $a \gg_1 b := \top, b \gg_1 c := \top, a \gg_1 c := \top$, abbreviated as $\gg_i := \top$.*

Example 8 *Another continuation of Example 4 is with declaring votes. If in state t voter 2 declares his vote, i.e., fixes d as the candidate of his choice, 1 votes a, because with the given tie $b \succ a \succ d \succ c$, her preference a now gets elected. We can simulate this assignment as the sequence of $d \gg_2 c := \top$, $d \gg_2 b := \top$, $d \gg_2 a := \top$ (or as the assignment of preference to the declared vote: $\gg_2 := \succ_2$). For simplicity this is depicted as making d bold.*



t, P u, P' t, P' u, P'

We have no results yet for the interaction of declaring votes and revealing voter preference, but Stackelberg games are the obvious games of interest here.

Axiomatization and completeness. All four logics proposed in this work have sound and complete axiomatizations with respect to the class of profile models. However, this is not remarkable. We have therefore omitted these axiomatizations, for that see the cited references.

8. CHAIR AND COALITIONS

We have some modelling results concerning matters relevant for social choice theory that we have chosen not to incorporate in the main story, as not to lose focus there: how to model the central authority, and group notions of preference and knowledge.

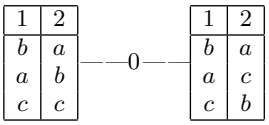
8.1 Central authority

Apart from the n voters, it seems convenient to distinguish yet another agent: a designated agent named 0, the *central authority*, or *chair*. We recall that the tie-breaking preference $\succ_{\mathsf{tie}}$ is a linear order on candidates. Apart from applying the tie, the central authority may perform other kinds of actions such as fixing the agenda. This also opens the door to the logical modelling of well-studied problems in computational social choice, such as control by the chair, or determining possible winners. The main reason *not* to model the chair it that her role is uniform throughout the model (throughout any knowledge profile model). We assume that there is no uncertainty on what the voting rule (and the tie-breaking preference) is. So in that sense it is exogenous.

The universal relation $S \times S$ on a knowledge profile model can be seen as the indistinguishability relation of the agent 0, the central authority. On a connected model (i.e., when there is always a path between any two states in the model) this is the same as common knowledge of the voters. The computational tasks of the central authority, be it determining the possible winners or finding strategic actions such as agenda fixing or any other form of control, can only be harder on knowledge profiles as it has to take uncertainty into account. By identifying the central authority with an agent with universal ignorance we can be precise about how much harder.

A partial profile in the social choice literature corresponds in a profile model to the set of profiles completing it, with identity access for all voters, and indistinguishable for the central authority, as in the following example. (The set of partial profiles then seems to consist of such disconnected parts.)

Example 9 *The following depicts the partial profile $(b \succ_1 a \succ_1 c, a \succ_2 \{b, c\})$. Voters 1 and 2 have identity access on the profile model. The central authority is agent 0.*



8.2 Coalitional manipulation

Group notions play an important role in social choice theory. We consider coalitions $G \subseteq \mathcal{N}$. As straightforward generalizations of (individual) preference $\succ_i$, (individual) manipulation, (weak) equilibrium, and (weak) equilibrium of a conditional voting game, we can also define: *coalitional preference* $\succ_G$, and *successful manipulation by a coalition* G. A profile P' is a *strong equilibrium profile* iff no coalition has a successful manipulation.

Group notions also play an important role in epistemic logic. Two notions useful in our setting are common knowledge and distributed knowledge. Given a knowledge profile, a proposition is commonly known if it is true in all states reachable (from the actual state of the knowledge profile) by arbitrarily long finite paths in the model (reflexive transitive closure of access for all voters in the coalition). With the interpretation of common knowledge of coalition G we

can thus associate an equivalence relation $\sim_G$ (defined as $(\bigcup_{i\in G} \sim_i)^*$). A proposition is distributedly known in a knowledge profile, if it is true in the intersection of accessibility relations in the actual state (the relation $\bigcap_{i\in G} \sim_i$).

If there is no uncertainty about the profile, the voters have common knowledge about the profile. This assumption is almost always made in social choice theory. It is important to observe that in the presence of uncertainty this strong form of common knowledge disappears, but that still some form of common knowledge remains: all agents have common knowledge of the structure of the profile model. This means that they have common knowledge of the set of states, the accessibility relations of the knowledge model, and what profiles these states stand for. The only thing they do (or rather, may) not know is the designated point of the profile model: what the preferences (truthful votes) are.

Coalitions play a big role in voting, partly because in realistic settings the power of individual voters is very limited. Now by analogy, just as the vote of an individual agent depends on her knowledge, the vote of a coalition would seem to depend on the common knowledge of that coalition. But that seems wrong. In voting theory, the power of a coalition means the power of a set of agents that can decide on a joint action as a result of communication between them. Communication makes the uncertainty about each others' profiles disappear. In terms of knowledge profiles, this means that we are talking about another model, namely the model where for all agents $i \in G$, $\sim_i$ is refined to $\bigcap_{i\in G} \sim_i$. What determines the voting power of a coalition seems rather its distributed knowledge.

We are still exploring the implications of these observation, and should note that also other choices can be made to model the power of a coalition in voting.

Knowledge of manipulation and equilibria of conditional voting games can also be defined for coalitions but have been left out of this presentation.

9. CONCLUSION, FURTHER RESEARCH

We presented a formal logical semantics for the interaction of voting and knowledge. The semantic primitive is the knowledge profile: a profile including uncertainty of voters about what the actual profile is. This reveals different notions for knowledge of manipulation, such as de re knowledge of manipulation and de dicto knowledge of manipulation, and novel notions for equilibria, such as conditional equilibrium for risk-averse voters. Dynamic operations on such knowledge profiles can also be modelled, and their effects on manipulation, where we distinguished public announcements, such as revealing true preferences, from public assignments, i.e., declaring votes.

As far as the formalization is concerned, our setting is very similar to that of the recent literature on robust mechanism design [7], which generalizes classical mechanism design by weakening the common knowledge assumptions of the environment among the players and the planner. In [7] uncertainty is modelled with information partitions. The main technical difference is that in our setting, as in classical social choice theory, preferences are ordinal, whereas in (robust) mechanism design preferences are numerical payoffs, which allows for payments (which we don't). This connection with mechanism design, however, is certainly worth exploring further. (We are very grateful to an anonymous reviewer for pointing this connection to us.)

The logical setting defined in the paper allows us to represent various classes of situations already studied specifically in (computational) social choice, thus offering a general representation framework in which, of course, new classes of problems will be representable as well, thus providing an homogeneous, unified representation framework. In some of the classes of problems we need one more agent, the chair. The chair may have preferences, but does not vote. In some classes of problems the dynamics plays a crucial role in defining these problems, both as announcements (revealing preference) and assignments (declaring votes). Here are a few such problems:

1. *possible and necessary winners* [17]: there is one more agent (the chair), who has an incomplete knowledge of each of the votes; the voters' knowledge is does not matter. x is a possible winner if the chair does not know that x is not a (co)winner, and a necessary winner if the chair knows that x is a (co)winner

2. *Stackelberg voting games* [30]: voters express their votes in sequence, in a commonly known order. Their preferences are common knowledge. The votes are announced publicly and each voter thus know the vote of the voters which speak before him.

3. *sequential voting games with abstention* [10]: voters express their votes in sequence, preferences are common knowledge; the voting rule is plurality; voters have the choice to vote or to abstain; voting is costly.

4. *control by adding or removing voters or candidates* [6]: the chair has a perfect knowledge of the voters' preferences; voters have no knowledge (and thus are supposed to vote truthfully); the chair may add or remove some candidates as well as register or unregister voters.

5. *sequential voting on multi-issue domains* [18]: the set of alternatives is a combinatorial domains, therefore the valuations are preference relations over tuples of values; voters vote in sequence, issue by issue, and the value for the (binary) issue is chosen by majority, and then communicated to the voters.

10. ACKNOWLEDGMENTS

We thank the TARK reviewers for their comments. The AAMAS poster [28] has the same content as this work, and it was also presented at the ESSLLI 2012 Opole workshop 'Strategies for Learning, Belief Revision and Preference Change'. The work was done while Hans van Ditmarsch was employed by the University of Seville, Spain. Hans van Ditmarsch is also affiliated to IMSc, Chennai, as a research associate.

11. REFERENCES

[1] T. Ågotnes and H. van Ditmarsch. What will they say? - Public announcement games. *Synthese*, 179(S.1):57–85, 2011.

[2] R. Aumann and A. Brandenburger. Epistemic conditions for nash equilibrium. *Econometrica*, 63:1161–1180, 1995.

[3] A. Baltag and S. Smets. A qualitative theory of dynamic interactive belief revision. In *Proc. of 7th LOFT*, Texts in Logic and Games 3, pages 13–60. Amsterdam University Press, 2008.

[4] S. Barbera, A. Bogomolnaia, and H. van der Stel. Strategy-proof probabilistic rules for expected utility maximizers. *Mathematical Social Sciences*, 35(2):89–103, 1998.

[5] J. Bartholdi, C. Tovey, and M. Trick. The computational difficulty of manipulating an election. *Social Choice and Welfare*, 6(3):227–241, 1989.

[6] J. Bartholdi III, C. Tovey, and M. Trick. How hard is it to control an election? *Mathematical and Computer Modelling*, 16(8/9):27–40, 1992.

[7] D. Bergemann and S. Morris. Robust mechanism design. *Econometrica*, 73(6):1771–1813, 2005.

[8] S. Chopra, E. Pacuit, and R. Parikh. Knowledge-theoretic properties of strategic voting. In *Proc. of 9th JELIA*, pages 18–30, 2004. LNCS 3229.

[9] V. Conitzer, T. Walsh, and L. Xia. Dominating manipulations in voting with partial information. In *Proc. of AAAI*, 2011.

[10] Y. Desmedt and E. Elkind. Equilibria of plurality voting with abstentions. In *ACM Conference on Electronic Commerce*, pages 347–356, 2010.

[11] J. Duggan and T. Schwartz. Strategic manipulability without resoluteness or shared beliefs: Gibbard-Satterthwaite generalized. *Social Choice and Welfare*, 17(1):85–93, 2000.

[12] R. Fagin, J. Halpern, Y. Moses, and M. Vardi. *Reasoning about Knowledge*. MIT Press, Cambridge MA, 1995.

[13] A. Gibbard. Manipulation of voting schemes: A general result. *Econometrica*, 41:587–601, 1973.

[14] J. Harsanyi. Games with Incomplete Information Played by 'Bayesian' Players, Parts I, II, and III. *Management Science*, 14:159–182, 320–334, 486–502, 1967–1968.

[15] N. Hazon, Y. Aumann, S. Kraus, and M. Wooldridge. Evaluation of election outcomes under uncertainty. In *Proc. of AAMAS '08*, pages 959–966, 2008.

[16] W. Jamroga and W. van der Hoek. Agents that know how to play. *Fundamenta Informaticae*, 63:185–219, 2004.

[17] K. Konczak and J. Lang. Voting procedures with incomplete preferences. In *Proc. IJCAI Multidisciplinary Workshop on Advances in Preference Handling*, 2005.

[18] J. Lang and L. Xia. Sequential composition of voting rules in multi-issue domains. *Mathematical Social Sciences*, 57(3):304–324, 2009.

[19] F. Liu. *Reasoning about Preference Dynamics*. Springer, 2011. Synthese Library, Vol. 354.

[20] M. Osborne and A. Rubinstein. *A Course in Game Theory*. MIT Press, 1994.

[21] R. Parikh, C. Tasdemir, and A. Witzel. The power of knowledge in games. In *Proc. of the Workshop on Reasoning About Other Minds*, 2011. CEUR Workshop Proceedings. Volume: 751.

[22] A. Perea. *Epistemic game theory*. Cambridge University Press, 2012.

[23] J. Plaza. Logics of public communications. In *Proc. of the 4th ISMIS*, pages 201–216. Oak Ridge National Laboratory, 1989.

[24] M. A. Satterthwaite. Strategy-proofness and Arrow's conditions: Existence and correspondence theorems for voting procedures and social welfare functions. *Journal of Economic Theory*, 10(2):187–217, April 1975.

[25] A. Slinko and S. White. Is it ever safe to vote strategically? Technical report, Auckland University, 2008. Dep. of Math. Research Report 563.

[26] J. van Benthem. Games in dynamic epistemic logic. *Bulletin of Economic Research*, 53(4):219–248, 2001.

[27] J. van Benthem, J. van Eijck, and B. Kooi. Logics of communication and change. *Information and Computation*, 204(11):1620–1662, 2006.

[28] H. van Ditmarsch, J. Lang, and A. Saffidine. Strategic voting and the logic of knowledge. In *Proc. of 11th AAMAS*, pages 1247–1248, 2012.

[29] H. van Ditmarsch, W. van der Hoek, and B. Kooi. Dynamic epistemic logic with assignment. In *Proc. of 4th AAMAS*, pages 141–148. ACM, 2005.

[30] L. Xia and V. Conitzer. Stackelberg voting games: Computational aspects and paradoxes. In *Proc. of AAAI*, 2010.

PDL as a Multi-Agent Strategy Logic

Extended Abstract*

Jan van Eijck
CWI and ILLC
Science Park 123
1098 XG Amsterdam, The Netherlands
jve@cwi.nl

ABSTRACT

Propositional Dynamic Logic or PDL was invented as a logic for reasoning about regular programming constructs. We propose a new perspective on PDL as a multi-agent strategic logic (MASL). This logic for strategic reasoning has group strategies as first class citizens, and brings game logic closer to standard modal logic. We demonstrate that MASL can express key notions of game theory, social choice theory and voting theory in a natural way, we give a sound and complete proof system for MASL, and we show that MASL encodes coalition logic. Next, we extend the language to epistemic multi-agent strategic logic (EMASL), we give examples of what it can express, we propose to use it for posing new questions in epistemic social choice theory, and we give a calculus for reasoning about a natural class of epistemic game models. We end by listing avenues for future research and by tracing connections to a number of other logics for reasoning about strategies.

Categories and Subject Descriptors

F.4.1 [**Mathematical Logic**]: Modal Logic; F.4.1 [**Mathematical Logic**]: Proof Theory; I.2.3 [**Artificial Intelligence**]: Deduction and Theorem Proving

Keywords

Strategies, Strategic Games, Coalition Logic, Modal Logic, Dynamic Logic, Voting Theory

1. INTRODUCTION

In this paper we propose a simple and natural multi-agent strategy logic, with explicit representations for individual and group strategies. The logic can be viewed as an extension of the well-known propositional logic of programs PDL. We show that the logic can express key notions of game theory and voting theory, such as Nash equilibrium, and the properties of voting rules that are used to prove the Gibbard-Satterthwaite theorem.

Unlike most other game logics, our logic uses explicit representations of group strategies in N-player games, with $N \geq 2$, and treats coalitions as a derived notion.

The logic we propose follows a suggestion made in Van Benthem [4] (in [11]) to apply the general perspective of action logic to reasoning about strategies in games, and links up to propositional dynamic logic (PDL), viewed as a general logic of action [29, 19]. Van Benthem takes individual strategies as basic actions and proposes to view group strategies as intersections of individual strategies (compare also [1] for this perspective). We will turn this around: we take the full group strategies (or: full strategy profiles) as basic, and construct individual strategies from these by means of strategy union.

A fragment of the logic we analyze in this paper was proposed in [10] as a logic for strategic reasoning in voting (the system in [10] does not have current strategies).

The plan of the paper is as follows. In Section 2 we review key concepts from strategic game theory, and hint at how these will show up in our logic. Section 3 does the same for voting theory. Section 4 gives a motivating example about coalition formation and strategic reasoning in voting. Section 5 presents the language of MASL, and gives the semantics. Next we show, in Section 6, that the key concepts of strategic game theory and voting theory are expressible in MASL. Section 7 extends the proof system for PDL to a sound and complete proof system for MASL. Section 8 gives an embedding of coalition logic into MASL. Section 9 extends MASL to an epistemic logic for reasoning about knowledge in games, Section 10 gives examples of what EMASL can express, and Section 11 sketches a calculus for EMASL. Section 12 concludes.

Key contributions of the paper are a demonstration of how PDL can be turned into a game logic for strategic games, and how this game logic can be extended to an epistemic game logic with PDL style modalities for game strategies and for epistemic operators. This makes all the logical and model checking tools for PDL available for analyzing properties of strategic games and epistemic strategic games.

2. GAME TERMINOLOGY

A strategic game form is a pair

$$(n, \{S_i\}_{i \in \{1,\ldots,n\}})$$

where $\{1, \ldots, n\}$ with $n > 1$ is the set of players, and each S_i is a non-empty set of strategies (the available actions for player i). Below we will impose the restriction that the game forms are finite: each S_i is a finite non-empty set.

We use N for the set $\{1, \ldots, n\}$, and S for $S_1 \times \cdots \times S_n$, and we call a member of S a strategy profile. Thus, a strategy

*A full version of this paper is available at `www.cwi.nl/~jve/papers/13`

TARK 2013, Chennai, India.

profile s is an n-tuple of strategies, one for each player. If s is a strategy profile, we use s_i or $s[i]$ for its i-th component. Strategy profiles are in one-to-one correspondence to game outcomes, and in fact we can view $s \in S$ also as a game outcome [22].

Consider the prisoner's dilemma game PD for two players as an example. Both players have two strategies: c for cooperate, d for defect. The possible game outcomes are the four strategy profiles $(c,c),(c,d),(d,c),(d,d)$.

	c	d
c	c,c	c,d
d	d,c	d,d

It is useful to be able to classify game outcomes. A P-outcome function for game form (N,S) is a function $o : S \to P$.

For the example of the PD game, o could be a function with range $\{x,y,z,u\}^2$, as follows:

	c	d
c	x,x	y,z
d	z,y	u,u

If $C \subseteq N$, we let $S_C = \prod_{i \in C} S_i$ be the set of group strategies for C. If $s \in S_C$ and $t \in S_{N-C}$ we use (s,t) for the strategy profile that results from combining s and t, i.e., for the strategy profile u given by

$$u[i] = s[i] \text{ if } i \in C, u[i] = t[i] \text{ otherwise.}$$

The group strategies for the PD game coincide with the strategy profiles.

An abstract game G is a tuple

$$(N, S, \{\geq_i\}_{i \in N}),$$

where (N,S) is a game structure, and each $\geq_i$ is a preference relation on $S_1 \times \cdots \times S_n$. These preference relations are assumed to be transitive, reflexive, and complete, where completeness means that for all different $s,t \in S$, one of $s \geq_i t$, $t \geq_i s$ holds.

In the PD game example, with the output function as above, the preferences could be fixed by adding the information that $z > x > u > y$.

The preference relations may also be encoded as numerical utilities. A payoff function or utility function for a player i is a function u_i from strategy profiles to real numbers. A payoff function u_i *represents* the preference ordering $\geq_i$ of player i if $s \geq_i t$ iff $u_i(s) \geq u_i(t)$, for all strategy profiles s,t.

A strategic game G is a tuple

$$(N, \{S_i\}_{i \in N}, \{u_i\}_{i \in N})$$

where $N = \{1,\ldots,n\}$ and $u_i : S_1 \times \cdots \times S_n \to \mathbb{R}$ is the function that gives the payoff for player i. Aim of players in the game is to maximize their individual payoffs. We will use u for the utility function, viewed as a payoff vector.

As an example, the PD game with payoffs as in the following picture, is a representation of the abstract version above.

	c	d
c	$2,2$	$0,3$
d	$3,0$	$1,1$

It should be noted that payoff functions are a special case of output functions. In the example of PD with payoffs, we can view the payoff function as an output function with range $\{0,1,2,3\}^2$.

Below, we will assume that output functions are of type $o : S \to P$, and we will introduce proposition letters to range over P. This allows us to view the game forms as modal frames, and the games including the output functions as models, with the output function fixing the valuation by means of "the valuation makes p true in a state s iff $s \in o^{-1}(p)$."

A special case of this is the case where the P are payoff vectors. Valuations that are payoff vectors allow us to express preferences of the players for an outcome as boolean formulas (see below).

Let (s_i', s_{-i}) be the strategy profile that is like s for all players except i, but has s_i replaced by s_i'. A strategy s_i is a *best response* in s if

$$\forall s_i' \in S_i \ u_i(s) \geq u_i(s_i', s_{-i}).$$

A strategy profile s is a (pure) Nash equilibrium if each s_i is a best response in s:

$$\forall i \in N \ \forall s_i' \in S_i \ u_i(s) \geq u_i(s_i', s_{-i}).$$

A game G is *Nash* if G has a (pure) Nash equilibrium.

These key notions of game theory will reappear below when we discuss the expressiveness of MASL.

3. VOTING AS A MULTI-AGENT GAME

Voting can be seen as a form of multi-agent decision making, with the voters as agents [14]. Voting is the process of selecting an item or a set of items from a finite set A of alternatives, on the basis of the stated preferences of a set of voters. See [7] for a detailed account.

We assume that the preferences of a voter are represented by a ballot, where a ballot is a linear ordering of A. Let $\mathbf{ord}(A)$ be the set of all ballots on A.

If there are three alternatives a,b,c, and a voter prefers a over b and b over c, then her ballot is abc.

Assume the set of voters is $N = \{1,\ldots,n\}$. If we use $\mathbf{b}, \mathbf{b}'$ to range over ballots, then a profile $\mathbf{P}$ is a vector $(\mathbf{b}_1,\ldots,\mathbf{b}_n)$ of ballots, one for each voter. If $\mathbf{P}$ is a profile, we use $\mathbf{P}_i$ for the ballot of voter i in $\mathbf{P}$.

The following represents the profile $\mathbf{P}$ where the first voter has ballot abc, the second voter has ballot abc, the third voter has ballot bca, and so on:

$$(abc, abc, bca, abc, cab, acb).$$

A *voting rule* V for set of alternatives A is a function from A-profiles to $\mathcal{P}^+(A)$ (the set of non-empty subsets of A). If $V(\mathbf{P}) = B$, then the members of B are called the winners of $\mathbf{P}$ under V. A voting rule is *resolute* if $V(\mathbf{P})$ is a singleton for any profile $\mathbf{P}$.

Absolute majority is the voting rule that selects an alternative with more than 50 % of the votes as winner, and returns the whole set of alternatives otherwise. This is not the same as plurality, which selects an alternative that has the maximum number of votes as winner, regardless of whether more than half of the voters voted like this or not.

Strategizing is replacing a ballot $\mathbf{b}$ by a different one, $\mathbf{b}'$, in the hope or expectation to get a better outcome, where better is "closer to $\mathbf{b}$" in some sense. There are many ways to

interpret 'better', and the particular choice does not matter. The way we will adopt (suggested in [32]) is to stipulate that X is better than Y if X weakly dominates Y, that is, if every $x \in X$ is at least as good as every $y \in Y$ and some $x \in X$ is better than some $y \in Y$.

Formally: If $X, Y \subseteq A$ $X \neq \emptyset$, $Y \neq \emptyset$, and $\mathbf{b} \in \mathbf{ord}(A)$, then $X >_{\mathbf{b}} Y$ if $\forall x \in X \forall y \in Y$: $x = y$ or x is above y in $\mathbf{b}$, and $\exists x \in X \exists y \in Y$: x is above y in $\mathbf{b}$.

Let $\mathbf{P} \sim_i \mathbf{P}'$ express that profiles $\mathbf{P}$ and $\mathbf{P}'$ differ only in the ballot of voter i.

A voting rule is *strategy-proof* if $\mathbf{P} \sim_i \mathbf{P}'$ implies $V(\mathbf{P}) \geq_{\mathbf{b}} V(\mathbf{P}')$, where $\mathbf{b} = \mathbf{P}_i$ (so $\geq_{\mathbf{b}}$ expresses 'betterness' according to the i-ballot in $\mathbf{P}$).

To analyze voting as a game, think of casting an individual vote as a strategy. If we assume that the voting rule is fixed, this fixes the game outcome for each profile. The definition of 'betterness' determines the pay-off.

Player strategies are the votes the players can cast, so the set of individual strategies is the set A, for each player. Strategy profiles are the vectors of votes that are cast. Outcomes are determined by the voting rule; if the voting rule is resolute, outcomes are in A, otherwise in $\mathcal{P}^+(A)$. Preferences are determined by the voter types, plus some stipulation about how voters value sets of outcomes, given their type, in the case of non-resolute voting rules.

4. GROUP ACTION IN VOTING GAMES

To illustrate strategic reasoning and coalition formation in voting, we give an extended example. Suppose there are three voters $1, 2, 3$ and three alternatives a, b, c. Suppose the voting rule is plurality. Then each player or voter has the choice between actions a, b, and c.

Suppose 1 is the row player, 2 the column player, and 3 the table player. Then the voting outcomes are given by:

a:

	a	b	c
a	a	a	a
b	a	b	a, b, c
c	a	a, b, c	c

b:

	a	b	c
a	a	b	a, b, c
b	b	b	b
c	a, b, c	b	c

c:

	a	b	c
a	a	a, b, c	c
b	a, b, c	b	c
c	c	c	c

To determine the payoff function, we need information about the types of the voters. Suppose voter 1 has type (true ballot) *abc*. Then the betterness relation for 1 for the possible outcomes of the vote is given by:

$$a > b > c \text{ and } a > \{a, b, c\} > c.$$

Observe that neither $\{a, b, c\} > b$ nor $b > \{a, b, c\}$. So let's assume these give the same payoff, and fix the payoff function for voters of type *abc* as

$$f(a) = 2, f(b) = f(\{a, b, c\}) = 1, f(c) = 0.$$

If we do similarly for the other voter types, then this fixes the strategic game for voting according to the plurality rule over the set of alternatives $\{a, b, c\}$.

So suppose 1 has ballot *abc*, 2 has ballot *bca*, and 3 has ballot *cab*. This gives the following strategic game form:

a:

	a	b	c
a	(2, 0, 1)	(2, 0, 1)	(2, 0, 1)
b	(2, 0, 1)	(1, 2, 0)	(1, 1, 1)
c	(2, 0, 1)	(1, 1, 1)	(0, 1, 2)

b:

	a	b	c
a	(2, 0, 1)	(1, 2, 0)	(1, 1, 1)
b	(1, 2, 0)	(1, 2, 0)	(1, 2, 0)
c	(1, 1, 1)	(1, 2, 0)	(0, 1, 2)

c:

	a	b	c
a	(2, 0, 1)	(1, 1, 1)	(0, 1, 2)
b	(1, 1, 1)	(1, 2, 0)	(0, 1, 2)
c	(0, 1, 2)	(0, 1, 2)	(0, 1, 2)

If the voters all cast their vote according to their true ballot, then 1 votes a, 2 votes b and 3 votes c, and the outcome is a tie, $\{a, b, c\}$, with payoff $(1, 1, 1)$. This is a Nash equilibrium: the vote cast by each player is a best response in the strategy profile.

Now let's change the voting rule slightly, by switching to plurality voting with tie breaking, where *abc* as the tie breaking order. This changes the plurality rule into a resolute voting rule. The new strategic game becomes:

a:

	a	b	c
a	(2, 0, 1)	(2, 0, 1)	(2, 0, 1)
b	(2, 0, 1)	(1, 2, 0)	(2, 0, 1)
c	(2, 0, 1)	(2, 0, 1)	(0, 1, 2)

b:

	a	b	c
a	(2, 0, 1)	(1, 2, 0)	(2, 0, 1)
b	(1, 2, 0)	(1, 2, 0)	(1, 2, 0)
c	(2, 0, 1)	(1, 2, 0)	(0, 1, 2)

c:

	a	b	c
a	(2, 0, 1)	(2, 0, 1)	(0, 1, 2)
b	(2, 0, 1)	(1, 2, 0)	(0, 1, 2)
c	(0, 1, 2)	(0, 1, 2)	(0, 1, 2)

If the players all vote according to their true preference, the outcome is a because of the tie breaking, with payoff given by $(2, 0, 1)$. But this is no longer a Nash equilibrium, for player 2 can improve his payoff from 0 to 1 by casting vote c, which causes the outcome to change into c, with payoff $(0, 1, 2)$. The strategy triple (a, c, c) is a Nash equilibrium.

So we are in a situation where the voting rule seems to favour voter 1 with ballot *abc*, because the tie breaking rule uses this order for tie breaking, and still the voter with this ballot ends up losing the game, because the other two players have an incentive to form a coalition against player 1.

5. A LANGUAGE FOR MASL

We will now turn to the description of strategic games like the PD game and the voting game in terms of actions in the spirit of PDL. We will take as our basic actions the full strategy profiles.

The reader is urged to think of a state in a game as a strategy vector where each player has determined her strategy. Strategy expressions in the MASL language are interpreted as relations on the space of all game states. Individual strategies emerge as unions of group strategies. An example

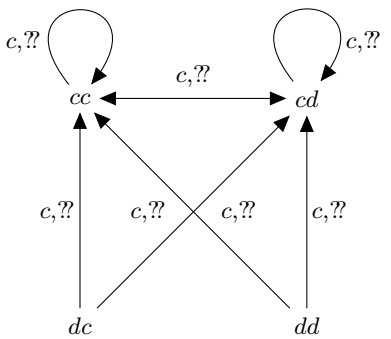


Figure 1: Cooperation Strategy for Player 1 in PD Game

is the strategy for the first player in the PD game to cooperate. This individual strategy is represented as $(c, ??)$, and interpreted as in Figure 1.

Strategy terms of MASL are:

$$t_i \quad ::= \quad a \mid ?? \mid !!$$

Here i ranges over the set of players N; and a ranges over the set of all strategies S_i for player i. A random term "$??$" denotes an individual strategy for an adversary player, and "$!!$" denotes the current strategy of a player. Random terms serve to model what adversaries do, and current terms serve to model what happens when players stick to a previous choice.

As will become clear below, terms of the form $??$ are used for succinctness; they could be dispensed with in favour of explicit enumerations of individual strategies.

From strategy terms we construct MASL strategy vectors, as follows:

$$\mathbf{c} \quad ::= \quad (t_1 \ldots, t_n)$$

The MASL strategy vectors occur as atoms and as modalities in MASL formulas. Allowing strategy terms as atomic formulas allows for succinct classification of game situations.

We assume that p ranges over a set of game outcome values, that is: we assume an outcome function $o : S \to P$. The language is built in the usual PDL manner by mutual recursion of action expressions and formulas:

$$\phi \quad ::= \quad \top \mid \mathbf{c} \mid p \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid [\gamma]\phi$$

$$\gamma \quad ::= \quad \mathbf{c} \mid ?\phi \mid \gamma_1; \gamma_2 \mid \gamma_1 \cup \gamma_2 \mid \gamma^*$$

We will employ the usual abbreviations for $\bot$, $\phi_1 \vee \phi_2$, $\phi_1 \to \phi_2$, $\phi_1 \leftrightarrow \phi_2$ and $\langle\gamma\rangle\phi$.

Let $s \in S$. Then s is a strategy profile, with individual strategies for player i taken from S_i. We refer to the i-component of s as $s[i]$. Thus, if $s = (a, b, b)$, then $s[1] = a$.

Let $i \in N$. Then $[\![\cdot]\!]^{S_i,s,i}$ is a function that maps each t_i to a subset of S_i, and $[\![\cdot]\!]^{S,s}$ is a function that maps each strategy vector to a set of strategy profiles $\subseteq S$, as follows:

$$\begin{aligned}
[\![a]\!]^{S_i,s,i} &= \{a\} \\
[\![??]\!]^{S_i,s,i} &= S_i \\
[\![!!]\!]^{S_i,s,i} &= \{s[i]\} \\
[\![(t_1 \ldots, t_n)]\!]^{S,s} &= [\![t_1]\!]^{S_1,s,1} \times \cdots \times [\![t_n]\!]^{S_n,s,n}
\end{aligned}$$

For example, let the set of individual strategies for each player be $A = \{a, b, c\}$, and let $n = 3$ (as in the voting example in Section 4). Then a strategic change by the first player to b, while both other players stick to their vote is expressed as $(b, !!, !!)$. In a game state (a, b, b) this is interpreted as $\{((a, b, b), (b, b, b))\}$.

A strategic change by the first player to b, given that the second player sticks to her vote, while the third player may or may not change, is expressed by $(b, !!, ??)$. In the context of a strategy profile $s = (a, b, c)$, this is interpreted as follows:

$$[\![(b, !!, ??)]\!]^{A,s} = \{b\} \times \{b\} \times \{a, b, c\}.$$

$(??, c, c)$ represents the group strategy where players 2 and 3 both play c. This is a strategy for the coalition of 2 and 3 against 1.

The formula that expresses that the coalition of 2 and 3 can force outcome c by both voting c is (abbreviating the singleton outcome $\{c\}$ as c):

$$[(??, c, c)]c.$$

The strategy $(??, ??, c)$ is different from $(!!, !!, c)$, for the latter expresses the individual strategy for player 3 of playing c, in a context where the two other players do not change their strategy.

The relational interpretation for coalition strategies follows the recipe proposed in [4], but with a twist. We interpret a strategy for an individual player as a relation on a set of game states, by taking the union of all full strategy relations that agree with the individual strategy. So the strategies for the individual players are choices that emerge from taking unions of vectors that determine the game outcome completely. If we assume that the players move together, without information about moves of the other players, then the individual strategies are choices, but an individual choice does not determine an outcome. Only the joint set of all choices does determine an outcome.

So if we represent a strategy for player i as a relation, then we have to take into account that the individual choice of i does need information about how the others move to determine the outcome. The relation for the individual choice a of player i is given by

$$[\![(??, \cdots, ??, a, ??, \cdots, ??)]\!]^{S,s}$$

$$= S_1 \times \cdots \times S_{i-1} \times \{a\} \times S_{i+1} \times \cdots \times S_n.$$

This relation is computed from all choices that the other players could make (all strategies for the other players).

Compare this with

$$[\![(!!, \cdots, !!, a, !!, \cdots, !!)]\!]^{S,s} =$$

$$\{s[1]\} \times \cdots \{s[i-1]\} \times \{a\} \times \{s[i+1]\} \times \cdots \{s[n]\}.$$

This is the action where player i switches to a, while all other players stick to their strategies.

The picture in Figure 2 gives the interpretation of the $(c, !!)$ strategy vector in the PD game.

This generalizes to coalitions, as follows. A strategy for a coalition is a choice for each of the coalition members, and the corresponding relation is the union of all full strategy relations that agree with the coalition strategy. Compare the definition of the $[\![\cdot]\!]^{S,s}$ function for strategy vectors above.

This gives an obvious recipe for turning strategic game forms with outcome functions into Kripke models. Let (N, S)

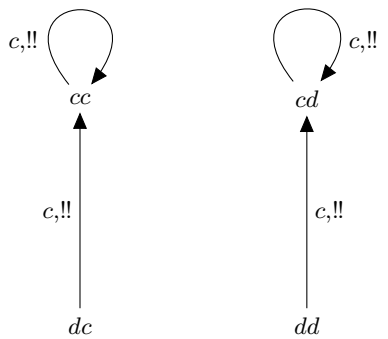


Figure 2: Interpretation of $(c, !!)$ in PD Game

be a strategic game form and let $o : S \to P$ be an outcome function, and let $s \in S$ be a strategy profile.

Then the truth definition for MASL, with respect to $M = (N, S, o)$ and s is given by:

$$\begin{array}{rcl} M, s \models \top & & \text{always} \\ M, s \models \mathbf{c} & \text{iff} & s \in [\![\mathbf{c}]\!]^{S,s} \\ M, s \models p & \text{iff} & s \in o^{-1}(p) \\ M, s \models \neg\phi & \text{iff} & M, s \not\models \phi \\ M, s \models \phi_1 \wedge \phi_2 & \text{iff} & M, s \models \phi_1 \text{ and } M, s \models \phi_2 \\ M, s \models [\gamma]\phi & \text{iff} & \text{for all } t \text{ with } (s,t) \in [\![\gamma]\!]^M : \\ & & \quad M, t \models \phi \end{array}$$

$$\begin{array}{rcl} [\![\mathbf{c}]\!]^M & = & \{(s,t) \mid t \in [\![\mathbf{c}]\!]^{S,s}\} \\ [\![?\phi]\!]^M & = & \{(s,s) \mid M, s \models \phi\} \\ [\![\gamma_1;\gamma_2]\!]^M & = & [\![\gamma_1]\!]^M \circ [\![\gamma_2]\!]^M \\ [\![\gamma_1 \cup \gamma_2]\!]^M & = & [\![\gamma_1]\!]^M \cup [\![\gamma_2]\!]^M \\ [\![\gamma^*]\!]^M & = & ([\![\gamma]\!]^M)^*, \end{array}$$

where $\circ$ is used for relation composition, and * for reflexive transitive closure.

Note that it is assumed that the signature of the language matches that of the model: for interpretation in $M = (N, S, o)$ with $o : S \to P$, we assume that strategy vectors of the language have length n, that the terms t_i of the language get interpreted as subsets of S_i, and that the propositionional atoms range over P.

6. EXPRESSIVENESS OF MASL

We give examples to demonstrate that MASL expresses key concepts of game theory, voting theory, social choice theory and iterated game playing, in a natural way.

Abbreviations.

Let $(i_a, \overline{!!})$ abbreviate the strategy vector

$$(!!, \cdots, !!, a, !!, \cdots, !!),$$

with a in i-th position, and $!!$ everywhere else.

Using this, let $[(i, \overline{!!})]\phi$ abbreviate $\bigwedge_{a \in S_i} [(i_a, \overline{!!})]\phi$. Then $[(i, \overline{!!})]\phi$ expresses that all strategies to which player i can switch from the current strategy profile result in a strategy profile where ϕ holds (provided that the other players keep their strategies fixed).

Let $(i_a, \overline{?})$ abbreviate the strategy vector

$$(?, \cdots, ?, a, ?, \cdots, ?),$$

with a in i-th position, and $?$ everywhere else.

Using this, let $[(i, \overline{?})]\phi$ abbreviate $\bigwedge_{a \in S_i} [(i_a, \overline{?})]\phi$. Then $[(i, \overline{?})]\phi$ expresses that all strategies for i guarantee ϕ, no matter what the other players do.

Let $(\overline{?})$ abbreviate $(?, \cdots, ?)$ (the strategy vector that everywhere has $?$). Then $\langle(\overline{?})\rangle\phi$ expresses that in some game state ϕ holds.

Representing Payoffs.

To represent payoffs, we will assume that basic propositions are payoff vectors u, and that the payoff values are in a finite set U (the set of all utilities that can be assigned in the game). Next, define $u_i \geq v$ as $\bigvee_{w \in U, w \geq v} u[i] = w$ and $u_i > v$ as $\bigvee_{w \in U, w > v} u[i] = w$. Then $u_i \geq v$ expresses that player i gets at least v, and $u_i > v$ expresses that player i gets more than v (compare [34] for a similar approach).

Weak Dominance.

Using the above abbreviations, we can express what it means for an i-strategy a to be *weakly dominant*. Intuitively, it means that a is as least as good for i against any moves the other players can make as any alternative b for a. In our logic:

$$\bigwedge_{v \in U} \bigwedge_{b \in A - \{a\}} [(i_b, \overline{?})](u_i \geq v \to \langle(i_a, \overline{!!})\rangle u_i \geq v).$$

Nash Equilibrium.

The following formula expresses that the current strategy profile is a Nash equilibrium:

$$\bigwedge_{i \in N} \bigvee_{v \in U} (u_i \geq v \wedge [(i, \overline{!!})]\neg u_i > v).$$

The following formula expresses that the game is Nash:

$$\langle(\overline{?})\rangle \bigwedge_{i \in N} \bigvee_{v \in U} (u_i \geq v \wedge [(i, \overline{!!})]\neg u_i > v).$$

Plurality Voting.

For the application of MASL to voting, assume the output function produces an ordered pair consisting of the outcome of the voting rule for a profile, plus the utility vector for the players for that profile.

Let A be the set of alternatives. Let P_a be the set of all full strategy vectors where a gets more votes than any other alternative. Then

$$\bigwedge_{x \in A} \bigwedge_{\mathbf{c} \in P_x} [\mathbf{c}]x$$

expresses that the game is a voting game with plurality rule. This is easily extended to a formula that expresses the rule of plurality voting with tie breaking.

Resoluteness.

Assume that the proposition a expresses that a is among the winners given the current profile. A voting rule is *resolute* if there is always exactly one winner. Viewing voting

according to a voting rule as a game, the following formula expresses that the game is resolute:

$$[(\overline{?\!?})] \bigvee_{a \in A} (a \wedge \bigwedge_{b \in A - \{a\}} \neg b).$$

Strategy-Proofness.

A voting rule is *strategy proof* if it holds for any profile S and for any player (voter) i that changing his vote (action) does not give an outcome that is better (according to the preferences of i in S) that the outcome in S. This is expressed by the following formula:

$$[(\overline{?\!?})] \bigwedge_{i \in N} \bigvee_{v \in U} (u_i \geq v \wedge \neg \langle (i, \overline{!\!!}) \rangle u_i > v).$$

Non-Imposedness.

A voting rule is (weakly) *non-imposed* if at least three outcomes are possible. Viewing voting as a game, we can use the following formula to express this:

$$\bigvee_{a \in A} \bigvee_{b \in A - \{a\}} \bigvee_{c \in A - \{a,b\}} (\langle (\overline{?\!?}) \rangle a \wedge \langle (\overline{?\!?}) \rangle b \wedge \langle (\overline{?\!?}) \rangle c).$$

Dictatorship.

In a multi-agent game setting, a dictator is a player who can always get what he wants, where getting what you want is getting a payoff that is at least as good as anything any other player can achieve. Here is the formula for that, using the abbreviation $(i, \overline{?\!?})$:

$$\bigvee_{v \in U} \bigwedge_{j \in N - \{i\}} [(?\!?)](\neg u_j > v \wedge \langle (i, !\!!) \rangle u_i \geq v).$$

Gibbard-Satterthwaite.

The classic Gibbard-Satterthwaite theorem [15, 30] states that all reasonable voting rules allow strategizing, or put otherwise, that no reasonable voting rule is strategy-proof.

Resoluteness, strategy-proofness, non-imposedness and dictatorship are the four properties in terms of which the Gibbard-Satterthwaite theorem is formulated, and in fact, the theorem can be stated and proved in our logic. What the theorem says semantically is:

$$\text{Res}, \text{SP}, \text{NI} \models \text{Dict}.$$

It follows from the completeness of the logic (Section 7 below) that for every choice of MASL language (where the choice of language fixes the number of players/voters N and the set of alternatives A, with $|A| > 3$), the following can be proved:

$$\text{Res}, \text{SP}, \text{NI} \vdash \text{Dict}.$$

Meta-Strategies: Tit-for-Tat.

Tit-for-tat as a meta-strategy for the PD game [3] is the instruction to copy one's opponents last choice, thereby giving immediate, and rancour-free, reward and punishment. Figure 3 gives a picture of the tit-for-tat meta-strategy for player 2, with the states indicating the outcomes of the last play of the game.

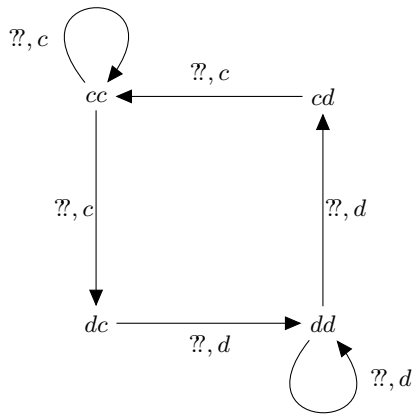


Figure 3: Tit-for-tat Meta-Strategy for Player 2 in PD Game

This works because we may think of the current state of the game as the result of the last play of PD, remembered in the state. Testing the state yields the clue for whether the reward action $(?\!?, c)$ or the punishment action $(?\!?, d)$ has to be executed. Thus, the following MASL action expression describes this meta-strategy for player 2.

$$(?(c, ?\!?); (?\!?, c) \ \cup \ ?(d, ?\!?); (?\!?, d))^*$$

What this says is: if the last action by the opponent was a c, then reward, otherwise (the last action by the opponent was a d) punish. To turn this into a meta-strategy for player 1, just swap all pairs:

$$(?(?\!?, c); (c, ?\!?) \ \cup \ ?(?\!?, d); (d, ?\!?))^*$$

Note that tit-for-tat for the PD game boils down to the same thing as the *copycat* meta-strategy, where a player always copies the last move of the opponent. So the player 1 copycat and player 2 copycat meta-strategies for the PD game are also given by the above strategy expressions.

7. A COMPLETE CALCULUS FOR MASL

To axiomatize this logic, we can use the well-known proof system for PDL [31, 23], with appropriate axioms for the strategy vectors added.

Call a strategy vector $\mathbf{c} = (t_1 \ldots, t_n)$ *determined* if for no $i \in N$ it is the case that $t_i = ?\!?$.

Vector axioms are:

1. Effectivity:

$$[\mathbf{c}]\mathbf{c}.$$

2. Seriality:

$$\langle \mathbf{c} \rangle \top.$$

3. Functionality:

$$\langle \mathbf{c} \rangle \phi \rightarrow [\mathbf{c}] \phi$$

for all determined strategy vectors $\mathbf{c}$.

4. Adversary power:

Let $\mathbf{c}$ have ⁇ in position i, and let $\mathbf{c}^i_a$ be the result of replacing ⁇ in position i in $\mathbf{c}$ by a. Then:

$$[\mathbf{c}]\phi \leftrightarrow \bigwedge_{a \in S_i} [\mathbf{c}^i_a]\phi.$$

Note that this uses the assumption that the set S_i of available actions for player i s finite.

5. Determinate current choice:

   Let $\mathbf{c}$ have ‼ in position i, and let $\mathbf{c}^i_a$ be the result of replacing ‼ at position i in $\mathbf{c}$ by a. Then:

$$(i_a, \overline{‼}) \rightarrow (\mathbf{c} \leftrightarrow \mathbf{c}^i_a).$$

The effectivity axiom says that execution of a strategy vector always makes the vector true.

The seriality axiom says that every strategy vector can be executed.

The functionality axiom says that determined strategy vectors are functional. This expresses that the outcome is determined if every player makes a determinate choice. This axiom does not hold for vectors that are not determined. The vector $(c, ⁇)$ has $|S_2|$ possible outcomes.

The adversary power axiom spells out what an adversary player can do. This defines the meaning of ⁇ terms.

The determinate current choice axiom fixes the meaning of ‼ terms.

These axioms are sound for the intended interpretation. Completeness can be shown by the usual canonical model construction for PDL (see [19, 6]):

THEOREM 1. *The calculus for MASL is complete.*

MASL has the same complexity for model checking and satisfiability as PDL: Model checking for PDL and MASL is PTIME-complete [20]. Sat solving for PDL and MASL is EXPTIME-complete [6]. Model checking for formulas that use only the modal fragment of MASL (modalities without Kleene star) can be done more efficiently, e.g., by using the algorithm of [13] that runs in time $O(|M| \times |\phi|)$.

The important thing to note is that the standard model checking tools for modal logic and PDL can now be used for strategic games, using the MASL extension of PDL.

8. CONNECTION TO COALITION LOGIC

Our approach links directly to coalition logic [27] (see also [4] for this connection). Coalition logic has the following syntax:

$$\phi \quad ::= \quad \top \mid p \mid \neg\phi \mid \phi \wedge \phi \mid [C]\phi$$

where p ranges over basic propositions, and $C \subseteq N$, with N the set of agents. Intended meaning of $[C]\phi$ is that the coalition C is able to force the game outcome to be in ϕ.

Again, let (N, S) be a strategic game form, let $o : S \rightarrow P$ be an outcome function, and let $s \in S$ be a strategy profile. Assume models M of the form (N, S, o). Coalition logic, the way it is presented in [27], is a bit mysterious about how valuations enter into game forms, but we can fix this by using the output functions in the same manner as in the semantics of MASL. Formulas of coalition logic are interpreted in strategy profiles s of M, as follows.

Recall that S_C is the set of group strategy functions for C, and that if $s \in S_C$ and $t \in S_{N-C}$, then (s, t) is the strategy profile where members of C choose according to s and all others choose according to t.

$$\begin{aligned}
M, s \models p \quad &\text{iff} \quad s \in o^{-1}(p).\\
M, s \models \neg\phi \quad &\text{iff} \quad M, s \not\models \phi.\\
M, s \models \phi_1 \wedge \phi_2 \quad &\text{iff} \quad M, s \models \phi_1 \text{ and } M, s \models \phi_2.\\
M, s \models [C]\phi \quad &\text{iff} \quad \exists t \in S_C \forall u \in S_{N-C}\\
&\qquad M, (t, u) \models \phi.
\end{aligned}$$

Let $\dot{C}$ be the set of all strategies for coalition C against all other players.

If we assume that for each player i the set S_i of possible strategies for i is finite, then $\dot{C}$ is finite as well, and $\dot{C}$ is defined by

$$\{(t_1, \ldots, t_n) \mid t_i \in S_i \text{ if } i \in C, t_i = ⁇ \text{ otherwise } \}.$$

This means we can construct the formula

$$\bigvee_{\mathbf{c} \in \dot{C}} [\mathbf{c}]\phi.$$

The translation instruction Tr for turning coalition logic into MASL becomes:

$$\begin{aligned}
\mathrm{Tr}(p) \quad &:= \quad p\\
\mathrm{Tr}(\neg\phi) \quad &:= \quad \neg\mathrm{Tr}(\phi)\\
\mathrm{Tr}(\phi_1 \wedge \phi_2) \quad &:= \quad \mathrm{Tr}(\phi_1) \wedge \mathrm{Tr}(\phi_2)\\
\mathrm{Tr}([C]\phi) \quad &:= \quad \bigvee_{\mathbf{c} \in \dot{C}} [\mathbf{c}]\mathrm{Tr}(\phi).
\end{aligned}$$

Induction on formula structure now proves:

THEOREM 2. *$M, s \models_{CL} \phi$ iff $M, s \models_{MASL} Tr(\phi)$.*

This assumes that the set of strategies for each agent is finite, as this is a basic assumption of MASL. This finiteness restriction aside, the main difference between coalition logic and MASL is that MASL is explicit about coalition strategies where coalition logic is not. Many key concepts of strategic game theory and voting theory that MASL can express are beyond the reach of coalition logic.

9. EPISTEMIC MASL

MASL uses PDL as an action logic for game actions. It is well-known that PDL also can be given an epistemic interpretation [5]. The language of Epistemic Multi Agent Strategy Logic (EMASL) combines the strategy interpretation of PDL with the epistemic interpretation of PDL. For that, a new set of PDL actions is thrown in, but this time with an epistemic/doxastic interpretation. Here is the extended language:

$$\begin{aligned}
\phi \quad &::= \quad \top \mid \mathbf{c} \mid p \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid [\gamma]\phi \mid [\alpha]\phi\\
\gamma \quad &::= \quad \mathbf{c} \mid ?\phi \mid \gamma_1; \gamma_2 \mid \gamma_1 \cup \gamma_2 \mid \gamma^*\\
\alpha \quad &::= \quad i \mid i^{\smile} \mid ?\phi \mid \alpha_1; \alpha_2 \mid \alpha_1 \cup \alpha_2 \mid \alpha^*
\end{aligned}$$

i ranges over the set N of agents. $i^{\smile}$ denotes the converse of the i relation.

The interpretations of the i operators (the atoms of α actions) can be arbitrary. Define $\mathbf{i}$ as $(i \cup i^{\smile})^*$, and you have a reflexive, symmetric and transitive knowledge operator (see [12]).

Note that tests appear both in the action expressions and in the epistemic expressions. Thus, actions can be conditioned by knowledge, and knowledge can refer to action. This allows the representation of strategies like "If I know that playing a results in ϕ, then play a, else play b" (action conditioned by knowledge), and the representation of epistemic relations expressing what will become known as a result of a certain strategy (knowledge referring to action).

To interpret this language, we define *intensional game forms* from (extensional) game forms. An intensional game form is a tuple

$$(N, W, R_1, \ldots, R_n)$$

where

- W is a set of pairs (G, s) where $G = (N, S)$ is a game form with $s \in S$,
- each R_i is a binary relation on W.

These intensional game forms can be viewed as Kripke frames. As before, they can be turned into models by using an output function $o : S \to P$ to define the valuation. For that, extend o to W by means of the stipulation saying that the output of a game-profile pair is determined by its profile component:

$$o(G', s') = o(s').$$

Since $s' \in S' \subseteq S$ for each S', this is well-defined.

Let M be an intensional game form $(N, W, R_1, \ldots, R_n)$ based on $G = (N, S)$ and let $o : W \to P$ be an output function that is extended from an output function $o : S \to P$ for G. Let $w \in W$.

Then the truth definition of EMASL formulas in M, w is given by (only clauses that differ from the MASL version shown):

$$\begin{array}{rcl} M, w \models \mathbf{c} & \text{iff} & w = ((N, S), s) \text{ and } s \in [\![\mathbf{c}]\!]^{S,s} \\ M, w \models [\alpha]\phi & \text{iff} & \text{for all } w' \text{ with } (w, w') \in [\![\alpha]\!]^M : \\ & & \quad M, w' \models \phi \\ \\ [\![\mathbf{c}]\!]^M & = & \{(w, w') \mid w = ((N, S), s), w' = ((N, S), t) \\ & & \text{with } t \in [\![\mathbf{c}]\!]^{S,s}\} \\ [\![i]\!]^M & = & R_i \\ [\![i\check{}]\!]^M & = & (R_i)\check{} \end{array}$$

One way to base an intensional game form on a game form $G = (N, S)$ is by putting $W = \{(G, s) \mid s \in S\}$ and

$$R_i = \{((G, s), (G, s')) \mid s[i] = s'[i]\}$$

for all $i \in N$. Call this the *epistemic lift* of G, and denote it with $G^{\#}$.

Then in $G^{\#}$ the accessibility relations express that every player can distinguish between her own actions, but not between those of other players.

For the PD game, this gives a model where every player knows her move, and the two possible strategies for her opponent. Furthermore, it is common knowledge that there is no coordination between the actions of the two players: the relation $(R_1 \cup R_2)^*$, denoted by the EMASL expression $(\mathbf{1} \cup \mathbf{2})^*$, is the whole set of strategy profiles.

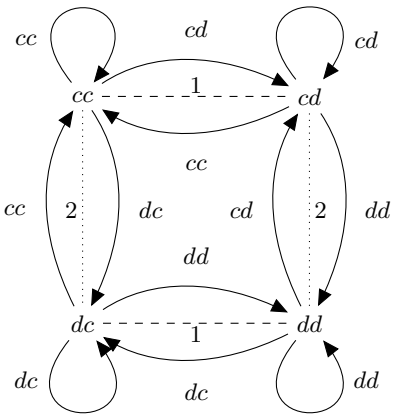


Figure 4: Epistemic PD Game Form

This is pictured in Figure 4, with dashed lines for the accessibilities of player 1, dotted lines for those of player 2, and reflexive epistemic arrows omitted.

In epistemic lifts of game forms it is common knowledge among all players what is the nature of the game; more in particular it is common knowledge what are the available strategic options for all players.

This assumption that the nature of the game is common knowledge is dropped for intensional game forms that are built by means of *strategy restrictions* from an (extensional) game form.

Let $G' \sqsubseteq G$ if $G = (N, \{S_i \mid i \in N\})$, $G' = (N, \{S'_i \mid i \in N\})$, and for all $i \in N$: $S'_i \subseteq S_i$. Call G' a strategy restriction of G.

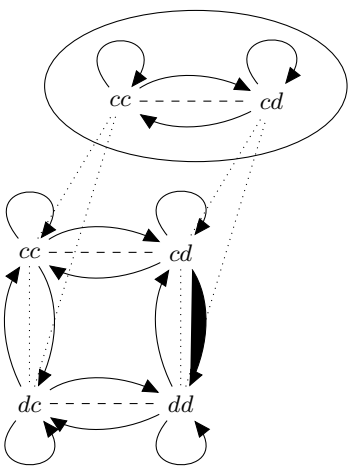


Figure 5: Restriction in PD Game.

An intensional game form built from strategy restriction from PD is given in Figure 5. This pictures a situation where the first player is committed to c, but the other player does not know this. The oval indicates the actual game; this is confused by player 2 with the full PD game (dotted lines for the accessibility relation of player 2).

10. EXPRESSIVENESS OF EMASL

EMASL extends MASL, so every concept from game theory, voting theory and social choice theory that is expressible in MASL is expressible in EMASL. Many concepts from social choice theory have epistemic versions. Here is one example.

Knowing Dictatorship.

A dictator in a multi agent game was defined as a player who is always able to get the best deal. A knowing dictator is a player who not only has this ability, but also *knows* that he has it:

$$[\mathbf{i}] \bigvee_{v \in U} \bigwedge_{j \in N-\{i\}} [(?)](\neg u_j > v \wedge \langle (i, !!) \rangle u_i \geq v).$$

As an example, consider player 2 in the game pictured in Figure 5, with an output function giving appropriate utilities for the PD game. Player 2 is a dictator for this game, for he can force the outcome *cd*, with best payoff for 2. But player 2 is not aware of this fact: for all he knows, he could end up in state *dd*, with worse payoff for him than *cd*.

Gibbard-Satterthwaite, Epistemically.

Resoluteness, strategy-proofness, non-imposedness, dictatorship and knowing dictatorship are all expressible in EMASL. Here is a new type of question. Consider the class of epistemic lifts $G^{\#}$ of strategic game forms G based on resolute, strategy-proof and non-imposed voting rules. Then the MASL proof of the GS theorem lifts to EMASL, so every such game has a dictator. But does every such game also have a knowing dictator? What are the minimum epistemic conditions to make the epistemic GS theorem go through in intensional games? It also make sense to formulate an epistemic version of strategy-proofness, stating that players do not know that they can improve their payoff by voting strategically. This is a weakening of strategy-proofness, and we can investigate under which epistemic conditions it is enough to derive GS, or to derive epistemic GS.

11. A CALCULUS FOR EMASL

There are various classes of intensional game models that one might want to axiomatize. As an example, we consider the class of epistemic lift models $(G^{\#}, o)$, where $G = (N, S)$ is a finite strategic game form and $o : S \to P$ is an output function for G.

Notice that the axioms of MASL are sound for this class, so that we can extend the calculus for MASL, to get a calculus for reasoning about epistemic lift models, as follows.

- Propositional axioms, modus ponens, necessitation for γ and α.
- PDL axioms for γ modalities.
- PDL axioms for α modalities.
- The five MASL vector axioms.
- $\phi \to [i]\langle i^{\smile} \rangle \phi$.
- $\phi \to [i^{\smile}]\langle i \rangle \phi$.
- $[(i_a, !!)][\mathbf{i}](i_a, !!)$.
- $\bigwedge_{j \in N-\{i\}} [(j_a, !!)] \neg [\mathbf{i}](j_a, !!)$.

The two axioms for $i^{\smile}$ are the standard modal axioms for converse. The first axiom for $[\mathbf{i}]$ expresses that player i can distinguish between his own actions, and the second axiom for $[\mathbf{i}]$ expresses that i cannot distinguish between the actions of other players. This gives a sound and complete system for reasoning about epistemic lift models.

12. RELATED AND FURTHER WORK

The present approach is closest to [4], to which it is indebted. Instead of constructing group strategies from individual strategies by relation intersection, we take complete group strategies as basic in the semantics, and construct strategies for subgroups and individuals by relation union.

Strategic reasoning is related to multimodal logic K_n with intersection in [1], where group strategies are constructed from individual strategies by means of relation intersection. Strategies are not explicit, and many key notions from game theory are not expressible.

The present approach is close to [18, 21] where PDL is taken as a starting point to formulate expressive STIT logics for analzying agency in games. In [33] a special purpose logic for reasoning about social choice functions is proposed, with an analysis to the concept of strategy-proofness, in terms of a modality for expressing that certain players stick to their current choice.

Coalition logic is a close kin of Alternating-time Temporal Logic [2, 16]. This has various extensions, of which CATL [34] deals explicitly with strategic reasoning. An important difference with the present approach is that ATL and CATL focus on extensive rather than strategic games.

Strategic reasoning is also the topic of game logics such as [24, 25, 28]. These logics focus on the theory of two-player games, and also use the regular operations for strategy construction. The imporant difference is that in game logic the regular operations are applied to single player strategies. We hope to study the connection with game logic and with game algebra [17, 35] more precisely in future work.

Our work provides a framework for extending the exploration of knowledge-theoretic properties of strategic voting in [8]. In [26] the notion of knowledge manipulation in games is discussed, which is in the compass of EMASL, as is the analysis of the role of knowledge and ignorance in voting manipulation in [9].

There are two important limitations of MASL and EMASL, in the versions presented here: the restriction to finite ranges of individual actions, and the restriction to strategic games.

The first restriction could be lifted by extending the language with quantification over i-strategies. To lift the second restriction, one could introduce a register for keeping track of the players that have made their move. This would get us closer to ATL, for such a register can be viewed as a clock. It remains to be seen whether this extension could still be handled naturally by a PDL-based approach.

In [5], epistemic PDL is used as a base system to which operators for communication and change are added. In future work, we hope to extend the framework of EMASL in a similar way with communication and changes operators. Communications change the epistemics of the game by informing players about strategies of other players. Change operations change the game by changing outcomes or utilities. The switch from plurality voting to plurality voting with tie breaking could be modelled as such a change.

Acknowledgements

This paper has benefitted from conversations with Johan van Benthem, Ulle Endriss, Floor Sietsma and Sunil Simon. We also wish to thank four anonymous TARK reviewers for comments and suggestions.

13. REFERENCES

[1] T. Agotnes and N. Alechina. Reasoning about joint action and coalitional ability in K_n with intersection. In J. Leite, P. Torroni, T. Agotnes, G. Boella, and L. van der Torre, editors, *Computational Logic in Multi-Agent Systems – 12th International Workshop (CLIMA XII)*, number 6814 in Lecture Notes in Computer Science, pages 139–156. Springer, 2011.

[2] R. Alur, T. Henzinger, and O. Kupferman. Alternating-time temporal logic. *Journal of the ACM*, 49:672–713, 2002.

[3] R. Axelrod. *The Evolution of Cooperation*. Basic Books, New York, 1984.

[4] J. v. Benthem. In praise of strategies. In J. v. Eijck and R. Verbrugge, editors, *Games, Actions, and Social Software*, volume 7010 of *Texts in Logic and Games, LNAI*, pages 105–125. Springer Verlag, Berlin, 2012.

[5] J. v. Benthem, J. van Eijck, and B. Kooi. Logics of communication and change. *Information and Computation*, 204(11):1620–1662, 2006.

[6] P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic*. Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, 2001.

[7] S. J. Brams and P. C. Fishburn. Voting procedures. In K. Arrow, A. Sen, and K. Suzumura, editors, *Handbook of Social Choice and Welfare*, volume I, chapter 4. Elsevier, 2002.

[8] S. Chopra, E. Pacuit, and R. Parikh. Knowledge-theoretic properties of strategic voting. In *Proceedings of the 9th European Conference on Logics in Artificial Intelligence (JELIA-2004)*, pages 18–30, 2004.

[9] V. Conitzer, T. Walsh, and L. Xia. Dominating manipulations in voting with partial information. In W. Burgard and D. Roth, editors, *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence*, pages 638–643. AAAI Press, 2011.

[10] J. v. Eijck and F. Sietsma. Strategic reasoning in social software. In J. van Benthem, S. Ghosh, and R. Verbrugge, editors, *Modeling Strategic Reasoning*, Texts in Logic and Games. Springer, to appear.

[11] J. v. Eijck and R. Verbrugge, editors. *Games, Actions, and Social Software*, volume 7010 of *Texts in Logic and Games, LNAI*. Springer Verlag, Berlin, 2012.

[12] J. v. Eijck and Y. Wang. Propositional Dynamic Logic as a logic of belief revision. In W. Hodges and R. de Queiros, editors, *Proceedings of Wollic'08*, number 5110 in Lecture Notes in Artificial Intelligence, pages 136–148. Springer, 2008.

[13] R. Fagin, J. Halpern, Y. Moses, and M. Vardi. *Reasoning about Knowledge*. MIT Press, 1995.

[14] R. Farquharson. *Theory of Voting*. Blackwell, 1969.

[15] A. Gibbard. Manipulation of voting schemes: A general result. *Econometrica*, 41:587–601, 1973.

[16] V. Goranko. Coalition games and alternating temporal logics. In *TARK '01: Proceedings of the 8th conference on Theoretical aspects of rationality and knowledge*, pages 259–272, 2001.

[17] V. Goranko. The basic algebra of game equivalences. *Studia Logica*, 75, 2003.

[18] A. Herzig and E. Lorini. A dynamic logic of agency I: STIT, abilities and powers. *Journal of Logic, Language and Information*, 19(1):89–121, 2010.

[19] D. Kozen and R. Parikh. An elementary proof of the completeness of PDL. *Theoretical Computer Science*, 14:113–118, 1981.

[20] M. Lange. Model checking propositional dynamic logic with all extras. *Journal of Applied Logic*, 4(1):39–49, March 2006.

[21] E. Lorini. A dynamic logic of agency II: Deterministic DLA coalition logic, and game theory. *Journal of Logic, Language and Information*, 19(3):327–351, 2010.

[22] M. Osborne and A. Rubinstein. *A Course in Game Theory*. MIT Press, 1994.

[23] R. Parikh. The completeness of propositional dynamic logic. In *Mathematical Foundations of Computer Science 1978*, pages 403–415. Springer, 1978.

[24] R. Parikh. Propositional game logic. In *IEEE Symposium on Foundations of Computer Science*, pages 195–200, 1983.

[25] R. Parikh. The logic of games and its applications. *Annals of Discrete Mathematics*, 24:111–140, 1985.

[26] R. Parikh, Tasdemir, and A. Witzel. The power of knowledge in games. In J. v. Eijck and R. Verbrugge, editors, *Proceedings of the Workshop on Reasoning About Other Minds: Logical and Cognitive Perspectives (RAOM-2011)*, number 755 in CEUR Workshop Proceedings, 2011.

[27] M. Pauly. *Logic for Social Software*. PhD thesis, ILLC, Amsterdam, 2001.

[28] M. Pauly and R. Parikh. Game logic — an overview. *Studia Logica*, 75(2):165–182, 2003.

[29] V. Pratt. Semantical considerations on Floyd–Hoare logic. *Proceedings 17th IEEE Symposium on Foundations of Computer Science*, pages 109–121, 1976.

[30] M. Satterthwaite. Strategy-proofness and Arrow's conditions: Existence and correspondence theorems for voting procedures and social welfare functions. *Journal of Economic Theory*, 10:187–217, 1975.

[31] K. Segerberg. A completeness theorem in the modal logic of programs. In T. Traczyck, editor, *Universal Algebra and Applications*, pages 36–46. Polish Science Publications, 1982.

[32] A. D. Taylor. *Social Choice and the Mathematics of Manipulation*. Cambridge University Press, 2005.

[33] N. Troquard, W. van der Hoek, and M. Woodridge. Reasoning about social choice functions. *Journal of Philosophical Logic*, 40:473–498, 2011.

[34] W. van der Hoek, W. Jamroga, and M. Wooldridge. A logic for strategic reasoning. In *Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems*, AAMAS '05, pages 157–164, New York, NY, USA, 2005. ACM.

[35] Y. Venema. Representing game algebras. *Studia Logica*, 75, 2003.

Game Theory with Translucent Players

Joseph Y. Halpern*
Cornell University
Dept. Computer Science
Ithaca, NY 14853, USA
halpern@cs.cornell.edu

Rafael Pass†
Cornell University
Dept. Computer Science
Ithaca, NY 14853, USA
rafael@cs.cornell.edu

ABSTRACT

A traditional assumption in game theory is that players are opaque to one another—if a player changes strategies, then this change in strategies does not affect the choice of other players' strategies. In many situations this is an unrealistic assumption. We develop a framework for reasoning about games where the players may be *translucent* to one another; in particular, a player may believe that if she were to change strategies, then the other player would also change strategies. Translucent players may achieve significantly more efficient outcomes than opaque ones.

Our main result is a characterization of strategies consistent with appropriate analogues of common belief of rationality. *Common Counterfactual Belief of Rationality (CCBR)* holds if (1) everyone is rational, (2) everyone counterfactually believes that everyone else is rational (i.e., all players i believe that everyone else would still be rational even if i were to switch strategies), (3) everyone counterfactually believes that everyone else is rational, and counterfactually believes that everyone else is rational, and so on. CCBR characterizes the set of strategies surviving iterated removal of *minimax dominated* strategies: a strategy σ_i is minimax dominated for i if there exists a strategy σ_i' for i such that $\min_{\mu'_{-i}} u_i(\sigma_i', \mu'_{-i}) > \max_{\mu_{-i}} u_i(\sigma_i, \mu_{-i})$.

Categories and Subject Descriptors

F.4.1 [**Mathematical Logic and Formal Languages**]: Mathematical Logic—*modal logic*; I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*multiagent systems*; J.4 [**Social and Behavioral Sciences**]: Economics

General Terms

Economics, Theory

*Halpern is supported in part by NSF grants IIS-0812045, IIS-0911036, and CCF-1214844, by AFOSR grant FA9550-08-1-0266, and by ARO grant W911NF-09-1-0281.

†Pass is supported in part by a Alfred P. Sloan Fellowship, Microsoft New Faculty Fellowship, NSF Award CNS-1217821, NSF CAREER Award CCF-0746990, NSF Award CCF-1214844, AFOSR YIP Award FA9550-10-1-0093, and DARPA and AFRL under contract FA8750-11-2- 0211. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the US Government

TARK 2013, Chennai, India.

Keywords

Epistemic logic, rationality, counterfactuals

1. INTRODUCTION

Two large firms 1 and 2 need to decide whether to *cooperate (C)* or *sue (S)* the other firm. Suing the other firm always has a small positive reward, but being sued induces a high penalty p; more precisely, $u(C, C) = (0, 0); u(C, S) = (-p, r); u(S, C) = (r, -p), u(S, S) = (r - p, r - p)$. In other words, we are considering an instance of the Prisoner's Dilemma.

But there is a catch. Before acting, each firms needs to discuss their decision with its board. Although these discussions are held behind closed doors, there is always the possibility of the decision being "leaked"; as a consequence, the other company may change its course of action. Furthermore, both companies are aware of this fact. In other words, the players are *translucent* to one another.

In such a scenario, it may well be rational for both companies to cooperate. For instance, consider the following situation.

- Firm i believes that its action is leaked to firm $2 - i$ with probability ϵ.
- Firm i believes that if the other firm $2 - i$ finds out that i is defecting, then $2 - i$ will also defect.
- Finally, $p\epsilon > r$ (i.e., the penalty for being sued is significantly higher than the reward of suing the other company).

Neither firm defects, since defection is noticed by the other firm with probability ϵ, which (according to their beliefs) leads to a harsh punishment. Thus, the possibility of the players' actions being leaked to the other player allows the players to significantly improve social welfare in equilibrium. (This suggests that it may be mutually beneficial for two countries to spy on each other!)

Even if the Prisoner's dilemma is not played by corporations but by individuals, each player may believe that if he chooses to defect, his "guilt" over defecting may be visible to the other player. (Indeed, facial and bodily cues such as increased pupil size are often associated with deception; see e.g., [Ekman and Friesen 1969].) Thus, again, the players may choose to cooperate out of fear that if they defect, the other player may detect it and act on it.

Our goal is to capture this type of reasoning formally. We take a Bayesian approach: Each player has a (subjective) probability distribution (describing the player's beliefs) over the states of the world. Traditionally, a player i is said to be rational in a state ω if the strategy σ_i that i plays at ω is a best response to the strategy profile μ_{-i} of the other players induced by i's beliefs in ω;[1] that is,

[1]Formally, we assume that i has a distribution on states, and at each

$u_i(\sigma_i, \mu_{-i}) \geq u_i(\sigma'_i, \mu_{-i})$ for all alternative strategies σ'_i for i. In our setting, things are more subtle. Player i may believe that if she were to switch strategies from σ_i to σ'_i, then players other than i might also switch strategies. We capture this using *counterfactuals* [Lewis 1973; Stalnaker 1968].[2] Associated with each state of the world ω, each player i, and $f(\omega, i, \sigma'_i)$ where player i plays σ'_i. Note that if i changes strategies, then this change in strategies may start a chain reaction, leading to further changes. We can think of $f(\omega, i, \sigma'_i)$ as the steady-state outcome of this process: the state that would result if i switched strategies to σ'_i. Let $\mu_{f(\omega,i,\sigma'_i)}$ be the distribution on strategy profiles of $-i$ (the players other than i) induced by i's beliefs at ω about the steady-state outcome of this process. We say that i is rational at a state ω where i plays σ_i and has beliefs μ_i if $u_i(\sigma_i, \mu_{-i}) \geq u_i(\sigma'_i, \mu_{f(\omega,i,\sigma'_i)})$ for every alternative strategy σ'_i for i. Note that we have required the closest-state function to be deterministic, returning a unique state, rather than a distribution over states. While this may seem incompatible with the motivating scenario, it does not seem so implausible in our context that, by taking a rich enough representation of states, we can assume that a state contains enough information about players to resolve uncertainty about what strategies they would use if one player were to switch strategies.

We are interested in considering analogues to rationalizability in a setting with translucent players, and providing epistemic characterizations of them. To do that, we need some definitions. We say that a player i *counterfactually believes* φ at ω if i believes φ holds even if i were to switch strategies. *Common Counterfactual Belief of Rationality (CCBR)* holds if (1) everyone is rational, (2) everyone counterfactually believes that everyone else is rational (i.e., all players i believe that everyone else would still be still rational even if i were to switch strategies), (3) everyone counterfactually believes that everyone else is rational, and counterfactually believes that everyone else is rational, and so on.

Our main result is a characterization of strategies consistent with CCBR. Roughly speaking, these results can be summarized as follows:

- If the closest-state function respects "unilateral deviations"—when i switches strategies, the strategies and beliefs of players other than i remain the same—then CCBR characterizes the set of rationalizable strategies.

- If the closest-state function can be arbitrary, CCBR characterizes the set of strategies that survive iterated removal of *minimax dominated* strategies: a strategy σ_i is minimax dominated for i if there exists a strategy σ'_i for i such that $\min_{\mu'_{-i}} u_i(\sigma'_i, \mu'_{-i}) > \max_{\mu_{-i}} u_i(\sigma_i, \mu_{-i})$; that is, $u_i(\sigma'_i, \mu'_{-i}) > u_i(\sigma_i, \mu_{-i})$ no matter what the strategy profiles μ_{-i} and μ'_{-i} are.

We also consider analogues of Nash equilibrium in our setting, and show that individually rational strategy profiles that survive iterated removal of minimax dominated strategies characterize such equilibria.

Note that in our approach, each player i has a *belief* about how the other players' strategies would change if i were to change strategies, but we do not require i to explicitly specify how he would respond to other people changing strategies. The latter approach, of having each player pick a "meta-strategy" that takes as input the strategy of other players, was explored by Howard [1971] in the 1970s. It led to complex formalisms involving infinite hierachies of meta-strategies: at the lowest level, each player specifies a strategy in the original game; at level k, each player specifies a "response rule" (i.e., a meta-strategy) to other players' $(k-1)$-level response rules. Such hierarchical structures have not proven useful when dealing with applications. Since we do not require players to specify reaction rules, we avoid the complexities of this approach.

Program equilibria [Tennenholz 2004] and *conditional commitments* [Kalai et al. 2010] provide a different approach to avoiding infinite hierarchies. Roughly speaking, each player i simply specifies a *program* Π_i; player i's action is determined by running i's program on input the (description of) the programs of the other players; that is, i' action is given by $\Pi_i(\Pi_{-i})$. Tennenholtz [2004] and Kalai et al. [2010] show that every (correlated) individually rational outcome can be sustained in a program equilibrium. Their model, however, assumes that player's programs (which should be interpreted as their "plan of action") are commonly known to all players. We dispense with this assumption. It is also not clear how to define common belief of rationality in their model; the study of program equilibria and conditional commitments has considered only analogues of Nash equilibrium.

Counterfactuals have been explored in a game-theoretic setting; see, for example, [Aumann 1995; Halpern 1999; Samet 1996; Stalnaker 1996; Zambrano 2004]. However, all these papers considered only structures where, in the closest state where i changes strategies, all other players' strategies remain the same; thus, these approaches are not applicable in our context.

2. COUNTERFACTUAL STRUCTURES

Given a game Γ, let $\Sigma_i(\Gamma)$ denote player i's pure strategies in Γ (we occasionally omit the parenthetical Γ if it is clear from context or irrelevant).

To reason about the game Γ, we consider a class of Kripke structures corresponding to Γ. For simplicity, we here focus on finite structures. A *finite probability structure M appropriate for* Γ is a tuple $(\Omega, \mathbf{s}, \mathcal{PR}_1, \ldots, \mathcal{PR}_n)$, where Ω is a finite set of states; $\mathbf{s}$ associates with each state $\omega \in \Omega$ a pure strategy profile $\mathbf{s}(\omega)$ in the game Γ; and, for each player i, $\mathcal{PR}_i$ is a *probability assignment* that associates with each state $\omega \in \Omega$ a probability distribution $\mathcal{PR}_i(\omega)$ on Ω, such that

1. $\mathcal{PR}_i(\omega)([\![\mathbf{s}_i(\omega)]\!]_M) = 1$, where for each strategy σ_i for player i, $[\![\sigma_i]\!]_M = \{\omega : \mathbf{s}_i(\omega) = \sigma_i\}$, where $\mathbf{s}_i(\omega)$ denotes player i's strategy in the strategy profile $\mathbf{s}(\omega)$;

2. $\mathcal{PR}_i(\omega)([\![\mathcal{PR}_i(\omega), i]\!]_M) = 1$, where for each probability measure π and player i, $[\![\pi, i]\!]_M = \{\omega : \mathcal{PR}_i(\omega) = \pi\}$.

state, a pure strategy profile is played; the distribution on states clearly induces a distribution on strategy profiles for the players other than i, which we denote μ_{-i}.

[2]A different, more direct, approach for capturing our original motivating example would be to consider and analyze an extensive-form variant G' of the original normal-form game G that explicitly models the "leakage" of players' actions in G, allows the player to react to these leakage signals by choosing a new action in G, which again may be leaked and the players may react to, and so on. Doing this is subtle. We would need to model how players respond to receiving leaked information, and to believing that there was a change in plan even if information wasn't leaked. To make matters worse, it's not clear what it would mean that a player is "intending" to perform an action a if players can revise what they do as the result of a leak. Does it mean that a player will do a if no information is leaked to him? What if no information is leaked, but he believes that the other side is planning to change their plans in any case? In addition, modeling the game in this way would require a distribution over leakage signals to be exogenously given (as part of the description of the game G'). Moreover, player strategies would have to be infinite objects, since there is no bound on the sequence of leaks and responses to leaks. In constrast, using counterfactuals, we can directly reason about the original (finite) game G.

These assumptions say that player i assigns probability 1 to his actual strategy and beliefs.

To deal with counterfactuals, we augment probability structures with a "closest-state" function f that associates with each state ω, player i, and strategy σ_i', a state $f(\omega, i, \sigma_i)$ where player i plays σ'; if σ' is already played in ω, then the closest state to ω where σ' is played is ω itself. Formally, a *finite counterfactual structure* M *appropriate for* Γ is a tuple $(\Omega, \mathbf{s}, f, \mathcal{PR}_1, \ldots, \mathcal{PR}_n)$, where $(\Omega, \mathbf{s}, \mathcal{PR}_1, \ldots, \mathcal{PR}_n)$ is a probability structure appropriate for Γ and f is a "closest-state" function. We require that if $f(\omega, i, \sigma_i') = \omega'$, then

1. $\mathbf{s}_i(\omega') = \sigma'$;
2. if $\sigma_i' = \mathbf{s}_i(\omega)$, then $\omega' = \omega$.

Given a probability assignment $\mathcal{PR}_i$ for player i, we define i's counterfactual belief at state ω ("what i believes would happen if he switched to σ_i' at ω) as

$$\mathcal{PR}^c_{i,\sigma_i'}(\omega)(\omega') = \sum_{\{\omega'' \in \Omega : f(\omega'', i, \sigma_i') = \omega'\}} \mathcal{PR}_i(\omega)(\omega'').$$

Note that the conditions above imply that each player i knows what strategy he would play if he were to switch; that is, $\mathcal{PR}^c_{i,\sigma_i'}(\omega)([\![\sigma_i']\!]_M) = 1$.

Let $Supp(\pi)$ denote the support of the probability measure π. Note that $Supp(\mathcal{PR}^c_{i,\sigma_i'}(\omega)) = \{f(\omega', i, \sigma_i') : \omega' \in Supp(\mathcal{PR}_i(\omega)\}$. Moreover, it is almost immediate from the definition that if $\mathcal{PR}_i(\omega) = \mathcal{PR}_i(\omega')$, then $\mathcal{PR}^c_{i,\sigma_i'}(\omega) = \mathcal{PR}^c_{i,\sigma_i'}(\omega')$ for all strategies σ_i' for player i. But it does *not* in general follow that i knows his counterfactual beliefs at ω, that is, it may not be the case that for all strategies σ_i' for player i, $\mathcal{PR}^c_{i,\sigma_i'}(\omega)([\![\mathcal{PR}^c_{i,\sigma_i'}(\omega), i]\!]_M) = 1$. Suppose that we think of a state as representing each player's *ex ante* view of the game. The fact that player $\mathbf{s}_i(\omega) = \sigma_i$ should then be interpreted as "i *intends* to play σ_i at state ω." With this view, suppose that ω is a state where $\mathbf{s}_i(\omega)$ is a conservative strategy, while σ_i' is a rather reckless strategy. It seems reasonable to expect that i's subjective beliefs regarding the likelihood of various outcomes may depend in part on whether he is in a conservative or reckless frame of mind. We can think of $\mathcal{PR}^c_{i,\sigma_i'}(\omega)(\omega')$ as the probability that i ascribes, at state ω, to ω' being the outcome of i switching to strategy σ_i'; thus, $\mathcal{PR}^c_{i,\sigma_i'}(\omega)(\omega')$ represents i's evaluation of the likelihood of ω' when he is in a conservative frame of mind. This may not be the evaluation that i uses in states in the support $\mathcal{PR}^c_{i,\sigma_i'}(\omega)$; at all these states, i is in a "reckless" frame of mind. Moreover, there may not be a unique reckless frame of mind, so that i may not have the same beliefs at all the states in the support of $\mathcal{PR}^c_{i,\sigma_i'}(\omega)$.

M is a *strongly appropriate counterfactual structure* if it is an appropriate counterfactual structure and, at every state ω, every player i knows his counterfactual beliefs. As the example above suggests, strong appropriateness is a nontrivial requirement. As we shall see, however, our characterization results hold in both appropriate and strongly appropriate counterfactual structures.

Note that even in strongly appropriate counterfactually structures, we may not have $\mathcal{PR}_i(f(\omega, i, \sigma_i')) = \mathcal{PR}^c_{i,\sigma_i'}(\omega)$. We do have $\mathcal{PR}_i(f(\omega, i, \sigma_i')) = \mathcal{PR}^c_{i,\sigma_i'}(\omega)$ in strongly appropriate counterfactual structures if $f(\omega, i, \sigma_i')$ is in the support of $\mathcal{PR}^c_{i,\sigma_i'}(\omega)$ (which will certainly be the case if ω is in the support of $\mathcal{PR}_i(\omega)$). To see why we may not want to have $\mathcal{PR}_i(f(\omega, i, \sigma_i')) = \mathcal{PR}^c_{i,\sigma_i'}(\omega)$ in general, even in strongly appropriate counterfactual structures, consider the example above again. Suppose that, in state ω, although i does not realize it, he has been given a drug that affects how he evaluates the state. He thus ascribes probability 0 to ω. In $f(\omega, i, \sigma_i')$ he has also been given the drug, and the drug in particular affects how he evaluates outcomes. Thus, i's beliefs in the state $f(\omega, i, \sigma_i')$ are quite different from his beliefs in all states in the support of $\mathcal{PR}^c_{i,\sigma_i'}(\omega)$.

2.1 Logics for Counterfactual Games

Let $\mathcal{L}(\Gamma)$ be the language where we start with $true$ and the primitive proposition RAT_i and $play_i(\sigma_i)$ for $\sigma_i \in \Sigma_i(\Gamma)$, and close off under the modal operators B_i (player i believes) and B_i^* (player i counterfactually believes) for $i = 1, \ldots, n$, CB (common belief), and CB^* (common counterfactual belief), conjunction, and negation. We think of $B_i\varphi$ as saying that "i believes φ holds with probability 1" and $B_i^*\varphi$ as saying "i believes that φ holds with probability 1, even if i were to switch strategies".

Let $\mathcal{L}^0$ be defined exactly like $\mathcal{L}$ except that we exclude the "counterfactual" modal operators B^* and CB^*. We first define semantics for $\mathcal{L}^0$ using probability structures (without counterfactuals). We define the notion of a formula φ being true at a state ω in a probability structure M (written $(M, w) \models \varphi$) in the standard way, by induction on the structure of φ, as follows:

- $(M, \omega) \models true$ (so $true$ is vacuously true).
- $(M, \omega) \models play_i(\sigma_i)$ iff $\sigma_i = \mathbf{s}_i(\omega)$.
- $(M, \omega) \models \neg\varphi$ iff $(M, \omega) \not\models \varphi$.
- $(M, \omega) \models \varphi \wedge \varphi'$ iff $(M, \omega) \models \varphi$ and $(M, \omega) \models \varphi'$.
- $(M, \omega) \models B_i\varphi$ iff $\mathcal{PR}_i(\omega)([\![\varphi]\!]_M) = 1$, where $[\![\varphi]\!]_M = \{\omega : (M, \omega) \models \varphi\}$.
- $(M, \omega) \models RAT_i$ iff $\mathbf{s}_i(\omega)$ is a best response given player i's beliefs regarding the strategies of other players induced by $\mathcal{PR}_i$.
- Let $EB\varphi$ ("everyone believes φ") be an abbreviation of $B_1\varphi \wedge \ldots \wedge B_n\varphi$; and define $EB^k\varphi$ for all k inductively, by taking $EB^1\varphi$ to be $EB\varphi$ and $EB^{k+1}\varphi$ to be $EB(EB^k\varphi)$.
- $(M, \omega) \models CB\varphi$ iff $(M, \omega) \models EB^k\varphi$ for all $k \geq 1$.

Semantics for $\mathcal{L}^0$ in counterfactual structures is defined in an identical way, except that we redefine RAT_i to take into account the fact that player i's beliefs about the strategies of players $-i$ may change if i changes strategies.

- $(M, \omega) \models RAT_i$ iff for every strategy σ_i' for player i,

$$\sum_{\omega' \in \Omega} \mathcal{PR}_i(\omega)(\omega') u_i(\mathbf{s}_i(\omega), \mathbf{s}_{-i}(\omega')) \geq \sum_{\omega' \in \Omega} \mathcal{PR}^c_{i,\sigma_i'}(\omega)(\omega') u_i(\sigma_i', \mathbf{s}_{-i}(\omega')).$$

The condition above is equivalent to requiring that

$$\sum_{\omega' \in \Omega} \mathcal{PR}_i(\omega)(\omega') u_i(\mathbf{s}_i(\omega), \mathbf{s}_{-i}(\omega')) \geq \sum_{\omega' \in \Omega} \mathcal{PR}_i(\omega)(\omega') u_i(\sigma_i', \mathbf{s}_{-i}(f(\omega', i, \sigma_i'))).$$

Note that, in general, this condition is different from requiring that $\mathbf{s}_i(\omega)$ is a best reponse given player i's beliefs regarding the strategies of other players induced by $\mathcal{PR}_i$.

To give the semantics for $\mathcal{L}$ in counterfactual structures, we now also need to define the semantics of B_i^* and CB^*:

- $(M,\omega) \models B_i^*\varphi$ iff for all strategies $\sigma_i' \in \Sigma_i(\Gamma)$, $\mathcal{PR}^c_{i,\sigma_i'}(\omega)([\![\varphi]\!]_M) = 1$.

- $(M,\omega) \models CB^*\varphi$ iff $(M,\omega) \models (EB^*)^k\varphi$ for all $k \geq 1$.

It is easy to see that, like B_i, B_i^* depends only on i's beliefs; as we observed above, if $\mathcal{PR}_i(\omega) = \mathcal{PR}_i(\omega')$, then $\mathcal{PR}^c_{i,\sigma_i'}(\omega) = \mathcal{PR}^c_{i,\sigma_i'}(\omega')$ for all σ_i', so $(M,\omega) \models B_i^*\varphi$ iff $(M,\omega') \models B_i^*\varphi$. It immediately follows that $B_i^*\varphi \Rightarrow B_iB_i^*\varphi$ is valid (i.e., true at all states in all structures).

The following abbreviations will be useful in the sequel. Let RAT be an abbreviation for $RAT_1 \wedge \ldots \wedge RAT_n$, and let $play(\vec{\sigma})$ be an abbreviation for $play_1(\sigma_1) \wedge \ldots \wedge play_n(\sigma_n)$.

2.2 Common Counterfactual Belief of Rationality

We are interested in analyzing strategies being played at states where (1) everyone is rational, (2) everyone counterfactually believes that everyone else is rational (i.e., for every player i, i believes that everyone else would still be rational even if i were to switch strategies), (3) everyone counterfactually believes that everyone else is rational, and counterfactually believes that everyone else is rational, and so on. For each player i, define the formulas $SRAT_i^k$ (player i is strongly k-level rational) inductively, by taking $SRAT_i^0$ to be *true* and $SRAT_i^{k+1}$ to be an abbreviation of

$$RAT_i \wedge B_i^*(\wedge_{j \neq i} SRAT_j^k).$$

Let $SRAT^k$ be an abbreviation of $\wedge_{j=1}^n SRAT_j^k$.

Define $CCBR$ (common counterfactual belief of rationality) as follows:

- $(M,\omega) \models CCBR$ iff $(M,\omega) \models SRAT^k\varphi$ for all $k \geq 1$.

Note that it is critical in the definition of $SRAT_i^k$ that we require only that player i counterfactually believes that everyone else (i.e., the players other than i) are rational, and believe that everyone else is rational, and so on. Player i has no reason to believe that his own strategy would be rational if he were to switch strategies; indeed, $B_i^*RAT_i$ can hold only if *every* strategy for player i is rational with respect to i's beliefs. This is why we do not define $CCBR$ as CB^*RAT.[3]

We also consider the consequence of just common belief of rationality in our setting. Define $WRAT_i^k$ (player i is weakly k-level rational) just as $SRAT_i^k$, except that B_i^* is replaced by B_i. An easy induction on k shows that $WRAT^{k+1}$ implies $WRAT^k$ and that $WRAT^k$ implies $B_i(WRAT^k)$.[4] It follows that we could have equivalently defined $WRAT_i^{k+1}$ as

$$RAT_i \wedge B_i(\wedge_{j=1}^n WRAT_j^k).$$

Thus, $WRAT^{k+1}$ is equivalent to $RAT \wedge EB(WRAT^k)$. As a consequence we have the following:

PROPOSITION 2.1: *$(M,\omega) \models CB(RAT)$ iff $(M,\omega) \models WRAT^k$ for all $k \geq 0$.*

[3]Interestingly, Samet [1996] essentially considers an analogue of CB^*RAT. This works in his setting since he is considering only events in the past, not events in the future.

[4]We can also show that $SRAT^{k+1}$ implies $SRAT^k$, but it is not the case that $SRAT_i^k$ implies $B_i^*SRAT_i^k$, since RAT does not imply B_i^*RAT.

3. CHARACTERIZING COMMON COUNTERFACTUAL BELIEF OF RATIONALITY

It is well known rationalizability can be characterized in terms of common belief of common belief of rationality in probability structures [**?**; **?**]. In the full version of the paper[5] we show that if we restrict to counterfactual structures that *respect unilateral deviations*—where in the closest state to ω where player i switches strategies, everybody else's strategy and beliefs remain same—common counterfactual belief of rationality characterizes rationalizable strategies. In a sense (which is made precise in the full version of the paper), counterfactual structures respecting unilateral deviations behave just like probability structures (without counterfactuals).

We now characterize common counterfactual belief of rationality without putting any restrictions on the counterfactual structures (other than them being appropriate, or strongly appropriate). Our characterization is based on ideas that come from the characterization of rationalizability. It is well known that rationalizability can be characterized in terms of an iterated deletion procedure, where at each stage, a strategy σ for player i is deleted if there are no beliefs that i could have about the undeleted strategies for the players other than i that would make σ rational [Pearce 1984]. Thus, there is a deletion procedure that, when applied repeatedly, results in only the rationalizable strategies, that is, the strategies that are played in states where there is common belief of rationality, being left undeleted. We now show that there is an analogous way of characterizing common counterfactual belief of rationality.

3.1 Iterated Minimax Domination

The key to our characterization is the notion of *minimax dominated* strategies.

DEFINITION 3.1: *Strategy σ_i for player i in game Γ is* minimax dominated with respect to $\Sigma'_{-i} \subseteq \Sigma_{-i}(\Gamma)$ *iff there exists a strategy $\sigma_i' \in \Sigma_i(\Gamma)$ such that*

$$\min_{\tau_{-i} \in \Sigma'_{-i}} u_i(\sigma_i', \tau_{-i}) > \max_{\tau_{-i} \in \Sigma'_{-i}} u_i(\sigma_i, \tau_{-i}).$$

∎

In other words, player i's strategy σ is minimax dominated with respect to Σ'_{-i} iff there exists a strategy σ' such that the worst-case payoff for player i if he uses σ' is strictly better than his best-case payoff if he uses σ, given that the other players are restricted to using a strategy in Σ'_{-i}.

In the standard setting, if a strategy σ_i for player i is dominated by σ_i' then we would expect that a rational player will never player σ_i, because σ_i' is a strictly better choice. As is well known, if σ_i is dominated by σ_i', then there are no beliefs that i could have regarding the strategies used by the other players according to which σ_i is a best response [Pearce 1984]. This is no longer the case in our setting. For example, in the standard setting, cooperation is dominated by defection in Prisoner's Dilemma. But in our setting, suppose that player 1 believes that if he cooperates, then the other player will cooperate, while if he defects, then the other player will defect. Then cooperation is not dominated by defection.

So when can we guarantee that playing a strategy is irrational in our setting? This is the case only if the strategy is minimax dominated. If σ_i is minimax dominated by σ_i', there are no counterfactual beliefs that i could have that would justify playing σ_i. Conversely, if σ_i is not minimax dominated by any strategy, then there

[5]Available at http://www.cs.cornell.edu/home/halpern/papers/minimax.pdf.

are beliefs and counterfactual beliefs that i could have that would justify playing σ_i. Specifically, i could believe that the players in $-i$ are playing the strategy profile that gives i the best possible utility when he plays σ_i, and that if he switches to another strategy σ_i', the other players will play the strategy profile that gives i the worst possible utility given that he is playing σ_i'.

Note that we consider only domination by pure strategies. It is easy to construct examples of strategies that are not minimax dominated by any pure strategy, but are minimax dominated by a mixed strategy. Our characterization works only if we restrict to domination by pure strategies. The characterization, just as with the characterization of rationalizability, involves iterated deletion, but now we do not delete dominated strategies in the standard sense, but minimax dominated strategies.

DEFINITION 3.2.: *Define $NSD_j^k(\Gamma)$ inductively: let $NSD_j^0(\Gamma) = \Sigma_j$ and let $NSD_j^{k+1}(\Gamma)$ consist of the strategies in $NSD_j^k(\Gamma)$ not minimax dominated with respect to $NSD_{-j}^k(\Gamma)$. Strategy σ* survives *k rounds of iterated deletion of minimax strategies for player i if $\sigma \in NSD_i^k(\Gamma)$. Strategy σ for player i survives iterated deletion of minimax dominated strategies if it survives k rounds of iterated deletion of strongly dominated for all k, that is, if $\sigma \in NSD_i^\infty(\Gamma) = \cap_k NSD_i^k(\Gamma)$.* ∎

In the deletion procedure above, at each step we remove *all* strategies that are minimax dominated; that is we perform a "maximal" deletion at each step. As we now show, the set of strategies that survives iterated deletion is actually independent of the deletion order.

Let $S^0, \ldots, S^m$ be sets of strategy profiles. $\vec{S} = (S^0, S^1, \ldots, S^m)$ is a *terminating deletion sequence* for Γ if, for $j = 0, \ldots, m-1$, $S^{j+1} \subset S^j$ (note that we use $\subset$ to mean proper subset) and all players i, S_i^{j+1} contains all strategies for player i not minimax dominated with respect to S_{-i}^j (but may also contain some strategies that are minimax dominated), and S_i^m does not contain any strategies that are minimax dominated with respect to S_{-i}^m. A set T of strategy profiles has *ambiguous* terminating sets if there exist two terminating deletion sequences $\vec{S} = (T, S_1, \ldots, S_m)$, $\vec{S}' = (T, S_1', \ldots, S_{m'}')$ such that $S_m \neq S_{m'}'$; otherwise, we say that T has a *unique terminating set*.

PROPOSITION 3.3.: *No (finite) set of strategy profiles has ambiguous terminating sets.*

Proof: Let T be a set of strategy profiles of least cardinality that has ambiguous terminating deletion sequences $\vec{S} = (T, S_1, \ldots, S_m)$ and $\vec{S}' = (T, S_1', \ldots, S_{m'}')$, where $S_m \neq S_{m'}'$. Let T' be the set of strategies that are not minimax dominated with respect to T. Clearly $T' \neq \emptyset$ and, by definition, $T' \subseteq S_1 \cap S_1'$. Since T', S_1, and S_1' all have cardinality less than that of T, they must all have unique terminating sets; moreover, the terminating sets must be the same. For consider a terminating deletion sequence starting at T'. We can get a terminating deletion sequence starting at S_1 by just appending this sequence to S_1 (or taking this sequence itself, if $S_1 = T'$). We can similarly get a terminating deletion sequence starting at S_1'. Since all these terminating deletion sequences have the same final element, this must be the unique terminating set. But $(S_1, \ldots, S_m)$ and $(S_1', \ldots, S_{m'}')$ are terminating deletion sequences with $S_m \neq S_{m'}'$, a contradiction. ∎

COROLLARY 3.4.: *The set of strategies that survivies interated deletion of minimax strategies is independent of the deletion order.*

REMARK 3.5.: *Note that in the definition of $NSD_i^k(\Gamma)$, we remove all strategies that are dominated by some strategy in $\Sigma_i(\Gamma)$, not just those dominated by some strategy in $NSD_i^{k-1}(\Gamma)$. Nevertheless, the definition would be equivalent even if we had considered only dominating strategies in $NSD_i^{k-1}(\Gamma)$. For suppose not. Let k be the smallest integer such that there exists some strategy $\sigma_i \in NSD_i^{k-1}(\Gamma)$ that is minimax dominated by a strategy $\sigma_i' \notin NSD_i^{k-1}(\Gamma)$, but there is no strategy in $NSD_i^{k-1}(\Gamma)$ that dominates σ_i. That is, σ_i' was deleted in some previous iteration. Then there exists a sequence of strategies $\sigma_i^0, \ldots, \sigma_i^q$ and indices $k_0 < k_1 < \ldots < k_q = k-1$ such that $\sigma_i^0 = \sigma_i'$, $\sigma_i^j \in NSD_i^{k_j}(\Gamma)$, and for all $0 \leq j < q$, σ_i^j is minimax dominated by σ_i^{j+1} with respect to $NSD_i^{k_j-1}(\Gamma)$. Since $NSD^{k-2}(\Gamma) \subseteq NSD^j(\Gamma)$ for $j \leq k-2$, an easy induction on j shows that σ_i^q minimax dominates σ^{q-j} with respect to NSD^{k-2} for all $0 < j \leq q$. In particular, σ^q minimax dominates $\sigma_i^0 = \sigma'$ with respect to NSD^{k-2}.* ∎

The following example shows that iteration has bite: there exist a 2-player game where each player has k actions and $k-1$ rounds of iterations are needed.

EXAMPLE 3.6.: *Consider a two-player game, where both players announce a value between 1 and k. Both players receive in utility the smallest of the values announced; additionally, the player who announces the larger value get a reward of $p = 0.5$.[6] That is, $u(x,y) = (y+p, y)$ if $x > y$, $(x, x+p)$ if $y > x$, and (x,x) if $x = y$. In the first step of the deletion process, 1 is deleted for both players; playing 1 can yield a max utility of 1, whereas the mininum utility of any other action is 1.5. Once 1 is deleted, 2 is deleted for both players: 2 can yield a max utility of 2, and the min utility of any other action (once 1 is deleted) is 2.5. Continuing this process, we see that only (k,k) survives.* ∎

3.2 Characterizing Iterated Minimax Domination

We now show that strategies surviving iterated removal of minimax dominated strategies characterize the set of strategies consistent with common counterfactual belief of rationality in (strongly) appropriate counterfactual structures. As a first step, we define a "minimax" analogue of rationalizability.

DEFINITION 3.7.: *A strategy profile $\vec{\sigma}$ in game Γ is* minimax rationalizable *if, for each player i, there is a set $\mathcal{Z}_i \subseteq \Sigma_i(\Gamma)$ such that*

- $\sigma_i \in \mathcal{Z}_i$;
- *for every strategy $\sigma_i' \in \mathcal{Z}_i$ and strategy $\sigma_i'' \in \Sigma_i(\Gamma)$,*

$$\max_{\tau_{-i} \in \mathcal{Z}_{-i}} u_i(\sigma_i', \tau_{-i}) \geq \min_{\tau_{-i} \in \mathcal{Z}_{-i}} u_i(\sigma_i'', \tau_{-i}).$$

∎

THEOREM 3.8.: *The following are equivalent:*

(a) $\vec{\sigma} \in NSD^\infty(\Gamma)$;

(b) $\vec{\sigma}$ is minimax rationalizable in Γ;

[6] This game can be viewed a a reverse variant of the Traveler's dilemma [Basu 1994], where the player who announces the smaller value gets the reward.

(c) there exists a finite counterfactual structure M that is strongly appropriate for Γ and a state ω such that

$$(M, \omega) \models play(\vec{\sigma}) \wedge_{i=1}^{n} SRAT_i^k$$

for all $k \geq 0$;

(d) for all players i, there exists a finite counterfactual structure M that is appropriate for Γ and a state ω such that

$$(M, \omega) \models play_i(\sigma_i) \wedge SRAT_i^k$$

for all $k \geq 0$.

The proof of Theorem 3.8 can be found in the full version of the paper. In the full version of the paper, we additionally characterize analogues of Nash equilibrium in counterfactual structures. These results allow us to more closely relate our model to those of Tennenholtz [2004] and Kalai et al. [2010].

4. DISCUSSION

We have introduced a game-theoretic framework for analyzing scenarios where a player may believe that if he were to switch strategies, this intention to switch may be detected by the other players, resulting in them also switching strategies. Our formal model allows players' counterfactual beliefs (i.e., their beliefs about the state of the world in the event that they switch strategies) to be arbitrary—they may be completely different from the players' actual beliefs.

We may also consider a more restricted model where we require that a player i's counterfactual beliefs regarding other players' strategies and beliefs is ϵ-close to player i's actual beliefs in total variation distance[7]—that is, for every state $\omega \in \Omega$, player i, and strategy σ_i' for player i, the projection of $\mathcal{PR}_{i,\sigma_i'}^{c}(\omega)$ onto strategies and beliefs of players $-i$ is ϵ-close to the projection of $\mathcal{PR}_i(\omega)$ onto strategies and beliefs of players $-i$.

We refer to counterfactual structures satisfying this property as ϵ-counterfactual stuctures. Roughly speaking, ϵ-counterfactual structures restrict to scenarios where players are not "too" transparent to one another; this captures the situation when a player assigns only some "small" probability to its switch in action being noticed by the other players.

As we show in the full paper, 0-counterfactual structures behave just as counterfactual structures that respect unilateral deviations: common counterfactual belief of rationality in 0-counterfactual structures characterizes rationalizable strategies. The general counterfactual structures investigated in this paper are 1-counterfactual structures (that is, we do not impose any conditions on players' counterfactual beliefs). We remark that although our characterization results rely on the fact that we consider 1-counterfactual structures, the motivating example in the introduction (the translucent prisoner's dilemma game) shows that even considering ϵ-counterfactual structures with a small ϵ can result in there being strategies consistent with common counterfactual belief of rationality that are not rationalizable. We leave an exploration of common counterfactual belief of rationality in ϵ-counterfactual structures for future work.

References

Aumann, R. J. (1995). Backwards induction and common knowledge of rationality. *Games and Economic Behavior 8*, 6–19.

Basu, K. (1994). The traveler's dilemma: paradoxes of rationality in game theory. *American Economic Review 84*(2), 391–395.

Brandenburger, A. and E. Dekel (1987). Rationalizability and correlated equilibria. *Econometrica 55*, 1391–1402.

Ekman, P. and W. Friesen (1969). Nonverbal leakage and clues to deception. *Psychiatry 32*, 88–105.

Halpern, J. Y. (1999). Hypothetical knowledge and counterfactual reasoning. *International Journal of Game Theory 28*(3), 315–330.

Howard, N. (1971). *Paradoxes of Rationality: Theory of Metagames and Political Behavior*. The MIT Press, Cambridge.

Kalai, A., E. Kalai, E. Lehrer, and D. Samet (2010). A commitment folk theorem. *Games and Economic Behavior 69*(1), 127–137.

Lewis, D. K. (1973). *Counterfactuals*. Cambridge, Mass.: Harvard University Press.

Pearce, D. G. (1984). Rationalizable strategic behavior and the problem of perfection. *Econometrica 52*(4), 1029–1050.

Samet, D. (1996). Hypothetical knowledge and games with perfect information. *Games and Economic Behavior 17*, 230–251.

Stalnaker, R. C. (1968). A semantic analysis of conditional logic. In N. Rescher (Ed.), *Studies in Logical Theory*, pp. 98–112. Oxford University Press.

Stalnaker, R. C. (1996). Knowledge, belief and counterfactual reasoning in games. *Economics and Philosophy 12*, 133–163.

Tan, T. and S. Werlang (1988). The Bayesian foundation of solution concepts of games. *Journal of Economic Theory 45*(45), 370–391.

Tennenholz, M. (2004). Program equilibrium. *Games and Economic Behavior 49*(12), 363–373.

Zambrano, E. (2004). Counterfactual reasoning and common knowledge of rationality in normal form. *Topics in Theoretical Economics 4*(1).

[7]Recall that two probability distribution are ϵ-close in total variation distance if the probabilities that they assign to any event E differ by at most ϵ.

Reasoning Under the Principle of Maximum Entropy for Modal Logics K45, KD45, and S5

Tivadar Papai
University of Rochester
papai@cs.rochester.edu

Henry Kautz
University of Rochester
kautz@cs.rochester.edu

Daniel Stefankovic
University of Rochester
stefanko@cs.rochester.edu

ABSTRACT

We propose modal Markov logic as an extension of propositional Markov logic to reason under the principle of maximum entropy for modal logics *K45*, *KD45*, and *S5*. Analogous to propositional Markov logic, the knowledge base consists of weighted formulas, whose weights are learned from data. However, in contrast to Markov logic, in our framework we use the knowledge base to define a probability distribution over non-equivalent epistemic situations (pointed Kripke structures) rather than over atoms, and use this distribution to assign probabilities to modal formulas. As in all probabilistic representations, the central task in our framework is inference. Although the size of the state space grows doubly exponentially in the number of propositions in the domain, we provide an algorithm that scales only exponentially in the size of the knowledge base. Finally, we briefly discuss the case of languages with an infinite number of propositions.

1. INTRODUCTION

The central reasoning task for probabilistic logics is to infer the probability of a query formula given a knowledge base. One such logic is propositional Markov logic [4], where the knowledge base consists of weighted propositional formulas. While the weighted formulas define a probability distribution over possible worlds, and increasing the weight of a formula increases the probability mass assigned to worlds where the formula is true, the weights are not true probabilities. Weights can be learned from data or from assertions about the subjective probabilities of statements, or from both using data and explicit assertions of subjective probabilities [20]. In any case, the information obtained from the training data or from an expert can be interpreted as probabilities of the propositional formulas in the KB. Hence, propositional Markov logic defines the probability of formulas in two steps: first learn the weights of formulas in the KB given data and/or subjective probabilities of these propositional formulas, and second, use the learned parameters to infer the probability of query formulas. Out of all the possible distributions which satisfy the probabilistic constraints imposed by the training data or domain expert, the one defined by Markov logic networks is the maximum entropy distribution [17], which makes Markov logic an appealing choice. Markov logic is not the first framework that has been proposed for doing inference under the principle of maximum entropy. For example, a first-order logic language is used in [12, 2] to reason under maximum entropy and the maximum entropy distribution is found using conditional probability constraints in [7, 25].

One of the common approaches for combining probabilities and modal logic builds on a probability distribution defined over possible worlds [15, 22]. Although efficient inference algorithms for probabilistic modal logic have appeared in the past [23], they have been based on using a probabilistic Kripke structure that is explicitly given, not learned from data or assertions about the probabilities of formulas. In contrast, our approach generalizes maximum entropy reasoning for propositional logics to allow both the formulas in the knowledge base and the queries to be propositional modal logic formulas. *K45*, *KD45* and *S5* are the modal logics typically referred to as the logics of beliefs and knowledge. Zero-one laws have been established for such logics [14, 21], which can make probabilistic reasoning challenging if the state space is not chosen carefully; hence, we restrict our domain to be a finite set of epistemic situations (pointed Kripke structures, *i.e.*, Kripke structures with a distinguished real world state). The advantage of these modal logics is that to enumerate all the non-equivalent epistemic situations, it suffices to iterate over a finite set as long as our set of propositional formulas Ω is finite. Although the number of non-equivalent epistemic situations is finite in our problem formulation, their number grows $2^{O(2^{|\Omega|})}$. The main contributions of the paper are to show how one can reason under the principle of maximum entropy when simple propositional logic is replaced by either single agent modal logic *K45*, *KD45* or *S5*, and to provide an exact inference algorithm based on counting, which can drastically reduce the doubly exponential cost of a naively implemented inference algorithm to one singly exponential in the size of the knowledge base. We briefly discuss the case of languages with an infinite number of propositions.

2. BACKGROUND

2.1 Markov Logic

Propositional Markov logic [4] is a knowledge representation language that uses weighted propositional formulas to define probability distributions over truth assignments to propositions. A propositional *Markov logic network* consists of a knowledge base $KB = \{(w_i, F_i)|i = 1, \ldots, m\}$, where $w_i \in \mathbb{R}$ and F_i is a propositional formula over a fixed set of propositions $\Omega = \{p_1, \ldots, p_{|\Omega|}\}$, and defines a probability

TARK 2013, Chennai, India.

distribution over truth assignments X to Ω as follows:

$$\Pr(X = x) = \frac{1}{Z} \exp(\sum_i w_i f_i(x)) \ , \quad (1)$$

where $f_i(x) = 1$ if F_i is true under x, otherwise $f_i(x) = 0$, and where $Z = \sum_{x \in \mathcal{X}} \exp(\sum_i w_i f_i(x))$ is the partition function, and $\mathcal{X}$ denotes the set of all possible truth assignments to Ω, (*i.e*, $|\mathcal{X}| = 2^{|\Omega|}$). Note that, (1) defines an exponential family of distributions (see *e.g.* [26]). Exponential families have the property that for a given set of f_i they describe the maximum entropy distribution that satisfies the set of consistent constraints $\mathbb{E}[f_i] = c_i$. Consistent here means that there exists a probability distribution that satisfies all the constraints simultaneously. We can interpret c_i as the probability of the propositional formula being satisfied under a randomly chosen truth assignment x, hence (1) defines the maximum entropy distribution over the state space of truth assignments to the propositions with the constraints $\mathbb{E}[f_i] = c_i$. The probability of an arbitrary propositional formula F over Ω is defined to be the probability of F being true under a randomly chosen truth assignment X, *i.e.*:

$$\Pr(F) = \sum_{x \in \mathcal{X}: F \text{ is satisfied under } x} \Pr(X = x) = \mathbb{E}[f_i] \ . \quad (2)$$

2.2 Modal Logics K45, KD45 and S5

Modal logics *K45*, *KD45* and *S5* [3] extend propositional or first-order logic by adding a non-truth-functional sentential operators; we will again only discuss the propositional case here. We use the symbol B to represent the modal operator in the language. Where α is a well formed sentence, then $B\alpha$ is a well formed sentence. For example, if we take B to mean "the agent knows that", then the formula $Bp \vee B\neg p$ means that agent i knows whether or not p holds. Note that this is quite different from the tautology $Bp \vee \neg Bp$.

Different modal operators for concepts such as belief, knowledge, desire, obligation, *etc.*, can be specified by the *axiom schemas* that they satisfy. In this paper, we will consider only modal logics *K45*, *KD45*, and *S5*. The properties of each of these logics is the subset of the following axioms and rules [6]:

R1. From ϕ and $\phi \supset \psi$ infer ψ (Modus ponens)

R2. From ψ infer $B\psi$ (Knowledge Generalization)

A1. All tautologies of propositional calculus

A2. $(B\phi \wedge B(\phi \supset \psi)) \supset B\psi$ (Distribution Axiom)

A3. $B\phi \supset \phi$ (Knowledge Axiom)

A4. $B\phi \supset BB\phi$ (Positive Introspection Axiom)

A5. $\neg B\phi \supset B\neg B\phi$ (Negative Introspection Axiom)

A6. $\neg Bfalse$ (Consistency Axiom)

We get *K45* if we take R1, R2, A1, A2, A4, and A5. Besides the axioms of *K45*, *KD45* contains A6 and *S5* contains A3. *S5* is generally used to represent knowledge, and *KD45* beliefs. *K45* is similar to *KD45*; however, it allows believing in contradicting statements.

The common property of these logics is that every formula has an equivalent representation that has depth one, *i.e.*, if $B\phi$ is a subformula then ϕ does not contain any other modal operators. In the rest of the paper we will always assume that we are only dealing with depth one modal formulas.

A Kripke structure over a set of propositions Ω is a tuple $\mathcal{M} = (S, \pi, \mathcal{K})$ where $S \neq \emptyset$ is the set of states, $\pi : S \rightarrow \mathcal{X}$, where $\mathcal{X}$ is the set of truth assignments over Ω and $\mathcal{K} \subseteq S \times S$. If $s \in S$ then for a propositional formula F, we have $\mathcal{M}, s \models F$ if F is satisfied under $\pi(s)$. For a formula BF, we have $\mathcal{M}, s \models BF$ iff $\forall (s, r) \in \mathcal{K} : \mathcal{M}, r \models F$. Moreover, $\mathcal{M}, s \models F_1 \wedge F_2$ iff $\mathcal{M}, s \models F_1$ and $\mathcal{M}, s \models F_2$, and $\mathcal{M}, s \models \neg F$ iff $\mathcal{M}, s \not\models F$.

For each different modal logic, Kripke structures with different properties are associated. Reflexive, symmetric, and transitive relations (equivalence relations) are associated with modal operators that satisfy *S5*. Euclidean, serial, and transitive relations are associated with *KD45*. While Euclidean, and transitive relations are associated with *K45*. For a more detailed description of Kripke structures see, *e.g.*, [3, 6].

A Kripke structure with a distinguished state (generally denoting the real world) is called a pointed Kripke structure or (epistemic) situation, hence an epistemic situation $\sigma = (s, S, \pi, \mathcal{K})$ where $s \in S$. We call two epistemic situations σ_1 and σ_2 equivalent if for every formula F we have $\sigma_1 \models F$ if and only if $\sigma_2 \models F$. Using this definition of equivalence, we can partition situations into equivalence classes.

We can enumerate all the non-equivalent epistemic situations for *K45*, *KD45* and *S5*, *i.e.*, we can select a member from each equivalence class by storing the worlds the agent considers possible and a distinguished real world state [6]. The Kripke structures have a fully connected sub-graph belonging to the possible worlds, and there is a special state s; in the case of *S5*, s is included in the fully connected states, and in *KD45*, there is an outgoing arc from s to every node representing a possible world. In both cases, the set of possible worlds is never empty. The difference between *KD45* and *K45* is that the set of possible worlds in *K45* can be empty. Let Σ_{K45}, Σ_{KD45} and Σ_{S5} denote the set of all possible situations we can construct using the previous descriptions for modal logics *K45*, *KD45* and *S5*, respectively. According to [6], if a formula is satisfiable it must be satisfiable in one of the situations in our Σ, and since not any two members of Σ are equivalent it is enough to consider the members of Σ to count every non-equivalent epistemic situations exactly once. For *K45*, *KD45*, and *S5*, if the set of propositions (Ω) is fixed, then $|\Sigma_{K45}| = 2^{2^{|\Omega|}} 2^{|\Omega|}$, $|\Sigma_{KD45}| = (2^{2^{|\Omega|}} - 1)2^{|\Omega|}$, $|\Sigma_{S5}| = 2^{2^{|\Omega|}-1} 2^{|\Omega|}$, respectively. In each case, the real world can be chosen from the possible $2^{|\Omega|}$ truth assignments. In *K45* the worlds the agent can consider to be possible can be any subset of all the possible worlds, *i.e.*, can have $2^{2^{|\Omega|}}$ values; in *KD45*, the subset cannot be empty; and in *S5*, since the real world must be considered possible, we can only pick a subset of the remaining truth assignments. We see that $|\Sigma| \leq 2^{|\Omega|} 2^{2^{|\Omega|}}$ in all the three cases. We will denote the above mentioned sets by $\Sigma_{\mathcal{L}}(\Omega)$, where $\mathcal{L} \in \{K45, KD45, S5\}$, when we want to emphasize their dependence on Ω.

3. DEFINING THE MAXIMUM ENTROPY DISTRIBUTION

Since the maximum entropy distribution can be sensitive to the choice of the state space (see, *e.g.*, [13, 16]), we have

to be careful when we choose our state space in order to avoid non-intuitive results. *E.g.*, if $\Omega = \{p\}$, and we want to reason about the knowledge of someone using modal logic *S5*, a straightforward extension might seem to be to add a "modal atom" Bp to Ω and define a probability distribution over the modally consistent truth assignments to this set $\{p, Bp\}$, ruling out *e.g.*, the case when p is assigned *false* and Bp is assigned *true*. However, it is easy to see that with an empty KB, $\Pr(p) = \frac{1}{3}$, which seems counter-intuitive, since we have no reason to believe that p is more likely to be true than to be false (analogous examples in a different domain are given in [13]). Moreover, if Ω contains more propositions, selection of modal atoms becomes more complicated, *e.g.*, if $\Omega = \{p, q\}$ should we choose only Bp, Bq, $B\neg p$ and $B\neg q$ as modal atoms, or should we also include $B(p \vee q)$? Without including the latter, its probability can only be bounded but not determined, because a truth assignment to the rest of the modal atoms would not be sufficient to decide its truth value.

Based on the above mentioned problems our goals should be as follows:

(i) Assign probabilities to arbitrary modal or non-modal formulas over a fixed set of propositions Ω based on a set of weighted formulas $KB = \{(w_i, F_i)\}$ in a well-defined way.

(ii) If KB contains only weighted non-modal formulas, we should obtain the distribution that propositional Markov logic would define.

(iii) If KB does not contain infinite weights and ψ subsumes ϕ, and ϕ and ψ are non-equivalent, then $\Pr(\psi) < \Pr(\phi)$.

These criteria can be achieved by assigning probabilities to epistemic situations rather than to modal atoms. Given a non-empty set of epistemic situations Σ over a fixed Ω propositions, we define the probability of $\sigma \in \Sigma$ as:

$$\Pr(\sigma) = \frac{1}{Z} \exp\Big(\sum_{i:\sigma \models f_i} w_i\Big) , \tag{3}$$

where the partition function is defined as:

$$Z = \sum_{\sigma \in \Sigma} \exp\Big(\sum_{i:\sigma \models f_i} w_i\Big) . \tag{4}$$

The probability of a formula ϕ (modal or non-modal) is defined as:

$$\Pr(\phi) = \sum_{\sigma \in \Sigma : \sigma \models \phi} \Pr(\sigma) . \tag{5}$$

Property (i) clearly holds, no matter how we choose Σ. To satisfy Property (ii) it must be true that $c(x) = |\{(\mathcal{M}, s) \in \Sigma | \pi(s) = x\}|$ has the same value for every truth assignment x over Ω. If Σ contains every non-equivalent situations then Property (iii) is clearly satisfied as well. Hence, if we choose the state space to be Σ_{K45}, Σ_{KD45} or Σ_{S5}, all the desired three conditions are satisfied.

Note that we could define the same distribution using modal atoms as we do by defining distribution over Σ_{K45}, Σ_{KD45} or Σ_{S5}. *E.g.*, we define the same distribution if we choose the modal atoms to be all the propositional atoms, and all the depth one formulas in the form Bc, where c is a conjunction which contains every proposition either as positive or a negative literal. However, for our goals we found the approach to define the distribution over epistemic situations more general, because in this way Property (i) always holds, we do not have to account for modally inconsistent states, moreover, it is easier to decide whether Properties (ii) and (iii) hold.

4. INFERENCE

The computationally expensive part of determining (3) and (5) can both be reduced to the computation of a partition function (4). *E.g.*, to infer the probability of a formula F not present in the knowledge base KB, we first have to create a new knowledge base $KB' = KB \cup \{(\infty, F)\}$. If Z and Z' denote the partition functions corresponding to the knowledge bases KB and KB', respectively, then it follows from (3), (4) and (5) that $\Pr(F) = \frac{Z'}{Z}$.

Computing the partition function is challenging because the size of the state space for *K45*, *KD45* and *S5* are all doubly exponential in $|\Omega|$ as mentioned in Sec. 2.2. On the other hand, there is much symmetry in the domain, *i.e.*, many situations have the same probability; hence Σ can be divided into equivalence classes. Similar to lifted inference techniques for quantified probabilistic logics (a highly active research area today, *e.g.* [24, 18, 11]), we show how one can compute the partition function without explicitly iterating through every state in the domain. Although our exact inference algorithm is exponential in a quantity describing the complexity of the knowledge base, it is vastly faster than iterating through the $2^{O(2^{|\Omega|})}$ epistemic situations in Σ_{K45}, Σ_{KD45}, or Σ_{S5}. We are going to assume that we have access to a propositional model counter, *i.e.*, for any propositional formula we can tell the number of its satisfying truth assignments. (Exhaustive solvers run in exponential time, which is sufficient for our claimed bounds, but heuristic/approximate solvers such as, *e.g.*, SampleSearch [10], may be more useful in practice.) We now show how to reduce the computation of a partition function to counting epistemic situations that satisfy a given set of modal logic formulas. We first introduce truth assignments to formulas in the knowledge base. If $KB = \{(w_1, F_1), \ldots, (w_n, F_n)\}$, let $\mathcal{T}$ be the set of length n Boolean vectors. For $t \in \mathcal{T}$ let $\Phi(t)$ be a conjunction where the i-th term is F_i if $t_i = \text{true}$, and it is $\neg F_i$ if $t_i = \text{false}$. Members of $\mathcal{T}$ will partition the space of epistemic situations Σ into disjoint sets. $\sigma_1, \sigma_2 \in \Sigma$ will be in the same partition if for every $t \in \mathcal{T}$ we have $\sigma_1 \models \Phi(t)$ iff $\sigma_2 \models \Phi(t)$. If σ_1 and σ_2 are in the same partition then $\Pr(\sigma_1) = \Pr(\sigma_2)$. To simplify notation, let $w(t) = \sum_{i:t_i=\text{true}} w_i$, *i.e.*, $w(t)$ is the sum of the weights of the formulas to which t assigns true. Hence, we can rewrite (4) as:

$$Z = \sum_{t \in \mathcal{T}} N(\Phi(t)) \exp(w(t)) , \tag{6}$$

where $N(\phi)$ denotes the number of epistemic situations where ϕ holds, *i.e.*, $N(\phi) = |\{\sigma \in \Sigma | \sigma \models \phi\}|$.

The probability of any query formula F can be written as:

$$\Pr(F) = \frac{1}{Z} \sum_{t \in \mathcal{T}} N(\Phi(t) \wedge F) \exp(w(t)) . \tag{7}$$

Next, we show how to compute $N(F)$ for different formulas. Table 1 contains the simple counts for different basic formulas in *K45*, *KD45*, or *S5*. We use the notation $c(\phi)$ for

the number of satisfying truth assignments of propositional formula ϕ. Using the rules in (8) we can compute $N(F)$ for any formula which is in CNF normal form, where each term is either a propositional formula, or in the form $B\phi$ or $\neg B\phi$, where ϕ is a propositional formula. The most general form of a conjunction is $C = \phi_0 \wedge B\psi \wedge (\wedge_{i=1}^{k} \neg B\phi_i)$ (we call such conjunctions simple), since $B\phi_1 \wedge B\phi_2 = B(\phi_1 \wedge \phi_2)$. The counting of the satisfying assignments of C is done by the inclusion-exclusion principle and by counting the members of the complement of sets.

EXAMPLE 1. *If p and q are propositions and $F = (p \supset q) \wedge B(p \vee q) \wedge \neg Bp \wedge \neg Bq$:*

$$\begin{aligned} N(F) &= N((p \supset q) \wedge B(p \vee q) \wedge \neg Bp \wedge \neg Bq) \\ &= N((p \supset q) \wedge B(p \vee q)) - \\ &\quad \big[N((p \supset q) \wedge B(p \vee q) \wedge Bp) + \\ &\quad N((p \supset q) \wedge B(p \vee q) \wedge Bq) - \\ &\quad N((p \supset q) \wedge B(p \vee q) \wedge Bp \wedge Bq)\big] \\ &= N((p \supset q) \wedge B(p \vee q)) - N((p \supset q) \wedge Bp) - \\ &\quad N((p \supset q) \wedge Bq) + N((p \supset q) \wedge B(p \wedge q)) \end{aligned}$$

Hence, a CNF formula with this general type of conjunctions can represent any modal formula in *K45*, *KD45*, and *S5* since every formula in *K45*, *KD45*, and *S5* have an equivalent depth one representation (which can possibly increase the size of the formula drastically). (To see why a depth one representation for every formula F exists, consider the set of modal atoms presented at the end of Sec. 3. Since every epistemic situation can be characterized by a conjunction of these modal atoms where each literals is either a positive or negative modal atom, we can form a depth one formula by taking the disjunction of the situations where F holds.) The final piece of computation of $N(F)$ for a CNF formula F again uses the inclusion-exclusion principle, replacing the computation for disjunctions with (exponentially) many conjunctions.

EXAMPLE 2. *In modal logic S5, p and q being propositions, the use of inclusion-exclusion principle to reduce the computation of $N(F)$ for the CNF formula $F = (p \supset q) \vee Bp) \wedge (p \vee B(p \vee q))$ proceeds as follows:*

$$\begin{aligned} N(F) &= N(((p \supset q) \vee Bp) \wedge (p \vee B(p \vee q))) \\ &= N((p \supset q) \wedge (p \vee B(p \vee q))) + \\ &\quad N(Bp \wedge (p \vee B(p \vee q))) - \\ &\quad N((p \supset q) \wedge Bp \wedge (p \vee B(p \vee q))) \\ &= N((p \supset q) \wedge p) + N((p \supset q) \wedge B(p \vee q)) - \\ &\quad N((p \supset q) \wedge p \wedge B(p \vee q)) + N((Bp \wedge p) + \\ &\quad N(Bp \wedge B(p \vee q)) - N(Bp \wedge p \wedge B(p \vee q)) - \\ &\quad (N((p \supset q) \wedge Bp \wedge p) + \\ &\quad N((p \supset q) \wedge Bp \wedge B(p \vee q)) - \\ &\quad N((p \supset q) \wedge Bp \wedge p \wedge B(p \vee q))) \end{aligned}$$

We see that each term can be easily computed using the rules in Table 1 and the expressions in (8).

$$N(\vee_{i=1}^{k} B\phi_i) = \sum_{i=1}^{k} (-1)^{i+1} \sum_{1 \le j_1 < j_2 < \ldots < j_i \le k} N(B(\phi_{j_1} \wedge \ldots \wedge \phi_{j_i})) \tag{8}$$

$$\begin{aligned} N(\wedge_{i=1}^{k} \neg B\phi_i) &= N(true) - \\ &\quad \sum_{i=1}^{k} (-1)^{i+1} \sum_{1 \le j_1 < j_2 < \ldots < j_i \le k} N(B(\phi_{j_1} \wedge \ldots \wedge \phi_{j_i})) \end{aligned}$$

$$\begin{aligned} N(\phi_0 \wedge B\psi \wedge (\wedge_{i=1}^{k} \neg B\phi_i)) &= N(\phi_0 \wedge B\psi) - \\ &\quad \sum_{i=1}^{k} (-1)^{i+1} \sum_{1 \le j_1 < j_2 < \ldots < j_i \le k} N(\phi_0 \wedge B\psi \wedge B(\phi_{j_1} \wedge \ldots \wedge \phi_{j_i})) \end{aligned}$$

$$\begin{aligned} N((\vee_{i=1}^{k} F_i) \wedge F) &= \\ &\quad \sum_{i=1}^{k} (-1)^{i+1} \sum_{1 \le j_1 < j_2 < \ldots < j_i \le k} N(F \wedge F_{j_1} \wedge \ldots \wedge F_{j_i}) \end{aligned}$$

Using the established rules of counting we could give a time complexity result of our inference algorithm for CNF formulas, but instead we give results for formulas in a more general form. The most general language $\mathcal{L}(\Omega)$ we use is defined as follows:

- Every propositional formula is a member of $\mathcal{L}(\Omega)$,
- If ϕ is a propositional formula, then $B\phi \in \mathcal{L}(\Omega)$,
- If $\phi \in \mathcal{L}(\Omega)$, then $\neg\phi \in \mathcal{L}(\Omega)$,
- If $\phi_1, \phi_2 \in \mathcal{L}(\Omega)$, then $\phi_1 \wedge \phi_2 \in \mathcal{L}(\Omega)$.

Hence, we only allow depth one modal formulas, however, since in modal logics *K45*, *KD45*, and *S5*, every formula has an equivalent depth one representation, we can allow this restriction without the loss of generality.

THEOREM 1. *Counting the non-equivalent epistemic situations that satisfy a depth one formula $F \in \mathcal{L}(\Omega)$ in K45, KD45, or S5 can be accomplished in time $2^{O(|F|+|\Omega|)}$.*

We use the following definition and lemma to prove Theorem 1.

DEFINITION 1. *For a formula F let $\mathbb{I}_F : \Sigma \to \{0,1\}$ denote the characteristic function of F in the space of all non-equivalent epistemic situations, i.e., $\mathbb{I}_F(\sigma) = 1$ iff $\sigma \models F$.*

The next result shows that the characteristic function of a depth one formula can be expressed as a combination of characteristic functions of simple conjunctions.

LEMMA 1. *For a depth one modal logic formula $F \in \mathcal{L}(\Omega)$ in $K45, KD45$ or S5, it is always possible to represent $\mathbb{I}_F = \sum_{i=1}^{K} w_i \mathbb{I}_{C_i}$ where every $w_i \in \{-1, +1\}$, $|C_i| \le |F|$, every C_i is a simple conjunction (i.e., in the form $\phi_0^i \wedge B\psi^i \wedge (\wedge_{k=1}^{l_i} \neg B\phi_k^i)$), and $K \le 2^{|F|}$.*

PROOF. We prove the lemma by induction on the structure of F. For the base cases where F is either a propositional formula, or in the form $B\phi$ where ϕ is a propositional formula, the claim clearly holds. Suppose that $F = \neg F_1$.

F	$N(F)$		
	K45	KD45	S5
true	$\lvert\Sigma_{K45}\rvert = 2^{\lvert\Omega\rvert}2^{2^{\lvert\Omega\rvert}}$	$\lvert\Sigma_{KD45}\rvert = 2^{\lvert\Omega\rvert}(2^{2^{\lvert\Omega\rvert}} - 1)$	$\lvert\Sigma_{S5}\rvert = 2^{\lvert\Omega\rvert}2^{2^{\lvert\Omega\rvert}-1}$
propositional formula ϕ	$c(\phi)2^{2^{\lvert\Omega\rvert}}$	$c(\phi)(2^{2^{\lvert\Omega\rvert}} - 1)$	$c(\phi)2^{2^{\lvert\Omega\rvert}-1}$
$B\phi$	$2^{\lvert\Omega\rvert}2^{c(\phi)}$	$2^{\lvert\Omega\rvert}(2^{c(\phi)} - 1)$	$c(\phi)2^{c(\phi)-1}$
$\phi_0 \wedge B\phi$	$c(\phi_0)2^{c(\phi)}$	$c(\phi_0)(2^{c(\phi)} - 1)$	$c(\phi_0 \wedge \phi)2^{c(\phi)-1}$
$\neg B\phi$	$\lvert\Sigma_{K45}\rvert - N(B\phi)$	$\lvert\Sigma_{KD45}\rvert - N(B\phi)$	$\lvert\Sigma_{S5}\rvert - N(B\phi)$

Table 1: Basic counting rules for *K45*, *KD45*, and *S5*

The claim of the lemma holds for F_1 (by induction hypothesis), and hence:

$$\mathbb{I}_{F_1} = \sum_{i=1}^{K} w_i \mathbb{I}_{C_i} \ . \tag{9}$$

Then we have

$$\mathbb{I}_F = \mathbb{I}_{true} - \mathbb{I}_{F_1} = \mathbb{I}_{true} - \sum_{i=1}^{K} w_i \mathbb{I}_{C_i} \ . \tag{10}$$

Now suppose $F = F_1 \wedge F_2$. Then:

$$\mathbb{I}_F = \mathbb{I}_{F_1}\mathbb{I}_{F_2} = \sum_{i=1}^{K_1}\sum_{j=1}^{K_2} w_i^1 w_j^2 \mathbb{I}_{C_i^1}\mathbb{I}_{C_j^2} = \sum_{i=1}^{K_1}\sum_{j=1}^{K_2} w_i^1 w_j^2 \mathbb{I}_{C_i^1 \wedge C_j^2} \ . \tag{11}$$

Notice that $|C_1 \wedge C_2| \leq |F_1| + |F_2| + 1 \leq |F|$ and $K_1 K_2 \leq 2^{|F_1|+|F_2|} \leq 2^{|F|}$. □

COROLLARY 1. *Since $N(F) = \sum_{\sigma\in\Sigma} \mathbb{I}_F(\sigma)$, we have*

$$N(F) = \sum_{i=1}^{K}\sum_{\sigma\in\Sigma} \mathbb{I}_{C_i}(\sigma) = \sum_{i=1}^{K} w_i N(C_i) \ , \tag{12}$$

i.e., the problem of counting epistemic situations in which F is satisfied has been reduced to counting epistemic situations in which the basic conjunctions C_i are satisfied.

Now we are ready to prove Theorem 1

PROOF PROOF OF THEOREM 1. According to Corollary 1, $N(F) = \sum_{i=1}^{K} w_i N(C_i)$. We first note that using the inclusion-exclusion principle (see (8)) for every i we can compute $N(C_i)$ in time $2^{O(|C_i|+|\Omega|)}$. (Note that the bound on the running time is large enough to allow counting the satisfying assignments of the necessary propositional formulas.) Since $K \leq 2^{|F|}$ and for every i we have $|C_i| \leq |F|$, computing $N(F)$ can be accomplished in $2^{O(|F|+|\Omega|)}$. □

COROLLARY 2. *Computing the partition function in* (6) *for a knowledge base consisting of formulas with depth at most one can be accomplished in time $2^{O(|F|+|\Omega|)}$.*

5. INFINITE DOMAINS

Although the main focus of the paper is finite domains, we briefly discuss here the case of infinite domains. The source of finiteness in our formulation is that there are only a finite set of non-equivalent epistemic situations over a given set of propositions (Ω). We now consider the questions regarding the effect of increasing the size of Ω where the state space is, as before, the set of non-equivalent epistemic situations:

1. Do zero-one laws hold for infinite domains?
2. Given a knowledge base of equality constraints on the probabilities of formulas, are there formulas the probability of which have to be either 0 or 1?

The existence of zero-one laws is well-known for first-order logic [9, 5] and for modal logic [14]. In the modal logic setting, the zero-one law states that given an arbitrary formula, the probability of it being valid in a randomly chosen Kripke structure with N number of states converges to 1 or to 0 as $N \to \infty$. In [14], the state space can contain multiple Kripke structures with N states that are equivalent; hence, the size of the state space is not bounded, despite Ω being finite. Moreover, the focus of their paper is on the probability of a formula being valid in a randomly chosen Kripke structure, while we are interested in the probability of a formula being satisfied in a randomly chosen epistemic situation. To show the contrast, consider the case of an empty knowledge base which defines a uniform distribution over the situations. The probability of a proposition p being true will always be 0.5 regardless of $|\Omega|$, hence its probability is not going to converge to 0 or to 1. However, *e.g.* $\Pr(B\phi) \to 0$ if ϕ is not a tautology, otherwise $\Pr(B\phi) \to 1$ (we can verify this by taking the limit of $\frac{N(true)}{N(B\phi)}$ using Table 1 and that adding k more propositions to Ω changes the value of $c(\phi)$ to $c(\phi)2^k$). More generally:

THEOREM 2. *If C is a consistent simple conjunction, i.e., $C = \phi_0 \wedge B\psi \wedge (\wedge_{i=1}^{k} \neg B\phi_i)$ where ψ and every ϕ_i is a propositional formula and β is a propositional formula s.t. $C \wedge B\beta$ is consistent as well, then $\lim_{|\Omega|\to\infty} \frac{N(B\beta\wedge C)}{N(C)} = 0$ if $\psi \not\models \beta$, otherwise $\lim_{|\Omega|\to\infty} \frac{N(B\beta\wedge C)}{N(C)} = 1$.*

The proof of Theorem 2 makes use of the following lemmas (which we prove only for *K45*).

LEMMA 2. *If ϕ_0, ψ and β are propositional formulas and $\phi_0 \wedge \psi$ is satisfiable then $\lim_{|\Omega|\to\infty} \frac{N(\phi_0\wedge B\psi\wedge B\beta)}{N(\phi_0\wedge B\psi)} = 0$ if $\psi \not\models \beta$, otherwise $\lim_{|\Omega|\to\infty} \frac{N(\phi_0\wedge B\psi\wedge B\beta)}{N(\phi_0\wedge B\psi)} = 1$.*

PROOF. We only prove the lemma for *K45*. Similar proof works for *KD45* and *S5*. Assume ϕ_0, ψ and β only build on propositions from a set Ω' and let $k = |\Omega| - |\Omega'|$. For a propositional formula F that builds only on propositions from Ω' let $c'(F)$ denote the number of its satisfying truth assignments over Ω'. We have

$$\begin{aligned} \frac{N(\phi_0 \wedge B\psi \wedge B\beta)}{N(\phi_0 \wedge B\psi)} &= \frac{2^k c'(\phi_0 \wedge \psi \wedge \beta) 2^{2^k c'(\psi\wedge\beta)}}{2^k c'(\phi_0 \wedge \psi) 2^{2^k c'(\psi)}} \\ &= \frac{c'(\phi_0 \wedge \psi \wedge \beta)}{c'(\phi_0 \wedge \psi)} 2^{2^k (c'(\psi\wedge\beta) - c'(\psi))} \end{aligned} \tag{13}$$

(note that N counts epistemic situations over Ω whereas c' counts satisfying assignments over Ω'). Since if $\psi \not\models \beta$ then $c'(\psi \wedge \beta) - c'(\phi) < 0$, hence the ratio converges to 0 as $k \to \infty$. It is easy to verify that if $\psi \models \beta$ this ratio is 1 for every k. □

The next result means that as the number of extra propositions increases we can remove terms in the form $\neg B\phi$ from simple conjunctions.

LEMMA 3. *If C is a consistent simple conjunction, i.e., $C = \phi_0 \wedge B\psi \wedge (\wedge_{i=1}^{k} \neg B\phi_i)$ where ψ and every ϕ_i is a propositional formula then* $\lim_{|\Omega|\to\infty} \frac{N(\phi_0 \wedge B\psi)}{N(C)} = 1$.

PROOF. If we expand C according to the inclusion-exclusion principle (equation (8)) we can conclude that

$$\lim_{|\Omega|\to\infty} \frac{N(\phi_0 \wedge B\psi)}{N(C)} = 1$$

since for all the other terms

$$\lim_{|\Omega|\to\infty} \frac{N\left(\phi_0 \wedge B\psi \wedge B(\phi_{j_1} \wedge \ldots \wedge \phi_{j_i})\right)}{N(\phi_0 \wedge B\psi)} = 0$$

according to Lemma 2. □

We can prove now Theorem 2.

PROOF OF THEOREM 2. Theorem 2 immediately follows from the following telescopic product

$$\frac{N(B\beta \wedge C)}{N(\phi_0 \wedge B\psi \wedge B\beta)} \frac{N(\phi_0 \wedge B\psi \wedge B\beta)}{N(\phi_0 \wedge B\psi)} \frac{N(\phi_0 \wedge B\psi)}{N(C)} \quad (14)$$

where the first and third terms converge to 1 (using Lemma 3). The claim now follows from Lemma 2 applied to the second term. □

Using Theorem 2, we can simplify the computation of $N(\phi)$ in the limit $|\Omega| \to \infty$ for any formula ϕ by dropping terms in the form $\neg B\phi$ whenever we encounter a conjunction in the most general form; in addition, we can expect to neglect the majority of the terms when using the inclusion-exclusion rule. Unfortunately, one consequence of Theorem 2 is that the weight of certain formulas in the knowledge base will go to infinity as $|\Omega| \to \infty$. Consider, *e.g.*, the simple formula Bp in the knowledge base. If its weight is w, then $\Pr(Bp) = \frac{N(Bp)\exp(w)}{N(Bp)\exp(w)+N(\neg Bp)}$ which converges to 0 for any finite w, hence $0 < \Pr(Bp) < 1$ cannot be captured with any finite w as we increase $|\Omega|$.

One way we can avoid this phenomenon is to define w as a function of $|\Omega|$ as [16] suggests for first-order Markov logic. Another approach would be to choose Σ not to include every non-equivalent episetemic situations. If this were done, any learned model would not be able to satisfy Property (iii), but could achieve non-zero and non-one probabilities for every modal formula with finite w values.

6. RELATED WORK

Reasoning with a knowledge base of statistical information have been approached in many different ways. The ones making use of the principle of maximum entropy [17] seem to be more natural, because when multiple distributions are consistent with our knowledge base, then there is no reason to prefer one over the other. In [12], first-order logic is the representational language and a connection between maximum entropy reasoning is presented when unary predicates are used. Markov logic [4] is one of the most popular choices in the statistical relational learning community for reasoning under the maximum entropy with a first-order logic knowledge base. Propositional Markov logic is generalized in [7] by using different features that are capable of capturing conditional probabilities. Although we do not mention representation of conditional probabilities, our framework could be generalized in this direction. Maximum entropy models are sensitive to the choice of domain, but whether this is a property of other kinds of models is discussed in [13]. [16] proposes to make the weights dependent on the size of the domain to counter act against the change of marginals when the domain size changes. Hence, it is not surprising that we eventually encountered the issue of changing marginals in both of our chosen state spaces. Zero-one laws for first-order logic are long known [9, 5]; for modal logics, they were established in [14]; and for conditional probabilities in [21]. In our setting when we have a finite number of propositions our state space is always finite, hence we only experience the convergence of the probability of certain formulas to 0 and to 1 when we started increasing the number of propositions. Probabilistic modal logic has been proposed in [22] and an efficient inference algorithm in [23]. Although their proposed framework is capable of answering queries using given a probabilistic Kripke structure, but not suitable for learning the probabilistic model given a probabilistic knowledge base. In contrast, our approach defines an exponential family or probability distributions, hence the learning of the parameters of the distribution is a convex optimization problem (see e.g. [19]).

7. CONCLUSIONS AND FUTURE WORK

In this paper we showed a way to extend propositional Markov logic with modal operators using epistemic situations (pointed Kripke structures) as basic building blocks of the domain. The modal logics we focused on were *K45*, *KD45*, and *S5* for a single agent. The common theme in the modal logics we considered is that the number of non-equivalent epistemic situations is finite, but grows doubly exponentially with the number of propositions in the domain. However, we provided an exact inference algorithm where complexity is only singly exponential in the size of the knowledge base.

Although we only provided an exact inference algorithm, the three main parts of computations we need to perform for exact inference can all be approximated. Bonferroni inequalities can provide an approximation when we use the inclusion-exclusion principle in our computations; in addition, our discussion of infinite domains suggest that the bounds of the approximation will be sharp as we increase the size of the domain. Heuristics for approximately counting satisfying assignments of propositional formulas exist, and toolboxes are readily available [10]. Finally, iterating through all the possible truth assignments to the formulas in the knowledge base can be avoided by sampling from the assignments using importance sampling (the idea is described in [11]).

We discussed the challenges of extending the framework to infinite domains, where the number of propositions is unbounded. Further examination of infinite domains is one of our future goals.

We only discussed modal logics with a single agent, since for multiple agents the number of non-equivalent epistemic

situations even over a fixed set of propositions is unbounded. Another future goal is to explore if we can do inference efficiently in the multi agent setting despite the infinite state space.

8. ACKNOWLEDGMENTS

This research was supported by grants from ARO (W991NF-08-1-0242), ONR (N00014-11-10417), NSF (IIS-1012017), DOD (N00014-12-C-0263), and a gift from Intel.

9. REFERENCES

[1] *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence, July 22-26, 2007, Vancouver, British Columbia, Canada*. AAAI Press, 2007.

[2] F. Bacchus, A. J. Grove, J. Y. Halpern, and D. Koller. From statistical knowledge bases to degrees of belief. *Artif. Intell.*, 87(1-2):75–143, 1996.

[3] B. F. Chellas. *Modal logic: an introduction*. Cambridge University Press, 1980.

[4] P. Domingos and D. Lowd. *Markov Logic: An Interface Layer for Artificial Intelligence*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers, 2009.

[5] R. Fagin. Probabilities on finite models. *J. Symb. Log.*, 41(1):50–58, 1976.

[6] R. Fagin, J. Y. Halpern, Y. Moses, and M. Y. Vardi. *Reasoning About Knowledge*. MIT Press, 1995.

[7] J. Fisseler. Toward markov logic with conditional probabilities. In *FLAIRS Conference*, pages 643–648, 2008.

[8] D. Fox and C. P. Gomes, editors. *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence, AAAI 2008, Chicago, Illinois, USA, July 13-17, 2008*. AAAI Press, 2008.

[9] Y. Glebskii, D. Kogan, M. Liogon'kii, and V. Talanov. Range and degree of realizability of formulas in the restricted predicate calculus. *Cybernetics*, 5:142–154, 1969.

[10] V. Gogate and R. Dechter. Approximate counting by sampling the backtrack-free search space. In *AAAI* [1], pages 198–203.

[11] V. Gogate and P. Domingos. Formula-based probabilistic inference. In P. Grünwald and P. Spirtes, editors, *UAI*, pages 210–219. AUAI Press, 2010.

[12] A. J. Grove, J. Y. Halpern, and D. Koller. Random worlds and maximum entropy. *J. Artif. Intell. Res. (JAIR)*, 2:33–88, 1994.

[13] J. Halpern and D. Koller. Representation dependence in probabilistic inference. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1852–1860, Montreal, Canada, August 1995.

[14] J. Y. Halpern and B. Kapron. Zero-one laws for modal logic. *Annals of Pure and Applied Logic 69*, 69:157–193, 1994.

[15] J. C. Harsanyi. Games with Incomplete Information Played by "Bayesian" Players, I-III. Part I. The Basic Model. *Management Science*, 14(3):159–182, Nov. 1967.

[16] D. Jain, A. Barthels, and M. Beetz. Adaptive Markov Logic Networks: Learning Statistical Relational Models with Dynamic Parameters. In *19th European Conference on Artificial Intelligence (ECAI)*, pages 937–942, 2010.

[17] E. T. Jaynes. *Where do we stand on Maximum Entropy?*, pages 15–118. The MIT Press, 1979.

[18] K. Kersting, B. Ahmadi, and S. Natarajan. Counting belief propagation. In *In Proc. UAI-09*, 2009.

[19] D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, 2009.

[20] T. Papai, S. Ghosh, and H. A. Kautz. Combining subjective probabilities and data in training markov logic networks. In P. A. Flach, T. D. Bie, and N. Cristianini, editors, *ECML/PKDD (1)*, volume 7523 of *Lecture Notes in Computer Science*, pages 90–105. Springer, 2012.

[21] R. Rosati and G. Gottlob. Asymptotic conditional probability in modal logic: A probabilistic reconstruction of nonmonotonic logic. In L. P. Kaelbling and A. Saffiotti, editors, *IJCAI*, pages 1378–1383. Professional Book Center, 2005.

[22] A. Shirazi and E. Amir. Probabilistic modal logic. In *Proceedings of the 22nd AAAI Conference on Artificial Intelligence.*, pages 489–495. Elsevier, 2007.

[23] A. Shirazi and E. Amir. Factored models for probabilistic modal logic. In *Proceedings of the 23rd national conference on Artificial intelligence*, AAAI'08, pages 541–547. AAAI Press, 2008.

[24] P. Singla and P. Domingos. Lifted first-order belief propagation. In Fox and Gomes [8], pages 1094–1099.

[25] M. Thimm, G. Kern-Isberner, and J. Fisseler. Relational probabilistic conditional reasoning at maximum entropy. In *ECSQARU*, pages 447–458, 2011.

[26] M. J. Wainwright and M. I. Jordan. *Graphical Models, Exponential Families, and Variational Inference*. Now Publishers Inc., Hanover, MA, USA, 2008.

Facebook and the epistemic logic of friendship

Jeremy Seligman
Department of Philosophy
The University of Auckland
Auckland, New Zealand

Fenrong Liu
Department of Philosophy
Tsinghua University
Beijing, China

Patrick Girard
Department of Philosophy
The University of Auckland
Auckland, New Zealand

This paper presents a two-dimensional modal logic for reasoning about the changing patterns of knowledge and social relationships in networks organised on the basis of a symmetric 'friendship' relation, providing a precise language for exploring 'logic in the community' [11]. Agents are placed in the model, allowing us to express such indexical facts as 'I am your friend' and 'You, my friends, are in danger'.

The technical framework for this work is general dynamic dynamic logic (GDDL) [4], which provides a general method for extending modal logics with dynamic operators for reasoning about a wide range of model-transformations, starting with those definable in propositional dynamic logic (PDL) and extended to allow for the more subtle operators involved in, for example, private communication, as represented in dynamic epistemic logic (DEL) and related systems. We provide a hands-on introduction to GDDL, introducing elements of the formalism as we go, but leave the reader to consult [4] for technical details.

Instead, the purpose of this paper is to investigate a number of conceptual issues that arise when considering communication between agents in such networks, both from one agent to another, and broadcasts to socially-defined groups of agents, such as the group of my friends. All three components of the communication (the sender, the message, and the receivers) can be specified in a variety of ways that need to be distinguished. For example, Charlie may tell Bella 'you are in danger' or 'I am in danger'. He may broadcast to all 'my friends are in danger', which if Bella is a friend, will mean that that she is in danger, or send a message only to his friends that they are in danger. All such possibilities, together with their epistemic consequences, will be examined.

We extend the treatment of announcements to questions, in which agents are taken to be sincere and cooperative interlocutors, and consider network structure changing operations such as adding and deleting friends (with the permission of other agents) and, finally, explore the effect of all this on the concept of common knowledge, which is more varied and rich in the social network setting.

These issues are illustrated by a number of examples about office gossip, cold-war spy networks and Facebook.

1. A LANGUAGE OF SOCIAL KNOWING

TARK 2013, Chennai, India.

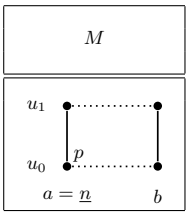


Figure 1: A simple EFL model

We start with a language $\mathcal{L}$ of *epistemic friendship logic* EFL based on atoms of two types: propositional variables $\rho \in \mathsf{Prop}$ representing indexical propositions such as 'I am in danger', and (a finite set of) agent nominals $n \in \mathsf{ANom}$ which stand for indexical propositions asserting identification: 'I am n'. The language is then inductively defined as:

$$\varphi \quad ::= \quad \rho \mid n \mid \neg\varphi \mid (\varphi \wedge \varphi) \mid K\varphi \mid F\varphi \mid A\varphi$$

We read K as 'I know that' and F as 'all my friends' and A as 'every agent'. Models for this language are Kripke models of the form $M = \langle W, A, k, f, V\rangle$, where W is a set (of epistemic states), A is a set (of agents), and

1. k is a family of equivalence relations k_a for each agent $a \in A$, representing the ignorance of a in distinguishing epistemic possibilities (as for standard S5 epistemic logic)

2. f is a family of symmetric and irreflexive relations f_w for each $w \in W$, representing the friendship relation in state w.

3. g is a function mapping each agent nominal $n \in \mathsf{ANom}$ to the agent $g(n) \in A$ named by n. We abbreviate $g(n)$ to $\underline{n}$ when the model is clear from the context.

4. V is a valuation function mapping propositional variables Prop to subsets of $W \times A$, with $(w, a) \in V(p)$ representing that the indexical proposition p holds of agent a in state w.

For example, Figure 1 illustrates a simple model for a language in which there is only one propositional variable p and one agent name n. The set of states is $W = \{u_0, u_1\}$ and the set of agents is $A = \{a, b\}$, with $g(n) = a$, n naming agent a. Both agents are ignorant about which state they are in, so $k_a = k_b$ is the universal relation. These are indicated by the two columns of the diagram. The left column displays the

k_a relation with a thick line; the right column displays the k_b relation, similarly. The lines are non-directional because the relations are assumed to be symmetric. In more complex diagrams, we will assume that the relations depicted are the reflexive, transitive closures of what is shown explicitly. The rows of the diagram show the relations f_{u_0} (first row) and f_{u_1} (second row) with dotted lines. This represents the two agents being friends in both states of W. Again these are non-directional because we assume symmetry. But for these lines we do *not* take the reflexive, transitive closure, since we assume that f_w is irreflexive and may or may not be transitive. Finally, that p holds only of agent a in state u_0, i.e., that $V(p) = \{(u_0, a)\}$ is shown by labelling the lower left node of the diagram with p.

Models are used to interpret $\mathcal{L}$ in a double-indexical way, as follows:

$M, w, a \models \rho$	iff	$(w, a) \in V(\rho)$, for $\rho \in \mathsf{Prop}$
$M, w, a \models n$	iff	$g(n) = a$, for $n \in \mathsf{ANom}$
$M, w, a \models \neg\varphi$	iff	$M, w, a \not\models \varphi$
$M, w, a \models (\varphi \wedge \psi)$	iff	$M, w, a \models \varphi$ and $M, w, a \models \psi$
$M, w, a \models K\varphi$	iff	$M, v, a \models \varphi$ for every $v \in W$ such that $\langle w, v\rangle \in k_a(w)$
$M, w, a \models F\varphi$	iff	$M, w, b \models \varphi$ for every $b \in A$ such that $\langle a, b\rangle \in f_w(a)$
$M, w, a \models A\varphi$	iff	$M, w, b \models \varphi$ for every $b \in A$.

As usual in modal logic, we can define the duals of the operators, which we write inside angle brackets: $\langle K\rangle = \neg K\neg$ 'it is epistemically possible for me that', $\langle F\rangle = \neg F\neg$ 'I have a friend who', and $\langle A\rangle = \neg A\neg$ 'there is someone who'. The English glosses are not so exact and require some manipulation to get proper translations, because of the way pronouns work in English. For example, if d represents 'I am in danger' then $\langle F\rangle Kd$ means 'I have a friend who knows that he is in danger' rather than 'I have a friend who I know that I am in danger' which is not even grammatically correct.

We also use abbreviations for the hybrid-logic-like operators $@_n\varphi = A(n \to \varphi)$ (equivalently, $\langle A\rangle(n \wedge \varphi)$).[1] So, for example, if $\underline{n}$ is Charlie then the operator $@_n$ simply shifts the indexical subject to Charlie, so that $@_n d$ means 'Charlie is in danger'.

We say that M is a *named agent* model, if every agent in M has a name, i.e., for each $a \in A$, there is an $n \in \mathsf{ANom}$ such that $g(n) = a$. The model depicted in Figure 1 is *not* a named agent model because agent b has no name. In what follows we will assume that all agents are named, and so use the letters representing the agents in the diagram also as names in the language, abusing the distinction between n and $\underline{n}$.

The advantage of working with named agent models is that we can define an operator $\downarrow n$ by

$$\downarrow n\ \varphi \quad := \quad \bigvee_{m \in \mathsf{ANom}} (m \wedge \varphi[^n_m])$$

where $\varphi[^n_m]$ is the result of replacing agent nominal n by m in φ. This provides a way of referring to 'me' inside the scope of other operators, by shifting the referent of n to the current agent. When M is a named agent model,

$$M, w, a \models \downarrow n\ \varphi \qquad \text{iff} \quad M[^n_a], w, a \models \varphi.$$

where $M[^n_a]$ is the result of changing M so that n now names a.[2] This allows us to express such propositions as, $\downarrow n\, FK\langle F\rangle n$, which says 'all my friends know they are friends with me', at least on the assumption that every agent has a name. The assumption is not so restrictive, since in all applications we have so far considered, we can assume that a finite set of agents is specified in advance.[3]

Relations and change.

We will define a class of operators $\mathcal{D}$ and corresponding actions on models such that for each $\Delta \in \mathcal{D}$ and each M model for $\mathcal{L}$, there is an $\mathcal{L}$ model ΔM, and for each state w of M, a state Δw of ΔM. We then extend $\mathcal{L}$ to a language $\mathcal{L}(\mathcal{D})$ of *dynamic epistemic friendship logic* (DEFL) by adding the elements of $\mathcal{D}$ as propositional operators and defining

$$M, w, a \models \Delta\varphi \qquad \text{iff} \quad \Delta M, \Delta w, a \models \varphi$$

To define $\mathcal{D}$, we use the language of propositional dynamic logic (PDL) with basic programs K, F and A, given by

$$\begin{array}{ll} \mathcal{T} & \pi ::= K \mid F \mid A \mid \varphi? \mid (\pi;\pi) \mid (\pi \cup \pi) \mid \pi^* \\ \mathcal{F} & \varphi ::= \rho \mid n \mid \neg\varphi \mid (\varphi \vee \varphi) \mid \langle\pi\rangle\varphi \end{array}$$

for $\rho \in \mathsf{Prop}$ and $n \in \mathsf{ANom}$. The denotation of program terms $\pi \in \mathcal{T}$ and formulas $\varphi \in \mathcal{F}$ in a model M are defined in the manner shown in Table 1. Note in particular, the clauses for K, F and A, in which these program terms refer to the accessibility relations of the corresponding operators of EFL, when interpreted two-dimensionally. Complex program terms are built up in the usual way: $(\pi_1;\pi_2)$ for the

$[\![\rho]\!]^M$	=	$V(\rho)$, for $\rho \in \mathsf{Prop}$
$[\![n]\!]^M$	=	$W \times \{g(n)\}$, for $n \in \mathsf{ANom}$
$[\![(\varphi \wedge \psi)]\!]^M$	=	$[\![\varphi]\!]^M \cap [\![\psi]\!]^M$
$[\![\neg\varphi]\!]^M$	=	$W \setminus [\![\varphi]\!]^M$
$[\![\langle\pi\rangle\varphi]\!]^M$	=	$\{w \in W \mid w[\![\pi]\!]^M v$ and $v \in [\![\varphi]\!]^M$ for some $v \in W\}$
$[\![K]\!]^M$	=	$\{\langle(w,a),(v,a)\rangle \mid k_a(w,v)\}$
$[\![F]\!]^M$	=	$\{\langle(w,a),(w,b)\rangle \mid f_w(a,b)\}$
$[\![A]\!]^M$	=	$\{\langle(w,a),(w,b)\rangle \mid a,b \in A,\ w \in W\}$
$[\![\varphi?]\!]^M$	=	$\{\langle w,w\rangle \mid w \in [\![\varphi]\!]^M\}$
$[\![\pi_1;\pi_2]\!]^M$	=	$\{\langle w,v\rangle \mid w[\![\pi_1]\!]^M s$ and $s[\![\pi_2]\!]^M v$ for some $s \in W\}$
$[\![\pi_1 \cup \pi_2]\!]^M$	=	$[\![\pi_1]\!]^M \cup [\![\pi_2]\!]^M$
$[\![\pi^*]\!]^M$	=	$\{\langle w,v\rangle \mid w = v$ or $w_i[\![\pi]\!]^M w_{i+1}$ for some $n \geq 0$, $w_0, \ldots, w_n \in W$, $w_0 = w$ and $w_n = v\}$

Table 1: Semantics of PDL terms and formulas

[1] Although reminiscent of hybrid logic, the 'agent nominals' n, binder $\downarrow n$ and now the operator $@_n$ are not exactly the same as their hybrid-logic namesakes, but are rather some sort of two-dimensional cousins. A true nominal, for example, is a proposition that is logically compelled to be satisfied by exactly one evaluation index, which in the case of our models, would have to be the pair $\langle w, a\rangle$.

[2] More precisely, $M[^n_a] = \langle W \times A, k, f, g[^n_a], V\rangle$ where for $m \in \mathsf{ANom}$, $g[^n_a](m) = a$ if $m = n$ and $g(m)$, otherwise.

[3] $\downarrow n$ can be introduced as a primitive, but without the restriction to named agent models, the resulting logic can be shown to be undecidable by encoding tiling problems (in the manner of [2]).

relational composition of π_1 and π_2, $(\pi_1 \cup \pi_2)$ for their union (or choice), $\varphi?$ for the 'test' consisting of a link from (w, a) to itself iff $M, w, a \models \varphi$, and π^* for the reflexive, transitive closure of π, which is understood as a form of iteration.
Note also that we have abused notation so that formulas φ of EFL, written with existential operators $\langle K\rangle$, $\langle F\rangle$ and $\langle A\rangle$, are also programs formulas (in F). This is justified by the obvious semantic equivalence:

$$M, w, a \models \varphi \text{ iff } (w, a) \in [\![\varphi]\!]^M$$

Now the class of dynamic operators will be defined using the theory of General Dynamic Dynamic Logic (GDDL) given in [4], which applies to any language of PDL. We refer the reader to that paper for full technical details, but we will introduce those parts of the theory that are required for present purposes.
The simplest GDDL operators are called PDL-*transformations*. These consist of assignment statements which transform models by redefining the basic programs. For example, the operator $[K := \pi]$ acts on model M to produce a new model $[K := \pi]M$ such that

$$[\![K]\!]^{[K:=\pi]M} = [\![\pi]\!]^M$$

On states, there is no change: $[K := \pi]w = w$, so the resulting DEFL operator has the following semantics:

$$M, w, a \models [K{:=}\pi]\varphi \quad \text{iff} \quad [K{:=}\pi]M, w, a \models \varphi$$

We must be a little careful in the choice of π so as to ensure that the resulting model $[K := \pi]M$ is still a model for EFL. For example, consider the program term $n?; K$. In M, this relates (u_0, a) to (u_1, b) in case $(u_0, a) \in [\![n]\!]^M$ and $(u_0, a)[\![K]\!]^M(u_1, b)$, which only holds when $g(n) = a$, $a = b$, and $k_a(u_0, u_1)$. Then $[K := n?; K]M$ is the structure $\langle W, A, k', f, V\rangle$ in which $k'_a = k_a$ and $k'_b = \emptyset$, for $b \neq a$. This is *not* a model for EFL. To make it into a model for EFL, we need to make each k_a reflexive. This can be done with the program term $\top?$, since $[\![\top?]\!]^M$ is the identity relation. Thus taking π to be $(n?; K) \cup \top?$ we get the model $[\![K]\!]^{[K:=(n?;K)\cup\top?]M}$ which is the structure $\langle W, A, k'', f, V\rangle$ in which $k''_a = k_a$ and k''_b is the identity relation for all $b \neq a$. The application of $[K := (a?; K) \cup \top?]$ to a particular model is illustrated in Figure 2. Here, M is a named agent model,

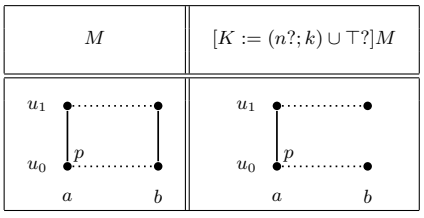


Figure 2: A simple PDL-transformation.

so we allow ourselves the abuse of notation involved in writing a for the name of a. In this model there are two friends, a and b, who are both ignorant about whether they are in state u_0 or u_1. p holds only of agent a in state u_0, so in particular, $M, u_0, b \models (K\neg p \wedge \neg K\langle F\rangle p)$, which means that agent b knows that she is not p but does not know whether she has a friend who is p. After the action $[K := (n?; K) \cup \top?]$ we get the model shown on the right, in which k_a is as before but now k_b is the identity relation. In the transformed model, agent b now knows that she has a friend who is p. Thus we get the dynamic fact:

$$M, u_0, b \models [K := (n?; K) \cup \top?]K\langle F\rangle p$$

In effect, the PDL-transformation, $[K := (n?; K) \cup \top?]$ is the action of revealing everything to every agent other than $\underline{n}$. We will consider more subtle forms of epistemic change in subsequent sections. Now it is time for a more extended example.

The Spy Network.

To take a Cold War example, suppose we are reasoning about the effect of a spy network being exposed.

> Bella (b) is friends with Charlie (c) and Erik (e), neither of whom are friends with each other. Unknown to the others is that Erik is a spy (s). The others are not spies, and Erik knows that because all spies know who else is a spy (we suppose). Bella knows that Charlie is not a spy, but Charlie does not know about her. After the network is exposed, all the spies and their friends will be interrogated by the police. But just before this happens a message is relayed to all agents revealing whether or not they are in danger, that is, whether they are a spy (which they would know in any case) or a friend of a spy.

A model M of the initial situation is depicted in Figure 3, with u_0 representing the actual state. In EFL we can state pertinent facts such as $@_b(K\neg s \wedge \neg K\langle F\rangle s)$ 'Bella knows that she is not a spy but doesn't know if a friend of hers is a spy'. We will write d 'I am in danger' as an abbreviation for $(s \vee \langle F\rangle s)$ 'either I'm a spy or I have a spy as a friend', and, for convenience, we have labelled those state-agent pairs at which d holds. Thus we can read that $@_b(d \wedge \neg Kd)$ 'Bella is in danger but doesn't know it', whereas $@_b K @_c \neg d$ 'Bella knows that Charlie is not in danger'.

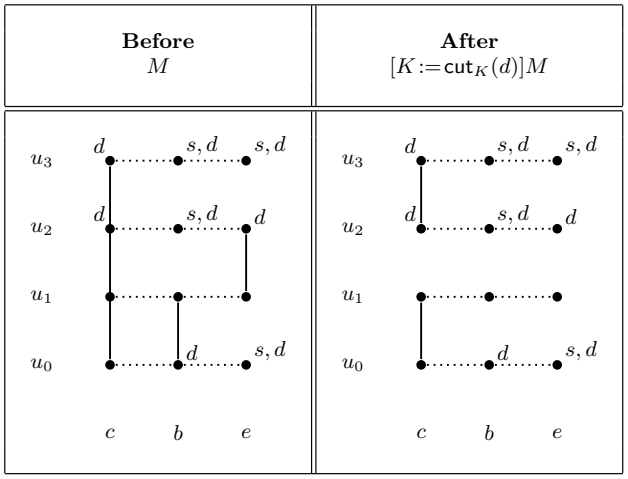


Figure 3: Spy Network

Now consider the PDL-term $\mathsf{cut}_K(\varphi)$ defined by

$$(\varphi?; K; \varphi?) \cup (\neg\varphi?; K; \neg\varphi?)$$

This relates $\langle w, a\rangle$ to $\langle v, b\rangle$ iff $a = b$, $k_a(w, v)$, and either φ is true of a in both states w and v or false of a in both

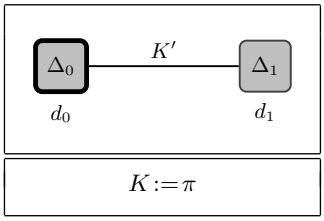
states. Thus the operator $[K := \mathsf{cut}_K(\varphi)]$ produces a new model $[K := \mathsf{cut}_K(\varphi)]M$ from M by removing the k_a links between states with conflicting values for φ (about a). Effectively, this 'reveals' to each agent whether or not φ holds (for them). This operator was first introduced in [14].
In our example, the situation after the revelation of d 'you are in danger' is given by the model $[K := \mathsf{cut}_K(d)]M$, shown in the right part of Figure 3. Notice that the k_c link between u_1 and u_2 are cut because $M, u_1, c \not\models d$ but $M, u_2, c \models d$; Charlie finds out that he is not in danger. Similarly, the k_b link between u_0 and u_1 is cut because Bella finds out that she *is* in danger ($@_b Kd$). Finally, the k_e link between u_1 and u_2 is cut because everyone now knows that Erik knows whether he is in danger (although only Bella knows which). Moreover, in the language of DEFL we can represent reasoning about these changes, such as the valid schema

$$[K := \mathsf{cut}_K(\varphi)]A(K\varphi \vee K\neg\varphi)$$

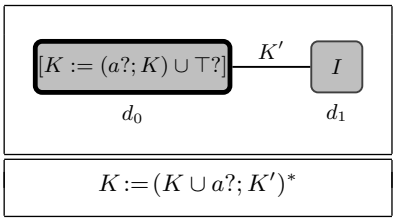
which states (for non-epistemic facts φ such as $d = \langle F\rangle s$) that after φ is revealed, everyone knows whether φ or not.

GDDL *operators.*

More complicated operators can be constructed from finite relational structures whose elements are each associated with a PDL transformation, and whose combined effect on the a model is calculated by 'integrating' them according to a further such transformation. A GDDL operator Δ is something that looks like this:



This represents an action d_0 (highlighted as the action that is actual performed) whose effect on the model is given by the PDL-transformation Δ_0. There is also an action d_1 with associated PDL-transformation Δ_1, and the relationship between d_0 and d_1 is marked as K'.[4] The effect of the operator on an EFL model M with domain W is computed by forming a product model M' (in the manner of [1]) whose domain is $W \times \{d_0, d_1\}$, in which the elements (w, d_i) represent the state resulting from action d_0 when the initial state is w. The model M' consists of copies of two models $[\Delta_0]M$ with domain $W \times \{d_0\}$ and $[\Delta_1]M$ with domain $W \times \{d_1\}$, and a duplication of the model occurring in Δ itself, with, in this case, $(w, d_0)[\![K']\!]^{M'}(w, d_1)$ for each $w, v \in W$. Finally, the model $[\Delta]M$ is computed by applying the 'integrating' transformation $[K := \pi]$ to M'. This uses a PDL program term π to compute the new value for K from a combination of relations in the copied models $[\Delta_0]M$ and $[\Delta_1]M$ and the new relation K' from Δ itself.[5]
This somewhat complex operation is best explained by looking at a simple example. Consider the case in which Δ_0 is the PDLtransformation $[K := (a?; K) \cup \top?]$ considered earlier, and Δ_1 is the identity transformation, I. We will also take π to be $(K \cup a?; K')^*$.



The action of this GDDL operator on the model M considered earlier, is show in Figure 4. It represents a situation in which an action d_0 gives complete information to all agents other than a. The occurrence of d_0 is known to all agents other than a, who stays completely in the dark. Not only is k_a unchanged in both $[\Delta_0]M$ (the top half of the diagram) and $[\Delta_1]M$ (the bottom half), but a is also ignorant about which of these two submodels she is in, as represented by the vertical lines in connecting the two halves of the a column: $(w, d_0)k_a(w, d_1)$ for all $w \in W$. Once again, we must check

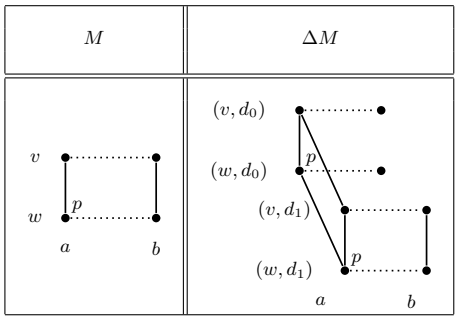


Figure 4: A simple GDDL operator in action.

that the resulting model is an EFL model. In this case, it is. The k_a and k_b relations are transitive thanks to the application of the $*$ operator in the integrating transformation $[K := (K \cup a?; K')^*]$.
We'll say that a GDDLtransformation Δ is a *general* EFL *dynamic operator* if it is in the language of PDL terms defined above, possibly augmented with internal relations such as K' and also preserves the property of being a EFL-model: whenever M is a EFL-model, so is ΔM.

2. SOCIAL ANNOUNCEMENTS

We now turn to direct communications, or 'announcements', within a social network. In the standard analysis of public announcement (PAL [7]), only the effect of announcement is modelled without reference to the agent who made the announcement and with the simplifying assumption that the message is received by all agents. In dynamic epistemic logic (following [1]), private announcements, in which a message is received by a limited set of agents are also considered. In the general case, within a social network, an announcement consists of an agent (the sender) transmitting some information (the message) to one or more other agents (the receivers) and each of these three components can be described in different ways, from different perspectives.[6] In this section, we will map out some of the subtleties.

[4]In the general case, as explained in [4], there may be many actions and many new relation symbols; also, propositional variables.
[5]Again, the general case is more flexible, allowing any of the basic expressions K, F, agent nominal and propositional variable to be reinterpreted at the integrating stage.
[6]We are aware of the attempts by others in this respect. [8]

As a starting point, we ignore the sender and define a basic act of communication in which a message ψ is sent (anonymously, we suppose) to a group of agents θ by

$$\mathsf{send}_\theta(\psi) = [K := (\theta?; \mathsf{cut}_K(\psi)) \cup (\neg\theta?; K)]$$

The action $\mathsf{send}_\theta(\psi)$ reveals the truth or falsity of ψ (which may be different for different agents) to all agents satisfying θ, and leaves the k_a relation unchanged for agents a not satisfying θ.

To see how this works, consider $\mathsf{send}_{\langle F\rangle b}(d)$ in the case of our spy network. This is an anonymous announcement to the friends of Bella (but not to Bella herself) whether or not they are in danger. The effect of this action is shown in Figure 5. The formula θ describing the receivers of the message is $\langle F\rangle b$, which is satisfied by Charlie and Erik in the actual state u_0. Thus only the relations k_c and k_e are changed; k_b remains the same. This is by no means our final analysis of

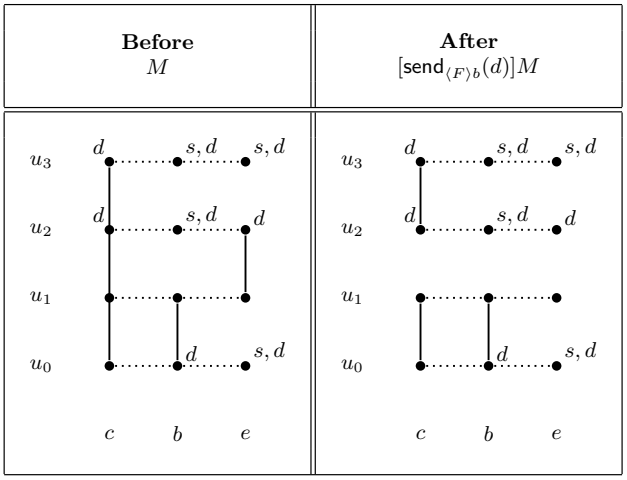


Figure 5: Restricting to Bella's friends

communication. For one thing, actions of this sort are only 'semi-private', i.e., directed at particular individuals, but with others not involved in the communication still aware that it has occurred. Later, we will need to make the analysis more complex to cope with a great degree of privacy, in which only the sender and receivers are aware that the communication has occurred. For example, after the communication to Bella's friends, Bella knows something that she didn't know before: before she knew that Charlie was not in danger, now she knows that Charlie knows this:

$$M, u_0, b \models [\mathsf{send}_{\langle F\rangle b}(d)]K@_cK\neg d$$

Yet before we get to the issue of privacy, we will bring the sender into our model, and explore some subtle distinctions about the nature of the message itself.

analysed specific types of communication network (i.e., communications that take place between one agent and another, or between an agent and a group of agents) when considering the issue of how distributed knowledge can be established by a group of agents through communication. Communication graphs were adopted by [6] to study communication between agents. Agent i directly receiving information from agent j is represented by an edge from agent i to agent j in such graph. Neither approach considers groups of agents described in terms of social relations.

Announcements about the sender.

The first case is that of a message sent by agent n to agents described by θ with a message ψ, which is understood to be about the sender, for example 'I am in danger'. We define $[n \triangleleft \psi! : \theta]\varphi$, the statement that φ holds such a communication, as

$$(@_nK\psi \to [\mathsf{send}_\theta(@_n\psi)]\varphi)$$

To make sense of this, we will look at a progression of simpler cases. First, with $\theta = \top$, the formula $[n\triangleleft\psi! : \top]\varphi$ means that φ holds after agent n publicly announces that ψ, noting that it simplifies to $(@_nK\psi \to [K :=\mathsf{cut}_K(@_n\psi)]\varphi)$.

We make the rather strong assumption that the message is known by the sender.[7] Suppose, for example, that Erik, unable to keep his secret any longer, told everyone that he is a spy. After this, everyone would know that he is a spy (and Bella, his friend, would know that she is in danger). This follows from the validity of $[e \triangleleft s! : \top]AK@_es$.[8] Note that $[b \triangleleft s! : \top]AK@_bs$ is also true (since it is valid!). This says that everyone would know that Bella is a spy after she announced it. But the reason is quite different: Bella could not announce that she is a spy, because she knows that she isn't.[9]

The second case is an announcement to a particular agent. In this case, θ is an agent nominal m and the formula $[n \triangleleft \psi! : m]\varphi$ means that φ holds after agent n announces to m that ψ. For example, Erik may be more cautious in his admission, telling only Bella, after which she, but not Charlie would know: $[e \triangleleft s! : b]@_bK@_es$ and $(\neg(b \vee K@_es) \to [e \triangleleft s! : b]\neg K@_es)$ are both valid, and the latter says that an agent who is neither Bella nor (already) knows that Erik is a spy, still doesn't know this after he announces it to Bella.

In the most general case, θ is a description of a group of agents. For example, $[b \triangleleft \neg s! : \langle F\rangle b]\varphi$ states that φ would hold after Bella tells her friends that she is not a spy. Again we have a useful validity: $[b \triangleleft \neg s! : \langle F\rangle b]@_bFK@_b\neg s$, which says that if Bella were to tell her friends that she is not a spy then they would all know that she isn't a spy.

Announcements about the receivers.

Announcements that are indexical about the receiver such as 'you are in danger' (announced to Bella by Erik) or 'you are my friends' (announced by Bella to her friends) can be expressed with a slight change that captures the different preconditions for announcements. We define $[n : \psi! \triangleright \theta]\varphi$, the statement that φ holds after agent n announces message ψ (about θ) to agents satisfying θ as

$$(@_nA(\theta \to \psi) \to [\mathsf{send}_\theta(\psi)]\varphi)$$

Again, we first consider the simple case of public announcement, represented by $[n : \psi! \triangleright \top]\varphi$, which can be seen to be equivalent to $(@_nKA\psi \to [K := \mathsf{cut}_K(\psi)]\varphi)$. Consider, for example, my announcing to everyone 'you are in danger'.

[7] The standard assumption of PAL that announcements are true is thus equivalent to supposing that they are made by God, or some other omniscient entity. [5] studied different types of agent (truth-teller, liar and bluffer) , how they make announcements, and are subsequently interpreted in communication.

[8] In fact, the information that Erik is a spy becomes common knowledge, as we will see in Section 6.

[9] It would be enough for Bella merely not to know that she is a spy for the announcement to be impossible.

The precondition that I know everyone is in danger is captured by the antecedent KAd, and after the announcement everyone knows that she is in danger, as is represented by the validity of $\downarrow n\ [n : d! \triangleright \top]AKd$.
The case of agent-to-agent announcement displays a nice symmetry between the two kinds of indexical message. Agent n announcing 'you are in danger' to agent m is equivalent to announcing (again to m) that m is in danger. More generally, the following equivalences are valid

$$[n : \psi! \triangleright m]\varphi \ \leftrightarrow\ [n \triangleleft @_m\psi! : m]\varphi$$
$$[n \triangleleft \psi! : m]\varphi \ \leftrightarrow\ [n : @_n\psi! \triangleright m]\varphi$$

This symmetry between announcements is more delicate when announcing to groups. Announcing 'you are in danger' to each of my friends is only the same as announcing to them 'all my friends are in danger' on the assumption that each friend knows only that she is my friend, and knows nothing about the others. Without this assumption,

$$[n : \psi! \triangleright \langle F\rangle n]\varphi \ \leftrightarrow\ [n \triangleleft @_nF\psi! : \langle F\rangle n]\varphi$$

is not always valid.[10]
For announcement to friends, an interesting new phenomenon arises. Consider the case of my announcing 'you are my friend' to my friends. That φ holds after such an announcement is represented by $[n : \langle F\rangle n! \triangleright \langle F\rangle n]$. The message is the same as the description of the set of receivers, so when this is expanded, we find that the precondition for the announcement is $\downarrow n\ KA(\langle F\rangle n \rightarrow \langle F\rangle n)$, which is valid, so the announcement can always be made, by anyone. But nonetheless, it can be informative, as can be seen from the validity of $\downarrow n\ [n : \langle F\rangle n! \triangleright \langle F\rangle n]FK\langle F\rangle n$, which says that after my making this announcement, my friends all know that they are my friends, something they may not have known before.
Finally, we note that any sender-indexical announcement to a group θ is equivalent to a receiver-indexical announcement to the same group θ in the case that there is at least one receiver ($A\neg\theta$ is false). The trick is that the statement ψ about n (the sender) is then equivalent to the statement $@_n\psi$ about any (every) receiver. More formally, the following is valid:[11]

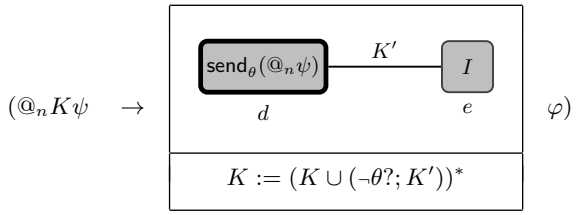
$$(\neg A\neg\theta \rightarrow [n \triangleleft \psi! : \theta]\varphi \ \leftrightarrow\ [n : @_n\psi! \triangleright \theta]\varphi)$$

Private announcements.

Communications of the form $[n \triangleleft \psi! : \theta]$ and $[n : \psi! \triangleright \theta]$ are only semi-private. Their effect on the model ensures that every agent will know that the announcement has occurred, if the sender satisfies the precondition, so, for example,

$$\downarrow n\ [n \triangleleft d! : m]AK(@_nKd \rightarrow @_mK@_nd)$$

is valid: after I announce to m that I am in danger, everyone will know that if I know I am in danger then m also knows it. This is (typically) an unjustified violation of the privacy of the communication between me and m.
To make the action $\mathsf{send}_\theta(\psi)$ private, we embed it in a GDDL operator similar to the one given in our earlier example. Thus, for the sender-indexical[12] version, that φ would hold after the private announcement of ψ by n to agents θ is be represented as



Call this formula $[\![n \triangleleft \psi! : \theta]\!]\varphi$. Inside the GDDL operator, the internal relation K' represents ignorance about whether the communication $\mathsf{send}^n_\theta(\psi)$ has occurred or not, the latter possibility represented by the identity transformation, I. The integrating transformation $[K := (K \cup (\neg\theta?; K'))^*]$ restricts ignorance of the K' kind to agents other than θ and factors this in to the new epistemic relation. The $*$ is needed to ensure that the result is an equivalence relation. We will see an example of this operator in action at the end of the next section.

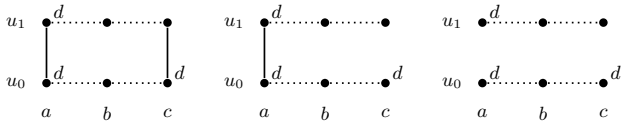
3. KNOWING YOUR FRIENDS

So far, the friendship relation in our models has been relatively tame, remaining fixed across epistemic states. We have used it to determine which group of agents receive a message, and even to specify the content of a message, but we have not yet considered ignorance about who is friends with whom. This is where it gets really interesting. We will explore some of the possibilities with an everyday example of infidelity and gossip.

> Peggy (p) knows that Roger (r) is cheating (c) on his wife, Mona (m). What's more, Roger knows that Peggy knows, because they met accidentally while he was with his mistress. Mona does not know about the affair, and both Peggy and Roger know this. The situation (for Roger) deteriorates when he discovers that Peggy is a terrible gossip. She is bound to have told all her friends about his affair. What Roger does not know is whether Mona is a friend of Peggy (she is).

We can represent the epistemic state of this network before Peggy's announcement with the model depicted in Figure 6, assuming that married couples are also friends. (The grey construction lines are only included to make the diagram easier to read; they have no epistemic or social significance.)

[10] For a simple counterexample, consider ψ to be d and the model M (shown left).



The precondition of $[b : d! \triangleright \langle F\rangle b]$ is $@_bKA(\langle F\rangle b \rightarrow d)$, which is equivalent to the precondition $@_bKFd$ of $[b \triangleleft @_bFd! : \langle F\rangle b]$ which is satisfied in M, and the resulting two models are shown middle and right. Yet these are easily distinguished, by taking φ to be $@_aK@_cd$.

[11] The key observation here is that the precondition for the sender-indexical announcement is $@_nK\psi$, which is equivalent to the precondition $@_nKU_A(\theta \rightarrow @_n\psi)$ when $U_A\neg\theta$ is false.

[12] The receiver-indexical version is obtained by changing the message and the precondition as in the simple semi-private case.

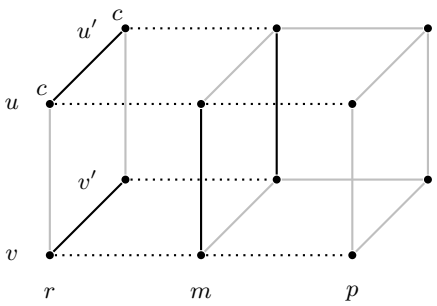


Figure 6: Roger's Quandry

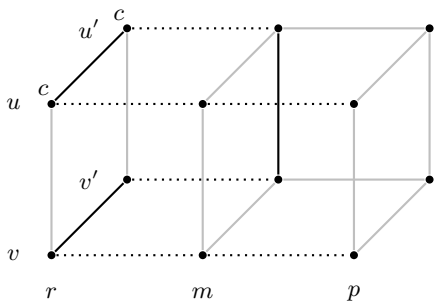


Figure 7: After Peggy's gossip

Note that the friendship relations are now different in different states. At u (the actual state) for Roger r, the statements listed in Table 2 are all true. As a result, we can compute that at w in the original model for Roger r, the formula

$$\downarrow n\ [p \triangleleft @_n c!:\langle F\rangle p] @_m K @_n c$$

is true, i.e., "I don't know that Mona will know about my cheating after Peggy tells her friends about it." That some

c	I'm cheating
$\downarrow n\ K(@_p K @_n c \wedge @_m \neg K @_n c)$	I know that Peggy (but not Mona) knows I am cheating.
$\downarrow n\ @_p K @_n K @_p K @_n c$	Peggy knows I know she knows I am cheating
$\neg K @_m \langle F\rangle p \wedge \neg K @_m \neg\langle F\rangle p$	I don't know whether Peggy and Mona are friends.
$\downarrow n\ @_p K @_n \neg K @_m \langle F\rangle p$	Peggy knows I don't know whether she and Mona are friends.

Table 2: Facts about Roger

proposition φ holds after the announcement 'Roger is cheating!' that Peggy makes to her friends is given by $[p \triangleleft @_r c!:\langle F\rangle p]\varphi$, which expands and simplifies to

$$(@_p K @_r c \rightarrow [K := (\langle F\rangle p?; \mathsf{cut}_K(@_r c)) \cup (\neg\langle F\rangle p?; K)]\varphi)$$

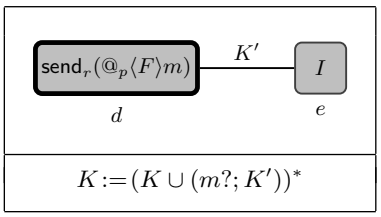
When evaluated at u, the presupposition that Peggy knows that Roger is cheating is satisfied, and so the formula φ is evaluated in the transformed model shown in Figure 7. (Note the missing vertical line in the middle.)
This is all very well, but Roger needs a little more privacy.

> Before returning home to face Mona, Roger is uneasy. He would really like to know whether or not she knows about his affair. He already knows that she knows if and only if she is friends with Peggy. So if Peggy told him that they are friends, he would be prepared for Mona's fury. But for his planned excuses to be convincing, Mona must not know that he knows she knows (about the affair). It is therefore very important that Peggy tells him in private.

Now let us suppose that the ever-loquacious Peggy announces to Robert privately that Mona is her friend, represented as $[\![p \triangleleft \langle F\rangle m!:r]\!]$. Now, whether the crucial proposition φ

$$(@_r K @_m K @_r c \wedge \neg @_m K @_r K @_m K @_r c)$$

(that Roger knows Mona knows he has been cheating but Mona doesn't know that he knows) holds must be determined by evaluating it in the model obtained by transforming the one in Figure 7 using the following GDDL operator, call it Δ:



The result is shown in Figure 8.
The upper half of the diagram represent the result of action d, Peggy telling Roger that she is friends with Mona ($\mathsf{send}_r(@_p\langle F\rangle m)$), whereas the lower half represent the result of action e, nothing (I); it is just a copy of the model in Figure 7. Mona is the only one of the three who doesn't know which action has taken place, and her ignorance is represented by the lines connected corresponding states in the upper and lower halves (in the m column). We see that $K @_m K @_r c$ holds of r in state (u, d), so Roger can meet Mona prepared.[13]
We may wonder about the accuracy of the model in representing Roger and Mona as friends after Peggy's announcement. Changes to the social network will be considered in Section 5.

4. ASKING QUESTIONS

As well as making announcements, agents in a social network can ask questions. Our approach to modelling questions will

[13]Even the additional level of privacy offered here is still not perfect, as it involves some change in Mona's knowledge. She goes from knowing that Roger doesn't know that she is friends with Peggy to not knowing this. However, one may just think that privacy is a matter of degree.

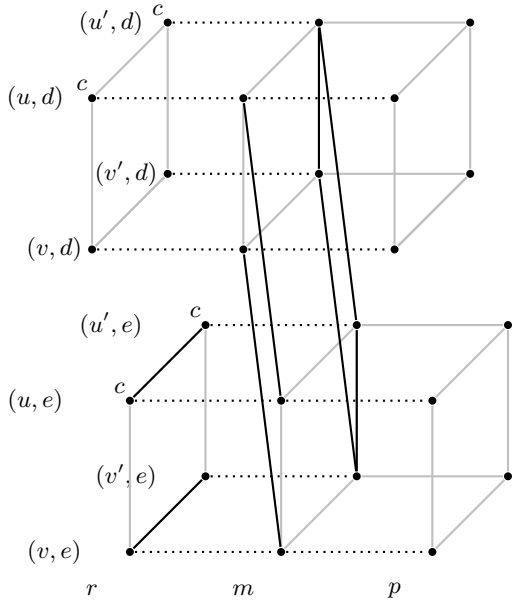


Figure 8: Peggy to Roger, privately.

assume that agents are cooperative to the extent that they answer those questions to which they know the answer.[14] A more elaborate model would consider the preferences of agents, but that is beyond the scope of the current paper. With this assumption, the effect of asking whether ψ of an agent a who knows that ψ is the same as an announcement by a that ψ. Likewise, the effect of asking whether ψ of an agent a who knows that $\neg\psi$ is the same as an announcement by a that $\neg\psi$. In the case that a does not know whether ψ, we assume that this also is communicated (possibly by the mere absence of an expected reply). With this in mind, we define $[n \triangleleft \psi? : m]\varphi$, the proposition that φ holds after agent n asks agent m whether ψ as

$$([m \triangleleft \psi! : n]\varphi \wedge [m \triangleleft \neg\psi! : n]\varphi \wedge [m \triangleleft \neg(K\psi \vee K\neg\psi)! : n]\varphi$$

In other words, φ holds after n asks m whether ψ just in case φ holds after in all three cases: (1) m answers 'yes', so announcing ψ to n (2) m answers 'no', so announcing $\neg\psi$ to n and (3) m answers 'I don't know', so announcing $\neg(K\psi \vee K\neg\psi)$ to n. This ensures that the following are valid:

$$(@_m K @_n p \to [n \triangleleft p? : m]@_n Kp)$$
$$(@_m K @_n \neg p \to [n \triangleleft p? : m]@_n K\neg p)$$
$$(@_m \neg(K @_n p \vee K @_n \neg p) \to [n \triangleleft p? : m]@_n K @_m \neg(K @_n p \vee K @_n \neg p))$$

So, for example, after Charlie c asks Erik e whether he (Charlie) is in danger, d, he will either know that he is in danger Kd or know that he is not in danger $K\neg d$, or know that Erik doesn't know whether or not he (Charlie) is in danger, $\downarrow n\ K@_e\neg(K@_n d \vee K@_n\neg d)$.

Sender-indexical questions can be distinguished from receiver-indexical questions in a way that parallels the distinction for announcements. The question 'Are you in danger?' from n to m, answered positively amounts to an announcement by m to n of 'I am in danger', and similarly with the 'you' and 'I' reversed.

As with announcements, this model of questions assumes that the answers are only semi-private. For example, after Charlie asks Erik whether he is in danger, a third-party will know that Charlie either knows whether he is in danger or knows that Erik doesn't know the answer. To make questioning more private, we need private announcements too. Here we will give one simple example.

> Roger approaches Peggy in private and asks her directly whether or not she and Mona are friends. Being sincere and cooperative, Peggy answers that they are. Mona, of course, knows nothing of their conversation.

This private question $[\![r : \langle F\rangle m? : p]\!]$ is defined by direct analogy with the semi-private question $[r \triangleleft \langle F\rangle m? : p]$ so that φ holds after the question is asked just in case

$$[\![p \triangleleft \langle F\rangle m! : r]\!]\varphi \wedge [\![p \triangleleft \neg\langle F\rangle m! : r]\!]\varphi \wedge [\![p \triangleleft \neg(K\langle F\rangle m \vee K\neg\langle F\rangle m)! : r]\!]\varphi$$

In this case, only the precondition of $[\![p \triangleleft \langle F\rangle m! : r]\!]$ is satisfied, and so the results are just as depicted in Figure 8.

Questions to groups present some further challenges. How would sincere and cooperative friends answer the question 'Am I in danger?'? For our present strategy to work they would have to do so by making an announcement. The problem is that if I have more than one friend who knows the answer, more than one announcement will follow. But in which order? Clearly, we must consider all possible orders, which in the general case involves quantification over an arbitrary number of friends. In finite named agent models this is possible, but a bit ugly, so we will pass over the details here.

5. CHANGING THE NETWORK

What makes networking intriguing is the dynamics of network changes. You can be friends with someone one day on Facebook, but you may drop him as a friend the following day or add someone else. Those acts, though simple, have a direct impact on information flow in communities. Consider the following:

> Roger, scared of the possibility that Mona will find out about his affair from Peggy, does all that he can to distance them. His smear campaign is designed to break their friendship and so protect his information.

To define the operation of deleting a friendship link, we first define the result of cutting the friendship link between agents n and m in one direction

$$\mathsf{cut}_F(n, m) = (\neg n?; F) \cup (F; \neg m?)$$

Then, to deleting the link between n and m we need to cut in both directions:[15]

$$[-F_{n,m}] = [F := \mathsf{cut}_F(n, m)][F := \mathsf{cut}_F(m, n)]$$

It is then fairly easy to show that $[\![F]\!]^{[-F_{nm}]M} = [\![F]\!]^M \setminus \{\langle n, m\rangle, \langle m, n\rangle\}$, as required.[16]

[14] For dealing with questions in terms of issue management in standard dynamic epistemic logic, we refer to [13]. Here we take a short-cut that reduces the action of asking a question to that of announcing the answer.

[15] It is also interesting to consider asymmetric relationships such as "following" on Twitter or "subscribing" on Facebook, as studied in [9].

[16] This follows from the fact that $a[\![F]\!]^{[F:=\mathsf{cut}_F(n,m)]M} b$ iff $a[\![F]\!]^M b$ and $\langle a, b\rangle \neq \langle n, m\rangle$.

Now how is this going to help Roger? Well, after the application of $[-F_{mp}]$ to the model of Figure **??**, Peggy's announcement to her friends that Roger is cheating has no effect; in fact, she has no friends to receive the message. So the model is unchanged. In other words, in this original model, it is true for Roger that

$$[-F_{mp}]\downarrow n\ [p \triangleleft @_n c!: \langle F\rangle p]@_m \neg K @_n c$$

'after Peggy loses Mona as a friend, even after she tells her friends that I am cheating, Mona won't know.'
Next we consider adding a friend. In the basic case, we can define the operation $[+F_{n,m}]$ by analogy with deletion, but more simply, as

$$[F =: F \cup (n?; A; m?)]$$

But a more interesting model of adding friends follows the protocol of Facebook and other online social networks, whereby one must first request friendship. To capture this aspect of network change, we need to represent whether or not an agent *wants* to be friends with another agent. In a fuller account, this could be done with a preference order, showing that the agent prefers states in which they are friends to those in which they are not. But for now, suppose that there is some additional indexical relation d_w in our models, with $d_w(a, b)$ interpreted to mean that in state w, agent a wants to become friends with agent b. Let D be the corresponding modal operator.
The question 'do you want to be my friend?' from n to m is thus represented by $[n \triangleleft \langle D\rangle n?: m]$, but as a *request* we interpret this as involving an action: if the answer is 'yes' then we become friends; otherwise, there is no change to the social network, thought there are consequent epistemic changes, such as my learning that you don't want to be my friend. That φ holds after this 'friend request' is therefore represented by

$$\begin{aligned}[\mathsf{add}(m)]\varphi = \quad & \downarrow n\ [n \triangleleft \langle D\rangle n?: m]((K@_m\langle D\rangle n \wedge [+F_{n,m}]\varphi) \\ & \vee(\neg K@_m\langle D\rangle n \wedge \varphi))\end{aligned}$$

A private version of this operation can be obtained by replacing the announcement and network change by a GDDL-based version.
The following validity shows some of the epistemic consequence of friend requests:

$$\begin{aligned}&\downarrow n\ ((\neg\langle F\rangle m \wedge \neg K@_m\langle D\rangle n) \rightarrow \\ &[\mathsf{add}(m)]((K@_m K\langle D\rangle n \wedge \langle F\rangle m) \vee (K@_m\neg K\langle D\rangle n \wedge \neg\langle F\rangle m))\end{aligned}$$

If I'm not friends with m and don't know that she wants to be my friend, then were I to ask her, I would either know that she knows she wants to be friends and we would be friends, or know that she doesn't know she wants to be friends and we wouldn't be friends.

6. COMMON KNOWLEDGE

In the context of social networks or communities, common knowledge is clearly an important notion. One can easily imagine the situations in which we want to reason about whether or not something is commonly known in some community or among my friends. There are at least two subtleties involved in making this precise. The first has to do with identifying the group of agents who are said to have common knowledge. This may be by means of a specific list ('Charlie, Bella, and Erik'), or a description ('Charlie's friends') or even an indexical description ('friends of mine'). Secondly, the information that is shared may be rigid ('it is common knowledge that Charlie is not a spy') or indexical (e.g. 'it is common knowledge among Charlie's friends that I am in danger' or 'it is common knowledge among my friends that they are in danger.')
To capture all these cases, first define $\overline{K}_a$ to be $(A; a?; K)$. Then $[\overline{K}_a]\varphi$ means that agent a knows that φ, as justified by the following equivalence:

$$M, w, b \models [\overline{K}_a]\varphi \quad \text{iff} \quad M, v, a \models \varphi \text{ for all } v \in W \text{ such that } k_a(w, v).$$

Here φ could be an indexical proposition, so, for example, 'Charlie knows that he is not a spy' would be represented by $[\overline{K}_c]\neg s$, whereas 'Bella knows that Charlie is not a spy' would have to be represented as $[\overline{K}_b]@_c\neg s$. Now, for common knowledge, define

$$\mathsf{c}_\theta = (A; \theta?; K)^*; A; \theta?$$

and interpret $[\mathsf{c}_\theta]\varphi$ to mean, roughly, that there is common knowledge among θ-agents that φ. So this enables us to talk, in our formal language, about the common knowledge of some group.This definition seems more general than the standard notion of common knowledge (see e.g. [3]) . It is justified by the following applications, each of which can be suitably generalised.

1. Common knowledge among an enumerated set of agents about a non-indexical proposition. For example, that there is common knowledge between Bella (b) and Charlie (c) that Charlie is not a spy (s) can be represented by $[\mathsf{c}_{(b\vee c)}]@_c\neg s$.[17] To justify this claim, first note that the standard way of defining common knowledge for a group of agents G is to introduce a new operator C_G such that

   $$M, w, a \models C_G\varphi \quad \text{iff} \quad M, v, a \models \varphi \text{ for all } \langle u, v\rangle \in (\textstyle\bigcup_{a'\in G} k_{a'})^*$$

   We can then prove that, for example, $[\mathsf{c}_{(b\vee c)}]@_c\neg s$ is equivalent to $C_{\{b,c\}}@_c\neg s$.[18]

2. Common knowledge among a non-indexically described group of agents about a non-indexical proposition. For example, that it is common knowledge among Peggy's (p) friends that Roger (r) is cheating (c) can be represented as $[\mathsf{c}_{\langle F\rangle p}]@_r c$. This implies that every friend of Peggy knows that Roger is cheating ($@_p FK@_r c$), but also that each of them knows that all of Peggy's friends know this ($@_p FK@_p FK@_r c$), and that each of them knows they all know *that* ($@_p FK@_p FK@_p FK@_r c$), and so on. As such, it is not equivalent to any statement of the form $C_G\varphi$. In particular, if, say, Peggy's only friends are Mona (m) and Nancy (n), it may

[17] Another concrete and interesting area of application is our ordinary email exchange, see an interesting analysis in [12].
[18] The argument is simple. First note that $(A; (b\vee c)?; K)^*$ is equivalent to $(\overline{K}_b \cup \overline{K}_c)^*$. Also, since $@_c\neg s$ is non-indexical, $[A; (b\vee c)?]@_c\neg s$ is equivalent to $@_c\neg s$. Thus $[\mathsf{c}_{(b\vee c)}]@_c\neg s$ is equivalent to $[(\overline{K}_b \cup \overline{K}_c)^*]@_c\neg s$, which is obviously equivalent to $C_{\{b,c\}}@_c\neg s$.

not have the same truth value as $C_{\{m,n\}}@_r c$, which is compatible with Mona's and Nancy's ignorance about what Peggy's friends (in general) know.

3. Common knowledge among a non-indexically described group of agents about a proposition that is indexical with respect to each member of the group. This is the subtlest case. For example, after the spy network has been exposed, that it is common knowledge among Erik's (e) friends that they are in danger (d) is represented by $[\mathsf{c}_{\langle F\rangle e}]d$. This implies that every friend of Erik (the spy) knows that s/he is in danger ($@_e FKd$), that each of them knows they all know this ($@_e FK@_e FKd$), and so on. Again, this is compatible with their ignorance about the friendship relation, so long as in all epistemically indistinguishable states, the friends of Erik (whoever they may be) are still in danger. The reason to have the final part $A;\theta?$ in the above definition of c_θ is this: when φ is indexical, we need to ensure that it is about the members of θ. When φ is not indexical, this part is redundant.

4. Common knowledge among an indexically described group of agents about a non-indexical proposition. For example, that it is common knowledge among my friends that Roger is cheating is represented by $\downarrow n\ [\mathsf{c}_{\langle F\rangle n}]@_r c$. This is a straightforward generalisation of the previous case to an indexically specified description, with the $\langle F\rangle n$ using the nominal n, which is bound to the speaker by $\downarrow n$.

5. Common knowledge among an indexically described group of agents about a proposition that is indexical with respect to the speaker. For example, that there is common knowledge among my friends that I am not a spy is represented by $\downarrow n\ [\mathsf{c}_{\langle F\rangle n}]@_n \neg s$. This is really no more complicated than the last case. Again, the indexical work is all done by $\downarrow n$ in creating a temporary name 'n' for the speaker. Within that context, both the description of group ($\langle F\rangle n$) and the content of the common knowledge $@_n \neg s$ are both non-indexical.

6. Common knowledge among an indexically described group of agents about a proposition that is indexical with respect to each member of the group. For example, that it is common knowledge among my friends that they are in danger represented by $\downarrow n\ [\mathsf{c}_{\langle F\rangle n}]d$. This is an obvious generalisation of the previous cases.

Other useful specifications of groups of agents as the subjects of common knowledge include 'common knowledge of φ in my community' ($\downarrow n\ [\mathsf{c}_{\langle f^*\rangle n}]\varphi$), 'common knowledge of φ among those who know they are in danger' ($[\mathsf{c}_{Kd}]\varphi$), 'common knowledge of φ among those who know they are my friends' ($\downarrow n\ [\mathsf{c}_{K\langle F\rangle n}]\varphi$).

7. CONCLUDING REMARKS

What has emerged from this study is an appreciation of the diversity of subtle logic distinctions when combining epistemic and social relations, especially when allowing indexical propositions, as are very common in the social setting. Although Facebook was an inspiration for this work, we have only scratched the surface. Facebook offers many interesting features that would be good to model, such as the wall, commenting, and liking. There are many directions in which the rather tight assumptions of epistemic friendship logic can be relaxed, such as by dropping symmetry for friendship, allowing degrees or hierarchies of friends (as in [10]), diluting knowledge to belief and adding preference.

8. REFERENCES

[1] A. Baltag, L. S. Moss, and S. Solecki. The logic of public announcements, common knowledge and private suspicious. Technical Report SEN-R9922, CWI, Amsterdam, 1999.

[2] P. Blackburn and J. Seligman. What are hybrid languages? In M. Kracht, M. de Rijke, H. Wansing, and M. Zakharyaschev, editors, *Advances in Modal Logic*, volume 1 of , pages 41–62. CSLI Publications, Stanford University, 1998.

[3] R. Fagin, J. Halpern, Y. Moses, and M. Vardi. *Reasoning about Knowledge*. The MIT Press, 1995.

[4] P. Girard, J. Seligman, and F. Liu. General dynamic dynamic logic. In T. Bolander, T. Braüner, S. Ghilardi, and L. S. Moss, editors, *Advances in Modal Logics Volume 9*, pages 239–260, 2012.

[5] F. Liu and Y. Wang. Reasoning about agent types and the hardest logic puzzle ever. *Minds and Machines*, 2013. To appear.

[6] E. Pacuit and R. Parikh. Reasoning about communication graphs. In D. G. Johan van Benthem, Benedikt Loewe, editor, *Interactive Logic*, pages 13–60. Amsterdam University Press, 2007.

[7] J. Plaza. Logics of public announcements. In *Proceedings of the 4th International Symposium on Methodologies for Intelligent Systems*, 1989.

[8] F. Roelofsen. Exploring logical perspectives on distributed information and its dynamics. Master's thesis, ILLC,The University of Amsterdam, 2005.

[9] J. Ruan and M. Thielscher. A logic for knowledge flow in social networks. In *Australasian Conference on Artificial Intelligence*, pages 511–520, 2011.

[10] J. Seligman, P. Girard, and F. Liu. Logical dynamics of belief change in the community. 2013. Under submission.

[11] J. Seligman, F. Liu, and P. Girard. Logic in the community. In M. Banerjee and A. Seth, editors, *ICLA*, volume 6521 of *Lecture Notes in Computer Science*, pages 178–188, 2011.

[12] F. Sietsma and K. Apt. Common knowledge in email exchanges. In J. Eijck and R. Verbrugge, editors, *Proceedings of the Workshop on Reasoning About Other Minds: Logical and Cognitive Perspectives*, pages 5–19, 2011.

[13] J. van Benthem and Ştefan Minică. Toward a dynamic logic of questions. In E. P. X. He, J. Horty, editor, *Logic Rationality and Interaction*, pages 27–41, Chongqing, China, August 2009. Springer.

[14] J. van Benthem and F. Liu. Dynamic logic of preference upgrade. *Journal of Applied Non-Classical Logic*, 17:157–182, 2007.

An Epistemic Approach to Compositional Reasoning about Anonymity and Privacy

Yasuyuki Tsukada
NTT Communication Science Laboratories, NTT Corporation
3-1 Morinosato-Wakamiya, Atsugi, 243-0198 Japan
tsukada.yasuyuki@lab.ntt.co.jp

Hideki Sakurada
NTT Communication Science Laboratories, NTT Corporation
3-1 Morinosato-Wakamiya, Atsugi, 243-0198 Japan
sakurada.hideki@lab.ntt.co.jp

Ken Mano
NTT Communication Science Laboratories, NTT Corporation
3-1 Morinosato-Wakamiya, Atsugi, 243-0198 Japan
mano.ken@lab.ntt.co.jp

Yoshifumi Manabe
NTT Communication Science Laboratories, NTT Corporation
2-4 Hikaridai, Seika-cho, Kyoto, 619-0237 Japan
manabe.yoshifumi@lab.ntt.co.jp

ABSTRACT

In this paper, we present an epistemic logic approach to the compositionality of several privacy-related information-hiding/disclosure properties. The properties considered here are anonymity, privacy, onymity, and identity. Our initial observation reveals that anonymity and privacy are not necessarily sequentially compositional; this means that even though a system comprising several sequential phases satisfies a certain unlinkability property in each phase, the entire system does not always enjoy a desired unlinkability property. We show that the compositionality can be guaranteed provided that the phases of the system satisfy what we call the independence assumptions. More specifically, we develop a series of theoretical case studies of what assumptions are sufficient to guarantee the sequential compositionality of various degrees of anonymity, privacy, onymity, and/or identity properties. Similar results for parallel composition are also discussed.

Categories and Subject Descriptors

F.4.1 [**Mathematical Logic and Formal Languages**]: Mathematical Logic—*Modal logic*; D.2.4 [**Software Engineering**]: Software/Program Verification—*Formal methods*

General Terms

Security, Theory, Verification

Keywords

Epistemic logic, anonymity, privacy, compositionality, modular reasoning

1. INTRODUCTION

An information system generally consists of a number of subsystems. If some subsystems are shown to have certain formal properties and some others shown to have different properties, the question arises as to how we can deduce that the total system has certain formal properties. Or, more complicatedly, the system may possibly consist of a variety of subsystems that have various degrees of multiple properties. Thus, the concept of *compositionality* plays a key role in a modular approach to formal reasoning about complex information systems.

This paper deals with a logical approach to the compositionality of several privacy-related information-hiding/disclosure properties. Since privacy and related properties such as those discussed in [21, 2] have become crucial requirements for today's information systems, the compositionality of those properties has also become a concern. The properties considered here are *anonymity*, *privacy*, *onymity*, and *identity* (Fig. 1). Intuitively, we can understand anonymity to be the property of *hiding who* performed a certain specific action, privacy that of *hiding what* was performed by a certain specific agent, onymity that of *disclosing who* performed a certain specific action, and identity that of *disclosing what* was performed by a certain specific agent. A series of previous studies by Halpern and O'Neill [12], Mano *et al.* [19], and Tsukada *et al.* [26] showed that these properties can be formulated concisely in terms of *epistemic logic* (or *the modal logic of knowledge*) for multiagent systems.

For example, *sender anonymity* can be formulated in terms of our epistemic logic as

$$\theta(i, send(m)) \Rightarrow \bigwedge_{i' \in I_A} P_j[\theta(i', send(m))].$$

Here, I_A, called an *anonymity set*, denotes a set of possible senders. We read this formula as "if an agent i sends a message m, then the observer j thinks that it is possible that every agent i' in I_A performs the sending action." In other words, this formula means that the observer j does not know who sends the message m. On the other hand, *message privacy* can be formulated as

$$\theta(i, send(m)) \Rightarrow \bigwedge_{a' \in A_I} P_j[\theta(i, a')].$$

Here, A_I, called a *privacy set*, denotes a set of possible sending actions, that is, $\{send(m') \mid m' \text{ is a possible message}\}$. This formula should be read as "if an agent i sends a mes-

TARK 2013, Chennai, India.

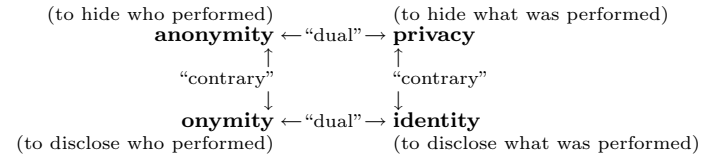


Figure 1: Privacy-related information-hiding/disclosure properties.

sage m, then the observer j thinks that it is possible that the agent i performs every sending action a' in A_I." In other words, this formula means that the observer j does not know what message is sent from the agent i. We may say that these two properties—sender anonymity and message privacy—are "dual" because each of the above two formulas can be obtained from the other by interchanging "who" with "what," or more specifically, I_A with A_I. We can also define onymity and identity as the "contrary" of anonymity and privacy, respectively, in terms of epistemic logic. Thus, epistemic logic enables us to succinctly describe formal specifications of various privacy-related information-hiding/disclosure properties of information systems.

In this paper, the epistemic logic approach developed in [12, 19, 26] is further exploited to discuss the compositionality of multiple properties comprising anonymity, privacy, onymity, and identity. More specifically, the contributions of this paper can be summarized as follows. First, we indicate that anonymity and privacy are not necessarily sequentially compositional. (This may be contrary to our intuition, because we might think that anonymity/privacy can be reinforced by sequentially connecting anonymous/private communication channels.) To show this indication, we introduce, as a motivating example, an abstract model of an anonymous members-only bulletin board system, which comprises two sequential phases, namely, the registration and posting phases. We show that the composition of anonymity in the registration phase and privacy in the posting phase does not necessarily induce anonymity or privacy in the entire system. If we regard anonymity and privacy as special cases of *unlinkability*, this indication can be paraphrased by saying that even though a system comprising several sequential phases satisfies a certain unlinkability property in each phase, the system as a whole does not always enjoy a desired unlinkability property. For example, our epistemic logic approach shows that a chain $M_1 * M_2$ of two *mix-servers* [5] does not necessarily guarantee unlinkability between incoming and outgoing messages even though both M_1 and M_2 do. This non-compositionality of unlinkability can be viewed as being analogous to the non-transitivity of inequality: $a \neq b$ and $b \neq c$ do not necessarily imply $a \neq c$. Second, we show that the sequential compositionality of anonymity and privacy can be guaranteed provided that the phases of the system satisfy what we call the *independence assumptions*. We develop a series of case studies of what assumptions are sufficient to guarantee the sequential compositionality of various degrees of anonymity, privacy, onymity, and/or identity properties. These compositionality results are summarized in Table 1. Third, we show that similar compositionality results can be obtained for parallel composition. We demonstrate that some variations of independence assumptions also play important roles in guaranteeing the parallel compositionality of anonymity and privacy.

Related Work

A considerable amount of substantial research on the measurement, characterization, and taxonomy of privacy and related information-hiding/disclosure properties has been undertaken from various standpoints [7, 23, 8, 25, 14, 17, 21, 30]. The present paper focuses on formal approaches to privacy-related properties, since our primary motivation is to contribute to the development of a new methodology for the formal verification of these properties.

Formal approaches to privacy-related information-hiding properties go back to the seminal work of Schneider and Sidiropoulos [22], who formulated the concept of *strong anonymity* in terms of a process calculus called CSP. Since then, this concept has been further developed and elaborated in various computational or logical frameworks such as ACP [20], applied π calculus [6], I/O-automata [16], category theory [13], and epistemic logic [24, 27, 10, 15, 29, 1, 28, 18, 4, 3].

Although the approach presented in this paper shares a common style of anonymity definitions with these epistemic logic approaches, it directly builds on the approach described by Halpern and O'Neill [12]. Within Halpern and O'Neill's framework, Mano *et al.* [19] formulated privacy as the dual of anonymity and showed that these two properties can be related by a newly proposed information-hiding property called *role interchangeability*. They proved the role-interchangeability property of a practical electronic voting protocol, thereby demonstrating the *voter anonymity* and *vote privacy* properties of the protocol. Further, Tsukada *et al.* [26] considered the logical contraries of anonymity and privacy, thereby giving formal definitions of onymity and identity. In particular, they showed that some weak forms of anonymity and privacy are compatible with some weak forms of onymity and identity, respectively. They also discussed the relationships between their proposed definitions and existing standard terminology, in particular Pfitzmann and Hansen's consolidated proposal [21]. The epistemic logic approach developed in [12, 19, 26] has recently been extended by Goriac [11], where a wider spectrum of privacy-related properties including *undetectability*, *unobservability*, and *pseudonymity* are formulated and discussed.

2. EPISTEMIC DEFINITIONS OF ANONYMITY AND PRIVACY

We briefly review epistemic logic for multiagent systems. Notions and terminologies are borrowed from [9, 12].

A *multiagent system* consists of n *agents* with their *local states* and develops over time. We assume that an agent's local state encapsulates all the information to which the agent has access. Let $I = \{i_1, \ldots, i_n\}$ be the set of n agents. A *global state* is defined as the tuple $(s_{i_1}, \ldots, s_{i_n})$ with all local states from i_1 to i_n. A *run* is a function from *time*, ranging over the natural numbers, to global states. A *point* is a pair

(r, m) comprising a run r and a time m, and the global state at a point (r, m) is denoted by $r(m)$. The function r_x of m is the projection of $r(m)$ to x's component, so that $r_x(m) = s_x$ if $r(m) = (s_{i_1}, \ldots, s_{i_n})$ for $x = i_1, \ldots, i_n$. A *system* is a set of runs. The set of all points in a system $\mathcal{R}$ is denoted by $\mathcal{P}(\mathcal{R})$.

In a multiagent system, we can define the knowledge of an agent on the basis of the indistinguishability of the state for the agent. Given a system $\mathcal{R}$ and an agent i, let $\mathcal{K}_i(r, m)$ be the set of points in $\mathcal{P}(\mathcal{R})$ that i thinks are possible at (r, m); that is, $\mathcal{K}_i(r, m) = \{(r', m') \in \mathcal{P}(\mathcal{R}) \mid (r', m') \sim_i (r, m)\}$, where $(r', m') \sim_i (r, m)$ means that $r'_i(m') = r_i(m)$. We can say that an agent i "knows" ϕ at a point (r, m) if ϕ is true at all points in $\mathcal{K}_i(r, m)$.

The *formulas* of epistemic logic are inductively constructed from a set Φ of *primitive propositions* (such as "the key is k" or "an agent i sent a message m to an agent j"), the usual logical connectives, and an epistemic operator K_i that represents the knowledge of agent i. The meaning of each formula can be determined when each primitive proposition is given an interpretation. An *interpreted system* $\mathcal{I}$ consists of a pair $(\mathcal{R}, \pi)$ comprising a system $\mathcal{R}$ and an *interpretation* π that maps each point to the truth-value assignment function for Φ for the point. In other words, $(\pi(r, m))(p) \in \{true, false\}$ for each $p \in \Phi$ and $(r, m) \in \mathcal{P}(\mathcal{R})$. Given an interpreted system $\mathcal{I} = (\mathcal{R}, \pi)$ and a point (r, m) in $\mathcal{R}$, we define what it means for a formula ϕ to be true at (r, m) in $\mathcal{I}$ by induction on the structure of formulas. Typical cases are as follows: $(\mathcal{I}, r, m) \models p$ if $(\pi(r, m))(p) = true$; $(\mathcal{I}, r, m) \models \neg\phi$ if $(\mathcal{I}, r, m) \not\models \phi$; $(\mathcal{I}, r, m) \models \phi \wedge \psi$ if $(\mathcal{I}, r, m) \models \phi$ and $(\mathcal{I}, r, m) \models \psi$; $(\mathcal{I}, r, m) \models K_i\phi$ if $(\mathcal{I}, r', m') \models \phi$ for all $(r', m') \in \mathcal{K}_i(r, m)$. In addition to $K_i\phi$, which means that i knows ϕ, we also use $P_i\phi$ as an abbreviation of $\neg K_i \neg\phi$, which means that i thinks that ϕ is possible. We also write $\mathcal{I} \models \phi$ if $(\mathcal{I}, r, m) \models \phi$ holds for every point (r, m) in $\mathcal{I}$.

In the rest of the paper, we consider that the set A of *actions* is also associated with each system. We assume that $i, i', j, j', \ldots$ range over agents while $a, a', b, b', \ldots$ range over actions. Following [12], we use a primitive proposition of the form $\theta(i, a)$, which denotes that "an agent i has performed an action a, or will perform a in the future." Note that the truth value of $\theta(i, a)$ depends on the run, but not on the time; that is, if $(\mathcal{I}, r, m) \models \theta(i, a)$ holds for some m, then $(\mathcal{I}, r, m') \models \theta(i, a)$ also holds for every m'.

Below we review the formal definitions of anonymity, privacy, onymity, and identity in terms of epistemic logic for multiagent systems. For full details, see [12, 19, 26].

Anonymity

We say that an action a performed by an agent i is *anonymous up to* an *anonymity set* $I_A \subseteq I$ with respect to an agent j in the interpreted system $\mathcal{I}$ if $\mathcal{I} \models \theta(i, a) \Rightarrow \bigwedge_{i' \in I_A} P_j[\theta(i', a)]$ holds. Intuitively, anonymity up to I_A means that, from j's viewpoint, a could have been performed by anybody in I_A. A typical example of anonymity of this form is *sender anonymity*, which is explained in Sect. 1.

We also say that an action a performed by an agent i is *minimally anonymous* with respect to an agent j in the interpreted system $\mathcal{I}$ if $\mathcal{I} \models \theta(i, a) \Rightarrow P_j[\neg\theta(i, a)]$ holds. Intuitively, minimal anonymity means that, from j's viewpoint, a could not have been performed by i. Consider that our built-in proposition $\theta(i, a)$ expresses a specific form of "link" between an agent i and an action a. Then, we can observe that minimal anonymity is very close to a specific form of the "unlinkability" property that was stipulated by Pfitzmann and Hansen [21]. This observation was elaborated in [26].

Privacy

Privacy properties can be obtained from anonymity properties by applying the operation of taking the agent/action reversal dual, that is, the operation that replaces a set of agents with a set of actions. For example, we say that an agent i performing an action a is *private up to* a *privacy set* $A_I \subseteq A$ with respect to an agent j in the interpreted system $\mathcal{I}$ if $\mathcal{I} \models \theta(i, a) \Rightarrow \bigwedge_{a' \in A_I} P_j[\theta(i, a')]$ holds. Intuitively, privacy up to A_I means that, from j's viewpoint, i could have performed any action in A_I. A typical example is *message privacy*, which is explained in Sect. 1.

We also say that an agent i performing an action a is *minimally private* with respect to an agent j in the interpreted system $\mathcal{I}$ if $\mathcal{I} \models \theta(i, a) \Rightarrow P_j[\neg\theta(i, a)]$ holds. Note that minimal privacy is equivalent to its dual, that is, minimal anonymity.

Role Interchangeability

Role interchangeability means that, as far as an agent j is concerned, two agents i and i' could interchange their roles, that is, the actions they performed. Specifically, a pair (i, a) comprising an agent i and an action a is *role interchangeable* with respect to an agent j in the interpreted system $\mathcal{I}$ if $\mathcal{I} \models \theta(i, a) \Rightarrow \bigwedge_{i' \in I/\{j\}} \bigwedge_{a' \in A}(\theta(i', a') \Rightarrow P_j[\theta(i', a) \wedge \theta(i, a')])$ holds. Despite the similarity between role interchangeability and anonymity/privacy, they are not equiexpressive. We can prove that role interchangeability implies both anonymity and privacy under some appropriate conditions [19].

Onymity

By the "contrary" of a formula of the form $\theta(i, a) \Rightarrow \Gamma$, we mean the formula $\theta(i, a) \Rightarrow \neg\Gamma$. By taking the contrary of the formulas defining anonymity, we can obtain definitions of onymity. We only show below the contrary of minimal anonymity. We say that an action a performed by an agent i is *maximally onymous* with respect to an agent j in the interpreted system $\mathcal{I}$ if $\mathcal{I} \models \theta(i, a) \Rightarrow K_j[\theta(i, a)]$ holds. Intuitively, maximal onymity means that j knows that i has performed a. This definition corresponds to our observation that onymity generally means that the agent who performs the action is disclosed. We can see that onymity is closely related to personal authentication.

Identity

Identity properties, which are closely related to attribute authentication, can be obtained as the contrary of privacy properties or as the dual of onymity properties. Below we only show the contrary of minimal privacy. We say that an agent i performing an action a is *maximally identified* with respect to an agent j in the interpreted system $\mathcal{I}$ if $\mathcal{I} \models \theta(i, a) \Rightarrow K_j[\theta(i, a)]$ holds. Note that maximal identity is equivalent to its dual, that is, maximal onymity.

The definitions of the properties presented above and their known relationships are summarized in Fig. 2. For example, role interchangeability implies anonymity up to I_A, which also implies minimal anonymity. Note that every implication

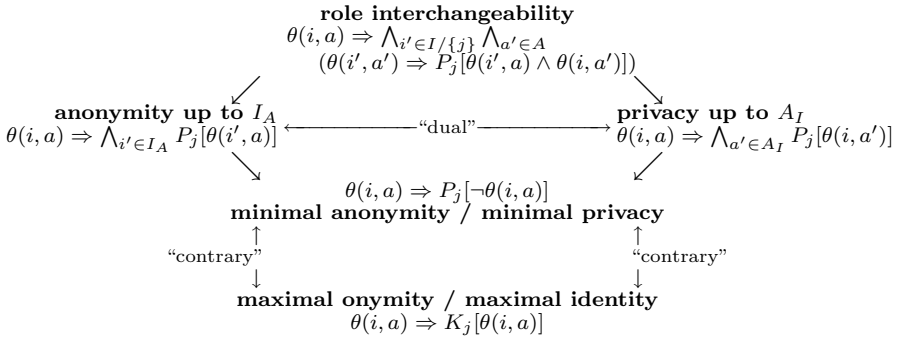


Figure 2: Formal definitions of some privacy-related information-hiding/disclosure properties.

described here is conditional. A more detailed version of this figure can be found in [26].

3. SEQUENTIAL COMPOSITIONALITY OF ANONYMITY AND PRIVACY

As a motivating example for discussion of sequential compositionality, consider an abstract model of an anonymous members-only bulletin board system (Fig. 3). Suppose that the set of agents includes two disjoint subsets I_R and I_P of *real names* and *pseudonyms*, respectively. Each real-name agent can register several pseudonyms to use; the correspondence between real names and pseudonyms is expressed by using $\theta(i, use(k))$, which means that a real i can use a pseudonym k. Besides I_R and I_P, we also introduce the domain C of possible articles. Each real-name agent uses some of its pseudonyms and posts some articles to a bulletin board. We express this as $\theta(k, post(c))$, which means that a pseudonym k posts an article c. When a real-name agent i uses a pseudonym k and k posts an article c, we say that i submits c. This is formulated as $\mathcal{I} \models \theta(i, submit(c)) \Leftrightarrow \bigvee_{k \in I_P} (\theta(i, use(k)) \wedge \theta(k, post(c)))$. Two sets $\{post(c) \mid c \in C\}$ and $\{submit(c) \mid c \in C\}$ of actions are denoted by A_P and A_S, respectively.

Although this is initially given as a model of an anonymous bulletin board system, it is quite abstract and can serve as a model for a more general class of systems, provided that it is appropriately modified. For example, if $\theta(i, use(k))$ is interpreted as meaning that a voter i is authorized to use a pseudonym k for voting and $\theta(k, post(c))$ is interpreted as meaning that k casts a ballot c for some candidate, then this will be regarded as a model of a voting system. (Of course, some appropriate assumptions will be required. For example, to guarantee eligibility, we must assume that each voter uses at most one pseudonym and each pseudonym also casts at most one ballot.) Furthermore, if $\theta(i, use(k))$ is interpreted as meaning that the first mix-server takes an incoming message i and produces an outgoing message k and if $\theta(k, post(c))$ is interpreted as meaning that the second mix-server takes an incoming message k and produces an outgoing message c, then this will be regarded as a model of a chain of two mix-servers.

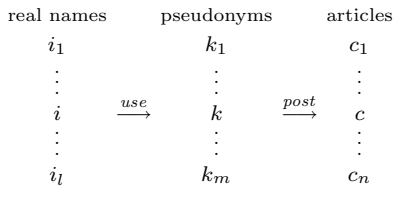


Figure 3: An anonymous members-only bulletin board system.

We shall consider several typical cases where different combinations of privacy-related properties are owned by each registration and posting phase (Table 1). Below we concentrate on some main specific cases (Cases 1 to 5). The other cases are discussed in Appendix A. Intuitively, when registration is anonymous and posting is private (Case 1), the entire system appears to have good anonymity/privacy properties. However, this conjecture is refuted. Indeed, assume that an observer has some presupposed background knowledge that a real-name agent i will never submit an improper article c. Then, even though the observer thinks that any real-name agents including i could have used a pseudonym k and that k could have posted any articles including c, the observer never thinks that i could have submitted c. More formally, the following holds.

CLAIM 3.1. *There is an interpreted system that satisfies the following: (1) every action use(k) performed by i is anonymous up to I_R with respect to an observer j; (2) every agent k performing post(c) is private up to A_P with respect to j; (3) some action submit(c) performed by i is not anonymous up to I_R; (4) some agent i performing submit(c) is not private up to A_S.*

PROOF. Suppose that $I_R = \{i_1, i_2\}$, $I_P = \{k_1, k_2\}$, $A_P = \{post(c_1), post(c_2)\}$, and $A_S = \{submit(c_1), submit(c_2)\}$. Consider an interpreted system consisting of two runs r_1 and r_2. In r_1, the following are true: $\theta(i_1, use(k_1))$, $\theta(k_1, post(c_1))$, $\theta(i_2, use(k_2))$, and $\theta(k_2, post(c_2))$. In r_2, the following are true: $\theta(i_1, use(k_2))$, $\theta(k_2, post(c_1))$, $\theta(i_2, use(k_1))$, and $\theta(k_1, post(c_2))$. We also assume that the two runs are indistinguishable from the observer j's viewpoint, that is, more precisely, $(r_1, m) \sim_j (r_2, m)$ holds for each m. Then, it is immediately seen that (1) and (2) hold. Furthermore, (3) and (4) also hold because $\theta(i_1, submit(c_2))$ is neither true in r_1 nor true in r_2 and because $\theta(i_2, submit(c_1))$ is neither true in r_1 nor true in r_2. In other words, the observer can have "presupposed background knowledge" that i_1 never submits c_2, and i_2 never submits c_1. □

Remark 1. The observations above, in particular, the construction of $\{r_1, r_2\}$ shown in the proof of Claim 3.1, can

Table 1: Sequential Compositionality: Twelve Cases

		Assumption	Registration	Posting	Total
Case 1	(Claim 3.1)	—	Anonymous up to I_R	Private up to A_P	—
Case 2	(Claim 3.2)	Independent	—	Private up to A_P	Private up to A_S
Case 3	(Claim 3.3)	Independent	Anonymous up to I_R	—	Anonymous up to I_R
Case 4	(Claim 3.4)	—	Maximally onymous	Private up to A_P	Private up to A_S
Case 5	(Claim 3.5)	—	Anonymous up to I_R	Maximally identified	Anonymous up to I_R
Case 6	(Claim A.1)	Pairwise independent	—	Role interchangeable	Role interchangeable
Case 7	(Claim A.2)	Pairwise independent	Role interchangeable	—	Role interchangeable
Case 8	(Claim A.3)	Independent & Exhaustive posting & Exclusive i and $post(c)$	—	Minimally private	Minimally private
Case 9	(Claim A.4)	Independent & Exhaustive registration & Exclusive i and $post(c)$	Minimally anonymous	—	Minimally anonymous
Case 10	(Claim A.5)	Exhaustive posting & Exclusive i and $post(c)$	Maximally onymous	Minimally private	Minimally private
Case 11	(Claim A.6)	Exhaustive registration & Exclusive i and $post(c)$	Minimally anonymous	Maximally identified	Minimally anonymous
Case 12	(Claim A.7)	—	Maximally onymous	Maximally identified	Maximally onymous/identified

be extended to consider other examples where anonymity/privacy properties are not sequentially compositional. For example, we can say that a chain $M_1 * M_2$ of two mix-servers does not necessarily guarantee unlinkability between incoming and outgoing messages even though M_1 and M_2 do individually. Indeed, if M_2 is the "inverse" ${M_1}^{-1}$ of M_1, then $M_1 * {M_1}^{-1}$ becomes an identity and thus provides obvious linkability, even though both M_1 and ${M_1}^{-1}$ guarantee unlinkability.

On the basis of the above discussion, we introduce "independence" assumptions so that anonymity/privacy in the entire system can be obtained quite directly from anonymity/privacy in the registration/posting phases. The registration and posting phases in an anonymous bulletin board system $\mathcal{I}$ are *independent* with respect to an observer j if

$$\mathcal{I} \models P_j[\theta(i, use(k))] \wedge P_j[\theta(k', post(c))] \Rightarrow P_j[\theta(i, use(k)) \wedge \theta(k', post(c))]$$

holds for every i, k, k', and c. This is analogous to the independence of two events in probability theory: two events A and B are independent if $\Pr(A)\Pr(B) = \Pr(A \cap B)$. The independence assumption can be regarded as meaning that the observer has no specific "presupposed background knowledge."

Example 1. In the system $\{r_1, r_2\}$ shown in the proof of Claim 3.1, the registration and posting phases are not independent. To guarantee independence, we can extend the system so that it has four indistinguishable runs $\{r_1, r_2, r_3, r_4\}$ (Fig. 4). In r_3, the following are true: $\theta(i_1, use(k_1))$, $\theta(k_1, post(c_2))$, $\theta(i_2, use(k_2))$, and $\theta(k_2, post(c_1))$. In r_4, the following are true: $\theta(i_1, use(k_2))$, $\theta(k_2, post(c_2))$, $\theta(i_2, use(k_1))$, and $\theta(k_1, post(c_1))$. Alternatively, we can also obtain a system $\{r_1, r_2, r_5, r_6, r_7, r_8\}$ of indistinguishable runs that has the independence property. Similarly, a system $\{r_1, r_2, r_9, r_{10}, r_{11}, r_{12}\}$ of indistinguishable runs also has the independence property.

We also discuss, in Appendix C, that independence could be viewed by itself as a "meta-level" abstraction of anonymity or privacy.

The following two lemmas are "dual" and show some obvious sufficient conditions for independence. Hereafter, the proofs of the "dual" of proved lemmas or claims are omitted, since they can be straightforwardly obtained from the original proofs via duality.

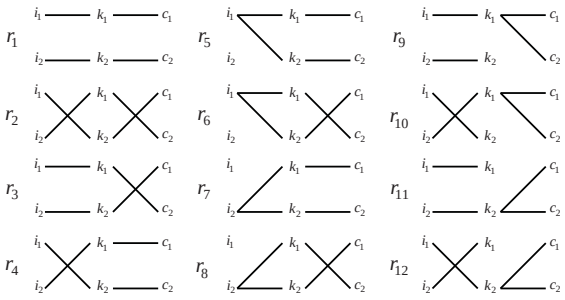


Figure 4: Systems $\{r_1, r_2, r_3, r_4\}$, $\{r_1, r_2, r_5, r_6, r_7, r_8\}$, and $\{r_1, r_2, r_9, r_{10}, r_{11}, r_{12}\}$ of runs satisfy the independence property.

LEMMA 3.1. *If every action $use(k)$ performed by i is maximally onymous with respect to an observer j, the registration and posting phases are independent with respect to j.*

PROOF. Suppose that $(\mathcal{I}, r, m) \models P_j[\theta(i, use(k))] \wedge P_j[\theta(k', post(c))]$. Then, $\theta(i, use(k))$ holds at some point (r', m') such that $(r', m') \sim_j (r, m)$, and $\theta(k', post(c))$ also holds at some point (r'', m'') such that $(r'', m'') \sim_j (r, m)$. Since $use(k)$ performed by i is maximally onymous and $\theta(i, use(k))$ holds at (r', m'), $\theta(i, use(k))$ also holds at (r'', m''). In other words, $(\mathcal{I}, r'', m'') \models \theta(i, use(k)) \wedge \theta(k', post(c))$ holds. Thus, we have proved that $(\mathcal{I}, r, m) \models P_j[\theta(i, use(k)) \wedge \theta(k', post(c))]$. □

LEMMA 3.2. *If every agent k performing $post(c)$ is maximally identified with respect to an observer j, the registration and posting phases are independent with respect to j.*

Case 2 in Table 1 indicates that if the posting phase guarantees privacy, then so does the entire system, provided that the posting and registration phases are independent.

CLAIM 3.2. *Assume that the registration and posting phases are independent with respect to an observer j. Also*

suppose that every agent k performing $post(c)$ is private up to A_P with respect to j. Then, every agent i performing $submit(c)$ is private up to A_S.

PROOF. Suppose that $(\mathcal{I}, r, m) \models \theta(i, submit(c))$. Then, there exists some k in I_P such that $(\mathcal{I}, r, m) \models \theta(i, use(k)) \wedge \theta(k, post(c))$. From $(\mathcal{I}, r, m) \models \theta(i, use(k))$, it is immediate to see that $(\mathcal{I}, r, m) \models P_j[\theta(i, use(k))]$. Because every k performing $post(c)$ is private up to A_P and because $(\mathcal{I}, r, m) \models \theta(k, post(c))$, we can say that for every possible article c', $(\mathcal{I}, r, m) \models P_j[\theta(k, post(c'))]$ holds. So, by virtue of the independence assumption, $(\mathcal{I}, r, m) \models P_j[\theta(i, use(k)) \wedge \theta(k, post(c'))]$ holds. That is, $(\mathcal{I}, r, m) \models P_j[\theta(i, submit(c'))]$ holds. Since c' is arbitrary, we have proved that $(\mathcal{I}, r, m) \models \bigwedge_{a' \in A_S} P_j[\theta(i, a')]$. □

Case 3 in Table 1 is a "dual" of Case 2. It means that if the registration phase guarantees anonymity, then so does the entire system, provided that the posting and registration phases are independent.

CLAIM 3.3. *Assume that the registration and posting phases are independent with respect to an observer j. Also suppose that every action $use(k)$ performed by i is anonymous up to I_R with respect to j. Then, every action $submit(c)$ performed by i is anonymous up to I_R.*

In the view of Lemma 3.1, Case 4 can be regarded as a special case of Case 2. More specifically, the following claim directly follows from Lemma 3.1 and Claim 3.2. It indicates that if the posting phase guarantees privacy, then so does the entire system, even though each registered pseudonym is linked to the corresponding real name.

CLAIM 3.4. *Suppose that every action $use(k)$ performed by i is maximally onymous with respect to an observer j. Also suppose that every agent k performing $post(c)$ is private up to A_P with respect to j. Then, every agent i performing $submit(c)$ is private up to A_S.*

Case 5 is a "dual" of Case 4. It can also be regarded, in the view of Lemma 3.2, as a special case of Case 3. It means that if the registration phase guarantees anonymity, then so does the entire system, even though each article is linked to the pseudonym who posted it.

CLAIM 3.5. *Suppose that every action $use(k)$ performed by i is anonymous up to I_R with respect to an observer j. Also suppose that every agent k performing $post(c)$ is maximally identified with respect to j. Then, every action $submit(c)$ performed by i is anonymous up to I_R.*

4. PARALLEL COMPOSITIONALITY OF ANONYMITY AND PRIVACY

By the *parallel composition* of $act_a(c)$ performed by i and $act_b(c)$ performed by i, we generally mean the action $act_p(c)$ performed by i that is introduced by $\theta(i, act_p(c)) \Leftrightarrow \theta(i, act_a(c)) \wedge \theta(i, act_b(c))$. We denote three sets $\{act_a(c) \mid c\}$, $\{act_b(c) \mid c\}$, and $\{act_p(c) \mid c\}$ of actions by A_a, A_b, and A_p, respectively.

Example 2. Consider the following situation. A special prosecution team has pursued their probe into the hideout of a radical and has found out a time bomb c that seems to have been provided by a sympathizer i. The urgent mission of the team is to determine i performing an action $give(c)$. The essential parts of the bomb c are a timer and gunpowder. The sympathizer seems to have bought the timer and have synthesized the gunpowder, thereby producing the time bomb. Thus, the following definition is obtained: $\theta(i, give(c)) \Leftrightarrow \theta(i, buy\_timer(c)) \wedge \theta(i, synthesize\_gunpowder(c))$. A concern here is how some (an)onymity property of $give(c)$ can be deduced from the (an)onymity properties of $buy\_timer(c)$ and $synthesize\_gunpowder(c)$.

Table 2 shows some cases where different combinations of privacy-related properties are owned by act_a and act_b. As for the case of sequential composition, the parallel compositionality of anonymity or privacy does not generally hold without some appropriate forms of independence assumptions. We say that act_a and act_b are *independent* with respect to an observer j in a system $\mathcal{I}$ if $\mathcal{I} \models P_j[\theta(i, act_a(c))] \wedge P_j[\theta(i, act_b(c))] \Rightarrow P_j[\theta(i, act_a(c)) \wedge \theta(i, act_b(c))]$ holds for every i and c. Roughly speaking, the independence means that act_a and act_b are not exclusive. Below we show that the independence assumption plays an essential role in Case I and its dual, Case II. The other cases are discussed in Appendix B.

CLAIM 4.1. *Assume that act_a and act_b are independent with respect to an observer j. Also suppose that i performing $act_a(c)$ is private up to A_a with respect to j and i performing $act_b(c)$ is private up to A_b with respect to j. Then, i performing $act_p(c)$ is private up to A_p with respect to j.*

PROOF. Suppose that $(\mathcal{I}, r, m) \models \theta(i, act_p(c))$. Then, $(\mathcal{I}, r, m) \models \theta(i, act_a(c)) \wedge \theta(i, act_b(c))$ holds. By the assumption of privacy, we have $(\mathcal{I}, r, m) \models P_j[\theta(i, act_a(c'))] \wedge P_j[\theta(i, act_b(c'))]$ for every c'. By the independence assumption, $(\mathcal{I}, r, m) \models P_j[\theta(i, act_a(c')) \wedge \theta(i, act_b(c'))]$, that is, $(\mathcal{I}, r, m) \models P_j[\theta(i, act_p(c'))]$ holds. Since c' is arbitrary, we have proved the claim. □

CLAIM 4.2. *Assume that act_a and act_b are independent with respect to an observer j. Also suppose that $act_a(c)$ performed by i and $act_b(c)$ performed by i are anonymous up to I_a and I_b, respectively. Then, $act_p(c)$ performed by i is anonymous up to $I_a \cap I_b$ with respect to j.*

Example 3. Consider the situation described in Example 2. Claim 4.2 indicates that $give(c)$ can be onymous even though both $buy\_timer(c)$ and $synthesize\_gunpowder(c)$ are anonymous. This can happen when buy_timer and $synthesize\_gunpowder$ are not independent, that is, when some suspect is considered to be unable to perform both actions for some reason.

5. CONCLUSION

Building on an epistemic-logic formalism, we have discussed the compositionality of several privacy-related information-hiding/disclosure properties. We have pointed out that anonymity and privacy are not necessarily sequentially compositional and have indicated that the independence assumptions can guarantee the compositionality. We have also developed a series of theoretical case studies on the conditions that are sufficient to guarantee the sequential compositionality of various degrees of anonymity, privacy, onymity, and/or identity. Similar compositionality results have also been shown for parallel composition.

Table 2: Parallel Compositionality: Five Cases

		Assumption	act_a	act_b	act_p (Total)
Case I	(Claim 4.1)	Independent	Private up to A_a	Private up to A_b	Private up to A_p
Case II	(Claim 4.2)	Independent	Anonymous up to I_a	Anonymous up to I_b	Anonymous up to $I_a \cap I_b$
Case III	(Claim B.1)	—	—	Minimally anonymous/private	Minimally anonymous/private
Case IV	(Claim B.1)	—	Minimally anonymous/private	—	Minimally anonymous/private
Case V	(Claim B.2)	—	Maximally onymous/identified	Maximally onymous/identified	Maximally onymous/identified

Future work will include a discussion of compositionality in terms of the probabilistic extension [12] of epistemic logic. To substantiate the practical value of our approach, a detailed analysis of real world examples should be carried out.

6. REFERENCES

[1] A. Baskar, R. Ramanujam, and S. P. Suresh. Knowledge-based modelling of voting protocols. In *Proc. TARK'07*, pages 62–71, 2007.

[2] E. Bertino and K. Takahashi. *Identity Management: Concepts, Technologies, and Systems.* Artech House, 2011.

[3] I. Boureanu, A. V. Jones, and A. Lomuscio. Automatic verification of epistemic specifications under convergent equational theories. In *Proc. AAMAS'12*, pages 1141–1148, 2012.

[4] R. Chadha, S. Delaune, and S. Kremer. Epistemic logic for the applied pi calculus. In *Proc. FMOODS/FORTE'09*, Springer LNCS, Vol. 5522, pages 182–197, 2009.

[5] D. Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Commun. ACM*, 24(2):84–88, 1981.

[6] S. Delaune, S. Kremer, and M. Ryan. Verifying privacy-type properties of electronic voting protocols. *J. Comput. Security*, 17(4):435–487, 2009.

[7] C. Díaz, S. Seys, J. Claessens, and B. Preneel. Towards measuring anonymity. In *Proc. PET'02*, Springer LNCS, Vol. 2482, pages 54–68, 2002.

[8] M. Edman, F. Sivrikaya, and B. Yener. A combinatorial approach to measuring anonymity. In *Proc. IEEE ISI'07*, pages 356–363, 2007.

[9] R. Fagin, J. Y. Halpern, Y. Moses, and M. Y. Vardi. *Reasoning About Knowledge*. The MIT Press, 1995.

[10] F. D. Garcia, I. Hasuo, W. Pieters, and P. van Rossum. Provable anonymity. In *Proc. ACM FMSE'05*, pages 63–72, 2005.

[11] I. Goriac. An epistemic logic based framework for reasoning about information hiding. In *Proc. ARES'11*, pages 286–293, 2011.

[12] J. Y. Halpern and K. R. O'Neill. Anonymity and information hiding in multiagent systems. *J. Comput. Security*, 13(3):483–512, 2005.

[13] I. Hasuo, Y. Kawabe, and H. Sakurada. Probabilistic anonymity via coalgebraic simulations. *Theoret. Comput. Sci.*, 411(22-24):2239–2259, 2010.

[14] D. Hughes and V. Shmatikov. Information hiding, anonymity and privacy: a modular approach. *J. Comput. Security*, 12(1):3–36, 2004.

[15] H. Jonker and W. Pieters. Receipt-freeness as a special case of anonymity in epistemic logic. In *WOTE'06*, 2006.

[16] Y. Kawabe, K. Mano, H. Sakurada, and Y. Tsukada. Theorem-proving anonymity of infinite-state systems. *Inform. Process. Lett.*, 101(1):46–51, 2007.

[17] D. Kelly. *A taxonomy for and analysis of anonymous communications networks*. Ph. D. Thesis, Air Force Institute of Technology, 2009.

[18] R. Küsters and T. Truderung. An epistemic approach to coercion-resistance for electronic voting protocols. In *Proc. IEEE S&P'09*, pages 251–266, 2009.

[19] K. Mano, Y. Kawabe, H. Sakurada, and Y. Tsukada. Role interchange for anonymity and privacy of voting. *J. Logic and Comput.*, 20(6):1251–1288, 2010.

[20] S. Mauw, J. Verschuren, and E. P. de Vink. Data anonymity in the FOO voting scheme. In *Proc. VODCA'06*, ENTCS, Vol. 168, pages 5–28, 2007.

[21] A. Pfitzmann and M. Hansen. A terminology for talking about privacy by data minimization: Anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management (Ver. v0.34), 2010.

[22] S. Schneider and A. Sidiropoulos. CSP and anonymity. In *Proc. ESORICS'96*, Springer LNCS, Vol. 1146, pages 198–218, 1996.

[23] A. Serjantov and G. Danezis. Towards an information theoretic metric for anonymity. In *Proc. PET'02*, Springer LNCS, Vol. 2482, pages 41–53, 2002.

[24] P. F. Syverson and S. G. Stubblebine. Group principals and the formalization of anonymity. In *Proc. FM'99*, Springer LNCS, Vol. 1708, pages 814–833, 1999.

[25] C. Troncoso, B. Gierlichs, B. Preneel, and I. Verbauwhede. Perfect matching disclosure attacks. In *Proc. PET'08*, Springer LNCS, Vol. 5134, pages 2–23, 2008.

[26] Y. Tsukada, K. Mano, H. Sakurada, and Y. Kawabe. Anonymity, privacy, onymity, and identity: A modal logic approach. *Transactions on Data Privacy*, 3(3):177–198, 2010.

[27] R. van der Meyden and K. Su. Symbolic model checking the knowledge of the dining cryptographers. In *Proc. 17th IEEE CSFW*, pages 280–291, 2004.

[28] R. van der Meyden and T. Wilke. Preservation of epistemic properties in security protocol implementations. In *Proc. TARK'07*, pages 212–221, 2007.

[29] J. van Eijck and S. Orzan. Epistemic verification of anonymity. In *Proc. VODCA'06*, ENTCS, Vol. 168, pages 159–174, 2007.

[30] M. Veeningen, B. de Weger, and N. Zannone. Modeling identity-related properties and their privacy strength. In *Proc. FAST'10*, Springer LNCS, Vol. 6561, pages 126–140, 2011.

APPENDIX

A. SEQUENTIAL COMPOSITIONALITY: MORE CASES

In this appendix, we discuss Cases 6 to 12 shown in Table 1.

We first introduce some additional conditions regarding our motivating example of an anonymous members-only bulletin board system. We say that an action $post(c)$ is *exclusive* if $post(c)$ is performed by at most one pseudonym in each run, that is, $\mathcal{I} \models \bigwedge_{k \neq k'} \neg[\theta(k, post(c)) \wedge \theta(k', post(c))]$ holds. For example, if we consider that each article c is labeled and identified with an article ID number, we will accordingly assume that each $post(c)$ is exclusive. We also say that an action $use(k)$ is *exclusive* if $use(k)$ is performed by at most one real-name agent in each run. For example, if we want to avoid the use of bogus pseudonyms, we will assume that each $use(k)$ is exclusive. Similarly, we say that a real-name agent i is *exclusive* if i performs at most one $use(k)$ action in each run, that is, $\mathcal{I} \models \bigwedge_{k \neq k'} \neg[\theta(i, use(k)) \wedge \theta(i, use(k'))]$ holds. We also say that a pseudonym k is *exclusive* if k performs at most one $post(c)$ action in each run.

We also say that the posting phase is *exhaustive* provided that every article $c \in C$ has been posted by some pseudonyms. This is formulated as $\mathcal{I} \models \bigwedge_{c \in C} \bigvee_{k \in I_P} \theta(k, post(c))$. Similarly, we say that the registration phase is *exhaustive* provided that every real-name agent $i \in I_R$ uses some pseudonyms. This is formulated as $\mathcal{I} \models \bigwedge_{i \in I_R} \bigvee_{k \in I_P} \theta(i, use(k))$.

We also extend the independence assumption so as to deal with Cases 6 to 11. First, the independence assumption can immediately be extended to a disjunctive form.

LEMMA A.1. *If the registration and posting phases in $\mathcal{I}$ are independent with respect to an observer j, then the following holds for arbitrary i_p, k_p, k'_q, and c_q:*

$$\mathcal{I} \models P_j[\textstyle\bigvee_p \theta(i_p, use(k_p))] \wedge P_j[\bigvee_q \theta(k'_q, post(c_q))] \Rightarrow P_j[(\textstyle\bigvee_p \theta(i_p, use(k_p))) \wedge (\bigvee_q \theta(k'_q, post(c_q)))].$$

PROOF. Suppose that $(\mathcal{I}, r, m) \models P_j[\vee_p \theta(i_p, use(k_p))]$ and $(\mathcal{I}, r, m) \models P_j[\vee_q \theta(k'_q, post(c_q))]$. This means that there exist some point (r', m') and p such that $\theta(i_p, use(k_p))$ holds at (r', m') and $(r', m') \sim_j (r, m)$. Further, there exist some point (r'', m'') and q such that $\theta(k'_q, post(c_q))$ holds at (r'', m'') and $(r'', m'') \sim_j (r, m)$. Then, by the independence assumption, there exists some point (r''', m''') such that $\theta(i_p, use(k_p)) \wedge \theta(k'_q, post(c_q))$ holds at (r''', m''') and $(r''', m''') \sim_j (r, m)$. This concludes the proof. □

Further, the independence assumption can be extended to "positive-negative" and "negative-positive" forms.

LEMMA A.2. *Assume that the registration and posting phases in $\mathcal{I}$ are independent with respect to an observer j. Also assume that the posting phase is exhaustive and that every posting action $post(c)$ is exclusive. Then, $\mathcal{I} \models P_j[\theta(i, use(k))] \wedge P_j[\neg\theta(k', post(c))] \Rightarrow P_j[\theta(i, use(k)) \wedge \neg\theta(k', post(c))]$ holds for every i, k, k', and c.*

PROOF. Since the posting phase is exhaustive, every c must have been posted by some pseudonyms in each run. Further, since $post(c)$ is exclusive, a uniquely determined pseudonym must have posted it in each run. In other words, $\neg\theta(k', post(c))$ can be equivalently expressed as a formula of the form $\vee_{k'_q \neq k'} \theta(k'_q, post(c))$. Hence, the lemma immediately follows from Lemma A.1. □

LEMMA A.3. *Assume that the registration and posting phases in $\mathcal{I}$ are independent with respect to an observer j. Also assume that the registration phase is exhaustive and that every real-name agent i is exclusive. Then, $\mathcal{I} \models P_j[\neg\theta(i, use(k))] \wedge P_j[\theta(k', post(c))] \Rightarrow P_j[\neg\theta(i, use(k)) \wedge \theta(k', post(c))]$ holds for every i, k, k', and c.*

In some cases, we require a stronger form of the independence assumption to prove compositionality results. Indeed, we need the binarily conjunctive form of the assumption. More specifically, the registration and posting phases in an anonymous bulletin board system $\mathcal{I}$ are *pairwise independent* with respect to an observer j if

$$\mathcal{I} \models P_j[\bigwedge_{m \in \{0,1\}} \theta(i_m, use(k_m))] \wedge P_j[\bigwedge_{n \in \{0,1\}} \theta(k'_n, post(c_n))] \Rightarrow P_j[(\bigwedge_{m \in \{0,1\}} \theta(i_m, use(k_m))) \wedge (\bigwedge_{n \in \{0,1\}} \theta(k'_n, post(c_n)))]$$

holds for every pair (i_0, i_1), (k_0, k_1), (k'_0, k'_1), and (c_0, c_1).

Example 4. In the system $\{r_1, r_2, r_3, r_4\}$ (Fig. 4), the registration and posting phases are pairwise independent. On the other hand, in the system $\{r_1, r_2, r_5, r_6, r_7, r_8\}$ or $\{r_1, r_2, r_9, r_{10}, r_{11}, r_{12}\}$, the registration and posting phases are not pairwise independent.

Cases 2 and 3 can be extended to show the sequential compositionality of role interchangeability. To obtain these results, we require the pairwise independence assumption.

CLAIM A.1. *Assume that the registration and posting phases are pairwise independent with respect to an observer j. Also suppose that every pair comprising an agent k and an action $post(c)$ is role interchangeable with respect to j. Then, every pair comprising an agent i and an action $submit(c)$ is role interchangeable as well.*

PROOF. Suppose that $(\mathcal{I}, r, m) \models \theta(i, submit(c))$ and $(\mathcal{I}, r, m) \models \theta(i', submit(c'))$. Then, there exist k and k' such that $(\mathcal{I}, r, m) \models \theta(i, use(k)) \wedge \theta(k, post(c))$ and $(\mathcal{I}, r, m) \models \theta(i', use(k')) \wedge \theta(k', post(c'))$. Because every pair comprising an agent k and an action $post(c)$ is role interchangeable and because $(\mathcal{I}, r, m) \models \theta(k, post(c)) \wedge \theta(k', post(c'))$, we can say that $(\mathcal{I}, r, m) \models P_j[\theta(k', post(c)) \wedge \theta(k, post(c'))]$ holds. On the other hand, we have $(\mathcal{I}, r, m) \models \theta(i', use(k')) \wedge \theta(i, use(k))$. That is, $(\mathcal{I}, r, m) \models P_j[\theta(i', use(k')) \wedge \theta(i, use(k))]$ holds. So, by virtue of the pairwise independence assumption, $(\mathcal{I}, r, m) \models P_j[\theta(i', use(k')) \wedge \theta(k', post(c)) \wedge \theta(i, use(k)) \wedge \theta(k, post(c'))]$ holds. That is, $(\mathcal{I}, r, m) \models P_j[\theta(i', submit(c)) \wedge \theta(i, submit(c'))]$. This concludes the proof. □

CLAIM A.2. *Assume that the registration and posting phases are pairwise independent with respect to an observer j. Also suppose that every pair comprising an agent i and an action $use(k)$ is role interchangeable with respect to j. Then, every pair comprising an agent i and an action $submit(c)$ is role interchangeable as well.*

Example 5. In the system $\{r_1, r_2, r_5, r_6, r_7, r_8\}$ or $\{r_1, r_2, r_9, r_{10}, r_{11}, r_{12}\}$ (Fig. 4), every pair comprising an

agent k and an action $post(c)$ is role interchangeable as well as every pair comprising an agent i and an action $use(k)$. However, the registration and posting phases are not pairwise independent. Consequently, in these systems, there exist some pairs comprising an agent i and an action $submit(c)$ such that they are not role interchangeable.

Cases 8, 9, 10, and 11 in Table 1 are respectively derived from Cases 2, 3, 4, and 5 by replacing "up-to" anonymity/privacy properties with minimal anonymity/privacy properties. There are two problems in obtaining these derivations. First, consider Case 8 and its dual, Case 9, which are derived from Cases 2 and 3, respectively. Since the definition of minimal privacy/anonymity involves negative formulas, independence assumptions in positive-negative and negative-positive forms are helpful in these cases. Thus, we will use Lemmas A.2 and A.3 in Cases 8 and 9, respectively.

Second, consider Case 10 (which is derived from Case 4) and an intended example system consisting of the two indistinguishable runs r_5 and r_6 (Fig. 4). In r_5, i_1 uses k_1 and k_2 to post c_1 and c_2, respectively. In r_6, i_1 uses k_1 and k_2 to post c_2 and c_1, respectively. Thus, in the system $\{r_5, r_6\}$, every $use(k)$ performed by i is maximally onymous and every k performing $post(c)$ is minimally private, but i performing $submit(c)$ is never minimally private. This is because although the posting actions performed by the pseudonyms k_1 and k_2 of i_1 are totally different, the submission actions performed by i_1 are defined using existential quantification over k and thus both $\theta(i_1, submit(c_1))$ and $\theta(i_1, submit(c_2))$ hold in both r_5 and r_6. To avoid this, we assume that every real-name agent can be allowed to use at most one pseudonym in each run, that is, each i is exclusive. This assumption will also be used in a generalization of Case 10, that is, Case 8. Note that to deal with Cases 9 and 11, we need a similar assumption that every possible article c can be posted by at most one pseudonym k in each run, that is, every $post(c)$ is exclusive, which is the "dual" of the assumption above.

Claim A.3. *Assume that the registration and posting phases are independent with respect to j. Suppose that the posting phase is exhaustive and that each $post(c)$ is exclusive as well as each i. Also suppose that every agent k performing $post(c)$ is minimally private with respect to j. Then, every agent i performing $submit(c)$ is minimally private.*

Proof. Suppose that $(\mathcal{I}, r, m) \models \theta(i, submit(c))$. Then, there exists some k in I_P such that $(\mathcal{I}, r, m) \models \theta(i, use(k)) \wedge \theta(k, post(c))$. From $(\mathcal{I}, r, m) \models \theta(i, use(k))$, it is immediately seen that $(\mathcal{I}, r, m) \models P_j[\theta(i, use(k))]$. Because every k performing $post(c)$ is minimally private and because $(\mathcal{I}, r, m) \models \theta(k, post(c))$, we can say that $(\mathcal{I}, r, m) \models P_j[\neg\theta(k, post(c))]$ holds. So, by virtue of Lemma A.2, $(\mathcal{I}, r, m) \models P_j[\theta(i, use(k)) \wedge \neg\theta(k, post(c))]$ holds. Since every real-name agent can be allowed to use at most one pseudonym in each run, this means that $(\mathcal{I}, r, m) \models P_j[\neg\theta(i, submit(c))]$ holds. □

Claim A.4. *Assume that the registration and posting phases are independent with respect to j. Suppose that the registration phase is exhaustive and that each i is exclusive as well as each $post(c)$. Also suppose that every action $use(k)$ performed by i is minimally anonymous with respect to j. Then, every action $submit(c)$ performed by i is minimally anonymous.*

Claim A.5. *Suppose that the posting phase is exhaustive and that each i is exclusive as well as each $post(c)$. Also suppose that every action $use(k)$ performed by i is maximally onymous with respect to j. Moreover assume that every agent k performing $post(c)$ is minimally private with respect to j. Then, every agent i performing $submit(c)$ is minimally private.*

Proof. This directly follows from Lemma 3.1 and Claim A.3. □

Claim A.6. *Suppose that the registration phase is exhaustive and that each $post(c)$ is exclusive as well as each i. Also suppose that every action $use(k)$ performed by i is minimally anonymous with respect to j. In addition assume that every agent k performing $post(c)$ is maximally identified with respect to j. Then, every action $submit(c)$ performed by i is minimally anonymous.*

The final case shown in Table 1 indicates that if both the registration and posting phases guarantee linkability, then so does the entire system.

Claim A.7. *Suppose that every action $use(k)$ performed by i is maximally onymous with respect to j and that every agent k performing $post(c)$ is maximally identified with respect to j. Then, every action $submit(c)$ performed by i is maximally onymous.*

Proof. Suppose that $(\mathcal{I}, r, m) \models \theta(i, submit(c))$. Then, there exists some k in I_P such that $(\mathcal{I}, r, m) \models \theta(i, use(k)) \wedge \theta(k, post(c))$. Because every action $use(k)$ performed by i is maximally onymous and because every agent k performing $post(c)$ is maximally identified, $(\mathcal{I}, r', m') \models \theta(i, use(k)) \wedge \theta(k, post(c))$ holds for every point (r', m') such that $(r', m') \sim_j (r, m)$. This means that $(\mathcal{I}, r, m) \models K_j[\theta(i, submit(c))]$. □

B. PARALLEL COMPOSITIONALITY: MORE CASES

In this appendix, we discuss Cases III to V shown in Table 1.

Cases III and IV are perfectly symmetric and deal with the parallel compositionality of minimal anonymity/privacy. Note that the independence assumption is unnecessary here.

Claim B.1. *Suppose that either i performing $act_a(c)$ or i performing $act_b(c)$ is minimally private with respect to j. Then, i performing $act_p(c)$ is also minimally private.*

Proof. Suppose that $(\mathcal{I}, r, m) \models \theta(i, act_p(c))$. Then, $(\mathcal{I}, r, m) \models \theta(i, act_a(c)) \wedge \theta(i, act_b(c))$ holds. Also assume that, say, i performing $act_a(c)$ is minimally private. Then, based on the assumption of minimal privacy, $(\mathcal{I}, r, m) \models P_j[\neg\theta(i, act_a(c))]$ holds. This immediately implies that $(\mathcal{I}, r, m) \models P_j[\neg\theta(i, act_a(c)) \vee \neg\theta(i, act_b(c))]$ holds. That is, $(\mathcal{I}, r, m) \models P_j[\neg\theta(i, act_p(c))]$ holds. □

Case V in Table 2 indicates a trivial result on the parallel compositionality of linkability.

Claim B.2. *Suppose that both i performing $act_a(c)$ and i performing $act_b(c)$ are maximally identified with respect to j. Then, i performing $act_p(c)$ is also maximally identified.*

C. INDEPENDENCE-AS-ANONYMITY/PRIVACY INTERPRETATION

In this appendix, we discuss that the independence assumption shown in Sect. 3 could be viewed by itself as a "meta-level" abstraction of the anonymity or privacy property.

We first introduce two additional conditions regarding our anonymous members-only bulletin board system. We say that the bulletin board system satisfies *backward causality* provided that if k posts c, then there exists some i such that i uses k. This is formulated as $\mathcal{I} \models \theta(k, post(c)) \Rightarrow \bigvee_{i \in I_R} \theta(i, use(k))$. Backward causality can be regarded as a natural assumption in that every posted article should be related by some real-name agent; however, it is not a mandatory assumption because in some cases, certain auxiliary pseudonyms may post some dummy articles to enhance the privacy of real-name agents. We may also assume *forward causality*, which means that if i uses k, then there exists some c such that k posts c.

It is immediately seen that the definition of independence is equivalent to stating that $\mathcal{I} \models \theta(k', post(c)) \Rightarrow \bigwedge_{i,k}(P_j[\theta(i, use(k))] \Rightarrow P_j[\theta(i, use(k)) \wedge \theta(k', post(c))])$ holds for every k' and c. If we assume backward causality, then this is also equivalent to that for every i', k', and c,

$$\mathcal{I} \models \theta(i', use(k')) \wedge \theta(k', post(c)) \Rightarrow \bigwedge_{i,k}(P_j[\theta(i, use(k))] \Rightarrow P_j[\theta(i, use(k)) \wedge \theta(k', post(c))])$$

holds. If we abuse the notation and write $\Theta(\theta(i, use(k)), coexist(\theta(k', post(c))))$ for $\theta(i, use(k)) \wedge \theta(k', post(c))$, which means a "meta-level" link between "first-class" links $\theta(i, use(k))$ and $\theta(k', post(c))$, then the above equivalent transformation indicates that the independence assumption can be viewed as a certain, abstract form of "anonymity." More specifically, the obtained, equivalent formula means that an "action" $coexist(\theta(k', post(c)))$ performed by an "agent" $\theta(i', use(k'))$ is anonymous up to a certain "anonymity set" with respect to j. Alternatively, if we assume forward causality, the independence assumption can be viewed as an abstract form of "privacy." When we apply our framework to the compositional verification of the anonymity or privacy property of a specific example, it will often be a key task to show that the independence assumption holds. The above remark suggests a possibility that we can use conventional proof methods for anonymity/privacy when showing the independence assumption, although we do not go into detail here.

Author Index

Theoretical Aspects of Rationality and Knowledge

Proceedings of the 14th Conference (TARK 2013)

Edited by Burkhard C. Schipper

The biannual TARK conferences bring together researchers from a wide variety of fields sharing a common interest in reasoning about rationality and knowledge. The impact of this tradition, going back to 1986, is apparent in many of today's research trends and in the growth of an intellectual community beyond traditional disciplinary boundaries. This volume documents the 14th TARK conference, held at the Institute of Mathematical Sciences, Chennai, India on January 7 to 9, 2013. It includes 18 contributed talks, 8 poster presentations, and 3 invited talks given at the conference. Like earlier volumes in this series, it gives a sense of the state of the art in studies of knowledge and rationality in areas such as game theory, decision theory, belief revision, language analysis, and computation. It should be of value to researchers, teachers, and students.

ISBN number: 978-0-615-74716-3