

Terminating Tableaux for Dynamic Epistemic Logics

Jens Ulrik Hansen^{1,2}

*Programming, Logic and Intelligent Systems/Philosophy and Science Studies
Roskilde University
Roskilde, Denmark*

Abstract

Throughout the last decade, there has been an increased interest in various forms of dynamic epistemic logics to model the flow of information and the effect this flow has on knowledge in multi-agent systems. This enterprise, however, has mostly been applicationally and semantically driven. This results in a limited amount of proof theory for dynamic epistemic logics.

In this paper, we try to compensate for a part of this by presenting terminating tableau systems for full dynamic epistemic logic with action models and for a hybrid public announcement logic (both without common knowledge). The tableau systems are extensions of already existing tableau systems, in addition to which we have used the reduction axioms of dynamic epistemic logic to define rules for the dynamic part of the logics. Termination is shown using methods introduced by Bräuner, Bolander, and Blackburn.

Keywords: Dynamic epistemic logic, public announcement logic, terminating tableau systems, decision procedures, hybrid logic, reduction axioms.

1 Introduction

Classic epistemic logic has played an important role in both philosophy and computer science. However, recent years have witnessed the importance of also looking at the dynamics of knowledge, i.e. how knowledge of different agents can change due to the development of a system. There are two ways of adding dynamics to epistemic logic. One can either combine it with a temporal logic or combine it with some dynamic logic of actions. The latter approach has become increasingly common and has resulted in what is now called Dynamic Epistemic Logic (DEL), which includes operators for so called *epistemic actions* (cf. [13]). The interest in DEL has mostly been related to applications, and has mainly been semantically

¹ Thanks to Torben Braner and Sine Zambach for comments on a draft of the paper. Also thanks to anonymous reviewers for valuable comments. The author is partially funded by the Danish Natural Science Research Council through the HYLOCORE project.

² jensuh@ruc.dk

driven. Thus, only very few attempts to develop a rich proof theory for DEL beyond standard Hilbert style systems have been performed. This work attempts to make up for some of this by discussing terminating tableau systems for different kinds of dynamic epistemic logics.

The simplest form of dynamics one can add to classical epistemic logic is a public announcement operator. The language is extended with formulas of the type $[\varphi]\psi$, which are read as “after public announcement of φ , ψ holds”. At the semantic level, the operator $[\varphi]$ corresponds to moving to the submodel consisting only of states where φ is true (thus we are exclusively concerned with truthful public announcements here). This simple extension, called Public Announcement Logic (PAL), is, nevertheless, quite useful as shown by the many applications presented in [13]. Having left out common knowledge, operators³, this logic is fairly simple and a few tableau systems do already exist, see [1] and [7]. The approach in this paper is different from these in the sense that we try to avoid constructing new complicated and tailor made tableau systems by instead using the existing systems. This is possible due to reduction axioms. Reduction axioms have, from the beginning of DEL’s short history, played an important role in showing completeness and expressiveness results. It was proved that Public Announcement Logic is no more expressive than the underlying epistemic logic. Using reduction axioms, it is possible to translate a public announcement formula into an equivalent one without any public announcement operators.

There are a lot of other possible epistemic actions, moving beyond bare public announcements: announcements to subgroups, private communications, secret announcements and more. The insight of Baltag, Moss and Solecki (in [2]) is that all these epistemic actions, considered as action modalities, can also be represented by a form of Kripke models. Using a general product operation on Kripke models, they can be given a semantic. More surprisingly, it was shown that also formulas with these more complicated action modalities can be reduced to basic epistemic formulas without any action modalities. This, of course, required more advanced reduction axioms than in the case of public announcements.

When one wants to prove a validity of DEL, one can simply translate the validity into an epistemic formula without action modalities and then use the existing tableau (or other) systems. However, the translation might result in an exponential increase in the size of the formula. As is shown in [10], this exponential increase cannot be avoided for public announcements. This fact provides another motivation for using DEL, since it offers us the opportunity to express things much more compact than in classical epistemic logic. It is also shown in [10] that the complexity of validity checking for PAL is no higher than for the underlying epistemic logic. Thus, the method of first translating and then using known proof methods for classical epistemic logic may be unfeasible. This justifies the direct tableau systems for PAL given in [1] and [7]. In DELs with arbitrary epistemic actions, the matter becomes much more complicated. Here, the challenge is how to represent the action modalities. Since PAL is part of DEL, [10] also shows that the blowup of the translation

³ In the rest of this paper, at least until the conclusion, we will disregard common knowledge.

may be exponential in the general DEL case. However, it is currently unknown to the author exactly what the complexity of deciding DEL validities is.⁴ When adding the global modality to the underlying epistemic logic, the complexity of this will already be exponential [11]. In this case, the exponential increase caused by the translation therefore does not destroy the worst-case complexity.

The work in this paper is based on the idea of using reduction axioms as rules to make the translation on the fly in the tableaux. In practice, this is more efficient than performing the whole translation at the beginning⁵, but in the case of public announcement, it might not be as fast as the methods of [1] and [7]. However, their tableau systems only work for public announcement logic, while the method presented here further works for full dynamic epistemic logic and for a hybrid version of public announcement logic. Our tableaux in this paper are kept terminating using the methods of Braüner, Bolander, and Blackburn ([5], [6], and [4]). The presentation here will be based on the approach in [4]. For basic modal logic, they show termination by noticing that the maximal formula complexity drops as new prefixes are introduced, which makes infinite tableaux impossible. In this paper we show that, essentially, this argument can be adapted in the setting where reduction axioms are used as extra tableau rules.

The paper is structured as follows: first we introduce public announcement logic, a hybrid public announcement logic, and full dynamic epistemic logic (section 2). Then, we present a terminating tableau system for full dynamic epistemic logic in section 3. Following this, we demonstrate how the approach can also be used to create a terminating tableau system for the hybrid public announcement logic. Finally, we present some concluding remarks and discuss further research.

2 Dynamic epistemic logic

We will first present the formal definitions of public announcement logic. Public announcements are added to the normal modal logic **K**, but it can easily be extended to the case of **S5**, which is often used for modeling knowledge. We will leave out common knowledge. First, we assume a finite set of agents \mathbb{A} and a countable infinite set of propositional variables PROP . Using the terminology of [13] the language of the Public Announcement Logic will be denoted by $\mathcal{L}_{K\Box}$, and is given by the following syntax:

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

⁴ When dealing with arbitrary formulas of DEL, the question is how to measure the size of the action modalities. On the one hand, an action modality could be counted as one symbol, but when deciding validity, the finer structure of the action modality is needed. Thus, this may result in a high complexity for validity checking in this size of the formula. On the other hand, using another size-measure of action modalities, it may become possible to decide validity in lower complexity in that size.

⁵ In worst case scenario doing the translation on the fly may not be more efficient. But there seems to be at least two cases where translation on the fly will speed up the process of deciding a formula. The first case is where only few steps of translation are needed to detect an inconsistency, as for instance in the formula $[\neg(q \wedge r)]K_a(p \rightarrow q)(p \vee r) \wedge \neg([\neg(q \wedge r)]K_a(p \rightarrow q)) \rightarrow (p \vee r)$. The other case is where the need for a translation may only occur at the very end of the tableau construction process, as for instance in the formula $K_a[p]p \wedge \neg K_a\neg[q]\neg q \wedge K_a([q]p \wedge [p]q)$.

where $p \in \text{PROP}$ and $a \in \mathbb{A}$. The connectives \vee , \rightarrow and \leftrightarrow are defined from \neg and \wedge in the usual way, and the dual operators $\neg K_a \neg$ and $\neg[\varphi] \neg$ are abbreviated by \hat{K}_a and $\langle \varphi \rangle$. The interpretation of $K_a \varphi$ is that “agent a knows that φ ” and of $[\varphi] \psi$ that “after (truthful) public announcement of φ , ψ is the case”. These interpretations are captured by the following formal semantic: A Kripke frame (or just a frame) is a pair $\mathcal{F} = \langle W, (R_a)_{a \in \mathbb{A}} \rangle$ consisting of a non-empty set W of states (or possible worlds) and for each $a \in \mathbb{A}$ a binary relation R_a on W (i.e. $R_a \subseteq W \times W$). A model \mathcal{M} is a pair consisting of a frame \mathcal{F} and a valuation V that assigns a set of states in W to every propositional variable of PROP (i.e. $V : \text{PROP} \rightarrow \mathcal{P}(W)$). Given a formula φ of $\mathcal{L}_{K\Box}$, a model $\mathcal{M} = \langle W, (R_a)_{a \in \mathbb{A}}, V \rangle$, and a state $w \in W$, the truth of φ at w , notation $\mathcal{M}, w \models \varphi$, is defined as standard in modal logic, taking K_a to be the box modality corresponding to R_a . In addition we add the following clause for $[\varphi] \psi$:

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

where the model $\mathcal{M}|_\varphi = \langle W|_\varphi, R|_\varphi, V|_\varphi \rangle$ is defined by:

$$W|_\varphi = \{v \in W \mid \mathcal{M}, v \models \varphi\}$$

$$R_a|_\varphi = R_a \cap (W|_\varphi \times W|_\varphi)$$

$$V|_\varphi(p) = V(p) \cap W|_\varphi.$$

We write $\mathcal{M} \models \varphi$ if $\mathcal{M}, w \models \varphi$ for all $w \in W$ and we say that φ is valid if $\mathcal{M} \models \varphi$ for all models \mathcal{M} . The logic of this semantic will be denoted by **PA** and be call Public Announcement Logic. It is not hard to prove the following validities in this logic:

- (1) $[\varphi] p \leftrightarrow (\varphi \rightarrow p)$
- (2) $[\varphi] \neg \psi \leftrightarrow (\varphi \rightarrow \neg[\varphi] \psi)$
- (3) $[\varphi] (\psi \wedge \chi) \leftrightarrow ([\varphi] \psi \wedge [\varphi] \chi)$
- (4) $[\varphi] K_a \psi \leftrightarrow (\varphi \rightarrow K_a [\varphi] \psi)$
- (5) $[\varphi] [\psi] \chi \leftrightarrow [\varphi \wedge [\varphi] \psi] \chi.$

These are the reduction axioms for public announcement ([13]). Adding these axioms together with necessitation of $[\varphi]$ to a Hilbert style proof system for multi modal **K** will result in a sound and complete proof system for **PA** (for details see [13]).

Note that the complexity of the formula occurring within the scope of the public announcement operator is greater on the left than on the right side of “ \leftrightarrow ” in these reduction axioms. This can be used to define a translation $T : \mathcal{L}_{K\Box} \rightarrow \mathcal{L}_K$, where \mathcal{L}_K is the standard multi-modal language. The translation “commute” with all logic operators beside the public announcement operator (e.g. $T(\neg \varphi) = \neg T(\varphi)$). In the case when the translation encounter a $[\varphi]$ operator it uses the reduction axioms to decrease the complexity of the formula within the scope of the operator, e.g. $T([\varphi](\neg \psi)) = T(\varphi) \rightarrow \neg T([\varphi] \psi)$. This translation can be shown to be a truth preserving translation of **PA** into multi modal **K**, which shows that adding the public announcement operator does not increase the expressiveness of

the language.⁶ This recursive translation is not recursive in the normal way, since going from left to right (of \leftrightarrow) in the reduction axioms (1)–(5) does not reduce the standard formula complexity. Therefore, to prove the correctness of the translation, a new complexity measure the formulas is needed. One possible measure is (taken from [13]):

Definition 2.1 Define a complexity measure $c : \mathcal{L}_{K\Box} \rightarrow \mathbb{N}$ by the inductive clauses:

$$\begin{aligned} c(p) &= 1 \\ c(\neg\varphi) &= 1 + c(\varphi) \\ c(\varphi \wedge \psi) &= 1 + \max\{c(\varphi), c(\psi)\} \\ c(K_a\varphi) &= 1 + c(\varphi) \\ c([\varphi]\psi) &= (4 + c(\varphi)) \cdot c(\psi) \end{aligned}$$

What can be shown about this complexity measure is that it decreases when moving to subformulas and, furthermore, that the c complexity of the left hand sides of the reduction axioms (1)–(5) are higher than the c complexity of the right hand sides. This fact will be important when we consider the tableau system in the next section.

We will not present a tableau system for **PA**, but for a hybrid extension of this, namely the Hybrid Public Announcement Logic of [8]. To obtain this new logic we will first extend the language. For this we fix a countable infinite set of nominals **NOM** disjoint from the propositional variables. The language of hybrid public announcement logic \mathcal{L}_{HPA} is defined by:

$$\varphi ::= p \mid i \mid \neg\varphi \mid \varphi \wedge \psi \mid K_a\varphi \mid [\varphi]\psi \mid @_i\varphi \mid E\varphi,$$

where $p \in \text{PROP}$, $i \in \text{NOM}$ and $a \in \mathbb{A}$.

The nominals will function as names for states. The formula $@_i\varphi$ states that φ is true at the state that i denotes and $E\varphi$ express that there is a state where φ is true. The semantic is specified somewhat different from what is standard in hybrid logic. The reason is that the semantic of the public announcement operator takes us to submodels where states denoted by nominals may disappear. To deal with this, we extend the class of models such that the valuation assigns *at most one* state instead of *exactly one* state to each nominal (for more on these issues, see [8]). The definition of a model $\mathcal{M} = \langle W, (R_a)_{a \in \mathbb{A}}, V \rangle$ is thus the same as for **PA**, but with the further requirement on the valuation $V : \text{PROP} \cup \text{NOM} \rightarrow \mathcal{P}(W)$ that $|V(i)| \leq 1$ for all $i \in \text{NOM}$. For the part of the language that coincide with $\mathcal{L}_{K\Box}$ the semantic clauses are the same as for **PA**. For the new part of the language we define:

$$\begin{aligned} \mathcal{M}, w \models i &\quad \text{iff} \quad w \in V(i) \\ \mathcal{M}, w \models @_i\varphi &\quad \text{iff} \quad \text{there is a } v \in V(i) \text{ such that } \mathcal{M}, v \models \varphi \\ \mathcal{M}, w \models E\varphi &\quad \text{iff} \quad \text{there is a } v \in W \text{ such that } \mathcal{M}, v \models \varphi. \end{aligned}$$

⁶ This also works when the underlying logic is **S5**, however, if one wants to models beliefs using the logic **KD45** a problem arise. The problem is that the frame properties defined by the axioms of **KD45** are not preserved under the operation of taking submodels. Thus one cannot get completeness with respect to the class of models where beliefs are always interpreted as **KD45**. In other words the given semantic for the public announcement operator can result in agents having inconsistent beliefs after a public announcement.

The logic, this semantic give rise to, will be called “Hybrid Public Announcement Logic” and will be denoted by **HPA**. The dual operators of E and $@_i$ will be denoted by A and $\bar{@}_i$. Note, that since nominals only partially denote states, $@_i$ is no longer its own dual. We still have the equivalences $@_i\varphi \equiv E(i \wedge \varphi)$ and $\bar{@}_i\varphi \equiv A(i \rightarrow \varphi)$ though, but now these might not be equivalent anymore.⁷ Thus the satisfaction operator has been split into an existential quantifier $@$ and an universal one $\bar{@}$. The fact, that the nominal i denotes something in a model (i.e. $|V(i)| = 1$) can be expressed by the formula Ei .

Completeness with respect to a Hilbert style proof system can also be shown using reduction axioms, as the following from [8]:

- (6) $[\varphi]i \leftrightarrow (\varphi \rightarrow i)$
- (7) $[\varphi]@_i\psi \leftrightarrow (\varphi \rightarrow @_i(\varphi \wedge [\varphi]\psi))$
- (8) $[\varphi]E\psi \leftrightarrow (\varphi \rightarrow E(\varphi \wedge [\varphi]\psi))$

Since we extended the language of $\mathcal{L}_{K\Box}$ we also need to extend the definition of the measure c . This done by adding the following clauses to definition 2.1:

$$\begin{aligned} c(i) &= 1 \\ c(@_i\varphi) &= 1 + c(\varphi) \\ c(E\varphi) &= 1 + c(\varphi). \end{aligned}$$

With this complexity measure, the left hand sides of the new reduction axioms (6)–(8) still have higher c complexity than the right hand sides.

Public announcements are just one kind of epistemic actions though. To deal with a larger amount of epistemic actions in a uniform way, the notion of action models was introduced by Baltag, Moss and Solecki ([2]). The intuition behind epistemic action models is that the agents may be unsure about exactly which action takes place and that each action has a precondition that has to be satisfied for that action to take place. Epistemic actions can be represented by Kripke structures where each state is an event/action and instead of a complete valuation each event is assigned a formula of the language as a precondition.

We now turn to the formal details. An action model $M = \langle S, (Q_a)_{a \in \mathbb{A}}, \text{pre} \rangle$ consists of a finite set of events S , accessibility relations Q_a on S for all agents $a \in \mathbb{A}$, and a precondition function $\text{pre} : S \rightarrow \mathcal{L}$ assigning a precondition to every event (for some logical language \mathcal{L}). The language of formulas, $\mathcal{L}_{K\Box}$, and the action model language, $\mathcal{L}_{K\Box}^{\text{act}}$, have to be defined at the same time using mutual recursion. The action model language $\mathcal{L}_{K\Box}^{\text{act}}$ is defined by:

$$\alpha ::= (M, s),$$

where $M = \langle S, (Q_a)_{a \in \mathbb{A}}, \text{pre} \rangle$ is an action model such that $s \in S$ and $\text{pre} : S \rightarrow \mathcal{L}_{K\Box}$. At the same time the formula language $\mathcal{L}_{K\Box}$ is defined by:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \psi \mid K_a\varphi \mid [\alpha]\psi,$$

⁷ These equivalences also show that we do not need the $@_i$ operator in the language, since it is definable in terms of E and i . However to ease the adaption of the tableau system from [4] we keep $@_i$ in the language.

where $p \in \text{PROP}$, $a \in \mathbb{A}$, and $\alpha \in \mathcal{L}_{K \otimes}^{\text{act}}$.⁸ The reading of the formula $[M, s]\varphi$ is “after the epistemic action (M, s) , φ is the case”. M represent the uncertainty among the agents about what event is taking place, and s is the event that actually takes place.

For general epistemic actions a little contemplation shows that they can actually result in an enlargement of a Kripke model. The way we reflect this in the semantic is by defining a product update between a Kripke model and an action model. For a Kripke model $\mathcal{M} = \langle W, (R_a)_{a \in \mathbb{A}}, V \rangle$ and an action model $M = \langle S, (Q_a)_{a \in \mathbb{A}}, \text{pre} \rangle$ define the restricted product $\mathcal{M} \otimes M = \langle W', (R'_a)_{a \in \mathbb{A}}, V' \rangle$ by:

$$\begin{aligned} W' &= \{(w, s) \in \mathcal{M} \times M \mid \mathcal{M}, w \models \text{pre}(s)\} \\ R'_a((w, s), (v, t)) &\text{ iff } R_a(w, v) \text{ and } Q_a(s, t) \\ V'(p) &= \{(w, s) \in W' \mid w \in V(p)\}. \end{aligned}$$

We can now define the semantic of the action modality $[M, s]$ as:

$$\mathcal{M}, w \models [M, s]\varphi \text{ iff } \mathcal{M}, w \models \text{pre}(s) \text{ implies that } \mathcal{M} \otimes M, (w, s) \models \varphi.$$

The other logical operators have the normal semantic and validity is also defined in the standard way. This logic will be called Dynamic Epistemic Logic and be denoted by **AM**. Note that we have now left out the hybrid machinery since it is not obvious how exactly to combine it with action models.⁹

As in the case of public announcement, adding action modalities does not increase the expressive power of the language. Again this is shown by providing reduction axioms (see for instance [13]). The reduction axioms, which are now a little more complex, are:

$$\begin{aligned} (9) \quad & [M, s]p \leftrightarrow (\text{pre}(s) \rightarrow p) \\ (10) \quad & [M, s]\neg\varphi \leftrightarrow (\text{pre}(s) \rightarrow \neg[M, s]\varphi) \\ (11) \quad & [M, s](\varphi \wedge \psi) \leftrightarrow ([M, s]\varphi \wedge [M, s]\psi) \\ (12) \quad & [M, s]K_a\varphi \leftrightarrow (\text{pre}(s) \rightarrow \bigwedge_{R_a(s, t)} K_a[M, t]\varphi) \\ (13) \quad & [M, s][M', s']\varphi \leftrightarrow [(M; M'), (s, s')]\varphi, \end{aligned}$$

where, in the last formula, the “;” operation is a semantic operation on action models. Given two action models, $M = \langle S, (Q_a)_{a \in \mathbb{A}}, \text{pre} \rangle$ and $M' = \langle S', (Q'_a)_{a \in \mathbb{A}}, \text{pre}' \rangle$, the composition $(M; M') = \langle S'', (Q''_a)_{a \in \mathbb{A}}, \text{pre}'' \rangle$ is defined by:

$$\begin{aligned} S'' &= S \times S' \\ Q''_a((s, s'), (t, t')) &\text{ iff } Q_a(s, t) \text{ and } Q'_a(s', t') \\ \text{pre}''((s, s')) &= \langle M, s \rangle \text{pre}'(s'). \end{aligned}$$

⁸ By this definition we load the syntax of the language with heavy semantic machinery, however, since we only deal with finite action models it is possible to list and name them all. For more on this discussion see [13].

⁹ The interpretation of nominals is none obvious when modalities capable of expanding states are present. Normally, nominals in hybrid logic are a special kind of propositional variables, which are true in *exactly one* state. However, when taking a product of an epistemic model with an action model, single states of the epistemic model can turn into several states in the resulting product model. Thus, if one keeps the original definition of the valuation for the product model, one breaks the requirement of nominals only being true in one state. On the other hand, there seems to be no obvious alternative definition of the valuation.

See [13] for the validity of the reduction axioms. As for **HPA** a new complexity measure is needed. A such, taken from [13], is:

Definition 2.2 The complexity measure $d : \mathcal{L}_{K\otimes} \rightarrow \mathbb{N}$ is defined inductively by:

$$\begin{aligned}
 d(p) &= 1 \\
 d(\neg\varphi) &= 1 + d(\varphi) \\
 d(\varphi \wedge \psi) &= 1 + \max\{d(\varphi), d(\psi)\} \\
 d(K_a\varphi) &= 1 + d(\varphi) \\
 d([M, s]\varphi) &= (4 + d(M, s)) \cdot d(\varphi) \\
 d(M, s) &= \max\{d(\text{pre}(t)) \mid t \in M\}.
 \end{aligned}$$

As for public announcement it can be shown that this complexity measure decreases when moving from the left hand sides to the right hand sides (of \leftrightarrow) of the reduction axioms (9)–(13), as well as when moving to subformulas.

3 A tableaux system for AM

In this section we introduce a tableau system for **AM** build on an existing tableau system for multi modal **K**, where we add the reduction axioms as tableau rules. This is done without violating termination or completeness of the original system. Formally, we will take a tableau to be a finitely branching tree, where each node is labeled by a formula of our language. As basic tableau system for the underlying multi modal **K** logic we will use the one from [4], which is a standard one. The tableau system is a prefixed tableau system, thus all formulas occurring on the tableaux have the form $\sigma\varphi$, where σ comes from some fixed countable infinite set of prefixes. The intuition behind the prefixes is that they represent states in a possible Kripke model. Thus the intuition behind $\sigma\varphi$ is that φ holds at σ . Additionally, we also have formulas of the form $\sigma R_a\tau$ on the tableaux representing that τ is accessible from σ by agent a . These will be called accessibility formulas. The rules of the tableau system applies to branches of tableaux and are presented in Figure 1.

In the rules ($[AM]$) and ($\neg[AM]$), t is the operation that uses the reduction axioms to translate the formula $[M, s]\varphi$ to a formula of less d -complexity. For instance $t([M, s](\varphi \wedge \psi)) = [M, s]\varphi \wedge [M, s]\psi$.¹⁰ Ignoring the accessibility formulas, the formula above the line in a rule will be called the *premise* and the formula(s) below the line the *conclusion(s)*. When constructing a tableau, we never add a formula to a branch if it already occurs on the branch, and we never apply the ($\neg K_a$) rule twice to the same formula on a branch. If a branch contains both $\sigma\varphi$ and $\sigma\neg\varphi$ for some formula φ and some prefix σ , then the branch is called *closed*, otherwise *open*. A closed tableau is one in which all branches are closed. A tableau proof of a formula φ is a closed tableau with $\sigma\neg\varphi$ as the root formula.

¹⁰ t is not to be confused with a full translation for the language $\mathcal{L}_{K\otimes}$ as discussed in section 2. Here t only translate/reduces one level.

$\frac{\sigma\neg\neg\varphi}{\sigma\varphi} (\neg\neg)$	$\frac{\sigma\varphi \wedge \psi}{\sigma\varphi} (\wedge)$	$\frac{\sigma\neg(\varphi \wedge \psi)}{\sigma\neg\varphi \mid \sigma\neg\psi} (\neg\wedge)$
$\frac{\sigma\neg K_a\varphi}{\sigma R_a\tau} (\neg K_a)^1$	$\frac{\sigma K_a\varphi \quad \sigma R_a\tau}{\tau\varphi} (K_a)$	
$\frac{\sigma[M, s]\varphi}{\sigma t([M, s]\varphi)} ([AM])$	$\frac{\sigma\neg[M, s]\varphi}{\sigma\neg t([M, s]\varphi)} (\neg[AM])$	

¹ The prefix τ is new to the branch.

Fig. 1. Tableau rules **AM**.

3.1 Termination of the tableau system

Two important properties for ensuring termination in the work of [4] are; *all formulas occurring on the tableau are subformulas or negation of subformulas of the root formula*, and *every rule only generates something of less formula complexity*. These two properties are essential for ensuring finiteness of the tableaux. However, these properties fails for our tableau system because the rules $([AM])$ and $(\neg[AM])$ can generate formulas that are not subformulas of the premise and may have higher formula complexity. But using the notion of d -complexity and stretching the notion of a subformula we can retain finiteness. Before a new notion of subformula can be defined a lemma is needed. For an action model $M = \langle S, (Q_a)_{a \in A}, \text{pre} \rangle$, let $\mathcal{D}(M)$ denote the domain, i.e. $\mathcal{D}(M) = S$.

Lemma 3.1 *Let $\sigma_0\varphi_0$ be the root formula of a tableau \mathcal{T} and assume that $[M, s]\varphi$ occurs on \mathcal{T} . Then, there are an $n \leq d(\varphi_0)$ and action models M_1, \dots, M_n occurring in φ_0 , such that $\mathcal{D}(M) = \mathcal{D}(M_1) \times \dots \times \mathcal{D}(M_n)$.*

Proof. The proof goes by induction on the construction of \mathcal{T} . The claim is obvious for $[M, s]\varphi$ being φ_0 . It is also obvious that when applying any rule, besides $([AM])$ and $(\neg[AM])$ to a formula of the form $[M, s][M', s']\psi$ or $\neg[M, s][M', s']\psi$, the action modalities in the conclusions and the premises have the same domains (or the action modality have completely been removed). Now for the rules $([AM])$ and $(\neg[AM])$ applied to two consecutive modalities $[M, s]$ and $[M', s']$. Assume that the claim of the lemma is true for $[M, s]$ and $[M', s']$. Then

$$(14) \mathcal{D}((M; M'), (s, s')) = \mathcal{D}(M_1) \times \dots \times \mathcal{D}(M_n) \times \mathcal{D}(M'_1) \times \dots \times \mathcal{D}(M'_m),$$

where M_i and M'_j occur in φ_0 for all $1 \leq i \leq n$ and $1 \leq j \leq m$. Thus, the only thing that remains to be shown is that $n+m \leq d(\varphi_0)$. Note, that the complexity measure d is an upper bound for how deep the action modalities can be nested. Furthermore, for every number of nested action modalities, only one more “ $\times \mathcal{D}(M_i)$ ” can be added

by the rules $([AM])$ and $(\neg[AM])$ to (14), which also decrease the number of nested modalities by one. Thus $n + m$ must be less than $d(\varphi_0)$. \square

Definition 3.2 A formula ψ is said to be a d -subformula of a formula φ if

- $d(\psi) \leq d(\varphi)$,
- Every propositional variable p that occurs in ψ also occurs in φ .
- If an action modality $[M, s]$ occurs in ψ , then there are action models $(M_1, s_1), \dots, (M_n, s_n)$ for which M_i occurs in φ for $1 \leq i \leq n \leq d(\varphi)$, and $\mathcal{D}(M) = \mathcal{D}(M_1) \times \dots \times \mathcal{D}(M_n)$,

Note that if the action modality $[M, s]$ occurs in a formula φ , then all the preconditions of M are also counted as occurring in φ and by definition of the d -complexity we automatically have that $d(\text{pre}(t)) < d(\varphi)$ for all $t \in \mathcal{D}(M)$. Using the reduction axioms as rules result in a decrease in d -complexity, since

$$d([M, s]\varphi) > d(t([M, s]\varphi))$$

Furthermore the d -complexity decreases when moving to a strict subformula. Thus we get the following lemma and from it a subformula property.

Lemma 3.3 *For every tableau rule the d -complexity of the conclusion is strictly less than the d -complexity of the premise.*

Lemma 3.4 (d -subformula property) *Let \mathcal{T} be a tableau with $\sigma_0\varphi_0$ as root formula. Then for every prefixed formula $\sigma\varphi$ on \mathcal{T} , φ is a d -subformula of φ_0 .*

Proof. Let \mathcal{T} be a tableau with $\sigma_0\varphi_0$ as root formula. The proof goes by induction on the tableau construction. By lemma 3.3 it follows that the d -complexity of any formula occurring on \mathcal{T} is less than $d(\varphi_0)$. Moreover it is obvious that none of the rules can introduce propositional variables that do not already occur in the root formula. The only rules that can introduce new action modalities are the $([AM])$ and $(\neg[AM])$ rule applied to formulas $[M, s]K_a\varphi$, $\neg[M, s]K_a\varphi$, $[M, s][M', s']\varphi$, and $\neg[M, s][M', s']\varphi$. For the first two cases, a new action modality of the form $[M, t]$ may be introduced, but M must be the same action model as in the premise. Thus, these cases are just special cases of the third bullet in definition 3.2. For the last two cases it follows from lemma 3.1 that also these two preserve d -subformulas. \square

Definition 3.5 Given a tableau branch Θ and a prefix σ that occurs on Θ , let

$$T^\Theta(\sigma) := \{\varphi \mid \sigma\varphi \text{ is on } \Theta\},$$

Lemma 3.6 *Let Θ be a branch of a tableau and σ a prefix occurring on it. Then the set $T^\Theta(\sigma)$ is finite.*

Proof. By lemma 3.4, all formulas on Θ are d -subformulas of the root formula. Thus, the lemma follows if we can show that for all formulas φ , the set of d -subformulas of φ is finite. This can be proved by induction on $n = d(\varphi)$ given a fixed number of propositional variables N . For $n = 1$: It is obvious that there can only be N many different d -subformulas of φ , for all formulas φ with complexity $c(\varphi) = 1$. For the induction step, assume there are only finitely many d -subformulas of φ , for

all φ with $d(\varphi) \leq n$. Given a formula φ with $d(\varphi) = n + 1$, it is easy to see that any d -subformula of φ is also a d -subformula of a formula with d -complexity less than or equal to n or constructed from one of these. By induction there can only be finitely many of the first kind. For the second kind, we divide into cases depending on the structure of φ . It is easy to see that given finitely many formulas only finitely many new formulas can be constructed using the logical connectives and the K_a operators (since there are only finitely many a 's). In the case of the action modalities, note that point 3 of definition 3.2 only allows for finitely many domains of action models, and the limitation on the d -complexity of the preconditions ensures that we can only construct finitely many different action modalities. This completes the proof of the lemma. \square

Definition 3.7 Let Θ be a tableau branch and σ a prefix occurring on Θ , then define $m_\Theta(\sigma)$ by

$$m_\Theta(\sigma) = \max\{d(\varphi) \mid \sigma\varphi \in \Theta\}.$$

Note that the d -subformula property justifies that this is well-defined. We can now adopt the method of [4] to show that **AM** tableaux always terminates.

Definition 3.8 When a prefix τ has been introduced on a branch Θ by the rule $(\neg K_a)$ to a formula $\sigma\varphi$, we say that τ is *generated* by σ and denote it by $\sigma \prec_\Theta \tau$.

Following this we can easily prove, as in [4], that:

Lemma 3.9 *If Θ is a tableau branch, then Θ is infinite if and only if there exist an infinite chain of prefixes on Θ*

$$\sigma_1 \prec_\Theta \sigma_2 \prec_\Theta \sigma_3 \prec_\Theta \dots$$

Proof. See [4]. \square

Lemma 3.10 *Let Θ be a tableau branch and σ and τ two prefixes occurring on Θ . Then $\sigma \prec_\Theta \tau$ implies that $m_\Theta(\sigma) > m_\Theta(\tau)$.*

Proof. The proof carries through just as in [4], once it have been noted that the rules $([AM])$ and $(\neg[AM])$ decrease the d -complexity from the premise to the conclusion, and that none of these rules introduce new prefixes. \square

As in [4] termination now easily follows from the lemmas 3.9 and 3.10:

Theorem 3.11 (Termination of the tableau system) *Any tableau constructed for a $\mathcal{L}_{K\otimes}$ -formula is finite.*

3.2 Soundness and completeness of the tableau system

Soundness is not hard to prove. The rules for the underlying multi modal logic are standard and easily seen to be sound. By the validity of the reduction axioms (9)–(13), the soundness of the rules $([AM])$ and $(\neg[AM])$ follows.

The completeness for the underlying multi modal logic using only the rules involving this part of the language is already well known (see for instance [4]). Given

an open saturated branch Θ , a canonical model \mathcal{M} is constructed from the prefixes occurring on Θ and the accessibility relations are defined by which accessibility formulas $\sigma R_a \tau$ occur on Θ . The valuation of a propositional variable p is defined relative to which of σp and $\sigma \neg p$ (if any) occurs on Θ . It is then straightforward to prove a truth lemma stating that; for all prefixed formulas $\sigma \varphi$ on Θ ,

$$\mathcal{M}, \sigma \models \varphi.$$

For our tableau system this construction and the formulation of the truth lemma are identical. However, instead of proving the truth lemma by induction on formula complexity, we prove it by induction on the d -complexity and add two new cases for $[M, s]\varphi$ and $\neg[M, s]\varphi$. These cases are, however, quite straightforward: Assume that $\sigma[M, s]\varphi$ occurs on Θ . Then by saturation of Θ , $\sigma t([M, s]\varphi)$ also occurs on Θ and since $t([M, s]\varphi)$ has less d -complexity than $\sigma[M, s]\varphi$, it follows by induction that $\mathcal{M}, \sigma \models t([M, s]\varphi)$. But then, by the validity of the reduction axioms (9)–(13), it follow that also $\mathcal{M}, \sigma \models [M, s]\varphi$. The case for $\neg[M, s]\varphi$ is similar. Thus we get:

Theorem 3.12 *The tableau system of figure 1 is sound and complete with respect to the logic **AM**.*

4 A tableau for hybrid public announcement logic

In this section, we introduce a tableau system for **HPA**. It is both simpler and more complicated than the tableau system of the previous section. The simplification consist in looking purely at public announcement, where as the complication consist in extending the underlying epistemic logic to a hybrid logic. The simplification shortens the proof of the lemmas 3.4 and 3.6 considerably, but the hybrid machinery makes us in need of a more advanced termination proof as in [4]. Our tableau system will be based on a small modification of the one in [4], further extended with reduction axiom rules for public announcement. We reuse all of the terminology of section 3. The tableau rules are given in figure 2.

In the rules (\Box) and ($\neg\Box$), the operation t is defined via the reduction axioms for **HPA**, in the same way as in the previous section. Compared to [4] one rule has also been left out, and the rule ($\neg\@$) has been alternated. Both changes have been made to deal with the fact that nominals in our logic only partially denote states. The rules ($\neg K_a$), ($\@$), ($\neg\@$) and (E) are called *prefix generating* rules. The construction of a tableau is done with the constraints that no prefix generation rule is applied twice to the same premise on the same branch, and a formula is never added to the branch if it already occurs on that branch. Furthermore, to make the tableaux terminate, we introduce (as in [4]) a loop-checking mechanism. Before this we need the notion of an “urfurther”.

Definition 4.1 Given a branch Θ , the prefix τ is an *urfather*¹¹ of the prefix σ if τ is the earliest introduced prefix on Θ such that $T^\Theta(\sigma) \subseteq T^\Theta(\tau)$. We denote this by $u_\Theta(\sigma) = \tau$.

¹¹ This notion of an urfather used here is called an inclusion urfather in [4].

$$\begin{array}{c}
\frac{\sigma \neg \neg \varphi}{\sigma \varphi} (\neg \neg) \quad \frac{\sigma \varphi \wedge \psi}{\sigma \varphi \quad \sigma \psi} (\wedge) \quad \frac{\sigma \neg(\varphi \wedge \psi)}{\sigma \neg \varphi \quad \sigma \neg \psi} (\neg \wedge) \\
\\
\frac{\sigma \neg K_a \varphi}{\sigma R_a \tau \quad \tau \neg \varphi} (\neg K_a)^1 \quad \frac{\sigma K_a \varphi \quad \sigma R_a \tau}{\tau \varphi} (K_a) \quad \frac{\sigma E \varphi}{\tau \varphi} (E)^1 \quad \frac{\sigma \neg E \varphi}{\tau \neg \varphi} (\neg E)^2 \\
\\
\frac{\sigma @_i \varphi}{\tau i \quad \tau \varphi} (@)^1 \quad \frac{\sigma \neg @_i \varphi}{\sigma \neg E i \quad \tau i \quad \tau \neg \varphi} (\neg @)^1 \quad \frac{\sigma \varphi \quad \sigma i \quad \tau i}{\tau \varphi} (Id) \\
\\
\frac{\sigma [\varphi] \psi}{\sigma t([\varphi] \psi)} ([\varphi]) \quad \frac{\sigma \neg [\varphi] \psi}{\sigma \neg t([\varphi] \psi)} (\neg [\varphi])
\end{array}$$

¹ The prefix τ is new to the branch. ² The prefix τ is already on the branch.

Fig. 2. Tableau rules for **HPA**.

The construction of a **HPA** tableau is subject to the following constraint:

A prefix generating rule is only allowed to be applied to a formula $\sigma \varphi$ on a branch if σ is an urfather on that branch.

4.1 Termination of **HPA** tableaux

As in the general action model case, we need an extended notion of subformulas based on the complexity measure c of definition 2.1.

Definition 4.2 A formula ψ is said to be a c -subformula of a formula φ if

- $c(\psi) \leq c(\varphi)$
- Every propositional variable and all the nominals that occur in ψ also occur in φ .

In the case of **HPA** tableaux, the following can straightforwardly be proven:

Lemma 4.3 *For every tableau rule the c -complexity of the conclusion is less than the c -complexity of the premise.*

Lemma 4.4 (c -subformula property) *Let \mathcal{T} be a tableau with root formula $\sigma \varphi$. If the prefixed formula $\tau \psi$ occurs on \mathcal{T} , then ψ is a c -subformula of φ .*

Proof. By lemma 4.3 and the fact that no rules can introduce new nominals or propositional variables, it is easy to check for all the rules that if they have c -subformulas as premises, the conclusions will also be c -subformulas.

Note that the rule $(\neg @)$ can only be applied if a prefixed formula $\tau \neg @_i \chi$ occurs on the tableau, in which case, by induction $c(\neg @_i \chi) \leq c(\varphi)$. Thus, it follows that $c(\neg Ei) \leq c(\varphi)$, and hence all formulas of the form $\tau \neg Ei$ occurring on \mathcal{T} will, also be c -subformulas of the root formula φ . \square

The following lemma, on the other hand, is easier to prove in the case of **HPA**.

Lemma 4.5 *For all tableau branches Θ and prefixes σ occurring on Θ , the set $T^\Theta(\sigma)$ is finite.*

Proof. From lemma 4.4 it follows that $T^\Theta(\sigma)$ is a subset of the set of all c -subformulas of the root formula of Θ . Thus the lemma follows if we can show that for all formulas φ , the set of c -subformulas of φ is finite. The proof of this is similar to the proof of lemma 3.6, but easier, since the public announcement operator is not as complicated as the action modalities. \square

We now extend the ordering \prec_Θ introduced in the previous section. Let Θ be a tableau branch. If a prefix τ has been introduced to the branch using a prefix generating rule on a formula of the form $\sigma\varphi$, we say that τ is *generated by* σ and write $\sigma \prec_\Theta \tau$. It is straightforward to show that lemma 3.9 remains true in this case.

The rest of the proof of termination is identical to the proof of Theorem 5.4 in [4]. The only difference is that their notion of quasi-subformula has to be replaced by our notion of c -subformula. Thus we get that:

Theorem 4.6 *Any tableau constructed using the given rules for **HPA** is finite, and the logic **HPA** is thus decidable.*

4.2 Soundness and completeness of the tableau system for **HPA**

Again, as for **AM**, the proof of soundness is simple. The completeness is also almost as in [4]. The only modification needed is because nominals only partly denote in **HPA**. The reduction axiom rules are dealt with as for the tableau system for **AM**. Given an open saturated branch Θ , a model $\mathcal{M}^\Theta = \langle W^\Theta, R^\Theta, V^\Theta \rangle$ is constructed as in [4] by:

$$\begin{aligned} W^\Theta &= \{\sigma \mid \sigma \text{ is an urfather on } \Theta\} \\ R_a^\Theta &= \{(\sigma, u_\Theta(\tau)) \in W^\Theta \times W^\Theta \mid \sigma R_a \tau \text{ occurs on } \Theta\} \\ V^\Theta(x) &= \{u_\Theta(\sigma) \in W^\Theta \mid \sigma x \text{ occurs on } \Theta\}, \text{ for all } x \in \text{PROP} \cup \text{NOM}. \end{aligned}$$

For V to be well-defined, we have to make sure that $|V^\Theta(i)| \leq 1$ for all nominals i . If there were two different urfathers σ and τ and a nominal i , such that both σi and τi occurred on Θ , then using the saturation of the branch and the rule (Id) , we would get that $T^\Theta(\sigma) = T^\Theta(\tau)$. However, since they were both urfathers, this would imply that $\sigma = \tau$, which is a contradiction. Thus V is well-defined. Now completeness follows from the theorem:

Theorem 4.7 *Let Θ be an open saturated branch for the tableau system. For any formula $\sigma\varphi$ occurring on Θ , with σ an urfather, it holds that $\mathcal{M}^\Theta, \sigma \models \varphi$.*

Proof. The proof goes by induction on the complexity of φ . The basic cases follow from the definition of V^Θ . The cases $\varphi = K_a\psi$, $\varphi = \neg K_a\psi$, $\varphi = E\psi$, $\varphi = \neg E\psi$, and $\varphi = @_i\psi$ are as in [4].

In the case of $\varphi = \neg @_i\psi$ a little more work is required. Assume that $\sigma \neg @_i\varphi$ occurs on Θ and that σ is an urfather. Then by the closure of the rule $(\neg @)$ either $\sigma \neg Ei$ or τi and $\tau \neg \varphi$ occur on the branch. In the first case, there can be no prefix σ' such that $\sigma' i$ is on Θ . This is because the rule $\neg E$ gives that $\sigma' \neg i$ is also on Θ , which contradicts the assumption that Θ is an open branch. But then there can be no state in \mathcal{M}^Θ , which i denotes. Thus by the semantic $\mathcal{M}^\Theta, \sigma \models \neg @_i\varphi$. On the other hand if τi and $\tau \neg \varphi$ are on Θ for a prefix τ , then by urfather closure, also $u_\Theta(\tau)i$ and $u_\Theta(\tau)\neg \varphi$ are on Θ , which by the induction hypothesis gives that $\mathcal{M}^\Theta, u_\Theta(\tau) \models i$ and $\mathcal{M}^\Theta, u_\Theta(\tau) \models \neg \varphi$. Thus we get that $\mathcal{M}^\Theta, \sigma \models \neg @_i\varphi$. \square

As a consequence of this we get that.

Theorem 4.8 *The tableau system of figure 2 is sound and complete with respect to the logic **HPA**.*

5 Concluding remarks and further research

In this paper, we have presented two tableau systems; one for dynamic epistemic logic with action models and one for a hybrid public announcement logic (both without common knowledge). These were based on already existing tableau systems to which we simply added tableau rules corresponding to the reduction axioms of the two logics. Following this, we showed that the method used to prove termination in [4], can also be extended to our new tableau systems.

There are already tableau systems for **PA**, [7] and [1], of which the one in [1] is shown to be optimal with respect to complexity. However, these only work for **PA** and cannot be generalized to other DELs in an obvious way. The aim of this paper has not been to construct complexity optimal tableau systems, but to show how tableau systems can be obtained in a more general way for various DELs.

Due to the unknown complexity status of **AM** and the problem of how exactly to measure the length of formulas, it is unknown whether the tableau method here presented is optimal with respect to complexity. However, it does seem to provide some kind of exponential upper bound. In the case of **HPA**, the underlying hybrid logic has an EXPTIME complexity as it contains the global modality [11]. Again, the system here presented seems also to yield an exponential upper bound in this case. The exact complexity details are left for further research.

Presently, there exist no tableau systems (known to the author) for DELs extended with common knowledge, and, due to the lack of reduction axioms, our method cannot be used. However, in [12] it is shown that a generalization of common knowledge called “relativized common knowledge” allows for reduction axioms for the public announcement operator. Thus, if tableau systems can be constructed for a multi-modal logic extended with relativized common knowledge, the method here presented may be extendable to give a terminating tableau system for a public

announcement logic with a form of common knowledge. The relativized common knowledge is actually nothing else than the until operator from temporal logics interpreted over arbitrary Kripke frames. Hence, it might be possible to extend tableau systems from temporal logics to public announcement logics with relativized common knowledge.

An even more general setting for reduction axioms has been given by Barteld Kooi in [9]. A further direction of research would be to extend the methods presented here in order to make them work in that setting.

A final matter of concern is the choice to only deal with logics where the underlying modal logic is multi modal **K**. In epistemic logics, you usually add extra requirements to the agents' accessibility relations, which causes the underlying modal logic to change into for instance **S5** or **KD45**. It is therefore important to be able to extend the presented tableau systems to also deal with these cases. The methods here presented are based on the paper [4], which fortunately has a follow-up paper ([3]) that deals with the problems of adding extra conditions to the accessibility relations. It seems possible to use that work in connection with the tableau systems presented in this paper, but the exact details are left for further research.

References

- [1] Balbiani, P., H. Van Ditmarsch, A. Herzig and T. De Lima, *Tableaux for Public Announcement Logic*, Journal of Logic and Computation (2008), p. exn060.
URL <http://logcom.oxfordjournals.org/cgi/content/abstract/exn060v1>
- [2] Baltag, A., L. S. Moss and S. Solecki, *The logic of public announcements, common knowledge and private suspicious*, Technical Report SEN-R9922, CWI, Amsterdam (1999).
- [3] Bolander, T. and P. Blackburn, *Terminating tableau calculi for hybrid logics extending K*, in: C. Areces and S. Demri, editors, *Workshop Proceedings of Methods for Modalities 5*, École Normale Supérieure de Cachan, France, 2007 pp. 157–175.
- [4] Bolander, T. and P. Blackburn, *Termination for hybrid tableaux*, Journal of Logic and Computation **17** (2007), pp. 517–554.
- [5] Bolander, T. and T. Braüner, *Two tableau-based decision procedures for hybrid logic*, in: *Proceedings of M4M (Methods For Modalities) 4*, Humboldt University, Germany, 2005 pp. 79–96.
- [6] Bolander, T. and T. Braüner, *Tableau-based decision procedures for hybrid logic*, Journal of Logic and Computation **16** (2006), pp. 737–763.
- [7] de Boer, M. S., *KE tableaux for public announcement logic*, in: *Proceedings of Formal Approaches to Multi-Agent Systems Workshop (FAMAS 07)*, Durham, UK, 2007 .
- [8] Hansen, J. U., *Hybrid public announcement logic*, submitted (2009).
- [9] Kooi, B., *Expressivity and completeness for public update logics via reduction axioms*, Journal of Applied Non-Classical Logics **17** (2007), pp. 231–153.
- [10] Lutz, C., *Complexity and succinctness of public announcement logic*, in: *AAMAS '06: Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems* (2006), pp. 137–143.
- [11] Spaan, E., “Complexity of modal logics,” Ph.D. thesis, Institute for logic, Language and Computation, Universiteit van Amsterdam, Amsterdam, The Netherlands (1993).
- [12] van Benthem, J., J. van Eijck and B. Kooi, *Logics of communication and change*, Information and Computation **204** (2006), pp. 1620–1662.
- [13] van Ditmarsch, H., W. van der Hoek and B. Kooi, “Dynamic Epistemic Logic,” Syntese Library volume 337, Springer, The Netherlands, 2008.