

Simulation and information: quantifying over epistemic events

Hans van Ditmarsch*, Tim French†

June 5, 2008

Abstract

We introduce a multi-agent logic of knowledge with time where $F\varphi$ stands for ‘there is an informative event after which φ ’. Formula $F\varphi$ is true in a model iff it is true in all its refinements (i.e., ‘atoms’ and ‘back’ are satisfied; the dual of simulation). The logic is ‘almost’ normal, and positive knowledge is preserved. The meaning of $F\varphi$ is also “after the agents become aware of new factual information, φ is true,” and on finite models it is also “there is an event model (M, s) after which φ .” The former provides a correspondence with bisimulation quantifiers in a setting with epistemic operators.

1 Introduction

If you know where you are and you know what’s going to happen, you *want* to know where you will end up. But it can also be that you know where you are and know where you would like to end up, and that you *want* to know how to make that happen. Or you might *want* to know where you can end up in the first place, disregarding how that may be brought about. In the setting of logics for information update [3, 12, 11], knowledge of where you are and where you end up is formalized in multi-agent epistemic logic and semantically represented by a pointed multi-agent Kripke model, and knowledge about what’s going to happen is formalized as a dynamic modal operation that is interpreted as a relation between such Kripke models. The standard focus in dynamic epistemic logic was on the first of the three issues above: precision about a specific information update and precision about the effects of that update. In this contribution we focus on the other two issues instead. As this is partly about what may happen after *any* event, this concerns quantification over events. Our work is a further generalization of works such as [8, 2] and our presentation of future event operators as temporal is motivated by works such as [9] linking temporal epistemic logic to dynamic epistemic logic.

We introduce a very succinct logic of future events: the multi-agent logic of knowledge with (only) an operation $G\varphi$ that stands for ‘ φ holds after all informative events’ — the diamond version $F\varphi$ stands for ‘there is an informative event after which φ .’ The semantics of $G\varphi$ employs the notion of simulation [1]. We demonstrate that this is useful notion for informative event by a number of technical results for this logic—the logic is ‘almost’ normal, positive knowledge is preserved—and by a number of equivalence results for alternative semantics: $F\varphi$ also means “there is an event model (M, s) after which φ ,” and it also means “after the agents become aware of new factual information, φ is true.” The last provides a correspondence with bisimulation quantifiers [13, 7] in a setting with epistemic operators, as in [5].

*Computer Science, University of Otago, New Zealand & IRIT, France; hans@cs.otago.ac.nz.

†Computer Science and Software Engineering, University of Western Australia; tim@csse.uwa.edu.au.

For standard notions such as epistemic model, epistemic state, bisimulation, simulation, refinement, and event model, and for standard abbreviations and other conventions, we refer to the appendix. Throughout our contribution, the set of agents A is finite and the set of atoms P is (infinitely) enumerable.

2 Simulation and information

In *future event logic* one can express what informative events can be expected in a given information state. The language and the semantics of future event logic are as follows.

Definition 1 (Language \mathcal{L}_{fel}) Given agents A and atoms P , the language \mathcal{L}_{fel} is inductively defined as

$$\varphi ::= p \mid \neg\varphi \mid (\varphi \wedge \psi) \mid K_a\varphi \mid G\varphi$$

where $a \in A$ and $p \in P$.

We write $F\varphi$ for $\neg G\neg\varphi$. We propose a dynamic epistemic modal way to interpret temporal operators. This means that our future is the *computable future*: $F\varphi$ is true now, iff there is an (unspecified) informative event after which φ is true.

In the semantics for $G\varphi$, now to follow, we use the structural notion of *refinement*. A bisimulation satisfies **atoms**, **forth** and **back**, a simulation **atoms** and **forth**, and a refinement **atoms** and **back**. Refinement is therefore the dual of simulation: if (M, s) is a simulation of (M', s') , then (M', s') is a refinement of (M, s) (and we write $(M', s') \supseteq (M, s)$ and $(M, s) \sqsubseteq (M', s')$, respectively). See the appendix for more details.

Definition 2 (Semantics of future event logic) Assume an epistemic model $M = (S, R, V)$. The interpretation of $\varphi \in \mathcal{L}_{fel}$ is defined by induction.

$$\begin{aligned} M, s \models p & \quad \text{iff} \quad s \in V_p \\ M, s \models \neg\varphi & \quad \text{iff} \quad M, s \not\models \varphi \\ M, s \models \varphi \wedge \psi & \quad \text{iff} \quad M, s \models \varphi \text{ and } M, s \models \psi \\ M, s \models K_a\varphi & \quad \text{iff} \quad \text{for all } t \in S : (s, t) \in R_a \text{ implies } M, t \models \varphi \\ M, s \models G\varphi & \quad \text{iff} \quad \text{for all } (M', s') \supseteq (M, s) : M', s' \models \varphi \end{aligned}$$

In other words, $G\varphi$ is true in an epistemic state iff φ is true in all of its *refinements*. Note the ‘wrong direction’ in the definition: the future epistemic state simulates the current epistemic state. Typical model operations that produce a refinement are: blowing up the model (to a bisimilar model) such as adding copies that are indistinguishable from the current model and one another for some agent(s), removing states, and removing pairs of the accessibility relation for an agent. Validity in a model, and validity, are defined as usual. For $\{s \mid M, s \models \varphi\}$ we write $\llbracket \varphi \rrbracket_M$.

Example 3 Given are two agents that are uncertain about the value of a fact p , and where this is common knowledge, and where p is true. We assume both accessibility relations are equivalence relations, and that the epistemic operators model the agents’ knowledge. An informative event is possible after which a knows that p but b does not know that: $M, 1 \models F(K_a p \wedge \neg K_b K_a p)$. In the figure, $(M, 1)$ is the structure on the left, and its refinement

validating the postcondition is on the right.

$$0 \text{ --- } ab \text{ --- } \underline{1} \quad \Rightarrow \quad \begin{array}{c} 0 \text{ --- } ab \text{ --- } 1 \\ | \\ b \\ | \\ \underline{1} \end{array}$$

Proposition 4 Some elementary validities are:

1. $\models G(\varphi \rightarrow \psi) \leftrightarrow (G\varphi \rightarrow G\psi)$
2. $\models G\varphi \rightarrow \varphi$
3. $\models G\varphi \rightarrow GG\varphi$
4. $\models \varphi$ implies $\models G\varphi$
5. $\models K_a G\varphi \rightarrow GK_a\varphi$

Proof

1. Obvious.
2. A model is a refinement of itself; this corresponds to the trivial event ‘announce true’.
3. Consider the diamond version $FF\varphi \rightarrow F\varphi$. The relational composition of two simulations is again a simulation.
4. Obvious.
5. Consider the diamond version. Choose an accessible state in a refinement of a model. By **back**, this accessibility step can also be taken in the initial model.

Proposition 4 makes clear that G comes close to being a normal modal operator. But it is not a normal modal logic: the validities of the logic are not closed under uniform substitution of atomic variables for other formulas. For example, given some atom p , $p \rightarrow Gp$ is valid, but $\neg Kp \rightarrow G\neg Kp$ is not valid. A countermodel of the latter is the typical two-state situation where there is uncertainty about the value of p and where p is true. In that case, restriction to the p -state (‘public announcement of p ’) makes it known. Another countermodel is provided by the example above, for the knowledge of agent b .

A standard check for our bold claim that G formalizes a notion of informative event is that

Proposition 5 Bisimilar epistemic states have the same logical theory.

Proof This is not completely trivial, because bisimilarity is with respect to the epistemic operators, whereas the same logical theory is with respect to the epistemic operators and the temporal operator. Both can be established easily by the observation that if an epistemic state is a refinement of one of two given bisimilar epistemic states, it is also a refinement of the other epistemic state, because the relational composition of a simulation relation and a bisimulation relation is a simulation relation. The inductive case $G\varphi$ of the proof is:

Assume $\mathfrak{R} : (M, s) \Leftrightarrow (M', s')$, and let $M, s \models G\varphi$. To show that $M', s' \models G\varphi$, let (M'', s'') be such that $\mathfrak{R}' : (M'', s'') \rightrightarrows (M', s')$. We now have that $\mathfrak{R}' \circ \mathfrak{R}^{-1} : (M'', s'') \rightrightarrows (M, s)$. From that and $M, s \models G\varphi$ follows $M'', s'' \models \varphi$.

The positive formulas are those in the inductively defined fragment

$$\varphi ::= p \mid \neg p \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid K_a \varphi \mid G\varphi.$$

The preserved formulas are those for which

$$\varphi \rightarrow G\varphi \text{ is valid.}$$

I.e., they preserve truth under model refinement as long as the refinement includes an image for the actual state; the better known setting is model restriction. The first real corroboration that the temporal operators formalize a notion of informative event is that they model *growth of information* in the sense that positive knowledge does not get lost:

Proposition 6 Positive formulas preserve truth under refinement of models.

Proof Elementary.

Further corroboration that the temporal operators are quantifying over informative events is provided by the observation that *a restricted modal product is a refinement of a model* if the valuations of the states in that model are preserved under the product operation. This entails that the execution of an event model in an epistemic state is a refinement of that epistemic state. This we will now address.

3 Quantifying over event models

An informative update is the execution of an event model in an epistemic state. We consider event models for the epistemic language \mathcal{L}_{el} .

Proposition 7 An informative update is a refinement.

Proof Let $(M, s) = ((S, R, \text{pre}), s)$ be an event model for language \mathcal{L}_{el} . Let $(M', s') = ((S', R', V'), s')$ be an epistemic state and suppose $M, s \models \text{pre}(s)$. Then $\mathfrak{R}(t, t') = t$ is a simulation between $((M \otimes M'), (s, s'))$ and (M, s) ; below we assume that $(M \otimes M') = (S', R', V')$.

- **atoms:** if $(t, t') \in V'(p)$ then $t \in V(p)$;
- **forth:** let $((t, t'), (t', t'')) \in R'_a$; then $(t, t') \in R_a$.

Subject to the restrictions that we also have common knowledge in the epistemic language (language \mathcal{L}_{el}^C) and that the epistemic models are finite, the fit is exact: refinements are informative updates.

Proposition 8 (On finite epistemic models, given common knowledge) A refinement is an informative update.

Proof Given are a finite epistemic state $((S, R, V), s)$ and a refinement $((S', R', V'), s')$ of that model (according to refinement \mathfrak{R}). Consider the event model that is isomorphic to that refinement (according to isomorphism \mathfrak{J}). Instead of valuations for states t , this event model has preconditions for events $\mathfrak{J}(t)$. We want the preconditions only to be satisfied in states s such that $(s, t) \in \mathfrak{R}$ —this we cannot guarantee, but we can come close enough. In a finite model, states can be distinguished from all other (except bisimilar) states by employing the characteristic formulas $\delta_{((S, R, V), s)}$. (Characteristic formulas satisfy the property that truth in the structure equals entailment from the formula: $M, s \models \varphi$ iff $\delta_{(M, s)} \models \varphi$. Finite models have characteristic formulas in \mathcal{L}_{el}^C [4].) These are the preconditions we need. Given a state $s \in S$:

$$\text{pre}(\mathfrak{J}(t)) = \bigvee_{(s, t) \in \mathfrak{R}} \delta_{((S, R, V), s)}$$

This may give us pairs $(s, \mathfrak{J}(t))$ with $(s, t) \notin \mathfrak{R}$, but in that case s will be bisimilar to some s' satisfying the same characteristic formula and such that $(s', t) \in \mathfrak{R}$. Of course, the composition of the total bisimulation on (S, R, V) with the refinement relation \mathfrak{R} will also be a refinement relation. Without loss of generality we assume that \mathfrak{R} is maximal in the sense that it is a fixed-point of composition with that total bisimulation. This makes the structure of the proof clearer.

We now show that the restricted modal product $((S'', R'', V''), (s, s'))$ resulting from executing the event model $((S', R', \text{pre}), s')$ in the given epistemic state $(S, R, V), s$ is bisimilar to its refinement $((S', R', V'), s')$. The bisimulation \mathfrak{R}' is as follows: all pairs $(t, \mathfrak{J}(u))$ in the restricted modal product are bisimilar to the state $u \in S'$ of which their second argument of the pair is the isomorphic image:

$$\mathfrak{R}'(t, \mathfrak{J}(u)) = u$$

Condition **atoms** is obvious, as refinement satisfies **atoms**. Condition **forth** is also obvious: if $((t, \mathfrak{J}(u)), (t', \mathfrak{J}(u'))) \in R''_a$, then by definition of the modal product $(\mathfrak{J}(u), \mathfrak{J}(u')) \in R'_a$, so $(u, u') \in R'_a$ (and, indeed, $((t', \mathfrak{J}(u')), u') \in \mathfrak{R}'$ by definition). Condition **back** is not obvious but also holds. Let $(u, u') \in R'_a$ and $((t, \mathfrak{J}(u)), u) \in \mathfrak{R}'$. There must be a $t' \in S$ (modulo bisimilarity) such that $(t', u') \in \mathfrak{R}$ so that $(t', \mathfrak{J}(u'))$ is in the modal product. We now have that from $(u, u') \in R'_a$ follows $(\mathfrak{J}(u), \mathfrak{J}(u')) \in R'_a$, and we also have that from $(u, u') \in R'_a$ follows $(t, t') \in R_a$ (as \mathfrak{R} is a refinement). From $(t, t') \in R_a$ and $(\mathfrak{J}(u), \mathfrak{J}(u')) \in R'_a$ follows by definition the requested $(t, \mathfrak{J}(u)), (t', \mathfrak{J}(u')) \in R''_a$.

We emphasize that the notion of event model relative to a language allows for *infinite* event models, unlike in a logic with an inductively defined language including (finite!) event models. That is to come next. This will also allow us to compare our proposal with a known method [2] for quantifying over events.

Definition 9 ([2]) The language \mathcal{L}_{aeml} of arbitrary event model logic is the language \mathcal{L}_{fel} of future event logic with an additional inductive construct $[M, s]\varphi$.

We can view $[M, s]\varphi$ as an inductive construct, because, given the (enumerable) set of event model frames, $[M, s]$ can be seen as an operation on $|\mathcal{D}(M)|$ arguments of type formula (similar to automata-PDL). These arguments are the preconditions of the events in the event model. The language \mathcal{L}_{aeml} can also be seen as extension with construct $G\varphi$ of the language \mathcal{L}_{eml} for event model logic shown in the appendix.

To distinguish future event logic from logics with the same language but other semantics for $G\varphi$, we also write $\models_{\underline{=}}$ instead of \models for the forcing relation in future event logic; we (always) write \models_{\otimes} for the forcing relation in arbitrary event model logic.

For the semantics of $G\varphi$ in terms of event models we need to restrict the preconditions of their events to G -free formulas, i.e. \mathcal{L}_{eml} formulas. This is to avoid circularity in the definition, as $G\varphi$ could itself be a precondition of such an event. An event model is G -free iff all preconditions of its events are G -free.

Definition 10 (Semantics of arbitrary event model logic) Where the preconditions of events in any M are G -free.

$$M, s \models_{\otimes} G\varphi \quad \text{iff} \quad \text{for all } G\text{-free } (M, s) : M, s \models_{\otimes} [M, s]\varphi$$

There are refinements of epistemic models that cannot be seen as the result of executing an event model. This is because event models (in the language) have by definition a finite domain. For example, given a finite epistemic model (M, s) , consider its unwinding as an infinite tree (representing the bisimulation class). This is a refinement of (M, s) . But the result of executing a finite event model in a finite epistemic model cannot be an infinite tree. Of course, that tree is bisimilar to the initial epistemic state so can be seen in another sense as the result of execution the trivial event. But:

Because of the restriction to G -free preconditions in event models, we will still not get precise correspondence between the two semantics. The crux is that there are more epistemic distinctions in models than can be enumerated by epistemic formulas, see [2] for a similar matter. (However, we do not have a counterexample.)

Restricted to the class of finite epistemic models we still have that:

Proposition 11 Let M be finite. Then: $M, s \models_{\underline{=}} \varphi$ iff $M, s \models_{\otimes} \varphi$.

Proof Directly from Propositions 8 and 7.

4 Bisimulation and information

Instead of validating $F\varphi$ in some (M, s) by finding a refinement of (M, s) , we can equivalently find a *model restriction of a bisimilar epistemic state*. This alternative semantics $\models_{\underline{=}}$ is interesting because of a relationship with bisimulation quantifiers [13], for which many theoretical results are known; and it is also interesting because it shows that every informative update is equivalent to public announcement of factual information ‘of which the agents may not have been aware’.

Definition 12 Below, S' is the domain of M' , and S'' is such that $s \in S''$:

$$M, s \models_{\underline{=}} G\varphi \quad \text{iff} \quad \text{for all } (M', s') \Leftrightarrow (M, s) \text{ and for all } S'' \subseteq S' : M'|S'', s' \models_{\underline{=}} \varphi$$

On first thought it might seem that there are more refinements of a given model than domain restrictions of bisimilar models. In a refinement we can both restrict the domain (remove states) and remove links between states (delete pairs of the accessibility relation for an agent). But removing links between states can also be seen as a domain restriction on a even larger bisimilar model.

Proposition 13 $M, s \models_{\leftarrow} \varphi$ iff $M, s \models_{\leftrightarrow} \varphi$.

Proof This can be shown by induction on the complexity of formulas. As \models_{\leftarrow} and $\models_{\leftrightarrow}$ agree on the interpretations of atoms and all operators except G , it is sufficient to show that given $(M, s \models_{\leftarrow} \varphi$ iff $M, s \models_{\leftrightarrow} \varphi)$ we have $(M, s \models_{\leftarrow} G\varphi$ iff $M, s \models_{\leftrightarrow} G\varphi)$. From left to right the latter is trivial, because the refinements of (M, s) include the bisimulations of (M, s) . For the direction from right to left, it suffices to show that any refinement (M', s') of model (M, s) is the restriction of a model (M'', s'') that is bisimilar to (M, s) . This model is constructed as follows:

Let $M = (S, R, V)$, $M' = (S', R', V')$, and suppose that the refinement relation is \mathfrak{R} . Consider $(M'', s'') = ((S \oplus S', R'', V''), (s', 1))$, where for all agents $a \in A$

$$\begin{aligned} ((s, 0), (t, 0)) \in R''_a & \text{ iff } (s, t) \in R_a \\ ((s', 1), (t', 1)) \in R''_a & \text{ iff } (s', t') \in R'_a \\ ((s', 1), (t, 0)) \in R''_a & \text{ iff } \exists s \in S : (s, s') \in \mathfrak{R} \text{ and } (s, t) \in R_a \end{aligned}$$

We can then define the relation \mathfrak{R}' between (M, s) and $(M'', (s, 0))$ as follows:

$$\begin{aligned} (s, (s', 1)) \in \mathfrak{R}' & \text{ iff } (s, s') \in \mathfrak{R} \\ (s, (s, 0)) \in \mathfrak{R}' & \text{ iff } s \in S \end{aligned}$$

This relation \mathfrak{R}' is a bisimulation: it *still* satisfies **back** since the states of S added to M' also satisfy **back**: any relation between them copied their relation in the original M . But it now also satisfies **forth**:

If $(s, (s, 0)) \in \mathfrak{R}'$ and $(s, t) \in R_a$ then by definition of the first clause of \mathfrak{R}' we have $(t, (t, 0)) \in \mathfrak{R}'$ and, trivially by the definition of R''_a we have $((s, 0), (t, 0)) \in R''_a$. If $(s, (s', 1)) \in \mathfrak{R}'$ and $(s, t) \in R_a$ then we have (as before) $(t, (t, 0)) \in \mathfrak{R}'$ and $((s', 1), (t, 0)) \in R''_a$. The latter holds because of the *third* clause in the definition of R''_a .

Since $M''|(S' \times \{1\})$ is isomorphic to M' this concludes the proof.

We proceed by explaining the stated relation of this semantics with bisimulation quantifiers.

5 Bisimulation quantifiers

Suppose that apart from the atoms in P we had an additional, reserved, atom r . The future temporal operator can be seen as (existential) bisimulation quantification over r . (See the appendix— withheld from this abstract—for bisimulation quantifier semantics.) This relation becomes clear if we consider the restricted bisimulation version of the semantics for F :

First choose a bisimilar epistemic state, then do a model restriction in that epistemic state that contains the actual state.

Given the class of models also valuing r we can replace this by

First choose a P -bisimilar epistemic state (but where the valuation of r may vary wildly), then do a model restriction in that epistemic state that contains the actual state.

Of course we can match the variation in the valuation of r , as long as it contains the actual state, with that model restriction so we get

First choose a P -bisimilar epistemic state, then do a model restriction to the r -states in that epistemic state, on condition that it contains the actual state.

The part ‘choose a P -bisimilar epistemic state’ of this informal description is the semantics of an existential bisimulation quantification.

Definition 14 Where V' is the valuation of M' .

$$M, s \models_{\forall r} G\varphi \quad \text{iff} \quad \text{for all } (M', s') \Leftrightarrow_P (M, s) : s' \in V'(r) \text{ implies } M'|r, s' \models_{\forall r} \varphi$$

Example 15 For an example, consider again the model with common uncertainty about the value of an atom p for agents a and b , where p is true. We now operate on models that also value the atom r , in the figure this is the value of the second digit: note that r is *not* part of the logical language! Given the bisimulation quantification, the initial value of r does not matter. In this model the formula $F(K_{ap} \wedge \neg K_a K_b p)$ is true. The first transition is to a model that is bisimilar with respect to p only. The second transition is a restriction to the states where r is true.

$$\begin{array}{ccc}
 & \begin{array}{cc} 01 - ab - 11 \\ | \qquad | \\ b \qquad b \\ | \qquad | \\ 00 - ab - \underline{10} \end{array} & \Rightarrow & \begin{array}{cc} 01 - ab - 11 \\ | \\ b \\ | \\ 00 - ab - \underline{11} \end{array} & \Rightarrow & \begin{array}{c} 01 - ab - 11 \\ | \\ b \\ | \\ \underline{11} \end{array}
 \end{array}$$

Proposition 16 $M, s \models_{\Leftrightarrow} \varphi$ iff $M, s \models_{\forall r} \varphi$

Corollary 17 On finite models and given common knowledge in the language, the four different semantics for G correspond. (I.e. $\models_{\Leftrightarrow}$, \models_{\Leftarrow} , \models_{\otimes} , and $\models_{\forall r}$.)

Note that the extra atom r does not disturb these results. As yet it is mere surplus luggage that we’re carrying along towards the next section where it will become more meaningful. Our fourth perspective of bisimulation quantifier semantics is useful for theoretical and for practical reasons. A theoretical consequence is that

Proposition 18 Future event logic is decidable.

Proof Consider some $\varphi \in \mathcal{L}_{fel}$. Replace all occurrences of G in φ by $\forall r[r]$. It is decidable whether $\varphi^{\forall r}$ is satisfiable. (The decidability of bisimulation quantified modal logics can be generalized to multi-agent logics. Note that it also holds for specific model classes such as $\mathcal{KD45}$, $\mathcal{S5}$ and the modal μ -calculus; see [10].)

This is a useful result. If we add dynamic event model operators to future event logic (the language \mathcal{L}_{aeml}) we obtain arbitrary event model logic (see Definition 9). The restriction of this arbitrary event model logic to events that are public announcements is the logic $APAL$ investigated in [2]. For that logic, the satisfiability problem is undecidable (see [6]). That result also motivated this current investigation, because it promised more decidable logics.

However, we may note that the translation given (replace all occurrences of G in φ by $\forall r[r]$) is an accurate translation for *all* logics that are closed under bisimulation quantifiers and announcement. From a recent result of van Benthem and Ikegami [10] we know that the modal μ -calculus is also closed under products with event models. Since future event

logic and arbitrary event model logic agree on the interpretation of $G\varphi$ over finite models (Proposition 11), we can conclude that the satisfiability problem for \mathcal{L}_{aeml} restricted to finite models is reducible to the satisfiability problem for the μ -calculus, and hence decidable.

Note that the G -operator in arbitrary event model logic is interpreted differently (see Definition 10), and it is unknown whether this logic is decidable.

Our current perspective also provides us with additional modelling insight, namely that every informative update corresponds to the public announcement of an atomic fact. Kind of. What kind of? So far, it is unclear how to interpret this new perspective: we compare semantics with respect to model classes for different sets of atomic propositions; we did not add the fresh atom r to the logical language \mathcal{L}_{fel} . Here is where some trouble seems to start. If we merely add r as a formula to the language, but, e.g., rule out $K_a r$, we cannot truly interpret a r -restriction of a model as a public announcement: what use is an announcement of r if we cannot express that an agent a knows r after its announcement? But if we add r as just another propositional variable to the base clause of our inductive language definition, we run into trouble of a different kind: an existential bisimulation quantification means that the value of p is scrambled. Even with the restriction that the value of r remains unchanged in the actual state, we may now still have that an agent a knew r before an event, but has forgotten it afterwards, or vice versa. This is highly undesirable!

Example 19 In the previous example, we have that initially agent b knows that r is false: $K_b \neg r$, but after the update he apparently has forgotten that: $\neg K_b \neg r$. For another example: $K_a r \rightarrow F \neg K_a r$ would be a validity.

A technical solution to this dilemma, that at least makes the public announcement clear, is to

replace all occurrences of G in formulas by occurrences of $\forall r[r]$,

where $\forall r$ is universal bisimulation over r and where $[r]$ stands for public announcement of r . Public announcement is a singleton event model, accessible to all agents, where there precondition of the event is the formula between brackets, in this case: r . If we also allow r as formula, we can now interpret formulas of form $[r]\varphi$ in the usual sense for such events. For example, in our running example it is initially true that $\exists r \langle r \rangle (K_a p \wedge \neg K_a K_b p)$, as this is the translation of $F(K_a p \wedge \neg K_a K_b p)$. (For $\neg[\varphi]\neg\psi$ we write $\langle \varphi \rangle \psi$.)

But the real solution to this seeming dilemma is to consider an existential bisimulation as ‘the agents become aware of an additional fact’, about which uncertainty is possible. From a modelling point of view this means that, before the bisimulation operation, the value of r should be ‘no care’, in other words, ‘the agents are unaware of r ’, the bisimulation quantification itself then means ‘the agents become aware of r ’. This is now in the proper sense that we move to a bisimilar model except for atom r , and (unlike before!) without the restriction that r should remain true in the actual state, because maybe it was false in the first place. And after that it should be possible for them to know that r , or know that $\neg r$: they are now aware of their uncertainty about r . Of course after that, there might be other facts the agents might become aware of. If we merely add r to the base of the inductive language definition we cannot express this. We need one more step. That final step we will now set in the next section.

6 Becoming aware of factual information

First, we add more structure: For each epistemic model M , the set of atoms P is the disjoint union of a set of *relevant facts* $P_r(M)$ and a set of *irrelevant facts* $P_i(M)$. The set of relevant facts is typically finite. Then, in a given model, the interpretation of formulas containing irrelevant facts is undefined, unless they are bound by a bisimulation quantifier: we can only interpret irrelevant facts *after* they have *become* relevant to the agents. The bisimulation quantifier ‘makes a fact relevant’: its interpretation involves removing it from the set of irrelevant facts and adding it to the set of relevant facts. The result of this is that the value of irrelevant facts in any model is now truly ‘don’t care’ from the perspective of the agents. But they can still reason about the consequences of new facts after they were to become relevant, i.e., after the agents were to become aware of those facts.

Definition 20 The language \mathcal{L}_{qel} of quantified event logic is inductively defined as

$$\varphi ::= p \mid \neg\varphi \mid (\varphi \wedge \varphi) \mid K_a\varphi \mid [M, s]\varphi \mid \forall p\varphi$$

where $a \in A$ and $p \in P$.

For the dual $\exists p\varphi$, read, “(there exists a fact p such that) after the agents have become aware of p , φ .” We emphasize that by ‘becoming aware of p ’ we do not mean ‘learning that p is true’. In the information state resulting from becoming aware of p , the agents may know that p is true, or that p is false, or have any epistemic uncertainty about its value, e.g., they may not know whether p is true, or one agent may know but not another, etc.

Definition 21 (Semantics) Assume an epistemic model $M = (S, R, V)$ for atoms $P = P_r(M) \cup P_i(M)$. The interpretation of $\varphi \in \mathcal{L}_{qel}$ is defined by induction. We only give the clauses for relevant atoms p and for $\forall p\varphi$. The interpretation of irrelevant atoms is undefined. In the clause for $\forall p\varphi$ it is required that (M', s') is such that $P_r(M') = P_r(M) + p$ and $P_i(M') = P_i(M) - p$.

$$\begin{array}{lll} M, s \models p & \text{iff } s \in V_p & \text{where } p \in P_r(M) \\ M, s \models \forall p\varphi & \text{iff for all } (M', s') \Leftarrow_{P-p}(M, s) : M', s' \models \varphi & \text{where } p \in P_i(M) \end{array}$$

We have not explored this version in greater detail yet. Unlike the logics with temporal operators and the proposal with a reserved atom r for bisimulation quantification, agents may in this logic become aware of several different facts.

We think this logic may help modellers construct epistemic models in steps. In this logic, if we say that agent a is uncertain about p , and we represent this in the two-state epistemic model, this now means that the agent *only* is uncertain about p . The value of other atoms in that epistemic state is ‘don’t care’: information on an additional fact q might become available later, we then ‘simply’ construct a p -but-not- q bisimulation of this current epistemic state that represents the agents’ current knowledge, that includes q . This is exactly the $\exists q$ -operation! We close this section with a suitable illustration of this.

Example 22 Initially the agents are only uncertain about p . Then, they become aware of q : in fact, a knows the value of q but b doesn’t. Finally, it is announced that $p \vee q$. In the resulting state, a knows that p but b does not know that. Initially, the formula $\exists r(p \vee q)(K_a p \wedge \neg K_b K_a p)$ is true. Observe that the bisimulation quantification is in this example different from the

subsequent announcement. We now cannot announce the value of an atom, but only that of a more complex formula (well, a disjunction, but it could have been an epistemic formula as well).

$$\begin{array}{ccc}
 & 01 - ab - 11 & 01 - ab - 11 \\
 & | & | \\
 & b & b \\
 & | & | \\
 0 - ab - \underline{1} & \Rightarrow & 00 - ab - \underline{10} \Rightarrow \underline{10} \\
 & & | \\
 & & \underline{10}
 \end{array}$$

7 Further research

We are currently investigating the axiomatization of these logics, their model checking complexities (relative to different model classes, such as $S5$), and expressivity issues.

Acknowledgements: The authors gratefully acknowledge the suggestions and insights from the anonymous LOFT referees. Hans van Ditmarsch acknowledges support of the Netherlands Institute of Advanced Study where he was Lorentz Fellow in 2008.

References

- [1] P. Aczel. *Non-Well-Founded Sets*. CSLI Publications, Stanford, CA, 1988. CSLI Lecture Notes 14.
- [2] P. Balbiani, A. Baltag, H.P. van Ditmarsch, A. Herzig, T. Hoshi, and T. De Lima. What can we achieve by arbitrary announcements? A dynamic take on Fitch’s knowability. In D. Samet, editor, *Proceedings of TARK XI*, Louvain-la-Neuve, Belgium, 2007. Presses Universitaires de Louvain.
- [3] A. Baltag and L.S. Moss. Logics for epistemic programs. *Synthese*, 139:165–224, 2004. Knowledge, Rationality & Action 1–60.
- [4] J. Barwise and L.S. Moss. *Vicious Circles*. CSLI Publications, Stanford, 1996.
- [5] T. French. *Bisimulation quantifiers for modal logic*. PhD thesis, University of Western Australia, 2006.
- [6] T. French and H.P. van Ditmarsch. Undecidability for arbitrary public announcement logic. To appear in *Proceedings of Advances in Modal Logic 2008*, Nancy, 2008.
- [7] M. Hollenberg. *Logic and bisimulation*. PhD thesis, University of Utrecht, 1998.
- [8] T. Hoshi. The logic of communication graphs for group communication and the dynamic epistemic logic with a future operator. Philosophy Department, Stanford University, 2006.
- [9] J.F.A.K. van Benthem, J.D. Gerbrandy, and E. Pacuit. Merging frameworks for interaction: DEL and ETL. In D. Samet, editor, *Proceedings of TARK 2007*, pages 72–81, 2007.

- [10] J.F.A.K. van Benthem and D. Ikegami. Modal fixed-point logic and changing models. In A. Avron, N. Dershowitz, and A. Rabinovich, editors, *Pillars of Computer Science*, volume 4800 of *Lecture Notes in Computer Science*, pages 146–165. Springer, 2008. Also available as ILLC Prepublication Series PP-2008-19.
- [11] J.F.A.K. van Benthem, J. van Eijck, and B.P. Kooi. Logics of communication and change. *Information and Computation*, 204(11):1620–1662, 2006.
- [12] H.P. van Ditmarsch, W. van der Hoek, and B.P. Kooi. *Dynamic Epistemic Logic*, volume 337 of *Synthese Library*. Springer, 2007.
- [13] A. Visser. Bisimulations, model descriptions and propositional quantifiers, 1996. Logic Group Preprint Series 161, Department of Philosophy, Utrecht University.

Appendix of technical terms

Structural notions Assume a finite set of agents A and a countably infinite set of atoms P .

Definition 23 (Structures) An *epistemic model* $M = (S, R, V)$ consists of a *domain* S of (factual) *states* (or ‘worlds’), *accessibility* $R : A \rightarrow \mathcal{P}(S \times S)$, and a *valuation* $V : P \rightarrow \mathcal{P}(S)$. For $s \in S$, (M, s) is an *epistemic state* (also known as a pointed Kripke model).

For $R(a)$ we write R_a ; accessibility R can be seen as a set of relations R_a , and V as a set of valuations $V(p)$. Given two states s, s' in the domain, $R_a(s, s')$ means that in state s agent a considers s' a possibility. We adopt the standard rules for omission of parentheses in formulas, and we also delete them in representations of structures such as (M, s) whenever convenient and unambiguous. (For $B \subseteq A$, write $R(B)$ (or R_B) for $(\bigcup_{a \in A} R(a))^*$. This accessibility relation is used to interpret common knowledge among agents in B , except in the reference to characteristic formulas not otherwise used in this contribution: $M, s \models C_B \varphi$ iff for all t : $(s, t) \in R_B$ implies $M, t \models \varphi$.)

Definition 24 (Bisimulation, simulation, refinement) Let two models $M = (S, R, V)$ and $M' = (S', R', V')$ be given. A non-empty relation $\mathfrak{R} \subseteq S \times S'$ is a bisimulation, iff for all $s \in S$ and $s' \in S'$ with $(s, s') \in \mathfrak{R}$:

atoms $s \in V(p)$ iff $s' \in V'(p)$ for all $p \in P$

forth for all $a \in A$ and all $t \in S$, if $R_a(s, t)$, then there is a $t' \in S'$ such that $R_a(s', t')$ and $(t, t') \in \mathfrak{R}$

back for all $a \in A$ and all $t' \in S'$, if $R_a(s', t')$, then there is a $t \in S$ such that $R_a(s, t)$ and $(t, t') \in \mathfrak{R}$

We write $(M, s) \Leftrightarrow (M', s')$, iff there is a bisimulation restricted to P' between M and M' linking s and s' . Then we call (M, s) and (M', s') bisimilar. We also say that (M, s) is similar to (M', s') and vice versa.

A relation that satisfies **atoms** and **forth** is a *simulation*, and in that case (M', s') is a *simulation* of (M, s) , and (M, s) is a *refinement* of (M', s') , and we write $(M, s) \supseteq (M', s')$ (or $(M', s') \preceq (M, s)$).

A bisimulation (simulation) that satisfies atoms for a subset $P' \subseteq P$ is a P' -bisimulation (P' -simulation); we write $(M, s) \Leftrightarrow_{P'} (M', s')$ ($(M, s) \supseteq_{P'} (M', s')$), etc.

Standard language notions The languages of propositional logic (\mathcal{L}_{pl}) and of epistemic logic (\mathcal{L}_{el}) — $a \in A, p \in P, B \subseteq A$.

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

$$\mathcal{L}_{el} : \quad \varphi ::= p \mid \neg\varphi \mid (\varphi \wedge \varphi) \mid K_a\varphi$$

$$\mathcal{L}_{el}^C : \quad \varphi ::= p \mid \neg\varphi \mid (\varphi \wedge \varphi) \mid K_a\varphi \mid C_B\varphi$$

Standard abbreviations include: $\varphi \vee \psi$ iff $\neg(\neg\varphi \wedge \neg\psi)$; $\varphi \rightarrow \psi$ iff $\neg\varphi \vee \psi$, $\hat{K}_a\varphi$ iff $\neg K_a\neg\varphi$.

Event model logic All the following are simultaneously defined:

Definition 25 Language \mathcal{L}_{eml} of event model logic:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid K_a\varphi \mid [M, s]\varphi$$

Definition 26 (Event model) An *event model* for a finite set of agents A and a language \mathcal{L} is a triple $M = (S, R, \text{pre})$ where

- *domain* S is a finite non-empty set of events,
- $R : A \rightarrow \mathcal{P}(S \times S)$ assigns an *accessibility relation* to each agent,
- $\text{pre} : S \rightarrow \mathcal{L}$ assigns to each event a *precondition*,

A pair (M, s) with a distinguished actual event $s \in S$ is called an *epistemic event*. An epistemic event with a singleton domain, accessible to all agents, and identity postcondition, is a *public announcement*.

Definition 27 (Semantics of event model logic) Let a model (M, s) with $M = (S, R, V)$ be given. Let $a \in A, B \subseteq A$, and $\varphi, \psi \in \mathcal{L}$.

$$(M, s) \models [M, s]\varphi \quad \text{iff} \quad (M, s) \models \text{pre}(s) \text{ implies } (M \otimes M, (s, s)) \models \varphi$$

Definition 28 (Execution of an event model) Given are an epistemic model $M = (S, R, V)$, a state $s \in S$, an event model $M = (S, R, \text{pre})$, and an event $s \in S$ with $(M, s) \models \text{pre}(s)$. The result of executing (M, s) in (M, s) is the model $(M \otimes M, (s, s)) = ((S', R', V'), (s, s))$ where

- $S' = \{(t, t) \mid (M, t) \models \text{pre}(t)\}$,
- $R'(a) = \{((t, t), (u, u)) \mid (t, t), (u, u) \in S' \text{ and } (t, u) \in R(a) \text{ and } (t, u) \in R(a)\}$,
- $V'(p) = \{(t, t) \mid (M, t) \models p\}$.

Bisimulation quantifiers and bisimulation quantified epistemic logic The language and semantics are as follows.

Definition 29 (Bisimulation quantified epistemic logic) Bisimulation quantified epistemic logic augments epistemic logic by additionally allowing formulas of the kind $\forall p\varphi$ in the recursive definition, where p is an atom of P , and φ is a formula. This is the language \mathcal{L}_{bqel} .

Given an epistemic model $M = (S, R, V)$ and a state $s \in S$ we say:

$$M, s \models \forall p\varphi \quad \text{iff} \quad \text{for every epistemic model } (M', s') \stackrel{\leftarrow P \setminus \{p\}}{\hookrightarrow} (M, s) : M', s' \models \varphi.$$