Logic Journal of IGPL Advance Access published October 27, 2011

Connecting dynamic epistemic and temporal epistemic logics

Hans van Ditmarsch[1,*], Wiebe van der Hoek[2,†] and Ji Ruan[3,‡]

[1] *University of Seville, Spain,* [2] *University of Liverpool, UK and* [3] *University of New South Wales, Australia*

Abstract

We give a relation between a logic of knowledge and change, with a semantics on Kripke models, and a logic of knowledge and time, with a semantics on interpreted systems. In particular, given an *epistemic state* (pointed Kripke model with equivalence relations) and a formula in a *dynamic epistemic logic* (a logic describing the consequences of epistemic actions), we construct an *interpreted system* relative to that epistemic state and that formula that satisfies the translation of the formula into a *temporal epistemic logic*. The construction involves that the *protocol* that is implicit in the dynamic epistemic formula, i.e. the set of sequences of actions being executed to evaluate the formula, is made explicit. We first focus on the logic of knowledge and change that is known as public announcement logic, then generalize our results to a dynamic epistemic logic.

Keywords: dynamic epistemic logic, temporal epistemic logic, model checking, interpreted systems, protocols.

1 Introduction

Epistemic logic is a formalization of knowledge. Seminal work in this area is Hintikka's [9], from 1962, and since then many philosophers have been interested in further developing the notions of knowledge and belief using a possible world semantics. In the late 1980s these approaches were picked up and further developed by computer scientists, cf. [4, 8]. This development was originally motivated by the need to reason about communication protocols. One is typically interested in what different parties to a protocol know before, during and after a run (an execution sequence) of the protocol. This interest in change of knowledge over time is already eminent in this area for twenty years. Fagin, Halpern, Moses and Vardi's seminal *Reasoning about Knowledge* [4] is a culmination of several earlier papers in this area, and also incorporates Halpern and Vardi's 1986 paper [8] *The Complexity of Reasoning about Knowledge and Time*. Apart from computer science, there is much interest in the temporal dynamics of knowledge and belief in areas as diverse as artificial intelligence [15], multi-agent systems [20], philosophy [1] and game theory [2].

The central notion in the work of Fagin *et al.* [4] is that of an *interpreted system*. When compared to Kripke (possible worlds) models, interpreted systems have at least two appealing features: a natural accessibility relation between domain objects, and an equally natural notion of dynamics, modelled by *runs*. The accessibility relation as we know it from the possible worlds model is in this case *grounded*; it has a direct and natural interpretation, as follows. In an interpreted system, the role of possible worlds is performed by global states,

*E-mail: hvd@us.es
†E-mail: Wiebe.Van-Der-Hoek@liverpool.ac.uk
‡E-mail: ji@jiruan.net

doi:10.1093/jigpal/jzr038

which are constituted by the agents' local states and the state of the environment. Each agent knows exactly its own local state and the possible local states of other agents: two global states are indistinguishable for an agent if his local compartment is the same. Secondly, an interpreted system defines a number of *runs* through such global states (i.e. a sequence of global states). Each run corresponds to a possible computation allowed by a protocol. In an object language with temporal and epistemic operators one can then express temporal properties such as *liveness* and temporal epistemic properties such as *perfect recall*.

The interpreted systems approach has proven its value far beyond the scope of communication protocols, and Temporal Epistemic Logic (TEL) that describes them has been studied and applied extensively. Rather than linear time, one may consider branching time, and apart from synchrony (roughly, the agents know what the time is) and perfect recall, one may consider properties with or without assuming a unique *initial state*, and with or without the principle of *no learning*. We have only mentioned a few of the parameters that one can vary; for a comprehensive overview of the linear case we refer to [7], and for the branching time case, to [27]. Moreover, apart from the interpreted systems stance there have been several other and related approaches to knowledge and time, such as the distributed processes approach of [17].

Dynamic Epistemic Logic (DEL) studies what kinds of actions are responsible for change of knowledge in a multi-agent setting. A quizmaster may publicly announce the winning lot, or whisper it in the ear of his assistant. Both result in a change of knowledge for everybody present, although the change is different in either case. Where belief revision [1] is interested in describing the effect of expansion, contraction and revision of a belief set of one agent, DEL treats all of knowledge, higher-order knowledge, and its dynamics on the same level, and it gives a fine-tuned analysis of the *way* the revision is brought about, ranging from a private insight by one agent to a public announcement in a group. Unlike TEL, where the meaning of a temporal shift only appears from the underlying model, in DEL this change is specified 'directly' in the dynamic operators. Starting with a few somewhat isolated contributions in the late 1980s [18, 23], the area strongly developed from the late 1990s onward [3, 6, 30]. A general theory emerges only now and in bits. We will base our treatment of DEL on [30].

1.1 Related work

The presented frameworks interact both on the level of logical languages and on the level of semantic objects—and it is precisely this interaction that is the subject of the underlying investigation. Various results have already been achieved. The relation between Kripke models and interpreted systems has been investigated by Lomuscio and Ryan in [13]. They focus on an interpreted system named hypercube that corresponds to the cartesian product of all local state values, and that has no dynamic features. Their approach suits Kripke models where all states have different valuations, which is not generally the case. Pacuit [16] compares the history-based approach by Parikh and Ramanujam [17] to interpreted systems, with runs. This addresses the relation between Kripke models *with histories consisting of event sequences* and interpreted systems. It handles *partial* observability of agents, when they perceive only some but not all of a sequence of events, but does not address the partial observability common in dynamic epistemics, where only an aspect of an event is observable. A more recent work by van Benthem *et al.* [26], rooted in older work [24, 25], gives a systematic and precise relation between TEL (corresponding to Epistemic Temporal

Logic in their notation) and DEL. They provide a representation theorem characterizing the largest class of temporal epistemic models corresponding to dynamic epistemic protocols in terms of notions of *Perfect Recall, No Miracles, and Bisimulation Invariance*. Also they give an axiomatization for a public announcement logic constrained by protocols, and study the issues of completeness and decidability of that logic. In their approach, starting from an initial (pointed) epistemic model, a TEL model is generated by repeatedly applying updates with event models (or, action models). Sequences of pointed event models closed under initial segment relations forms a DEL protocol. We have applied their elegant approach on representing protocols. We do not assume a parameter set of allowed protocols, as they do, but instead compute a protocol from a given DEL formula.

Other work on relating DEL with TEL focuses on the TEL feature to refer to past actions. Recent work by Hoshi and Yap [10] extends the public announcement logic constrained by protocols in [26] with full class of event models and a past-time operator; a new method is introduced for axiomatization. And Sack [21] also adds a past-time operator to public announcement logic and DEL, but without protocols.

1.2 Motivation

Our motivation to find links between DEL and TEL is model checking. Much recent work in model checking multi-agent systems is based on TEL. State-of-the-art model checkers are MCMAS [19], MCK [5], VerICS [12] and MCTK [22]. As far as we know, DEMO [32] is currently the only model checker based on DEL. In a previous study [31], we address a relation between TEL and DEL by specifying and checking a communication protocol that solves the Russian Cards problem [28] in three model checkers: MCK, MCMAS and DEMO. We encode public announcement logic into temporal epistemics by way of explicitly introducing boolean state variables for each announcement with values corresponding to unknown (i.e. before the announcement is made), and true (after a truthful announcement). In another study [29], we discuss an inherent difficulty of specifying the Sum-and-Product riddle in temporal epistemic model checkers such as MCK and MCMAS. In MCK, a state of the environment is an assignment to a set of variables declared in the environment section. These variables are usually assumed to be partially accessible to the individual agents, and agents could share some variables. The change of the state of the multi-agent system is either made by agents or the environment, in the form of changing these variables. In both cases, we need guarded statements to make the change. For example, a simple deterministic statement has the form:

$$if\ \ cond \rightarrow C\ [otherwise \rightarrow Co]fi$$

where command C is eligible for execution only if the corresponding condition *cond* evaluates to true in the current state, otherwise, the command *Co* will be executed. If we would like to model the Sum-and-Product riddle in MCK, the effect of a public announcement should be recorded in a variable that is accessible to all agents. But unfortunately, even though MCK can check *epistemic postconditions*, it only supports checking *propositional* formulas as *preconditions* in *cond* (e.g. an epistemic formula $K_i\varphi$ or a temporal formula $\bigcirc\varphi$ is not allowed in *cond*). An explanation lies in the process of constructing interpreted systems during model checking. While the truth of a propositional formula only depends on a single state, that of an epistemic or temporal formula relies on other possible states, which may

have not yet been generated during the process. So in[29], we only analyse the Sum-and-Product riddle in public announcement logic and solve it using the dynamic epistemic model checker DEMO.

Luo *et al.* [14] reduce a model checking problem in public announcement logic to a series of Binary Decision Diagram (BDD) based computations of the set of states that satisfies a given epistemic formula. They apply this to model checking the Sum-and-Product riddle. The result is checked in BDD-based model checker MCTK and the model checking time is 90 seconds, which is an impressive improvement over 1864 seconds obtained by using DEMO, which is not based on BDD method, on the same computer. This suggests that BDD-based temporal epistemic model checkers may have better efficiency than DEL-based methods.

The investigations in [14, 29, 31] motivated us to have a more systematic analysis between model checking in DEL versus model checking in TEL. We provide that correspondence in the following sense: given an epistemic state (pointed Kripke model with equivalence relations) and a formula in a DEL, we construct an interpreted system relative to that epistemic state and that formula that satisfies the translation of the formula into a TEL. The construction involves that the protocol that is implicit in the dynamic epistemic formula, i.e. the set of sequences of actions being executed to evaluate the formula, is made explicit.

This article is organized as follows. Section 2 contains the fundamentals of logical languages and structures involved. Section 3 presents our results for public announcement logic. We translate all formulas with announcements into a TEL, and prove a theorem that identifies the truth of such a dynamic epistemic formula in a world of a Kripke model, with the truth of a temporal epistemic formula in a global state of a corresponding interpreted system. Section 4 provides a generalization of these results from public announcement logic to DEL with detailed proofs.

2 Logical Preliminaries

We introduce *four* structural primitives and *two* languages. The structures are:

- *State models*, which are Kripke models with equivalence relations representing agents' knowledge about states;
- *Action models*, which are Kripke models with equivalence relations representing agents' knowledge about actions;
- *Forest models*, which are Kripke models with not only accessibility relations representing agents' knowledge of states but also accessibility relations representing state transitions;
- *Action-based interpreted systems*, which are interpreted systems with transitions labelled with actions.

The reason that we restrict state models and action models to have only equivalence relations is because the framework we want to relate it to, namely that of action-based interpreted systems, has equivalence relations.

The languages are those of DEL and a variant of TEL which one could think of as 'next-time temporal epistemic logic'. The former can be given meaning both on state models and on forest models; the latter both on forest models and on action-based interpreted systems. As global parameters to both the languages and the structures we have a set Ag of n agents, and a (countable) set Q of atoms q.

2.1 Structures

Definition 2.1 (State Model and Action Model [3])
A *state model* is a structure $\langle W, \{\sim_i | i \in Ag\}, \pi \rangle$ where W is a domain of possible states, $\sim_i$ is an equivalence relation on W^2 expressing the states that are indistinguishable from each other by agent $i \in Ag$, and $\pi : W \to \wp(Q)$ is a valuation (or interpretation) that determines for each state which atoms are true in that state. We call (M, w) a pointed state model where M is a state model and $w \in W$.

An *action model* A is a structure $\langle \mathsf{W}, \{\sim_i | i \in Ag\}, \mathsf{pre} \rangle$ where W is a domain of *actions*, and $\sim_i$ is an equivalence relation on W^2 expressing the *actions* that are indistinguishable from each other by agent $i \in Ag$, and $\mathsf{pre} : \mathsf{W} \to \mathcal{L}$ is a precondition function that assigns a *precondition* $\mathsf{pre}(\mathsf{a})$ in language $\mathcal{L}$ to each $\mathsf{a} \in \mathsf{W}$. We call (A, a) a pointed action model where A is an action model and $\mathsf{a} \in \mathsf{W}$.

A state model captures all the possible states that agents could think of at a particular moment and their knowledge about these states. It represents a static view of a system. An action model captures similar aspects in terms of actions, instead of states. An action has a precondition, which must be satisfied in a state if this action is executable in that state. An action model transforms the states of a system by executing actions on these states. This will be introduced as an *update product operation* in Section 2.3. A pointed state model (M, w) uses w to denote the actual state. Similar for pointed action models.

To represent both states and changes of a system in a single model, we introduce a structure that combines states and actions.

Definition 2.2 (Forest Model)
Given a set of actions W, a forest model is a structure

$$\langle W, \{\sim_i | i \in Ag\}, \{\to_{\mathsf{a}} | \mathsf{a} \in \mathsf{W}\}, \pi \rangle$$

where

- W is a set of states;
- $\sim_i \subseteq W \times W$ is an equivalence relation of agent i for each $i \in Ag$;
- $\to_{\mathsf{a}} \subseteq W \times W$ is a binary relation on states expressing the execution of action a with an *extra condition*: each state has at most one action predecessor (i.e. for each state $w \in W$, if there is $w_1, w_2 \in W$ and $\mathsf{a}', \mathsf{a}'' \in \mathsf{W}$ such that $w_1 \to_{\mathsf{a}'} w$ and $w_2 \to_{\mathsf{a}''} w$, then $w_1 = w_2$ and $\mathsf{a}' = \mathsf{a}''$);
- and π is a valuation function from W to $\wp(Q)$.

We may write a forest model as $\langle W, \{\sim_i | i \in Ag\}, \{\to_{\mathsf{a}}\}, \pi \rangle$ if W is clear from the context.

In this article, we represent all states of W in the form of $(w, \mathsf{a}^1, \ldots, \mathsf{a}^m)$ where w is a state and $\mathsf{a}^1, \ldots, \mathsf{a}^m$ is a sequence of actions. If $w_1 \to_{\mathsf{a}} w_2$ then w_2 can be written as (w_1, a). Note that according to the definition of $\to_{\mathsf{a}}$, there is at most one a predecessor for w_2, so such naming will not lead to ambiguity. For brevity, $((w, \mathsf{a}^1, \ldots, \mathsf{a}^m), \mathsf{a})$ and $(w, \mathsf{a}^1, \ldots, \mathsf{a}^m, \mathsf{a})$ are treated as the same state. Note that our forest model is comparable to the history-based temporal model in [26] in the sense that $(w, \mathsf{a}^1, \ldots, \mathsf{a}^m)$ can be seen as a (partial) history.

To associate a forest model with an interpreted system, we extend the interpreted system of [4] with actions.

DEFINITION 2.3 (Action-based Interpreted System)
Given a set of actions W, an action-based interpreted system is a structure

$$\mathcal{I} = \langle \mathcal{G}, \mathcal{R}, \{\rightarrow_{\mathsf{a}} | \mathsf{a} \in \mathsf{W}\}, \pi \rangle$$

where

- $\mathcal{G}$ is a set of global states; a global state $s \in \mathcal{G}$ is a tuple $s = (s_e, s_1, \ldots, s_n)$ where s_e is the state of the environment and for $i \in Ag$, s_i is the local state of agent i;
- $\mathcal{R}$ is a set of runs over $\mathcal{G}$; a *run* r is an infinite sequence of global states; the pair (r, m) consisting of a run and a time point is also referred to as a point, and $\mathsf{r}(m)$ is a global state associated with the point (r, m);
- $\rightarrow_{\mathsf{a}}$ is a binary relation on points in a run; for two consecutive points (r, m) and $(\mathsf{r}, m+1)$, there exists a unique action $\mathsf{a} \in \mathsf{W}$ such that $(\mathsf{r}, m) \rightarrow_{\mathsf{a}} (\mathsf{r}, m+1)$;
- π is a valuation function which decides for each global state s a set of atoms $P \subseteq Q$ that are true in s; the valuation of a point (r, m) is simply $\pi(\mathsf{r}(m))$.

Let $\mathsf{r}(m) = s$ be the global state at time m in run r, then with $\mathsf{r}_i(m)$ we mean local state s_i. Two points (r, m) and (r', m') are indistinguishable for i, written as $(\mathsf{r}, m) \sim_i (\mathsf{r}', m')$, if $\mathsf{r}_i(m) = \mathsf{r}'_i(m')$ (i.e. the local states of their associated global states are the same for i).

Note that a point is only associated with a global state, but a global state can be associated with multiple points. Action relation $\rightarrow_{\mathsf{a}}$ only relates the consecutive points in the same run, but two runs may overlap in the sense that $\mathsf{r}(m) = \mathsf{r}'(m')$ and $\mathsf{r}(m+1) = \mathsf{r}'(m'+1)$.

2.2 Languages

DEFINITION 2.4 (Language $\mathcal{L}_{\mathsf{DEL}}$)
The language $\mathcal{L}_{\mathsf{DEL}}$ of Dynamic Epistemic Logic is inductively defined as follows

$$\varphi \quad ::= \quad q \mid \neg\varphi \mid (\varphi \wedge \varphi) \mid K_i\varphi \mid C_B\varphi \mid [\mathsf{A}, \mathsf{a}]\varphi$$

where $q \in Q$, $i \in Ag$, $B \subseteq Ag$, (A, a) a pointed action model, and precondition function pre is defined on $\mathsf{W} \rightarrow \mathcal{L}_{\mathsf{DEL}}$ such that a can not occur inside $\mathsf{pre}(\mathsf{a})$. The usual abbreviations for $\top, \vee, \rightarrow$ are employed.

The language $\mathcal{L}_{\mathsf{DEL}}$ is an extension of the language of epistemic logic with dynamic modalities parameterized by action models. In this article, we assume that all points of all action models are differently named, so that we can uniquely associate a particular a with the pointed model (A, a) whenever convenient. The extra requirement on precondition function prevents self-reference, e.g. $\mathsf{pre}(\mathsf{a}) = [\mathsf{A}, \mathsf{a}]\varphi$, or $\mathsf{pre}(\mathsf{a}) = [\mathsf{A}, \mathsf{b}]\varphi$ and $\mathsf{pre}(\mathsf{b}) = [\mathsf{A}, \mathsf{a}]\varphi$, is not allowed.

For the special case of singleton action models with reflexive accessibility relations for all agents, i.e. public announcements, we write $[\varphi]\psi$ where φ is the precondition (the announced formula).

We want to connect $\mathcal{L}_{\mathsf{DEL}}$ to a temporal epistemic language. The dynamic part of $\mathcal{L}_{\mathsf{DEL}}$ is the action modality and it can be seen as representing one time step. Our approach is to associate each action modality with a one-step temporal modality. So we define the language of Next-time Temporal Epistemic Logic (NTEL).

Definition 2.5 (Language $\mathcal{L}_{\mathsf{NTEL}}$)
The language $\mathcal{L}_{\mathsf{NTEL}}$ of **NTEL** is inductively defined as follows.

$$\varphi \quad ::= \quad q \mid \neg\varphi \mid (\varphi \wedge \varphi) \mid K_i\varphi \mid C_B\varphi \mid \bigcirc_{\mathsf{a}}\varphi$$

where, $q \in Q$, $i \in Ag$, $B \subseteq Ag$, and $\mathsf{a} \in \mathsf{W}$. The usual abbreviations for $\top, \vee, \rightarrow$ are employed.

Note that both $\mathcal{L}_{\mathsf{DEL}}$ and $\mathcal{L}_{\mathsf{NTEL}}$ have common knowledge, while in [26], they left out common knowledge from their temporal public announcement logic **TPAL**. This difference reflects different focus. [26] focuses on studying axiomatization, completeness and decidability problems of **TPAL**, and common knowledge adds more complexity for these problems. While in this article, we focus on model checking, and common knowledge can be handled relatively easily.

2.3 Semantics

In the following, we give *meaning* to the formulas of the languages over the structures we have introduced. More specifically, we interpret **DEL** formulas over *state models* and *forest models*, and interpret **NTEL** formulas over *forest models* and *action-based interpreted systems*. See Figure 1.

We distinguish four different interpretations. $\models_{\mathsf{sd}}$ denotes the interpretation of a **DEL** formula over a state model; $\models_{\mathsf{fd}}$ denotes the interpretation of a **DEL** formula over a forest model; $\models_{\mathsf{ft}}$ denotes the interpretation of an **NTEL** formula over a forest model; and $\models_{\mathsf{it}}$ denotes the interpretation of an **NTEL** formula over an action-based interpreted system. All these interpretations are defined similarly in terms of atomic propositions, logical connectives and knowledge modalities, which we assume to be familiar [4, 30]. We focus on clauses of action executions, and the temporal connectives. For action executions, we also mention the special case of public announcement.

Definition 2.6 (Semantics $\models_{\mathsf{sd}}$)
The semantics of $[\mathsf{A},\mathsf{a}]\psi$ over a state model M is as follows,

$$M, w \models_{\mathsf{sd}} [\mathsf{A},\mathsf{a}]\psi \quad \text{iff} \quad M, w \models_{\mathsf{sd}} \mathsf{pre}(\mathsf{a}) \Rightarrow M \otimes \mathsf{A}, (w, \mathsf{a}) \models_{\mathsf{sd}} \psi$$

where $\otimes$ is the update operation, which we define next.

The update operation is a function mapping a state model and an action model to a new state model. Given a state model $M = \langle W, \{\sim_i \mid i \in Ag\}, \pi\rangle$ and an action model

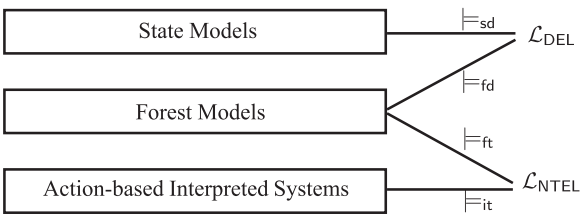


Fig. 1. Semantics of two languages over three models.

$\mathsf{A} = \langle \mathsf{W}, \sim''_1, \ldots, \sim''_n, \mathsf{pre} \rangle$, $M \otimes \mathsf{A} = \langle W', \{\sim'_i \mid i \in Ag\}, \pi' \rangle$ where

$$\begin{array}{lll} W' & = & \{(w, \mathsf{a}) \mid w \in W, \mathsf{a} \in \mathsf{W} \text{ and } M, w \models_{\mathsf{sd}} \mathsf{pre}(\mathsf{a})\} \\ (w, \mathsf{a}) \sim'_i (v, \mathsf{b}) & \text{iff} & w \sim_i v, \text{ and } \mathsf{a} \sim''_i \mathsf{b} \\ q \in \pi'(w, \mathsf{a}) & \text{iff} & q \in \pi(w) \end{array}$$

This definition is essentially the same as in [3]. In the special case of public announcement, in which the announcement of formula φ corresponds to a singleton action model, say $(\mathsf{A}_0, \mathsf{a}_0)$, we have:

$$M, w \models_{\mathsf{sd}} [\varphi]\psi \quad \text{iff} \quad M, w \models_{\mathsf{sd}} \varphi \Rightarrow M \otimes \mathsf{A}_0, (w, \mathsf{a}_0) \models_{\mathsf{sd}} \psi.$$

Definition 2.7 (Semantics $\models_{\mathsf{fd}}$)
The semantics of $[\mathsf{A}, \mathsf{a}]\psi$ over a forest model M is as follows,

$$M, w \models_{\mathsf{fd}} [\mathsf{A}, \mathsf{a}]\psi \quad \text{iff} \quad M, w \models_{\mathsf{fd}} \mathsf{pre}(\mathsf{a}) \Rightarrow \exists v \text{ s.t. } w \rightarrow_{\mathsf{a}} v \text{ and } M, v \models_{\mathsf{fd}} \psi$$

In the special case of public announcement, in which the announcement of formula φ corresponds to $(\mathsf{A}_0, \mathsf{a}_0)$, we have:

$$M, w \models_{\mathsf{fd}} [\varphi]\psi \quad \text{iff} \quad M, w \models_{\mathsf{fd}} \varphi \Rightarrow \exists v \text{ s.t. } w \rightarrow_{\mathsf{a}_0} v \text{ and } M, v \models_{\mathsf{fd}} \psi$$

The condition '$[\varphi]$ corresponds to a_0' is essential. If it does not hold, we can get into trouble, as the following example explains.

Example 2.8
Consider the formula $p \rightarrow [p]\top$. According to Definition 2.6, for any state model M and state w, we always have $M, w \models_{\mathsf{sd}} p \rightarrow [p]\top$. This is because if p is true in w, then a truthful public announcement could be made in w, and in the resulting state, $\top$ is trivially true. But if we interpret this formula in a forest model, it can be false. A simple example would be a forest model M' with only one state w' such that p is true in w', and there are no action successors of that state. So we have $M', w' \not\models_{\mathsf{fd}} [p]\top$, as the action relation corresponding to $[p]$ is empty; hence $M', w' \not\models_{\mathsf{fd}} p \rightarrow [p]\top$. This property of $\models_{\mathsf{fd}}$ may seem a bit strange. Essentially, this is because we do not yet enforce any connection of an announcement modality and an action in the forest model. We will see later (Definition 3.5) that formula $p \rightarrow [p]\top$ does hold in a special class of forest models that relate to this formula (see Example 3.10).

The reason to make a link between syntax and semantics is because later the syntactic translation from $\mathcal{L}_{\mathsf{DEL}}$ to $\mathcal{L}_{\mathsf{NTEL}}$ needs a correspondence so that we can associate it to the semantic transformation. Note that a similar point is made in [26] in state-dependent DEL protocols and in the context of their public announcement logic constrained with protocols.

Next, we define the meanings of the formulas with 'next-time' temporal operators in the following way.

Definition 2.9 (Semantics $\models_{\mathsf{ft}}$)
The semantics of a temporal formula $\bigcirc_{\mathsf{a}}\varphi$ on a forest model M is as follows:

$$M, w \models_{\mathsf{ft}} \bigcirc_{\mathsf{a}}\varphi \quad \text{iff} \quad \exists v \text{ s.t. } w \rightarrow_{\mathsf{a}} v \text{ and } M, v \models_{\mathsf{ft}} \varphi$$

Here, $\bigcirc$ stands for an existential branching next-time operator (see [27]), parameterized by actions a. The formula $\bigcirc_{\mathsf{a}}\varphi$ intuitively means that there exists a branch such that the transition to the next time point is labelled by a and φ is true there. According to this definition, $\bigcirc_{\mathsf{a}}\varphi \wedge \bigcirc_{\mathsf{a}}\neg\varphi$ is satisfiable, as one could imagine that there is a state w with two a successors in one of which φ is true and in the other φ is false. But we will show that this is not satisfiable in a special class of forest models, to be given in Definition 3.5.

Let $\mathcal{I}=(\mathcal{G},\mathcal{R},\{\rightarrow_{\mathsf{a}} | \mathsf{a}\in \mathsf{W}\},\pi)$ be an action-based interpreted system. 'Runs r and r' are equivalent up to time m' means that the initial segments of r and r' are the same from 0 to m, i.e. $\mathsf{r}(0)=\mathsf{r}'(0)$ up to $\mathsf{r}(m)=\mathsf{r}'(m)$. Choosing the bundle semantics as in [27], we now define the meaning of $\bigcirc_{\mathsf{a}}\varphi$ over an action-based interpreted system.

Definition 2.10 (Semantics $\models_{\mathsf{it}}$)
The semantics for $\bigcirc_{\mathsf{a}}\varphi$ on an action-based interpreted systems $\mathcal{I}$ is as follows,
$(\mathcal{I},\mathsf{r},m) \models_{\mathsf{it}} \bigcirc_{\mathsf{a}}\varphi$ iff there is a run r' that is equivalent to r up to time m, $(\mathsf{r}',m)\rightarrow_{\mathsf{a}} (\mathsf{r}',m+1)$ and $(\mathcal{I},\mathsf{r}',m+1) \models_{\mathsf{it}} \varphi$.

This shows a connection of action and time as in $(\mathsf{r}',m)\rightarrow_{\mathsf{a}}(\mathsf{r}',m+1)$. Note that action relation $\rightarrow_{\mathsf{a}}$ only relates two consecutive points in the same run, although it may seem to connect two points (r,m) and $(\mathsf{r}',m+1)$ in different runs. We show later that an action-based interpreted system is associated with a generated forest model, in which case a run corresponds to a branch in a forest model. Since an action 'a' may relate two worlds that belong to two branches in a forest model, there can be two runs going through the two corresponding global states, where an overlap occurs with '$\rightarrow_{\mathsf{a}}$'.

3 Public Announcements

In this section, we deal with the case of public announcement action models and the fragment of $\mathcal{L}_{\mathsf{DEL}}$ for public announcement, referred to as $\mathcal{L}_{\mathsf{PAL}}$. More concretely, $\mathcal{L}_{\mathsf{PAL}}$ is

$$\varphi \quad ::= \quad q \mid \neg\varphi \mid (\varphi\wedge\varphi) \mid K_i\varphi \mid C_B\varphi \mid [\varphi]\varphi$$

where $q\in Q$, $i\in Ag$, $B\subseteq Ag$.

Given a formula φ in $\mathcal{L}_{\mathsf{PAL}}$, and a pointed state model (M,w), we want to simulate checking the truth of φ in (M,w) by checking the truth of a corresponding next-time temporal epistemic formula in a corresponding action-based interpreted system. The action-based interpreted system should encode not only (M,w) but also the dynamics that is implicitly present in φ in the form of public announcement operators. It is therefore relative to both φ and (M,w). In other words, we are looking for a *syntactic translation* SYN (with type: $\mathcal{L}_{\mathsf{PAL}}\rightarrow\mathcal{L}_{\mathsf{NTEL}}$) and a *semantic transformation* SEM such that:

$$M,w \models_{\mathsf{sd}} \varphi \quad \text{iff} \quad \text{SEM}((M,w),\varphi) \models_{\mathsf{it}} \text{SYN}(\varphi).$$

The image of the actual world w under SEM (a global state s_w) is entirely determined by the role of w in M. It is therefore sufficient to determine $\text{SEM}(M,\varphi)$:

$$M,w \models_{\mathsf{sd}} \varphi \quad \text{iff} \quad \text{SEM}(M,\varphi), s_w \models_{\mathsf{it}} \text{SYN}(\varphi).$$

3.1 Syntactic translation

The $\mathcal{L}_{\mathsf{PAL}}$ formulas are translated to $\mathcal{L}_{\mathsf{NTEL}}$ formulas in the following way.

DEFINITION 3.1 ($\mathcal{L}_{\mathsf{PAL}}$ to $\mathcal{L}_{\mathsf{NTEL}}$)
Given that every announcement modality has a different action name, we define a translation SYN from $\mathcal{L}_{\mathsf{PAL}}$ to $\mathcal{L}_{\mathsf{NTEL}}$ as follows:

$$\begin{array}{lcl}
\mathrm{SYN}(q) & ::= & q \\
\mathrm{SYN}(\varphi \wedge \psi) & ::= & \mathrm{SYN}(\varphi) \wedge \mathrm{SYN}(\psi) \\
\mathrm{SYN}(\neg\varphi) & ::= & \neg \mathrm{SYN}(\varphi) \\
\mathrm{SYN}(K_i\varphi) & ::= & K_i \mathrm{SYN}(\varphi) \\
\mathrm{SYN}(C_B\varphi) & ::= & C_B \mathrm{SYN}(\varphi) \\
\mathrm{SYN}([\varphi]\psi) & ::= & \neg(\mathrm{SYN}(\varphi) \wedge \neg \bigcirc_{\mathsf{a}} \mathrm{SYN}(\psi))
\end{array}$$

where action a is the name of $[\varphi]$ within $[\varphi]\psi$.

The last clause associates the announcement modality $[\varphi]$ with the temporal modality $\bigcirc_{\mathsf{a}}$. It simulates the checking of an $\mathcal{L}_{\mathsf{PAL}}$ formula $[\varphi]\psi$ over a state model with semantics $\models_{\mathsf{sd}}$ (Definition 2.6) in the context of $\mathcal{L}_{\mathsf{NTEL}}$. Observe that $\neg(\mathrm{SYN}(\varphi) \wedge \neg \bigcirc_{\mathsf{a}} \mathrm{SYN}(\psi))$ is equivalent to the implication $\mathrm{SYN}(\varphi) \rightarrow \bigcirc_{\mathsf{a}} \mathrm{SYN}(\psi)$, and we will use the latter in the rest of our article due to its succinctness.

We assume that every announcement modality has a different action name, so even when two announcements are of the same formula, they still get different names. For this reason, we introduce a simple procedure to mark the announcement modalities so that they get a unique name. We mark the m announcements occurring in a formula with indexes from 1 to m in the order of occurrence of their left '[' bracket, when reading the formula from left to right. Then we associate each modality with index j with action a^j. Here is an example to explain this translation method.

EXAMPLE 3.2
Consider a formula $[q \wedge [r]K_2 r]C_{12} q \wedge [\top]\neg K_1 q$ with three announcements. We add indexes to the modalities as follows $[^1 q \wedge [^2 r]K_2 r]C_{12} q \wedge [^3\top]\neg K_1 q$, then associate them with three announcement variables as follows

$$\begin{array}{ll}
\mathsf{a}^1 & [^1 q \wedge [r]K_2 r] \\
\mathsf{a}^2 & [^2 r] \\
\mathsf{a}^3 & [^3 \top]
\end{array}$$

The translation $\mathrm{SYN}([q \wedge [r]K_2 r]C_{12} q \wedge [\top]\neg K_1 q)$ then is

$$((q \wedge (r \rightarrow \bigcirc_{\mathsf{a}^2} K_2 r)) \rightarrow \bigcirc_{\mathsf{a}^1} C_{12} q) \wedge (\top \rightarrow \bigcirc_{\mathsf{a}^3} \neg K_1 q).$$

The dynamics implicitly present in PAL formula φ can be *identified* with the set of all sequences of public announcements that may need to be evaluated in order to determine the truth of φ. As this is known as a protocol [16, 26], we call this the *protocol of a formula* φ. It can be determined from φ and is therefore another syntactic feature that we can address before applying it in the semantic transformation $\mathrm{SEM}((M, w), \varphi)$.

Definition 3.3 (Protocol of PAL formula)
The *protocol of a* PAL *formula* is defined by induction on the formula structure.

$$\begin{array}{lcl}
\textsc{prot}(q) & ::= & \emptyset \\
\textsc{prot}(\neg\varphi) & ::= & \textsc{prot}(\varphi) \\
\textsc{prot}(\varphi\wedge\psi) & ::= & \textsc{prot}(\varphi)\cup\textsc{prot}(\psi) \\
\textsc{prot}(K_i\varphi) & ::= & \textsc{prot}(\varphi) \\
\textsc{prot}(C_B\varphi) & ::= & \textsc{prot}(\varphi) \\
\textsc{prot}([\varphi]\psi) & ::= & \textsc{prot}(\varphi)\cup \mathsf{a}\textsc{prot}(\psi)
\end{array}$$

where, a is the name for the announcement of φ in $[\varphi]\psi$, and $\mathsf{a}\textsc{prot}(\psi)=\{\mathsf{a}\}$ if $\textsc{prot}(\psi)$ is empty, otherwise $\mathsf{a}\textsc{prot}(\psi)=\{\mathsf{a}\mathsf{a}^1\ldots\mathsf{a}^m \mid \mathsf{a}^1\ldots\mathsf{a}^m\in\textsc{prot}(\psi)\}$, i.e. the concatenation of a to all sequences in the non-empty set $\textsc{prot}(\psi)$. For a protocol variable we use P.

Note that a 'protocol' in this article is used differently from that in [26], where a DEL protocol is defined as a set of sequences of pointed event models closed under an initial segment relation. In our case, $\textsc{prot}(\psi)$ is not closed under an initial segment relation. The protocol of a formula would be closed under an initial segment relation if the last clause is changed to $\textsc{prot}([\varphi]\psi)::=\textsc{prot}(\varphi)\cup\mathsf{a}\textsc{prot}(\psi)\cup\{\mathsf{a}\}$. In the proposition linking DEL and TEL in [26], they assume the protocol of all finite sequences of DEL event models. We, instead, take a minimalistic approach, only reading a protocol off φ which is needed for model checking.

Example 3.4
We have that $\textsc{prot}([q][r](q\wedge r)\wedge[r]K_1r)=\{\mathsf{a}^1\mathsf{a}^2,\mathsf{a}^3\}$, and that $\textsc{prot}([q\wedge[r]K_2r]C_{12}q\wedge[\top]\neg K_1q)=\{\mathsf{a}^1,\mathsf{a}^2,\mathsf{a}^3\}$.

3.2 Semantic transformation

The required semantic transformation SEM in $\textsc{sem}(M,\varphi)$ is determined in two steps. First, we construct the *forest model* $\textsc{f}(M,\textsc{prot}(\varphi))$ from the state model M and the protocol $\textsc{prot}(\varphi)$ of the public announcement formula φ in a similar way as in [24, 26]. Then we determine an action-based interpreted system $\textsc{is}(M')$ corresponding to a forest model M'. We then simply define $\textsc{sem}(M,\varphi)$ as $\textsc{is}(\textsc{f}(M,\textsc{prot}(\varphi)))$. Figure 2 summarizes the syntactic translation and the semantic transformation.

Definition 3.5 (Generated Forest Models)
Given a state model $M=\langle W,\{\sim_i \mid i\in Ag\},\pi\rangle$, $w\in W$, and a protocol $\mathsf{P}=\textsc{prot}(\varphi)$ generated from PAL formula φ, a generated forest model $\textsc{f}(M,\mathsf{P})$ is defined in three steps.

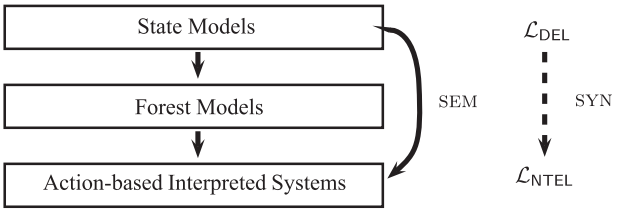


Fig. 2. Syntactic translation and semantic transformation.

(1) Let $\mathsf{a}^1 \cdots \mathsf{a}^m$ be a sequence of actions in protocol P, and suppose that these actions belong to public announcement models $\mathsf{A}_1, \mathsf{A}_2, \cdots, \mathsf{A}_m$ respectively. Let M_j be the state model $M \otimes \mathsf{A}_1 \cdots \otimes \mathsf{A}_j$, which is the result of announcing a^1 to a^j subsequently on M; let $M_0 = M$. Then $g(M, \mathsf{a}^1 \cdots \mathsf{a}^m)$ is a forest model $M' = \langle W', \{\sim'_i \mid i \in Ag\}, \{\rightarrow'_{\mathsf{a}}\}, \pi' \rangle$ where

- $W' = W_M \cup W_{M_1} \cup \cdots \cup W_{M_m}$ i.e. the set of states obtained from subsequent updates by announcements;
- $\sim'_i = \bigcup_{j \in [0..m]} \sim^j_i$, where $\sim^j_i$ is the epistemic relation of agent i in model M_j;
- $w \rightarrow_{\mathsf{a}} (w, \mathsf{a})$ for any $w, (w, \mathsf{a}) \in W'$;
- $\pi'(w) = \pi(w)$ for the unique M_k such that $w \in W_{M_k}$ & π belongs to M_k. Note that the uniqueness is guaranteed by the fact that $W_M, W_{M_1}, \cdots$, and W_{M_m} form a partition of W'.

(2) Define a union $\uplus$ of two forest models. Given forest model $M' = \langle W', \{\sim'_i \mid i \in Ag\}, \{\rightarrow'_{\mathsf{a}}\}, \pi' \rangle$, and $M'' = \langle W'', \{\sim''_i \mid i \in Ag\}, \{\rightarrow''_{\mathsf{a}}\}, \pi'' \rangle$,

$$M' \uplus M'' ::= \langle W''', \{\sim'''_i \mid i \in Ag\}, \{\rightarrow'''_{\mathsf{a}}\}, \pi''' \rangle$$

where $W''' = W' \cup W''$, $\sim'''_i = \sim'_i \cup \sim''_i$ for all $i \in Ag$, $\rightarrow'''_{\mathsf{a}} = \rightarrow'_{\mathsf{a}} \cup \rightarrow''_{\mathsf{a}}$ for all a, and $\pi'''(w) = \pi'(w) \cup \pi''(w)$ for all $w \in W' \cap W''$, $\pi'''(w) = \pi'(w)$ for all $w \in W' \setminus W''$, $\pi'''(w) = \pi''(w)$ for all $w \in W'' \setminus W'$.

(3) Finally $\mathrm{F}(M, \mathrm{PROT}(\varphi)) ::= \uplus_{\tau \in \mathrm{PROT}(\varphi)} g(M, \tau)$.

The construction can be seen as building a forest model by repeatedly merging a state model and the modal product of that model and a singleton 'action model' corresponding to an announcement, and then unifying these forest models. We refer to the next section for an example illustrating this procedure.

Next, from such a forest model we determine an action-based interpreted system. This is based on a fairly intuitive idea. Each *world* in a forest model is associated with a *global state*. This can be achieved by keeping that world as the value of the *environmental state* and for each agent the set of indistinguishable worlds as the value of that agent's *local state*. The valuation π remains as it was. For a world w in a state model $M = \langle W, \{\sim_i \mid i \in Ag\}, \pi \rangle$ this recipe delivers a corresponding global state $s = (w, w^{\sim_1}, \ldots, w^{\sim_n})$, where $w^{\sim_i}$ is the i-equivalence class containing w, i.e. $\{w' \in W \mid w' \sim_i w\}$. The same recipe applies, in principle, to worlds $(w, \mathsf{a}^1, \ldots, \mathsf{a}^m)$ in the forest model $\mathrm{F}(M, \mathrm{PROT}(\varphi))$, but here we can somewhat simplify matters by observing that (i) the environment is fully determined by the w in $(w, \mathsf{a}^1, \ldots, \mathsf{a}^m)$ because all events (such as announcements) are defined relative to their combined effect on the agents *only*, and by observing that (ii) public announcements are fully observable by all agents so we can represent them as global parameters. In the following we use $(w, w^{\sim_1}, \ldots, w^{\sim_n}, \mathsf{a}^1, \ldots, \mathsf{a}^m)$ to denote the global state

$$((w, \mathsf{a}^1, \ldots, \mathsf{a}^m), (w, \mathsf{a}^1, \ldots, \mathsf{a}^m)^{\sim_1}, \ldots, (w, \mathsf{a}^1, \ldots, \mathsf{a}^m)^{\sim_n}).$$

Definition 3.6 (Forest Model to Action-based Interpreted System)
Given a forest model $M = \langle W, \{\sim_i \mid i \in Ag\}, \{\rightarrow_{\mathsf{a}} \mid \mathsf{a} \in \mathsf{W}\}, \pi \rangle$, we associate M with an action-based interpreted system $\mathcal{I}$, also written as $\mathrm{IS}(M)$, which is a structure $\langle \mathcal{G}, \mathcal{R}, \{\rightarrow_{\mathsf{a}} \mid \mathsf{a} \in \mathsf{W} \cup \{\mathsf{a}_\perp\}\}, \pi \rangle$ defined as follows.

Every state $(w, \mathsf{a}^1, \ldots, \mathsf{a}^m)$ in the forest model M corresponds to a global state $(w, w^{\sim_1}, \ldots, w^{\sim_n}, \mathsf{a}^1, \ldots, \mathsf{a}^m)$, where the local state of environment is w and the local state of agent i is $(w^{\sim_i}, \mathsf{a}^1, \ldots, \mathsf{a}^m)$. For each state in forest model M that for no $\mathsf{a} \in \mathsf{W}$ has a $\rightarrow_{\mathsf{a}}$ successor, we define a run $\mathsf{r} \in \mathcal{R}$. Suppose $(w, \mathsf{a}^1, \ldots, \mathsf{a}^k)$ is such a state in M, then the associated run r is defined as follows:

- $\mathsf{r}(0) = (w, w^{\sim_1}, \ldots, w^{\sim_n})$;
- $\mathsf{r}(i) = (w, w^{\sim_1}, \ldots, w^{\sim_n}, \mathsf{a}^1, \ldots, \mathsf{a}^i)$ for all $1 \leq i \leq k$; $\mathsf{r}(i) = \mathsf{r}(i-1)$, otherwise;
- $\mathsf{r}(i-1) \rightarrow_{\mathsf{a}^i} \mathsf{r}(i)$ for all $1 \leq i \leq k$; $\mathsf{r}(i-1) \rightarrow_{\mathsf{a}_\perp} \mathsf{r}(i)$, otherwise.

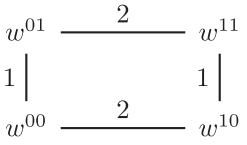
The valuations correspond: $\pi(\mathsf{r}(i)) = \pi(w)$, i.e. all the states in a run have the same valuation.

It is easy to see that each run is essentially a branch of the corresponding forest model. Since a branch of a forest mode is finite but a run is infinite, we introduce a special action $\mathsf{a}_\perp$ to represent the action of doing nothing, which does not change any global state.

3.3 Example

We illustrate the syntactic translation and the semantic transformations defined in the previous sections through the following example.

Consider two agents 1 and 2 and two facts q and r. Agent 1 knows whether q but is uncertain about the truth of r, whereas agent 2 knows whether r but is uncertain about the truth of q. The agents are commonly aware of each other's factual knowledge and ignorance. In fact, both q and r are true. This is modelled by the following state model (call it M_{init}).



We have named the four states of the model w^{00}, w^{10}, w^{01}, and w^{11}, where the index reveals the valuation of atoms q and r (in that order), e.g. to state w^{01}, we give $\pi(w^{01}) = \{r\}$. Accessibility relations are represented by lines labelled with agents' names. For example, agent 1 cannot distinguish states w^{00} and w^{01}, i.e. $w^{00} \sim_1 w^{01}$, so two states are linked by a line with label '1'.

Consider checking whether $M_{init}, w^{11} \models_{\mathsf{sd}} [q][r](q \wedge r) \wedge [r]K_1 r$. One could associate this formula with an indexed version $[^1q][^2r](q \wedge r) \wedge [^3r]K_1 r$ we proposed earlier, and the action variables $\mathsf{a}^1, \mathsf{a}^2$ and a^3 represent the three different announcements in this formula. Note that the first and second announcement r are named differently. The protocol $\textsc{prot}([q][r](q \wedge r) \wedge [r]K_1 r)$ is $\{\mathsf{a}^1\mathsf{a}^2, \mathsf{a}^3\}$.

We now apply the procedure introduced in Definition 3.5, and construct the forest model $\textsc{f}(M_{\text{init}}, \textsc{prot}([q][r](q \wedge r) \wedge [r]K_1 r))$ as follows.

First, consider $[q]$. An announcement of q results in a new model M_1 with two states (w^{11}, a^1) and (w^{10}, a^1). In the resulting model, agent 1 is still uncertain about r, but agent 2 now knows the value of q. After the announcement of q, in the sub-formula $[q][r](q \wedge r)$, atom r is subsequently announced, resulting in another state model M_2, which consists a single state $(w^{11}, \mathsf{a}^1, \mathsf{a}^2)$. In this model, both agents know that q and r are true. Now we

consider the second r. It is announced in the initial model and results in a third model M_3 with two states (w^{01},a^3) and (w^{11},a^3). In this model, agent 2 is still uncertain about q, but agent 1 knows whether q. Depicting the results of all three announcements at the same time in a matrix with 3 rows and 4 columns, we get

$$(w^{01},\mathsf{a}^3) \overset{2}{\text{———}} (w^{11},\mathsf{a}^3)$$

$$\begin{array}{lll} w^{01} \overset{2}{\text{———}} w^{11} & (w^{11},\mathsf{a}^1) & (w^{11},\mathsf{a}^1,\mathsf{a}^2) \\ 1 \,\big|\quad\quad 1\,\big| & 1\,\big| & \\ w^{00} \overset{2}{\text{———}} w^{10} & (w^{10},\mathsf{a}^1) & \end{array}$$

where M_1, with states (w^{11},a^1) and (w^{10},a^1), is in column 3 (counting from left to right), M_2 is in column 4 and M_3 is in row 1 (counting from top to bottom).

$\mathrm{F}(M_{\text{init}},\mathsf{a}^1\mathsf{a}^2)$ is depicted in the row 2 and 3 with seven states in total, and $\mathrm{F}(M_{\text{init}},\mathsf{a}^3)$ is depicted in the column 1 and 2 with six states in total. Easy to see that the common states of two forests above are exactly those of the initial model M_{init}. Merging these two forests results in $\mathrm{F}(M_{\text{init}},\mathsf{a}^1\mathsf{a}^2)\uplus\mathrm{F}(M_{\text{init}},\mathsf{a}^3)$.

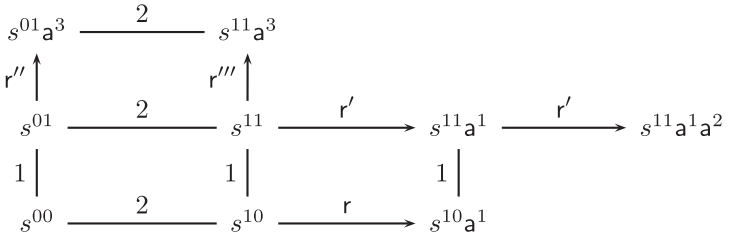
$$\begin{array}{llll} (w^{01},\mathsf{a}^3) \overset{2}{\text{———}} (w^{11},\mathsf{a}^3) & & & \\ \mathsf{a}^3\uparrow \quad\quad \mathsf{a}^3\uparrow & & & \\ w^{01} \overset{2}{\text{———}} w^{11} \xrightarrow{\mathsf{a}^1} (w^{11},\mathsf{a}^1) \xrightarrow{\mathsf{a}^2} (w^{11},\mathsf{a}^1,\mathsf{a}^2) & & & \\ 1\,\big|\quad\quad 1\,\big| & 1\,\big| & & \\ w^{00} \overset{2}{\text{———}} w^{10} \xrightarrow{\mathsf{a}^1} (w^{10},\mathsf{a}^1) & & & \end{array}$$

We now associate an action-based interpreted system with the forest model just given, following Definition 3.6. The above forest model consists of four trees with the roots w^{11}, w^{00}, w^{01} and w^{10}, and five states that have no action successors: (w^{10},a^1), $(w^{11},\mathsf{a}^1,\mathsf{a}^2)$, (w^{01},a^3) and (w^{11},a^3) and w^{00}.

The global state $(w^{10},\{w^{10},w^{11}\},\{w^{00},w^{10}\})$ is associated with the state w^{10}, and $(w^{10},\{w^{10},w^{11}\},\{w^{10}\},\mathsf{a}^1)$ (i.e. $(w^{10},\{(w^{10},\mathsf{a}^1),(w^{11},\mathsf{a}^1)\},\{(w^{10},\mathsf{a}^1)\})$) is associated with the state (w^{10},a^1) in the forest model, etc. Write s^{10} for the former global state and $s^{10}\mathsf{a}^1$ for the latter. The accessibility relations for agent 1 and 2 remain the same. Instead of action-labelled transitions we now have runs connecting the global states. There are five runs, (arbitrarily) named

$$\begin{array}{ll} \mathsf{r} & (s^{10},s^{10}\mathsf{a}^1,s^{10}\mathsf{a}^1,\cdots) \\ \mathsf{r}' & (s^{11},s^{11}\mathsf{a}^1,s^{11}\mathsf{a}^1\mathsf{a}^2,s^{11}\mathsf{a}^1\mathsf{a}^2,\cdots) \\ \mathsf{r}'' & (s^{01},s^{01}\mathsf{a}^3,s^{01}\mathsf{a}^3,\cdots) \\ \mathsf{r}''' & (s^{11},s^{11}\mathsf{a}^3,s^{11}\mathsf{a}^3,\cdots) \\ \mathsf{r}'''' & (s^{00},s^{00},\cdots) \end{array}$$

This interpreted system can now be depicted as (with repeated states being omitted)



The translation SYN of the formula $[q][r](q\wedge r)\wedge[r]K_1 r$, was, as we have already seen, $(q\rightarrow\bigcirc_{\mathsf{a}^1}(r\rightarrow\bigcirc_{\mathsf{a}^2}(q\wedge r)))\wedge(r\rightarrow\bigcirc_{\mathsf{a}^3}K_1 r)$. We can verify that both $M_{\text{init}}, w^{11}\models_{\mathsf{sd}}[q][r](q\wedge r)\wedge[r]K_1 r$ and $\text{IS}(M_{\text{init}},\text{PROT}([q][r](q\wedge r)\wedge[r]K_1 r)), s^{11}\models_{\mathsf{it}}(q\rightarrow\bigcirc_{\mathsf{a}^1}(r\rightarrow\bigcirc_{\mathsf{a}^2}(q\wedge r)))\wedge(r\rightarrow\bigcirc_{\mathsf{a}^3}K_1 r)$ hold.

This example explains both the syntactic translation and semantic transformation. In next section, we show their tight connection with ' $\models_{\mathsf{sd}}$ ' and ' $\models_{\mathsf{it}}$ '.

3.4 Theoretical results

We show, in three steps, the equivalence

$$M, w\models_{\mathsf{sd}}\varphi \text{ iff } \text{SEM}(M,\varphi), s_w\models_{\mathsf{it}}\text{SYN}(\varphi).$$

The *first step* is to show that given a state model M and a PAL formula φ, the interpretation of φ over (M,w) is equivalent to its interpretation over $\text{F}(M,\text{PROT}(\varphi))$ which is the forest model built from M and φ. The *second step* is to show that φ and its syntactic translation $\text{SYN}(\varphi)$ are equivalent when they are both interpreted over the forest model $\text{F}(M,\text{PROT}(\varphi))$. The *third, last, step* is to show that the interpretation of $\text{SYN}(\varphi)$ over an arbitrary forest model and its corresponding interpreted system are equivalent. We explain these steps in three propositions: Proposition 3.9, Proposition 3.11 and Proposition 3.12.

All the proofs are omitted here, as we will generalize these propositions in Section 4.1 and give full proofs there.

We first give a lemma about some important features of generated forest models.

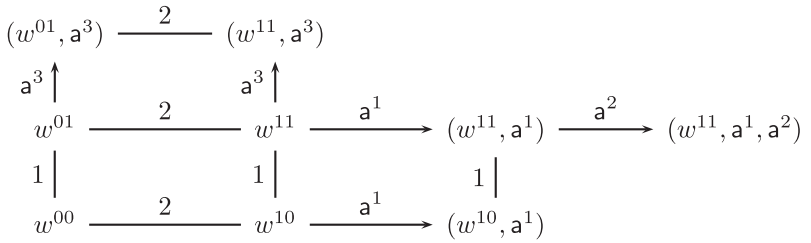
LEMMA 3.7
Given a state model M and PAL formulas φ,ψ, the following equivalences hold:

(i) $\text{F}(M,\text{PROT}(\varphi\wedge\psi)), w\models_{\mathsf{fd}}\varphi$ iff $\text{F}(M,\text{PROT}(\varphi)), w\models_{\mathsf{fd}}\varphi$;
(ii) $\text{F}(M,\text{PROT}([\varphi]\psi)), w\models_{\mathsf{fd}}\varphi$ iff $\text{F}(M,\text{PROT}(\varphi)), w\models_{\mathsf{fd}}\varphi$;
(iii) $\text{F}(M,\text{PROT}(\varphi\wedge\psi)), w\models_{\mathsf{ft}}\text{SYN}(\varphi)$ iff $\text{F}(M,\text{PROT}(\varphi)), w\models_{\mathsf{ft}}\text{SYN}(\varphi)$;
(iv) $\text{F}(M,\text{PROT}([\varphi]\psi)), w\models_{\mathsf{ft}}\text{SYN}(\varphi)$ iff $\text{F}(M,\text{PROT}(\varphi)), w\models_{\mathsf{ft}}\text{SYN}(\varphi)$;

This lemma shows special properties of the forest models generated from a state model and a PAL formula. In the case of formula $\varphi\wedge\psi$ and $[\varphi]\psi$, the truth value of φ is solely decided by the forest model $\text{F}(M,\text{PROT}(\varphi))$, which is a sub-model of both $\text{F}(M,\text{PROT}(\varphi\wedge\psi))$ and $\text{F}(M,\text{PROT}([\varphi]\psi))$.

We give the following example to explain the idea more intuitively.

EXAMPLE 3.8
As we have shown in Section 3.3, the forest model $\mathrm{F}(M_{\mathrm{init}}, \mathrm{PROT}([q][r](q \wedge r) \wedge [r]K_1 r))$ is as follows,



As a sub-model, the forest model $\mathrm{F}(M_{\mathrm{init}}, \mathrm{PROT}([q][r](q \wedge r)))$ consists of all the states in the lower two rows, and the forest model $\mathrm{F}(M_{\mathrm{init}}, \mathrm{PROT}([r]K_1 r))$ consists of all the states in the first and second columns. Clearly, the common states of these two forest models are w^{01}, w^{11}, w^{00} and w^{10}, which are exactly those states in model M_{init}. We evaluate the second conjunct of $[q][r](q \wedge r) \wedge [r]K_1 r$, namely $[r]K_1 r$, in the state w^{11} of the model $\mathrm{F}(M_{\mathrm{init}}, \mathrm{PROT}([q][r](q \wedge r)) \wedge K_1 r)$. It is easy to verify that r is true in w^{11} and there is an a^3-successor (w^{11}, a^3) in which $K_1 r$ is true. Since all a^3-successors can only be included in the forest $\mathrm{F}(M_{\mathrm{init}}, \mathrm{PROT}([r]K_1 r))$ and there are no epistemic links to the rest of the states, we conclude that the evaluation of $[r]K_1 r$ in the state w^{11} of the model $\mathrm{F}(M_{\mathrm{init}}, \mathrm{PROT}([r]K_1 r))$ is the same.

We can do a similar analysis for the evaluation of $\mathrm{SYN}([r]K_1 r))$, i.e. $r \to \bigcirc_{\mathsf{a}^3} K_1 r$, in model $\mathrm{F}(M_{\mathrm{init}}, \mathrm{PROT}([q][r](q \wedge r)))$.

PROPOSITION 3.9
Let M be a state model and $\varphi \in \mathcal{L}_{\mathsf{PAL}}$.

$$M, w \models_{\mathsf{sd}} \varphi \quad \text{iff} \quad \mathrm{F}(M, \mathrm{PROT}(\varphi)), w \models_{\mathsf{fd}} \varphi$$

This result shows that we can either evaluate a PAL formula in a state model, or alternatively construct a 'supermodel' that already contains all future dynamic developments labelled by actions. Our formulation, relative to a formula φ to be evaluated, is different from the standard semantic form. For a description of the technique see Venema's chapter 'Dynamic Models in their Logical Surroundings' (in particular page 122) in [25], or [24, 26].

As the following example shows, the 'special class of forest models' mentioned in Example 2.8 is in fact $\mathrm{F}(M, \mathrm{PROT}(\varphi))$. This also gives an intuitive explanation of why Proposition 3.9 holds.

EXAMPLE 3.10
Consider again the formula $p \to [p]\top$. On the one hand, for any state model M and state w, we always have $M, w \models_{\mathsf{sd}} p \to [p]\top$. On the other hand, as Example 2.8 shows, we can not guarantee that given any forest model M' and state w', we always have $M', w' \models_{\mathsf{fd}} p \to [p]\top$. But we do guarantee that for the forest model generated from M and $p \to [p]\top$, we have $\mathrm{F}(M, \mathrm{PROT}(p \to [p]\top)), w \models_{\mathsf{fd}} p \to [p]\top$. The reason is as follows. Assume that $[p]$ corresponds to action a_0. Easy to derive $\mathrm{PROT}(p \to [p]\top) = \{\mathsf{a}_0\}$. If $M, w \models_{\mathsf{sd}} p$, then the public announcement of p can be made on w; therefore there is a world (w, a_0) in $\mathrm{F}(M, \mathrm{PROT}(p \to [p]\top))$ such that $w \to_{\mathsf{a}_0} (w, \mathsf{a}_0)$ and $\mathrm{F}(M, \mathrm{PROT}(p \to [p]\top)), w \models_{\mathsf{fd}} [p]\top$. So the claim holds. The case for $M, w \not\models_{\mathsf{sd}} p$ is trivial.

The next result says that a formula $\varphi \in \mathcal{L}_{\mathsf{PAL}}$ and its translation $\textsc{syn}(\varphi) \in \mathcal{L}_{\mathsf{NTEL}}$ are equivalent when they are interpreted over the same forest model.

PROPOSITION 3.11
Given a state model M and a PAL formula φ:

$$\textsc{f}(M, \textsc{prot}(\varphi)), w \models_{\mathsf{fd}} \varphi \quad \text{iff} \quad \textsc{f}(M, \textsc{prot}(\varphi)), w \models_{\mathsf{ft}} \textsc{syn}(\varphi)$$

We now turn to the third result.

PROPOSITION 3.12
For every *executable* $\varphi \in \mathcal{L}_{\mathsf{NTEL}}$ (i.e. a formula of the form $\textsc{syn}(\psi)$ with $\psi \in \mathcal{L}_{\mathsf{PAL}}$), and every forest model M constructed from ψ with some initial state model, we have:

$$M, w \models_{\mathsf{ft}} \varphi \quad \text{iff} \quad \textsc{is}(M), (w, w^{\sim_1}, \ldots, w^{\sim_n}) \models_{\mathsf{it}} \varphi$$

We now have the main result from Propositions 3.9, 3.11 and 3.12.

THEOREM 3.13
Given a state model M, and a PAL formula φ,

$$M, w \models_{\mathsf{sd}} \varphi \quad \text{iff} \quad \textsc{sem}(M, \varphi), s_w \models_{\mathsf{it}} \textsc{syn}(\varphi)$$

4 Generalization

We now generalize the results in the previous section from public announcements as in $[\varphi]\psi$ to action models as in $[\mathsf{A}, \mathsf{a}]\psi$.

DEFINITION 4.1 ($\mathcal{L}_{\mathsf{DEL}}$ to $\mathcal{L}_{\mathsf{NTEL}}$)
We define a translation SYN from $\mathcal{L}_{\mathsf{DEL}}$ to $\mathcal{L}_{\mathsf{NTEL}}$ as follows:

$$\begin{array}{lcl}
\textsc{syn}(q) & ::= & q \\
\textsc{syn}(\varphi \wedge \psi) & ::= & \textsc{syn}(\varphi) \wedge \textsc{syn}(\psi) \\
\textsc{syn}(\neg\varphi) & ::= & \neg\textsc{syn}(\varphi) \\
\textsc{syn}(K_i\varphi) & ::= & K_i\textsc{syn}(\varphi) \\
\textsc{syn}(C_B\varphi) & ::= & C_B\textsc{syn}(\varphi) \\
\textsc{syn}([\mathsf{A}, \mathsf{a}]\psi) & ::= & \neg(\textsc{syn}(\mathsf{pre}(\mathsf{a})) \wedge \neg \bigcirc_{\mathsf{a}} \textsc{syn}(\psi))
\end{array}$$

It is easy to see that the clause for public announcement (see Definition 3.1) is a special case.

We then have the protocol of a DEL formula.

DEFINITION 4.2 (Protocol of DEL formula)
The protocol of a DEL formula is defined by induction on the formula structure. The cases of $q, \neg\varphi, \varphi \wedge \psi, K_i\varphi, C_B\varphi$ are the same as in Definition 3.3.

$$\textsc{prot}([\mathsf{A}, \mathsf{a}]\psi) \quad ::= \quad \bigcup_{\mathsf{b} \in \mathcal{D}(\mathsf{A})} (\textsc{prot}(\mathsf{pre}(\mathsf{b})) \cup \mathsf{b}\textsc{prot}(\psi))$$

where $\mathcal{D}(\mathsf{A})$ is the domain of the action model, which includes the point a, and $\mathsf{b}\textsc{prot}(\psi) = \{\mathsf{b}\}$ if $\textsc{prot}(\psi)$ is empty, otherwise, $\mathsf{b}\textsc{prot}(\psi) = \{\mathsf{b}\mathsf{a}^1 \ldots \mathsf{a}^m \mid \mathsf{a}^1 \ldots \mathsf{a}^m \in \textsc{prot}(\psi)\}$, i.e. the concatenation of b to all sequences in the set of $\textsc{prot}(\psi)$.

Here, we take a union for all $\mathsf{b} \in \mathcal{D}(\mathsf{A})$ because they will all be used in building the forest models that will be introduced shortly. It is easy to see that Definition 3.3 is a special case, as the public announcement model has a singleton domain.

Next, we generalize Definition 3.5 as follows.

DEFINITION 4.3 (Generated Forest Models: the General Case)
Given a state model $M = \langle W, \{\sim_i | i \in Ag\}, \pi\rangle$, $w \in W$, and a protocol $\mathsf{P} = \text{PROT}(\varphi)$ generated from **DEL** formula φ. The forest model $\text{F}(M, \mathsf{P})$ is defined in three steps.

(1) Let $\mathsf{a}^1 \cdots \mathsf{a}^m$ be a sequence of actions in protocol P, and suppose that these actions belongs to action models $\mathsf{A}_1, \mathsf{A}_2, \cdots, \mathsf{A}_m$, respectively. Let M_j be the state model $M \otimes \mathsf{A}_1 \cdots \otimes \mathsf{A}_j$, which is the result of updating A_1 to A_j subsequently on M; let $M_0 = M$. Then $g(M, \mathsf{a}^1 \cdots \mathsf{a}^m)$ is a forest model $M' = \langle W', \{\sim'_i | i \in Ag\}, \{\rightarrow'_{\mathsf{a}}\}, \pi'\rangle$ where

- $W' = W_M \cup W_{M_1} \cup \cdots \cup W_{M_m}$ i.e. the set of states from subsequent updates;
- $\sim'_i = \bigcup_{j \in [0..m]} \sim^j_i$, where $\sim^j_i$ is the epistemic relation of agent i in model M_j;
- $w \rightarrow_{\mathsf{a}} (w, \mathsf{a})$ for any $w, (w, \mathsf{a}) \in W'$;
- $\pi'(w) = \pi(w)$ for the unique M_k such that $w \in W_{M_k}$ & π belongs to M_k. Note that the uniqueness is guaranteed by the fact that $W_M, W_{M_1}, \cdots$ and W_{M_m} form a partition of W'.

(2) We define a union $\uplus$ of two forest models. Given forest model $M' = \langle W', \{\sim'_i | i \in Ag\}, \{\rightarrow'_{\mathsf{a}}\}, \pi'\rangle$, and $M'' = \langle W'', \{\sim''_i | i \in Ag\}, \{\rightarrow''_{\mathsf{a}}\}, \pi''\rangle$,

$$M' \uplus M'' ::= \langle W''', \{\sim'''_i | i \in Ag\}, \{\rightarrow'''_{\mathsf{a}}\}, \pi'''\rangle$$

where $W''' = W' \cup W''$, $\sim'''_i = \sim'_i \cup \sim''_i$ for all $i \in Ag$, $\rightarrow'''_{\mathsf{a}} = \rightarrow'_{\mathsf{a}} \cup \rightarrow''_{\mathsf{a}}$ for all a, and $\pi'''(w) = \pi'(w) \cup \pi''(w)$ for all $w \in W' \cap W''$, $\pi'''(w) = \pi'(w)$ for all $w \in W' \setminus W''$, $\pi'''(w) = \pi''(w)$ for all $w \in W'' \setminus W'$.

(3) Finally $\text{F}(M, \text{PROT}(\varphi)) ::= \uplus_{\tau \in \text{PROT}(\varphi)} g(M, \tau)$.

Definition 3.6, on the transformation from forest models to action-based interpreted systems, does not need to be generalized, as we do not put any special restriction on forest models in that definition.

4.1 Theoretical results

This section generalizes the results in Section 3.4 and presents detailed proofs. We first give a generalization of Lemma 3.7.

LEMMA 4.4
Given a state model M and **DEL** formulas φ, ψ, the following equivalences hold:

(i) $\text{F}(M, \text{PROT}(\varphi \wedge \psi)), w \models_{\mathsf{fd}} \varphi$ iff $\text{F}(M, \text{PROT}(\varphi)), w \models_{\mathsf{fd}} \varphi$;
(ii) $\text{F}(M, \text{PROT}([\mathsf{A}, \mathsf{a}]\psi)), w \models_{\mathsf{fd}} \mathsf{pre}(\mathsf{a})$ iff $\text{F}(M, \text{PROT}(\mathsf{pre}(\mathsf{a}))), w \models_{\mathsf{fd}} \mathsf{pre}(\mathsf{a})$;
(iii) $\text{F}(M, \text{PROT}(\varphi \wedge \psi)), w \models_{\mathsf{ft}} \text{SYN}(\varphi)$ iff $\text{F}(M, \text{PROT}(\varphi)), w \models_{\mathsf{ft}} \text{SYN}(\varphi)$;
(iv) $\text{F}(M, \text{PROT}([\mathsf{A}, \mathsf{a}]\psi)), w \models_{\mathsf{ft}} \text{SYN}(\mathsf{pre}(\mathsf{a}))$ iff $\text{F}(M, \text{PROT}(\mathsf{pre}(\mathsf{a}))), w \models_{\mathsf{ft}} \text{SYN}(\mathsf{pre}(\mathsf{a}))$;

PROOF. Let a state model M and **DEL** formulas φ, ψ be given.

Case i: we first prove the direction $\Rightarrow$.

Suppose $\mathrm{F}(M, \mathrm{PROT}(\varphi \wedge \psi)), w \models_{\mathsf{fd}} \varphi$, we have $\mathrm{F}(M, \mathrm{PROT}(\varphi) \cup \mathrm{PROT}(\psi)), w \models_{\mathsf{fd}} \varphi$. It follows that $\mathrm{F}(M, \mathrm{PROT}(\varphi)) \uplus \mathrm{F}(M, \mathrm{PROT}(\psi)), w \models_{\mathsf{fd}} \varphi$. Suppose the domain of $\mathrm{F}(M, \mathrm{PROT}(\varphi))$ is W_1, and that of $\mathrm{F}(M, \mathrm{PROT}(\psi))$ is W_2, then according to the forest model construction in Definition 4.3, we have $W_1 \cap W_2 = W_M$, i.e. the set of common states between these two forests is the set of the states in model M.

There are two cases for formula φ: either it contains no action modalities (then its truth value can be solely decided by the states in M), or it contains action modalities that correspond only to the actions in $\mathrm{F}(M, \mathrm{PROT}(\varphi))$, therefore its truth value can be decided solely by $\mathrm{F}(M, \mathrm{PROT}(\varphi))$. In both cases, the truth value of φ can be solely decided in $\mathrm{F}(M, \mathrm{PROT}(\varphi))$, so we have $\mathrm{F}(M, \mathrm{PROT}(\varphi)), w \models_{\mathsf{fd}} \varphi$.

The direction $\Leftarrow$ follows from a similar reasoning.

Case ii: for the direction $\Rightarrow$, suppose $\mathrm{F}(M, \mathrm{PROT}([\mathsf{A}, \mathsf{a}]\psi)), w \models_{\mathsf{fd}} \mathsf{pre}(\mathsf{a})$. We have $\mathrm{F}(M, \bigcup_{\mathsf{b} \in \mathcal{D}(\mathsf{A})} (\mathrm{PROT}(\mathsf{pre}(\mathsf{b})) \cup \mathsf{b}\mathrm{PROT}(\psi))), w \models_{\mathsf{fd}} \mathsf{pre}(\mathsf{a})$. It is easy to see that $\mathrm{F}(M, \mathrm{PROT}(\mathsf{pre}(\mathsf{a})))$ is a sub-model of $\mathrm{F}(M, \bigcup_{\mathsf{b} \in \mathcal{D}(\mathsf{A})} (\mathrm{PROT}(\mathsf{pre}(\mathsf{b})) \cup \mathsf{b}\mathrm{PROT}(\psi)))$.

We distinguish two cases of $\mathsf{pre}(\mathsf{a})$: either $\mathsf{pre}(\mathsf{a})$ contains no action modalities, or it contains action modalities that correspond only to the actions in $\mathrm{F}(M, \mathrm{PROT}(\mathsf{pre}(\mathsf{a})))$. In both cases, the truth value of $\mathsf{pre}(\mathsf{a})$ is solely decided by $\mathrm{F}(M, \mathrm{PROT}(\mathsf{pre}(\mathsf{a})))$, so we have $\mathrm{F}(M, \mathrm{PROT}(\mathsf{pre}(\mathsf{a}))), w \models_{\mathsf{fd}} \mathsf{pre}(\mathsf{a})$.

The direction $\Leftarrow$ follows from a similar reasoning.

Case iii: this follows from a similar reasoning as in Case i.

Case iv: this follows from a similar reasoning as in Case ii. ∎

This lemma shows special properties of the forest models generated from a state model and a **DEL** formula. In the case of formula $\varphi \wedge \psi$, the truth value of φ is solely decided by the forest model $\mathrm{F}(M, \mathrm{PROT}(\varphi))$, which is a sub-model of $\mathrm{F}(M, \mathrm{PROT}(\varphi \wedge \psi))$. In the case of formula $[\mathsf{A}, \mathsf{a}]\psi)$, the truth value of $\mathsf{pre}(\mathsf{a})$ is solely decided by the forest model $\mathrm{F}(M, \mathrm{PROT}(\mathsf{pre}(\mathsf{a})))$, which is a sub-model of $\mathrm{F}(M, \mathrm{PROT}([\mathsf{A}, \mathsf{a}]\psi))$.

We show, in three steps, the equivalence

$$M, w \models_{\mathsf{sd}} \varphi \text{ iff } \mathrm{SEM}(M, \varphi), s_w \models_{\mathsf{it}} \mathrm{SYN}(\varphi).$$

The main difference is that φ is now a **DEL** formula instead of a **PAL** formula.

The *first step* (Proposition 4.5) shows that given a state model M and a **DEL** formula φ, the interpretation of φ over (M, w) is equivalent to its interpretation over $\mathrm{F}(M, \mathrm{PROT}(\varphi))$, which is the forest model built from M and φ. The *second step* (Proposition 4.6) shows that φ and its syntactic translation $\mathrm{SYN}(\varphi)$ are equivalent when they are both interpreted over the forest model $\mathrm{F}(M, \mathrm{PROT}(\varphi))$. The *third step* (Proposition 4.7) shows that the interpretation of $\mathrm{SYN}(\varphi)$ over an arbitrary forest model and its corresponding interpreted system are equivalent.

PROPOSITION 4.5
Let M be a state model and $\varphi \in \mathcal{L}_{\mathsf{DEL}}$.

$$M, w \models_{\mathsf{sd}} \varphi \quad \text{iff} \quad \mathrm{F}(M, \mathrm{PROT}(\varphi)), w \models_{\mathsf{fd}} \varphi$$

PROOF. Given a state model M, and formula φ, we follow the procedure in Definition 4.3 and build a forest model $\mathrm{F}(M, \mathrm{PROT}(\varphi))$. We do an induction on the structure of φ. The cases of atomic proposition, negation, knowledge and common knowledge are trivial. We only show the case with action modality.

Case $[\mathsf{A},\mathsf{a}]\psi$:

$M, w \models_{\mathsf{sd}} [\mathsf{A},\mathsf{a}]\psi$

$\Leftrightarrow$

$M, w \models_{\mathsf{sd}} \mathsf{pre}(\mathsf{a}) \Rightarrow M \otimes \mathsf{A}, (w,\mathsf{a}) \models_{\mathsf{sd}} \psi$

$\Leftrightarrow$ By induction

$\mathrm{F}(M, \mathrm{PROT}(\mathsf{pre}(\mathsf{a}))), w \models_{\mathsf{fd}} \mathsf{pre}(\mathsf{a}) \Rightarrow \mathrm{F}(M \otimes \mathsf{A}, \mathrm{PROT}(\psi)), (w,\mathsf{a}) \models_{\mathsf{fd}} \psi$

$\Leftrightarrow$ By Lemma 4.4

$\mathrm{F}(M, \mathrm{PROT}([\mathsf{A},\mathsf{a}]\psi)), w \models_{\mathsf{fd}} \mathsf{pre}(\mathsf{a}) \Rightarrow \mathrm{F}(M \otimes \mathsf{A}, \mathrm{PROT}(\psi)), (w,\mathsf{a}) \models_{\mathsf{fd}} \psi$

$\Leftrightarrow$ By forest model construction

$\mathrm{F}(M, \mathrm{PROT}([\mathsf{A},\mathsf{a}]\psi)), w \models_{\mathsf{fd}} \mathsf{pre}(\mathsf{a}) \Rightarrow \mathrm{F}(M, \cup_{\mathsf{b}\in\mathcal{D}(\mathsf{A})} \mathsf{b}\mathrm{PROT}(\psi)), (w,\mathsf{a}) \models_{\mathsf{fd}} \psi$

$\Leftrightarrow$

$\mathrm{F}(M, \mathrm{PROT}([\mathsf{A},\mathsf{a}]\psi)), w \models_{\mathsf{fd}} \mathsf{pre}(\mathsf{a}) \Rightarrow$
$\mathrm{F}(M, \cup_{\mathsf{b}\in\mathcal{D}(\mathsf{A})} \mathsf{b}\mathrm{PROT}(\psi) \cup \mathrm{PROT}(\mathsf{pre}(\mathsf{a}))), (w,\mathsf{a}) \models_{\mathsf{fd}} \psi$

$\Leftrightarrow$ As $\cup_{\mathsf{b}\in\mathcal{D}(\mathsf{A})} \mathsf{b}\mathrm{PROT}(\psi) \cup \mathrm{PROT}(\mathsf{pre}(\mathsf{a})) = \mathrm{PROT}([\mathsf{A},\mathsf{a}]\psi)$

$\mathrm{F}(M, \mathrm{PROT}([\mathsf{A},\mathsf{a}]\psi)), w \models_{\mathsf{fd}} \mathsf{pre}(\mathsf{a}) \Rightarrow \mathrm{F}(M, \mathrm{PROT}([\mathsf{A},\mathsf{a}]\psi)), (w,\mathsf{a}) \models_{\mathsf{fd}} \psi$

$\Leftrightarrow$

$\mathrm{F}(M, \mathrm{PROT}([\mathsf{A},\mathsf{a}]\psi)), w \models_{\mathsf{fd}} \mathsf{pre}(\mathsf{a}) \Rightarrow \exists (w,\mathsf{a})$ s.t. $w \rightarrow_{\mathsf{a}} (w,\mathsf{a})$ and
$\mathrm{F}(M, \mathrm{PROT}([\mathsf{A},\mathsf{a}]\psi)), (w,\mathsf{a}) \models_{\mathsf{fd}} \psi$

$\Leftrightarrow$

$\mathrm{F}(M, \mathrm{PROT}([\mathsf{A},\mathsf{a}]\psi)), w \models_{\mathsf{fd}} [\mathsf{A},\mathsf{a}]\psi$ ∎

PROPOSITION 4.6
Given a state model M and a DEL formula φ:

$$\mathrm{F}(M, \mathrm{PROT}(\varphi)), w \models_{\mathsf{fd}} \varphi \quad \text{iff} \quad \mathrm{F}(M, \mathrm{PROT}(\varphi)), w \models_{\mathsf{ft}} \mathrm{SYN}(\varphi)$$

PROOF. Given a state model M and a DEL formula φ, we follow the procedure in Definition 4.3 and build a forest model $\mathrm{F}(M, \mathrm{PROT}(\varphi))$. We do an induction on the structure of φ. We only show the case with action modality.

Case $[\mathsf{A},\mathsf{a}]\psi$:

We have to show that

$$\mathrm{F}(M, \mathrm{PROT}([\mathsf{A},\mathsf{a}]\psi)), w \models_{\mathsf{fd}} [\mathsf{A},\mathsf{a}]\psi \text{ iff } \mathrm{F}(M, \mathrm{PROT}([\mathsf{A},\mathsf{a}]\psi)), w \models_{\mathsf{ft}} \mathrm{SYN}([\mathsf{A},\mathsf{a}]\psi).$$

In other words:

$$\mathrm{F}(M, \mathrm{PROT}([\mathsf{A},\mathsf{a}]\psi)), w \models_{\mathsf{fd}} \mathsf{pre}(\mathsf{a}) \Rightarrow \mathrm{F}(M, \mathrm{PROT}([\mathsf{A},\mathsf{a}]\psi)), (w,\mathsf{a}) \models_{\mathsf{fd}} \psi$$
$$\Leftrightarrow$$
$$\mathrm{F}(M, \mathrm{PROT}([\mathsf{A},\mathsf{a}]\psi)), w \models_{\mathsf{ft}} \mathrm{SYN}(\mathsf{pre}(\mathsf{a})) \Rightarrow \mathrm{F}(M, \mathrm{PROT}([\mathsf{A},\mathsf{a}]\psi)), w \models_{\mathsf{ft}} \bigcirc_{\mathsf{a}} \mathrm{SYN}(\psi)$$

First we show that both conditional parts are equivalent (i). Then we show that on the condition, both consequential parts are equivalent (ii). In the proof we use various times that $\mathrm{PROT}([\mathsf{A},\mathsf{a}]\psi) = \cup_{\mathsf{b}\in\mathcal{D}(\mathsf{A})} \mathsf{b}\mathrm{PROT}(\psi) \cup \mathrm{PROT}(\mathsf{pre}(\mathsf{a}))$.

(i) We show that

$$\mathrm{F}(M, \mathrm{PROT}([\mathsf{A},\mathsf{a}]\psi)), w \models_{\mathsf{fd}} \mathsf{pre}(\mathsf{a}) \text{ iff } \mathrm{F}(M, \mathrm{PROT}([\mathsf{A},\mathsf{a}]\psi)), w \models_{\mathsf{ft}} \mathrm{SYN}(\mathsf{pre}(\mathsf{a}))$$

by the following equivalence:

$\mathrm{F}(M, \mathrm{PROT}([\mathsf{A},\mathsf{a}]\psi)), w \models_{\mathsf{fd}} \mathsf{pre}(\mathsf{a})$

$\Leftrightarrow$ By Lemma 4.4
$\mathrm{F}(M, \mathrm{PROT}(\mathsf{pre}(\mathsf{a}))), w \models_{\mathsf{fd}} \mathsf{pre}(\mathsf{a})$
$\Leftrightarrow$ By induction
$\mathrm{F}(M, \mathrm{PROT}(\mathsf{pre}(\mathsf{a}))), w \models_{\mathsf{ft}} \mathrm{SYN}(\mathsf{pre}(\mathsf{a}))$
$\Leftrightarrow$ By Lemma 4.4
$\mathrm{F}(M, \mathrm{PROT}([\mathsf{A},\mathsf{a}]\psi)), w \models_{\mathsf{ft}} \mathrm{SYN}(\mathsf{pre}(\mathsf{a}))$

(*ii*) Next, we show that on condition of $\mathrm{F}(M, \mathrm{PROT}(\mathsf{pre}(\mathsf{a}))), w \models_{\mathsf{fd}} \mathsf{pre}(\mathsf{a})$:

$$\mathrm{F}(M, \mathrm{PROT}([\mathsf{A},\mathsf{a}]\psi)), (w,\mathsf{a}) \models_{\mathsf{fd}} \psi \quad \text{iff} \quad \mathrm{F}(M, \mathrm{PROT}([\mathsf{A},\mathsf{a}]\psi)), w \models_{\mathsf{ft}} \bigcirc_{\mathsf{a}} \mathrm{SYN}(\psi).$$

$\mathrm{F}(M, \mathrm{PROT}([\mathsf{A},\mathsf{a}]\psi)), (w,\mathsf{a}) \models_{\mathsf{fd}} \psi$
$\Leftrightarrow$
$\mathrm{F}(M, \cup_{\mathsf{b}\in\mathcal{D}(\mathsf{A})} \mathsf{b}\mathrm{PROT}(\psi) \cup \mathrm{PROT}(\mathsf{pre}(\mathsf{a}))), (w,\mathsf{a}) \models_{\mathsf{fd}} \psi$
$\Leftrightarrow$ $(w,\mathsf{a}) \notin \mathrm{F}(M, \mathrm{PROT}(\mathsf{pre}(\mathsf{a})))$
$\mathrm{F}(M, \cup_{\mathsf{b}\in\mathcal{D}(\mathsf{A})} \mathsf{b}\mathrm{PROT}(\psi)), (w,\mathsf{a}) \models_{\mathsf{fd}} \psi$
$\Leftrightarrow$ By forest model construction
$\mathrm{F}(M \otimes \mathsf{A}, \mathrm{PROT}(\psi)), (w,\mathsf{a}) \models_{\mathsf{fd}} \psi$
$\Leftrightarrow$ By induction
$\mathrm{F}(M \otimes \mathsf{A}, \mathrm{PROT}(\psi)), (w,\mathsf{a}) \models_{\mathsf{ft}} \mathrm{SYN}(\psi)$
$\Leftrightarrow$ By forest model construction
$\mathrm{F}(M, \cup_{\mathsf{b}\in\mathcal{D}(\mathsf{A})} \mathsf{b}\mathrm{PROT}(\psi)), (w,\mathsf{a}) \models_{\mathsf{ft}} \mathrm{SYN}(\psi)$
$\Leftrightarrow$ $(w,\mathsf{a}) \notin \mathrm{F}(M, \mathrm{PROT}(\mathsf{pre}(\mathsf{a})))$
$\mathrm{F}(M, \cup_{\mathsf{b}\in\mathcal{D}(\mathsf{A})} \mathsf{b}\mathrm{PROT}(\psi) \cup \mathrm{PROT}(\mathsf{pre}(\mathsf{a}))), (w,\mathsf{a}) \models_{\mathsf{ft}} \mathrm{SYN}(\psi)$
$\Leftrightarrow$
$\mathrm{F}(M, \mathrm{PROT}([\mathsf{A},\mathsf{a}]\psi)), (w,\mathsf{a}) \models_{\mathsf{ft}} \mathrm{SYN}(\psi)$
$\Leftrightarrow$
$\mathrm{F}(M, \mathrm{PROT}([\mathsf{A},\mathsf{a}]\psi)), w \models_{\mathsf{ft}} \bigcirc_{\mathsf{a}} \mathrm{SYN}(\psi)$ ∎

PROPOSITION 4.7
For every *executable* $\varphi \in \mathcal{L}_{\mathsf{NTEL}}$ (i.e. a formula of the form $\mathrm{SYN}(\psi)$ with $\psi \in \mathcal{L}_{\mathsf{DEL}}$), and every forest model M constructed from ψ with some initial state model, we have:

$$M, w \models_{\mathsf{ft}} \varphi \quad \text{iff} \quad \mathrm{IS}(M), (w, w^{\sim_1}, \ldots, w^{\sim_n}) \models_{\mathsf{it}} \varphi$$

PROOF. Let a forest model M be given. We construct an action-based interpreted system $\mathrm{IS}(M)$ according to Definition 3.6. Let s_w stand for $(w, w^{\sim_1}, \ldots, w^{\sim_n})$. We do an induction on φ. We only show the case with temporal modality.

Case $\mathsf{pre}(\mathsf{a}) \to \bigcirc_{\mathsf{a}}\psi$:
$M, w \models_{\mathsf{ft}} \mathsf{pre}(\mathsf{a}) \to \bigcirc_{\mathsf{a}}\psi$
$\Leftrightarrow$
$M, w \models_{\mathsf{ft}} \mathsf{pre}(\mathsf{a}) \Rightarrow M, w \models_{\mathsf{ft}} \bigcirc_{\mathsf{a}}\psi$
$\Leftrightarrow$ ($\natural$) on condition of $M, w \models \mathsf{pre}(\mathsf{a})$ a run exists
$M, w \models_{\mathsf{ft}} \mathsf{pre}(\mathsf{a}) \Rightarrow M, (w,\mathsf{a}) \models_{\mathsf{ft}} \psi$
$\Leftrightarrow$ By induction
$\mathrm{IS}(M), s_w \models_{\mathsf{it}} \mathsf{pre}(\mathsf{a}) \Rightarrow \mathrm{IS}(M), (s_w,\mathsf{a}) \models_{\mathsf{it}} \psi$
$\Leftrightarrow$ (@) a run always exists
$\mathrm{IS}(M), s_w \models_{\mathsf{it}} \mathsf{pre}(\mathsf{a}) \Rightarrow \mathrm{IS}(M), s_w \models_{\mathsf{it}} \bigcirc_{\mathsf{a}}\psi$

$\Leftrightarrow$
$\text{IS}(M), s_w \models_{\mathsf{it}} \mathsf{pre}(\mathsf{a}) \rightarrow \bigcirc_{\mathsf{a}} \psi$
In step ♯ of the proof, this is guaranteed by the condition $M, w \models_{\mathsf{ft}} \mathsf{pre}(\mathsf{a})$: as the precondition of a is true, it can be executed and there is an $\rightarrow_{\mathsf{a}}$ accessible state from w. This is not guaranteed if $\mathsf{pre}(\mathsf{a})$ is false.

In step @ of the proof the required path always exists, as runs in interpreted systems are infinite. In particular, if $s_w = (\mathsf{r}_w, i)$, then (s_w, a) (i.e. $(w, (w, \mathsf{a})^{\sim_1}, \ldots, (w, \mathsf{a})^{\sim_n})$) is of the form $(\mathsf{r}'_w, i+1)$ where r' is equivalent to r to time i. ∎

We emphasize that Proposition 4.7 does not hold for arbitrary formulas in our temporal epistemic fragment, because of the essential difference between forest models, where action sequences are finite, and corresponding interpreted systems, with infinite runs. More precisely: in case $\mathsf{pre}(\mathsf{a}) \rightarrow \bigcirc_{\mathsf{a}} \psi$ of the proof of Proposition 4.7 the precondition $\mathsf{pre}(\mathsf{a})$ is essential. States in forest models do not necessarily have an action successor, so that in such states all formulas of form $\bigcirc_{\mathsf{a}} \psi$ are false, whereas runs in interpreted systems keep looping after a finite meaningful prefix, e.g. $\bigcirc_{\mathsf{a}} q$ will always remain true if q is true.

Now the generalization of Theorem 3.13 also holds.

THEOREM 4.8
Given a state model M, and a DEL formula φ,

$$M, w \models_{\mathsf{sd}} \varphi \quad \text{iff} \quad \text{SEM}(M, \varphi), s_w \models_{\mathsf{it}} \text{SYN}(\varphi)$$

PROOF. It directly follows from Proposition 4.5, Proposition 4.6 and 4.7. ∎

5 Conclusion

Given an epistemic state (M, w) and a formula φ in a dynamic epistemic logic, we construct an action-based interpreted system $\text{SEM}(M, \varphi)$ relative to that epistemic state and that formula. The construction involves the protocol implicitly present in the dynamic epistemic formula φ, i.e. the set of sequences of actions being executed to evaluate the formula, is made explicit. Then we show that φ is satisfied in the epistemic state (M, w) *if and only if* its syntactic translation $\text{SYN}(\varphi)$ in a temporal epistemic logic is satisfied in the corresponding interpreted system $(\text{SEM}(M, \varphi), s_w)$.

We make two contributions in this article. First of all, we have formalized something that is vaguely clear on a conceptual level: a connection between epistemic actions and time in well-defined structures. We have adopted a more constructive approach in the sense that we made the protocols underlying a dynamic epistemic action, explicit in the corresponding interpreted system. And the mechanics of the syntactic translation and semantic transformation procedures are fully written out.

Secondly, our theorems provide a systematic approach to model check dynamic epistemic formulas using temporal epistemic formalisms. The large amount of work on model checking knowledge and time [4, 5, 8, 19, 22, 27] now becomes accessible for the community working on dynamic epistemic model checking as well. In particular, the temporal epistemic checkers mentioned [5, 19, 22] have all in common that they use interpreted systems as their semantics, and that they optimize search by implementing BDD techniques [11], which has yet not been employed by the DEL community. We did not undertake a systematic analysis and

comparison of the complexity of model checking in the different frameworks: this is surely important and work that needs to be addressed in the future.

Acknowledgements

We would like to thank two anonymous reviewers of the *Logic Journal of the IGPL*, and the reviewers and participants of the FAMAS'007 Workshop, for their helpful comments, suggestions and discussions. The submission was made during Ji Ruan's association with the University of Liverpool in UK, the revision was made during his association with St. Francis Xavier University in Canada, and the final correction was made and approved during his association with his current institution.

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] R. J. Aumann and A. Brandenburger. Epistemic conditions for nash equilibrium. *Econometrica*, **63**, 1161–1180, 1995.

[3] A. Baltag, L. S. Moss, and S. Solecki. The logic of public announcements, common knowledge, and private suspicions. In *Theoretical Aspects of Rationality and Knowledge*, I. Gilboa, ed., pp. 43–56, 1998.

[4] R. Fagin, J. Y. Halpern, Y. Moses, and M. Y. Vardi. *Reasoning about Knowledge*. MIT Press, 1995.

[5] P. Gammie and R. van der Meyden. MCK: model checking the logic of knowledge. In *Proceedings of the 16th International Conference on Computer Aided Verification (CAV 2004)*, R. Alur and D. Peled, eds, pp. 479–483. Springer, 2004.

[6] J. D. Gerbrandy and W. Groeneveld. Reasoning about information change. *Journal of Logic, Language, and Information*, **6**, 147–169, 1997.

[7] J. Y. Halpern, R. van der Meyden, and M. Y. Vardi. Complete axiomatizations for reasoning about knowledge and time. *SIAM Journal on Computing*, **33**, 674–703, 2004.

[8] J. Y. Halpern and M. Y. Vardi. The complexity of reasoning about knowledge and time. In *Proceedings 18th ACM Symposium on Theory of Computing*, pp. 304–315, 1986.

[9] J. Hintikka. *Knowledge and Belief*. Cornell University Press, 1962.

[10] T. Hoshi and A. Yap. Dynamic epistemic logic with branching temporal structures. *Synthese*, **169**, 259–281, 2009.

[11] M. R. A. Huth and M. D. Ryan. *Logic in Computer Science – Modelling and Reasoning about Systems*. Cambridge University Press, 2000.

[12] 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. *Fundamenta Informatica*, **85**, 313–328, 2008.

[13] A. R. Lomuscio and M. D. Ryan. On the relation between interpreted systems and kripke models. In *Proceedings of the AI97 Workshop on the Theoretical and Practical Foundations of Intelligent Agents and Agent-Oriented Systems*, C. Zhang M. Pagnucco, and W. R. Wobcke, eds, Vol. 1441 of *Lecture Notes in Articifial Intelligence*, pp. 46–59. Springer, 1998.

[14] X. Luo, K. Su, A. Sattar, and Y. Chen. Solving sum and product riddle via bdd-based model checking. In *WI-IAT '08: Proceedings of the 2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, pp. 630–633. IEEE Computer Society, 2008.

[15] R. C. Moore. A formal theory of knowledge and action. In *Readings in Planning*, J. F. Allen, J. Hendler, and A. Tate, eds, pp. 480–519. Morgan Kaufmann Publishers, 1990.

[16] E. Pacuit. Some comments on history based structures. *Journal of Applied Logic*, **5**, 613–624, 2007.

[17] R. Parikh and R. Ramanujam. Distributed processing and the logic of knowledge. In *Logic of Programs*, Vol. 193 of *Lecture Notes in Computer Science*, pp. 256–268. Springer, 1985. A newer version appeared in *Journal of Logic, Language and Information*, **12**, 453–467, 2003.

[18] J. A. Plaza. Logics of public communications. In *Proceedings of the 4th International Symposium on Methodologies for Intelligent Systems: Poster Session Program*, M. L. Emrich, M. S. Pfeifer, M. Hadzikadic, and Z. W. Ras, eds, pp. 201–216. Oak Ridge National Laboratory, 1989. ORNL/DSRD-24. A newer version appeared in Synthese (2007) **158**, 165–179, Knowledge, Rationality & Action, 255–269.

[19] F. Raimondi and A. R. Lomuscio. Verification of multi-agent systems via ordered binary decision diagrams: an algorithm and its implementation. In *Proceedings of the Autonomous Agents and Multi-Agent Systems (AAMAS) conference '04*, pp. 630–637. IEEE Computer Society, 2004.

[20] A. S. Rao and M. P. Georgeff. Modeling rational agents within a BDI-architecture. In *Proceedings of the 2nd International Conference on Principles of Knowledge Representation and Reasoning (KR'91)*, J. Allen, R. Fikes, and E. Sandewall, eds, pp. 473–484. Morgan Kaufmann, 1991.

[21] J. Sack. Temporal languages for epistemic programs. *Journal of Logic, Language and Information*, **17**, 183–216, 2009.

[22] K. Su, A. Sattar, and X. Luo. Model checking temporal logics of knowledge via obdds. *Computer Journal*, **50**, 403–420, 2007.

[23] J. F. A. K. van Benthem. Semantic parallels in natural language and computation. In *Logic Colloquium '87*. North-Holland, 1989.

[24] J. F. A. K. van Benthem. Games in dynamic epistemic logic. *Bulletin of Economic Research*, **53**, 219–248, 2001.

[25] J. F. A. K. van Benthem, P. Dekker, J. van Eijck, M. de Rijke, and Y. Venema. *Logic in Action*. ILLC, 2001.

[26] J. F. A. K. van Benthem, J. D. Gerbrandy, T. Hoshi, and E. Pacuit. Merging frameworks for interaction. *Journal of Philosophical Logic*, **38**, 491–526, 2009.

[27] R. van der Meyden and K. Wong. Complete axiomatisations for reasoning about knowledge and branching time. *Studia Logica*, **75**, 93–123, 2003.

[28] H. P. van Ditmarsch. The russian cards problem. *Studia Logica*, **75**, 31–62, 2003.

[29] H. P. van Ditmarsch, J. Ruan, and R. Verbrugge. Sum and product in dynamic epistemic logic. *Journal of Logic and Computation*, **18**, 563–588, 2008.

[30] H. P. van Ditmarsch, W. van der Hoek, and B. P. Kooi. *Dynamic Epistemic Logic*, Vol. 337 of *Synthese Library*. Springer, 2007.

[31] H. P. van Ditmarsch, W. van der Hoek, R. van der Meyden, and J. Ruan. Model checking russian cards. *Electronic Notes in Theoretical Computer Science*, **149**, 105–123, 2006. Presented at MoChArt 05 (Model Checking in Artificial Intelligence).

[32] J. van Eijck. Demo - a demo of epistemic modelling. In *Interactive Logic: Selected Papers from the 7th Augustus de Morgan Workshop, London*, J. van Benthem, B. Lowe, and D. Gabbay, eds, pp. 303–362. Amsterdam University Press, 2008. Texts in Logic and Games 1.

Received 27 May 2009