Undecidability in Epistemic Planning

Guillaume Aucher, Thomas Bolander

Undecidability in Epistemic Planning (extended version)

Guillaume Aucher
, Thomas Bolander

RESEARCH REPORT
N° 8310
April 2013
Project-Team S4

ISSN 0249-6399 ISRN INRIA/RR--8310--FR+ENG

Undecidability in Epistemic Planning (extended version)

Guillaume Aucher*
, Thomas Bolander†

Project-Team S4

Research Report n° 8310 — April 2013 — 21 pages

Abstract: Dynamic epistemic logic (DEL) provides a very expressive framework for multi-agent planning that can deal with nondeterminism, partial observability, sensing actions, and arbitrary nesting of beliefs about other agents' beliefs. However, as we show in this paper, this expressiveness comes at a price. The planning framework is undecidable, even if we allow only purely epistemic actions (actions that change only beliefs, not ontic facts). Undecidability holds already in the S5 setting with at least 2 agents, and even with 1 agent in S4. It shows that multi-agent planning is robustly undecidable if we assume that agents can reason with an arbitrary nesting of beliefs about beliefs. We also prove a corollary showing undecidability of the DEL model checking problem with the star operator on actions (iteration).

Key-words: Automated planning, dynamic epistemic logic, multi-agent systems, undecidability.

This research report is an extended version of the article published in the proceedings of the International Joint Conference on Artificial Intelligence (IJCAI), Beijing, August 2013.

* Université de Rennes 1 - INRIA, France
† DTU Copenhagen, Denmark

RESEARCH CENTRE
RENNES – BRETAGNE ATLANTIQUE

Campus universitaire de Beaulieu
35042 Rennes Cedex

Indécidabilité en planification épistémique
(version étendue)

Résumé : La logique épistémique dynamique (DEL) est un formalisme logique très expressif pour la planification épistémique qui permet de rendre compte du non-déterminisme, de l'observation partielle, des actions de perception, et de croyances d'ordre supérieur portant sur les croyances d'autres agents. Cependant, comme nous le montrons dans ce rapport, cette expressivité a un coût. Ce formalisme pour la planification est indécidable, et cela même si nous nous restreignons aux actions purement épistémiques (les actions qui changent seulement les croyances, pas les faits du monde). L'indécidabilité est déjà présente dans le cadre de la logique S5 avec au moins 2 agents, et même dans le cas d'un seul agent avec la logique S4. Cela montre que la planification multi-agent est indécidable de façon robuste si on suppose que les agents peuvent raisonner avec un enchevêtrement arbitraire de croyances d'ordre supérieur sur les croyances d'autres agents. Nous prouvons aussi un corollaire montrant que le problème du model checking de la logique épistémique dynamique (DEL) est indécidable avec l'opérateur étoile sur les actions (itération).

Mots-clés : Planification automatique, logique épistémique dynamique, systèmes multi-agents, indécidabilité.

Contents

1 Introduction

Recently a number of authors have independently started developing new and very expressive frameworks for automated planning based on dynamic epistemic logic [Bolander and Andersen, 2011, Löwe et al., 2011, Aucher, 2012, Pardo and Sadrzadeh, 2012]. Dynamic epistemic logic (DEL) extends ordinary modal epistemic logic [Hintikka, 1962] by the inclusion of *event models* to describe actions, and a *product update* operator that defines how epistemic models are updated as the consequence of executing actions described through event models [Baltag et al., 1998]. Using epistemic models as states, event models as actions, and the product operator as state transition function, one immediately gets a planning formalism based on DEL.

One of the main advantages of this formalism is expressiveness. Letting states of planning tasks be epistemic models implies that we have something that generalizes *belief states*, the classical approach to planning with nondeterminism and partial observability [Ghallab et al., 2004].[1] Compared to standard planning formalisms using belief states, the DEL-based approach has the advantage that actions (event models) encode both nondeterminism and partial observability [Andersen et al., 2012], and hence that observability can be action dependent, and we do not need observation functions on top of action descriptions. Active sensing actions are also expressible in the DEL-based framework. Another advantage of the DEL-based framework is that it generalizes immediately to the multi-agent case. Both epistemic logic and DEL are by default multi-agent formalisms, and the single-agent situation is simply a special case. Hence the formalism provides a planning framework for multi-agent planning integrating nondeterminism and partial observability. It can be used both for adversarial and cooperative multi-agent planning. Finally, the underlying epistemic logic also allows agents to represent their beliefs about the beliefs of other agents, hence allowing them to do Theory of Mind modeling. Theory of Mind (ToM) is a concept from cognitive psychology referring to the ability of attributing mental states (beliefs, intentions, etc.) to other agents [Premack and Woodruff, 1978]. Having a ToM is essential for successful social interaction in human agents [Baron-Cohen, 1997], hence can be expected to play an equally important role in the construction of socially intelligent artificial agents.

The flip side of the expressivity advantages of the DEL-based planning framework is that the plan existence problem is undecidable in the unrestricted framework. This was proven in [Bolander and Andersen, 2011] by an encoding of Turing machines as 3-agent planning tasks (leading to a reduction of the Turing machine halting problem to the 3-agent plan existence problem). The proof made essential use of actions with postconditions, that is, ontic actions that make factual changes to the world (e.g. writing a symbol to a tape cell of a Turing machine). One could speculate that undecidability relied essentially on the inclusion of ontic actions, but in the present paper we prove this not to be the case. We prove that plan existence is undecidable even when only allowing purely epistemic (non-ontic) actions, and already for 2 agents. This is by an encoding of two-counter machines as planning tasks. We also prove that even single-agent planning is undecidable on S4 frames.

Given that we deal with multi-agent situations, it is important to specify our modeling approach, and in particular whether the modeler/planner is one of the agents. A classification of the different modeling approaches and their respective formalisms can be found in [Aucher, 2010]. For ease of presentation, we follow in this article the perfect external approach of (dynamic) epistemic logic and model the situation from an external and omniscient point of view. This said, all our results in this article transfer to the other modeling approaches if we replace epistemic models with internal models or imperfect external models (*i.e.* multi-pointed models), which, as we said, generalize to a multi-agent setting the belief states of classical planning [Bolander and Andersen, 2011] .

The article is structured as follows. In Section 2, we recall the core of the DEL framework. In Section 3, we relate our DEL-based approach to the classical planning approach and provide an example of an

[1] A belief state can be modeled as a set of propositional valuations, which again can be modeled as a connected S5 model of epistemic logic.

epistemic multi-agent planning task. In Section 4, we introduce two-counter machines which are used in Sections 5 and 6 to prove our undecidability results. In Section 7, we derive from our results the undecidability of the DEL model checking problem (for the language with the star operator on actions). Finally, we discuss related work and end with some concluding remarks in Section 8.

2 Dynamic Epistemic Logic

In this section, we present the basic notions from DEL required for the rest of the article (see [Baltag et al., 1998, van Ditmarsch et al., 2007, van Benthem, 2011] for more details). Following the DEL methodology, we split our exposition into three subsections. In Section 2.1, we recall the syntax and semantics of the epistemic language. In Section 2.2, we define event models, and in Section 2.3, we define the product update. Finally, in Section 2.4, we define specific classes of epistemic and event models that will be studied in the sequel.

2.1 Epistemic Models

Throughout this article, P is a countable set of atomic propositions (propositional symbols) of cardinality at least two, and $\mathcal{A}$ is a non-empty finite set of agents. We will use symbols $p, q, r, \ldots$ for atomic propositions and numbers $0, 1, \ldots$ for agents. The epistemic language $\mathcal{L}(P, \mathcal{A})$ is generated by the following BNF:

$$\phi \ ::= \ p \ \mid \ \neg\phi \ \mid \ \phi \wedge \phi \ \mid \ \Box_i \phi$$

where $p \in P$ and $i \in \mathcal{A}$. As usual, the intended interpretation of a formula $\Box_i\phi$ is "agent i believes ϕ" or "agent i knows ϕ". The formulas $\Diamond_i\phi$, $\phi \vee \psi$ and $\phi \to \psi$ are abbreviations of $\neg\Box_i\neg\phi$, $\neg(\neg\phi \wedge \neg\psi)$, and $\neg\phi \vee \psi$ respectively. We define $\top$ as an abbreviation for $p \vee \neg p$ and $\perp$ as an abbreviation for $p \wedge \neg p$ for some arbitrarily chosen $p \in P$. The semantics of $\mathcal{L}(P, \mathcal{A})$ is defined as usual through Kripke models, here called *epistemic models*.

Definition 1 (Epistemic models and states). An *epistemic model of* $\mathcal{L}(P, \mathcal{A})$ is a triple $\mathcal{M} = (W, R, V)$, where

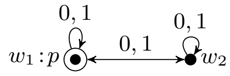
- W, the *domain*, is a finite set of *worlds*;
- $R : \mathcal{A} \to 2^{W \times W}$ assigns an *accessibility relation* $R(i)$ to each agent $i \in \mathcal{A}$;
- $V : P \to 2^W$ assigns a set of worlds to each atomic proposition; this is the *valuation* of that variable.

The relation $R(i)$ is usually abbreviated R_i, and we write $v \in R_i(w)$ or wR_iv when $(w, v) \in R(i)$. For $w \in W$, the pair $(\mathcal{M}, w)$ is called an *epistemic state of* $\mathcal{L}(P, \mathcal{A})$.

Definition 2 (Truth conditions). Let an epistemic model $\mathcal{M} = (W, R, V)$ be given. Let $i \in \mathcal{A}$, $w \in W$ and $\phi, \psi \in \mathcal{L}(P, \mathcal{A})$. Then

$$\begin{array}{lll} \mathcal{M}, w \models p & \text{iff} & w \in V(p) \\ \mathcal{M}, w \models \neg\phi & \text{iff} & \mathcal{M}, w \not\models \phi \\ \mathcal{M}, w \models \phi \wedge \psi & \text{iff} & \mathcal{M}, w \models \phi \text{ and } \mathcal{M}, w \models \psi \\ \mathcal{M}, w \models \Box_i\phi & \text{iff} & \text{for all } v \in R_i(w), \mathcal{M}, v \models \phi \end{array}$$

Example 1. Consider the following epistemic state of $\mathcal{L}(\{p\}, \{0, 1\})$.

$(\mathcal{M}, w_1) =$ 

Each world is marked by its name followed by a list of the propositional symbols being true at the world (which is possibly empty if none holds true). Edges are labelled with the name of the relevant accessibility relations (agents). We have e.g. $(\mathcal{M}, w_1) \models \neg\Box_i p \wedge \neg\Box_i \neg p$ for $i = 0, 1$: neither agent knows the truth-value of p. For epistemic states $(\mathcal{M}, w)$ we use the symbol $\odot$ to mark the *designated world* w.

2.2 Event Models

Dynamic Epistemic Logic (DEL) introduces the concept of *event model* (or *action model*) for modeling the changes to epistemic states brought about by the execution of actions [Baltag et al., 1998, Baltag and Moss, 2004]. Intuitively, in Definition 3 below, eQ_ie' means that while the possible event represented by e is occurring, agent i considers possible that the event represented by e' is in fact occurring.

Definition 3 (Event models and epistemic actions). An *event model of* $\mathcal{L}(P, \mathcal{A})$ is a triple $\mathcal{E} = (E, Q, pre)$, where

- E, the *domain*, is a finite non-empty set of *events*;
- $Q : \mathcal{A} \to 2^{E \times E}$ assigns an *accessibility relation* $Q(i)$ to each agent $i \in \mathcal{A}$;
- $pre : E \to \mathcal{L}(P, \mathcal{A})$ assigns to each event a *precondition*.

The relation $Q(i)$ is generally abbreviated Q_i, and we write $v \in Q_i(w)$ or wQ_iv when $(w, v) \in Q(i)$. For $e \in E$, $(\mathcal{E}, e)$ is called an *epistemic action of* $\mathcal{L}(P, \mathcal{A})$.

The event e in $(\mathcal{E}, e)$ is intended to denote the *actual event* that takes place when the action is executed. Note that we assume that events do not cause factual changes in the world. Hence, we only consider so-called *epistemic events* and not ontic events with postconditions, as in [van Ditmarsch et al., 2005, van Benthem et al., 2006]. Our assumptions for dealing with epistemic planning will therefore also differ from the assumptions used in [Bolander and Andersen, 2011].

2.3 Product Update

Definition 4 (Applicability). An epistemic action $(\mathcal{E}, e)$ is *applicable* in an epistemic state $(\mathcal{M}, w)$ if $(\mathcal{M}, w) \models pre(e)$.

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

Definition 5 (Product update). Given an epistemic action $(\mathcal{E}, e)$ applicable in an epistemic state $(\mathcal{M}, w)$, where $\mathcal{M} = (W, R, V)$ and $\mathcal{E} = (E, Q, pre)$. The *product update* of $(\mathcal{M}, w)$ with $(\mathcal{E}, e)$ is defined as the epistemic state $(\mathcal{M}, w) \otimes (\mathcal{E}, e) = ((W', R', V'), (w, e))$, where

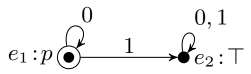
$$
\begin{aligned}
W' &= \{(w, e) \in W \times E \mid \mathcal{M}, w \models pre(e)\} \\
R'_i &= \{((w, e), (v, f)) \in W' \times W' \mid wR_iv \text{ and } eQ_if\} \\
V'(p) &= \{(w, e) \in W' \mid \mathcal{M}, w \models p\}.
\end{aligned}
$$

L	transitive	Euclidean	reflexive
K			
KT			✓
K4	✓		
K45	✓	✓	
S4	✓		✓
S5	✓	✓	✓

Fig. 1: L-epistemic states and actions

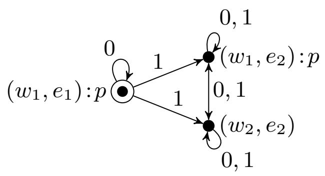
Example 2. Continuing Example 1, the following is an example of an applicable epistemic action of $\mathcal{L}(\{p\}, \{0, 1\})$ in $(\mathcal{M}, w_1)$:

$(\mathcal{E}_1, e_1) =$



It corresponds to a private announcement of p to agent 0, that is, agent 0 is told that p holds (event e_1), but agent 1 thinks that nothing has happened (event e_2). The product update is calculated as follows:

$(\mathcal{M}, w_1) \otimes (\mathcal{E}_1, e_1) =$



In the updated state, agent 0 knows p (since $\Box_0 p$ holds at (w_1, e_1)), but agent 1 didn't learn anything (doesn't know p and believes that 0 doesn't either).

2.4 Classes of Epistemic States and Actions

In this article, we consider epistemic states and actions where the accessibility relations satisfy specific properties, namely *transitivity* (for all w, v, u, wR_iv and vR_iu imply wR_iu, defined by the axiom 4: $\Box_i\phi \rightarrow \Box_i\Box_i\phi$), *Euclidicity* (for all w, v, u, wR_iv and wR_iu imply vR_iu, defined by the axiom 5: $\neg\Box_i\phi \rightarrow \Box_i\neg\Box_i\phi$) and *reflexivity* (for all w, wR_iw, defined by the axiom T: $\Box_i\phi \rightarrow \phi$).

Different conditions on the accessibility relations correspond to different assumptions on the notions of knowledge or belief [Fagin et al., 1995, Meyer and van der Hoek, 1995]. Modal logics of belief are usually considered to be at least as strong as K45, *i.e.*, they should validate at least modus ponens, necessitation (from ϕ, infer $\Box_i\phi$) and positive and negative introspection (Axioms 4 and 5 respectively). Modal logics of knowledge are usually considered to be at least as strong as S4, *i.e.*, they should validate modus ponens, necessitation, Truth (Axiom T) and positive introspection (Axiom 4). In the literature, it is often assumed that the logic of knowledge is S5, *i.e.*, it validates moreover negative introspection (Axiom 5). In the sequel we will refer to L-epistemic states and actions, where the conditions on these models is given in Figure 1.

These classes of epistemic and event models are *stable for the product update*, *i.e.*, for all $\mathsf{L} \in \{\mathsf{K}, \mathsf{KT}, \mathsf{K4}, \mathsf{K45}, \mathsf{S4}, \mathsf{S5}\}$, if $(\mathcal{M}, w)$ is a pointed L-epistemic model and $(\mathcal{E}, e)$ is a pointed L-event model applicable in $(\mathcal{M}, w)$ then $(\mathcal{M}, w) \otimes (\mathcal{E}, e)$ is a pointed L-epistemic model.

3 Classical and Epistemic Planning

In this section, we briefly relate the epistemic planning approach of DEL as propounded in [Bolander and Andersen, 2011, Löwe et al., 2011] with the classical planning approach [Ghallab et al., 2004]. For more detailed connections, we refer the reader to [Bolander and Andersen, 2011].

3.1 Classical Planning

Following [Ghallab et al., 2004], any classical planning domain can be represented as a restricted state-transition system.

Definition 6 (Restricted state-transition system). A *restricted state-transition system* is a tuple $\Sigma = (S, A, \gamma)$, where

- S is a finite or recursively enumerable set of *states*;
- A is a finite set of *actions*;
- $\gamma : S \times A \hookrightarrow S$ is a partial and computable *state-transition function*. The state-transition function is partial, *i.e.*, for any $(s, a) \in S \times A$, either $\gamma(s, a)$ is undefined or $\gamma(s, a) \in S$.

Definition 7 (Classical planning task). A *classical planning task* is represented as a triple (Σ, s_0, S_g), where

- Σ is a *restricted state-transition system*;
- s_0 is the *initial state*, a member of S;
- S_g is the set of *goal states*, a subset of S.

Definition 8 (Solution to a classical planning task). A *solution* to a classical planning task (Σ, s_0, S_g) is a finite sequence of actions (a *plan*) $a_1, a_2, \dots, a_n$ such that:

1. For all $i \leq n$, $\gamma(\gamma(\dots\gamma(\gamma(s_0, a_1), a_2), \dots, a_{i-1}), a_i)$ is defined;
2. $\gamma(\gamma(\dots\gamma(\gamma(s_0, a_1), a_2), \dots, a_{n-1}), a_n) \in S_g$.

Note that finding solutions to classical planning tasks is always at least semi-decidable: given a planning problem, we can compute its *state space* (the space of states reachable by a sequence of actions applied to the initial state) in a breadth-first manner, and if one of the goal states is reachable, we will eventually find it.

3.2 Epistemic Planning

Here is the definition of epistemic planning tasks, which are special cases of classical planning tasks:

Definition 9 (Epistemic planning tasks). An *epistemic planning task* is a triple (s_0, A, ϕ_g) where s_0 is a finite epistemic state, the *initial state*; A is a finite set of finite epistemic actions; ϕ_g is a formula in $\mathcal{L}(P, \mathcal{A})$, the *goal formula*.

Any epistemic planning task (s_0, A, ϕ_g) canonically induces a classical planning task $((S, A, \gamma), s_0, S_g)$ given by:

- $S = \{s_0 \otimes a_1 \otimes \dots \otimes a_n \mid n \geq 0, a_i \in A\}$.

- $S_g = \{s \in S \mid s \models \phi_g\}$.
- $\gamma(s, a) = \begin{cases} s \otimes a & \text{if } a \text{ is applicable in } s \\ \text{undefined} & \text{otherwise.} \end{cases}$

Hence, epistemic planning tasks are special cases of classical planning tasks. A *solution* to an epistemic planning task (s_0, A, ϕ_g) is a solution to the induced classical planning task.

Example 3. Let a_1 denote the epistemic action $(\mathcal{E}_1, e_1)$ of Example 2 and let a_2 denote the result of replacing 0 by 1 and 1 by 0 everywhere in a_1. The epistemic action a_2 is a private announcement of p to agent 1. Now consider an epistemic planning task (s_0, A, ϕ_g), where $s_0 = (\mathcal{M}, w_1)$ is the epistemic state from Example 1, and $A \supseteq \{a_1, a_2\}$. Let the goal be that both 0 and 1 know p, but don't know that each other knows: $\phi_g = \Box_0 p \wedge \Box_1 p \wedge \neg \Box_0 \Box_1 p \wedge \neg \Box_1 \Box_0 p$. It is easy to check that a solution to this epistemic planning task is the action sequence a_1, a_2, since we have $s_0 \otimes a_1 \otimes a_2 \models \phi_g$. Hence a solution to the task of both agents knowing p without suspecting that each other does, is to first announce p privately to 0 then privately to 1.

The following definition is adapted from [Erol et al., 1995].

Definition 10 (Plan existence problem). Let $n \geq 1$ and $\mathsf{L} \in \{\mathsf{K}, \mathsf{KT}, \mathsf{K4}, \mathsf{K45}, \mathsf{S4}, \mathsf{S5}\}$. $\mathsf{PlanEx}(\mathsf{L}, n)$ is the following decision problem: "Given an epistemic planning task $\mathcal{T} = (s_0, A, \phi_g)$ where s_0 is an L-epistemic state, A is a set of L-epistemic actions and $|\mathcal{A}| = n$, does $\mathcal{T}$ have a solution?"

4 Two-counter Machines

We will prove undecidability of the plan existence problem, $\textsc{PlanEx}(\mathsf{L}, n)$, for various classes of epistemic planning tasks. Each proof is by a reduction of the halting problem for two-counter machines to the plan existence problem for the relevant class of planning tasks. So, we first introduce two-counter machines [Minsky, 1967, Hampson and Kurucz, 2012].

Definition 11 (Two-counter machines). A *two-counter machine* M is a finite sequence of instructions $(I_0, \ldots, I_T)$, where each instruction I_t, for $t < T$, is from the set

$$\{\mathsf{inc}(i), \mathsf{jump}(j), \mathsf{jzdec}(i, j) \mid i = 0, 1, j \leq T\}$$

and $I_T = \mathsf{halt}$. A *configuration* of M is a triple $(k, l, m) \in \mathbb{N}^3$ with k being the index of the current instruction, and l, m being the current contents of counters 0 and 1, respectively. The *computation function* $f_M : \mathbb{N} \to \mathbb{N}^3$ of M maps time steps into configurations, and is given by $f_M(0) = (0, 0, 0)$ and if $f_M(n) = (k, l, m)$ then

$$f_M(n+1) = \begin{cases} (k+1, l+1, m) & \text{if } I_k = \mathsf{inc}(0) \\ (k+1, l, m+1) & \text{if } I_k = \mathsf{inc}(1) \\ (j, l, m) & \text{if } I_k = \mathsf{jump}(j) \\ (j, l, m) & \text{if } I_k = \mathsf{jzdec}(0, j) \text{ and } l = 0 \\ (j, l, m) & \text{if } I_k = \mathsf{jzdec}(1, j) \text{ and } m = 0 \\ (k+1, l-1, m) & \text{if } I_k = \mathsf{jzdec}(0, j) \text{ and } l > 0 \\ (k+1, l, m-1) & \text{if } I_k = \mathsf{jzdec}(1, j) \text{ and } m > 0 \\ (k, l, m) & \text{if } I_k = \mathsf{halt}\,. \end{cases}$$

We say that M *halts* if $f_M(n) = (T, l, m)$ for some $n, l, m \in \mathbb{N}$.

Theorem 1. *[Minsky, 1967] The halting problem for two-counter machines is undecidable.*

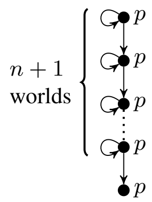


Fig. 2: CHAIN(p, n)

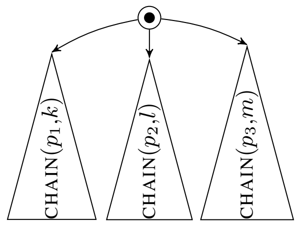


Fig. 3: $s_{(k,l,m)}$

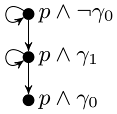


Fig. 4: INC(p)

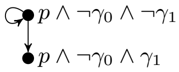


Fig. 5: DEC(p)

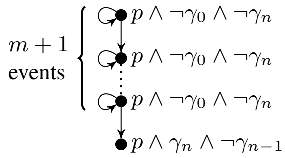


Fig. 6: REPL(p, n, m)

5 Single-agent Epistemic Planning

In this section, we assume that the set $\mathcal{A}$ is a singleton.

5.1 The General Case

We encode the halting problem of a two-counter machine M as an epistemic planning task in three steps:

1. We define epistemic models CHAIN(p, n) for encoding natural numbers, and epistemic states $s_{(k,l,m)}$ for encoding configurations;
2. We define a finite set of epistemic actions $\mathcal{F}_M$ for encoding the computation function f_M;
3. We encode the halting problem as an epistemic planning task using these models.

5.1.1 Encoding of Configurations

For each propositional symbol $p \in P$ and each $n \in \mathbb{N}$, we define an epistemic model CHAIN(p, n) as in Figure 2. For each $(k, l, m) \in \mathbb{N}^3$, we define the epistemic state $s_{(k,l,m)}$ as in Figure 3. It encodes the configurations (k, l, m) of two-counter machines.

5.1.2 Encoding of the Computation Function

First, we need some formal preliminaries:

Definition 12 (Path formulas). For every $n \in \mathbb{N}$, we define the *n-path formula* as follows: $\gamma_n := \Diamond^n \Box \bot$.

Lemma 1. *Let $n \in \mathbb{N}$ and let $(\mathcal{M}, w)$ be an epistemic state. Then $(\mathcal{M}, w) \models \gamma_n$ iff there is a path of length n starting in w and ending in a world with no successor (a sink).*

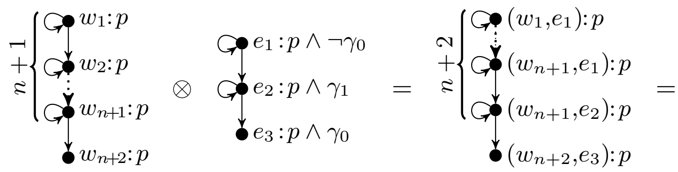
For each propositional symbol $p \in P$ and each $m, n \in \mathbb{N}$ we define three event models $\text{INC}(p)$, $\text{DEC}(p)$ and $\text{REPL}(p, n, m)$ as in Figures 4–6. We have omitted edge labels, as we are in the single-agent case.

Lemma 2. *For all $m, n \in \mathbb{N}$, for all $p \in P$,*

1. $\text{CHAIN}(p, n) \otimes \text{INC}(p) = \text{CHAIN}(p, n + 1)$
2. *if* $n > 0$, $\text{CHAIN}(p, n) \otimes \text{DEC}(p) = \text{CHAIN}(p, n - 1)$
3. $\text{CHAIN}(p, n) \otimes \text{REPL}(p, n, m) = \text{CHAIN}(p, m)$.

Proof. We only prove items 1 and 3. We first prove item 1. Introducing names for the nodes and events, we can calculate as follows:

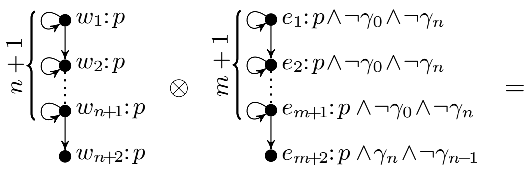
$\text{CHAIN}(p, n) \otimes \text{INC}(p) =$

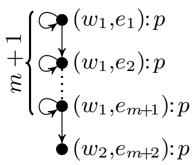


$\text{CHAIN}(p, n + 1)$.

Now, we prove item 3. Introducing names for the nodes and events, we can calculate as follows:

$\text{CHAIN}(p, n) \otimes \text{REPL}(p, n, m) =$





$= \text{CHAIN}(p, m)$.

□

Definition 13. For all $k \in \mathbb{N}$, we define the formulas ϕ_k as follows:

$$\phi_k := \Diamond(p_1 \wedge \neg\gamma_k \wedge \gamma_{k+1}). \tag{1}$$

Using Lemma 1 and the definition of $s_{(k,l,m)}$, we immediately get the following result:

Fact 1. *For all $k, l, m, k' \in \mathbb{N}$,*

$$s_{(k,l,m)} \models \phi_{k'} \text{ iff } k' = k. \tag{2}$$

Let $M = (I_0, \ldots, I_T)$ be a two-counter machine. For all $k < T$ and all $l, m \in \mathbb{N}$, we define an epistemic action $a_M(k, l, m)$ as in Figures 7–10 depending on the values of I_k, l and m. If $k, l, m, k', l', m' \in \mathbb{N}$, we write $(k, l, m) \approx (k', l', m')$ when the following holds:

$$k = k' \text{ and } \begin{cases} l = 0 \text{ iff } l' = 0 & \text{if } I_k = \mathsf{jzdec}(0, j) \\ m = 0 \text{ iff } m' = 0 & \text{if } I_k = \mathsf{jzdec}(1, j) \end{cases}$$

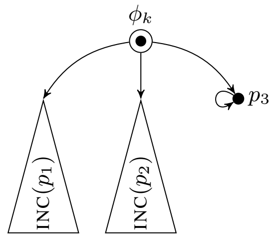


Fig. 7: The action $a_M(k,l,m)$ when $I_k = \mathsf{inc}(0)$. The case $I_k = \mathsf{inc}(1)$ is by replacing p_2 with p_3 everywhere.

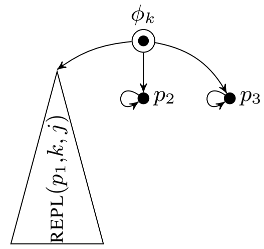


Fig. 8: The action $a_M(k,l,m)$ when $I_k = \mathsf{jump}(j)$.

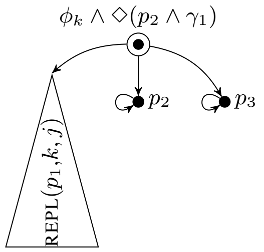


Fig. 9: The action $a_M(k,l,m)$ when $I_k = \mathsf{jzdec}(0,j), l = 0$. Case $I_k = \mathsf{jzdec}(1,j), m = 0$ is by replacing p_2 with p_3.

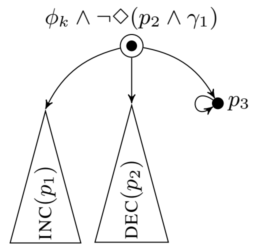


Fig. 10: The action $a_M(k,l,m)$ when $I_k = \mathsf{jzdec}(0,j), l > 0$. Case $I_k = \mathsf{jzdec}(1,j), m > 0$ is by replacing p_2 with p_3.

Note that when $(k,l,m) \approx (k',l',m')$ then $a_M(k,l,m) = a_M(k',l',m')$, hence the following set is *finite*:

$$\mathcal{F}_M := \{a_M(k,l,m) \mid k \in \{0,\ldots,T-1\}, l,m \in \mathbb{N}\}.$$

Lemma 3 below shows that $\mathcal{F}_M$ really encodes the computation steps of the computation function.

Lemma 3. *Let $M = (I_0,\ldots,I_T)$ be a two-counter machine, $l,m,n \in \mathbb{N}$ and $k < T$. Then, the following holds:*

1. *$a_M(k,l,m)$ is applicable in $s_{f_M(n)}$ iff $(k,l,m) \approx f_M(n)$;*
2. *$s_{f_M(n)} \otimes a_M(f_M(n)) = s_{f_M(n+1)}$.*

Proof sketch. Assume that $f_M(n) = (k',l',m')$. Item 1 is by case of I_k. We only consider the cases $I_k = \mathsf{inc}(0)$ and $I_k = \mathsf{jump}(j)$. In these cases, $a_M(k,l,m)$ is an epistemic action of the form $(\mathcal{E}, e)$ with $pre(e) = \phi_k$. Hence using Fact 1 we get:

$$\begin{array}{l} a_M(k,l,m) \text{ is applicable in } s_{f_M(n)} \Leftrightarrow \\ s_{(k',l',m')} \models \phi_k \Leftrightarrow \\ k = k' \Leftrightarrow \\ (k,l,m) \approx (k',l',m'), \end{array}$$

because $I_k = \mathsf{inc}(0)$ or $I_k = \mathsf{jump}(j)$. Item 2 is by case of $I_{k'}$. We only consider the case $I_{k'} = \mathsf{inc}(0)$:

$$s_{f_M(n)} \otimes a_M(f_M(n)) = s_{(k',l',m')} \otimes a_M(k',l',m') = s_{(k'+1,l'+1,m)} = s_{f_M(n+1)},$$

using Lemma 2 and that $a_M(k',l',m')$ is the epistemic action of Fig. 7. $\square$

5.1.3 Encoding of the Halting Problem

From Lemma 3, we derive the following lemma:

Lemma 4. *Let $M = (I_0, \dots, I_T)$ be a two-counter machine. Define $\mathcal{T}_M$ as the following single-agent epistemic planning task on* K*: $\mathcal{T}_M = (s_{(0,0,0)}, \mathcal{F}_M, \phi_T)$. Then $\mathcal{T}_M$ has a solution iff M halts.*

Proof. Using Lemma 3 and induction on m, we get that for all $m \in \mathbb{N}$,

(a) $a_M(f_M(m))$ is applicable in the state $s_{(0,0,0)} \otimes a_M(f_M(0)) \otimes \cdots \otimes a_M(f_M(m-1))$, and is the only applicable action from $\mathcal{T}_M$ in this state.

(b) $s_{(0,0,0)} \otimes a_M(f_M(0)) \otimes \cdots \otimes a_M(f_M(m)) = s_{f_M(m+1)}$.

For an action sequence $a_0, \dots, a_n$ to be a solution to $\mathcal{T}_M$ it must by definition (Section 3) satisfy:

(i) For all $m \leq n$, a_m is applicable in $s_{(0,0,0)} \otimes a_0 \otimes a_1 \otimes \cdots \otimes a_{m-1}$.

(ii) $s_{(0,0,0)} \otimes a_0 \otimes a_1 \otimes \cdots \otimes a_n \models \phi_T$.

First we prove that if $\mathcal{T}_M$ has a solution, M halts. Assume $a_0, \dots, a_n$ is solution to $\mathcal{T}_M$. Then (i) and (ii) holds. From (i) we get, using (a), that $a_m = a_M(f_M(m))$ for all $m \leq n$. This implies $s_{(0,0,0)} \otimes a_0 \otimes \cdots \otimes a_n = s_{f_M(n+1)}$, using (b), and hence (ii) gives us $s_{f_M(n+1)} \models \phi_T$. Fact 1 now implies that $f_M(n+1) = (T, l, m)$. Since T is the index of the halting instruction of M, this shows that M halts. We have now proven that if $\mathcal{T}_M$ has a solution then M halts.

Assume conversely that M halts. Then $f_M(n+1) = (T, l, m)$ for some n. Define $a_m = a_M(f_M(m))$ for all $m \leq n$. If we can prove (i) and (ii) we are done. (i) follows immediately from (a). From (b) we can conclude $s_{(0,0,0)} \otimes a_0 \otimes \cdots \otimes a_n = s_{f_M(n+1)}$. Since $f_M(n+1) = (T, l, m)$, we must have $s_{f_M(n+1)} \models \phi_T$, by Fact 1. Hence, we can conclude that (ii) holds. □

So, from Lemma 4 and Theorem 1, we obtain:

Theorem 2. PLANEX(K,1) *is undecidable.*

5.2 Epistemic Planning for K4, KT, K45, S4 and S5

We are going to prove an even stronger result than Theorem 2, namely that the plan existence problem for $\mathsf{S4}$ planning tasks is undecidable. The proof of this undecidability result generalizes the previous proof, but we need an extra atomic proposition r. For better readability, we use four atomic propositions p_1, p_2, p_3 and r, although we could use only two. The idea underlying the proof is to replace worlds with meta-worlds, which are in fact epistemic models.

5.2.1 Encoding of Configurations

The chains as defined in Figure 2 are replaced by the meta-chains of Figure 11. A *rectangle meta-world* is the epistemic model delimited in Figure 11 by a rectangle, and an *ellipse meta-world* is the epistemic model delimited in Figure 11 by the ellipse. Note that the worlds in an ellipse meta-world are related to each other in both directions, unlike the worlds in a rectangle meta-world where the arrow is in only one direction. Then, we define the epistemic model META-S$_{(k,l,m)}$ encoding the configuration (k, l, m) as in Figure 3. Note that the roots of the meta-chains and the root w_0 of META-S$_{(k,l,m)}$ are related to each other in *both* directions, unlike the previous proof for the logic K.

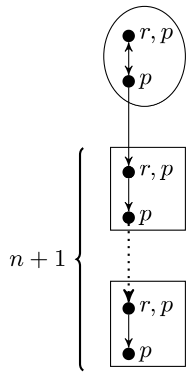


Fig. 11: META-CHAIN(p, n)

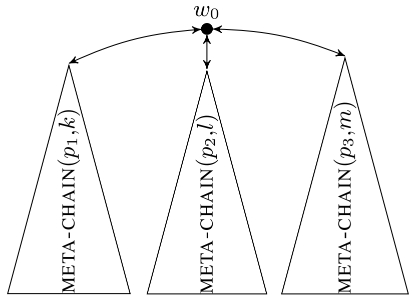


Fig. 12: META-S$_{(k,l,m)}$

5.2.2 Encoding of the Computation Function

The sub-model generated by a rectangle meta-world in a meta-chain is called a *meta-path*, and the *length* of a meta-path is the number of rectangle meta-worlds in such a sequence minus one.

Definition 14 (Meta-path formulas). For all $n \in \mathbb{N}$, we define the formulas χ_n inductively as follows:

$$\begin{array}{rcl} \chi_0 & := & \Box(\neg r \to \Box\neg r) \\ \chi_{n+1} & := & \Box(\neg r \to \Box(r \to \chi_n)) \end{array}$$

We then obtain a counterpart of Lemma 1:

Lemma 5. *Let $k, l, m \in \mathbb{N}$ and let $w \in$ META-S$_{(k,l,m)}$.*

1. *If w is in a rectangle meta-world and $n \in \mathbb{N}$, then,* META-S$_{(k,l,m)}, w \models \chi_n$ *iff there is a meta-path of length at least n starting at the rectangle meta-world containing w and ending in a rectangle meta-world with no successor (rectangle) meta-world.*

2. *w is in a rectangle meta-world of* META-S$_{(k,l,m)}$ *iff* META-S$_{(k,l,m)}, w \models \Box p$ *for some $p \in \{p_1, p_2, p_3\}$.*

Proof sketch. The proof of the first item is by induction on n and by observing that, in a meta-path, χ_0 holds only at the bottom last rectangle meta-world of the meta-path. The proof of the second item relies on the fact that the worlds of an ellipse meta-chain are all connected to the root of META-S$_{(k,l,m)}$, which satisfies neither p_1, p_2 nor p_3. $\Box$

Then, we define the event models META-INC(p), META-DEC(p) and META-REPL(p, n, m) as in Figures 13–15. Moreover, we define the following formulas which allow us to count the length of meta-chains:

Definition 15. For all $k \in \mathbb{N}$ and all $p \in P$, we define the formulas $\psi_k(p)$ as follows:

$$\psi_k(p) := \begin{cases} \Diamond\Box p \wedge \Box(\Box p \to \chi_0) & \text{if } k = 0; \\ \Diamond(\Box p \wedge \neg\chi_{k-1}) \wedge \Box(\Box p \to \chi_k) & \text{if } k > 0. \end{cases} \tag{3}$$

By Lemma 5, we easily get the following result:

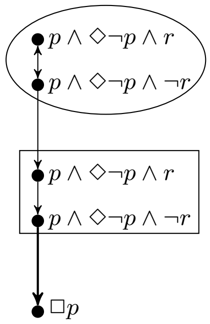


Fig. 13: META-INC(p)

Fig. 14: META-DEC(p)

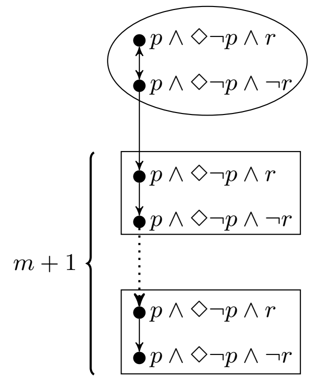


Fig. 15: META-REPL(p, n, m)

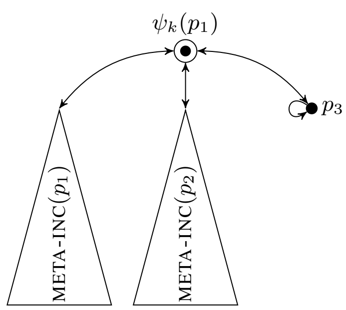


Fig. 16: The action $e_M(k, l, m)$ when $I_k = \mathsf{inc}(0)$. The case $I_k = \mathsf{inc}(1)$ is by replacing p_2 and p_3 everywhere.

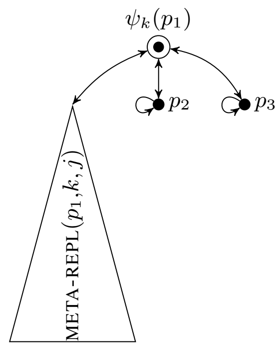


Fig. 17: The action $e_M(k, l, m)$ when $I_k = \mathsf{jump}(j)$.

Fact 2. *For all $k, l, m, k' \in \mathbb{N}$,*

$$\begin{array}{llll} \text{META-S}_{(k,l,m)}, w_0 \models & \psi_{k'}(p_1) & \textit{iff} & k' = k \\ \text{META-S}_{(k,l,m)}, w_0 \models & \psi_{l'}(p_2) & \textit{iff} & l' = l \\ \text{META-S}_{(k,l,m)}, w_0 \models & \psi_{m'}(p_3) & \textit{iff} & m' = m. \end{array}$$

Let $M = (I_0, \ldots, I_T)$ be a two-counter machine. For all $k < T$ and all $l, m \in \mathbb{N}$, we define an epistemic action $e_M(k, l, m)$ as in Figures 16–19 by replacing in Figures 7–10 INC, DEC and REPL with META-INC, META-DEC and META-REPL respectively and by replacing ϕ_k with ψ_k. Note again that the roots of META-INC, META-DEC and META-REPL and the root w_0 of META-S$_{(k,l,m)}$ are related to each other in *both* directions.

5.2.3 Encoding of the Halting Problem

If CHAIN, INC, DEC and the function a_M are replaced in Lemmata 2 and 3 with META-CHAIN, META-INC, META-DEC and the function e_M respectively, then these Lemmata still hold. Therefore, Lemma 4 and Theorem 2 also generalize to this S4 setting, and we finally obtain that:

Theorem 3. PLANEX$(S4, 1)$ *is undecidable.*

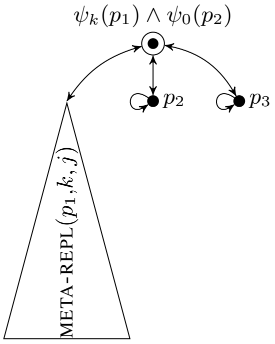


Fig. 18: The action $e_M(k,l,m)$ when $I_k = \mathsf{jzdec}(0,j), l = 0$. Case $I_k = \mathsf{jzdec}(1,j), m = 0$ is by replacing p_2 with p_3.

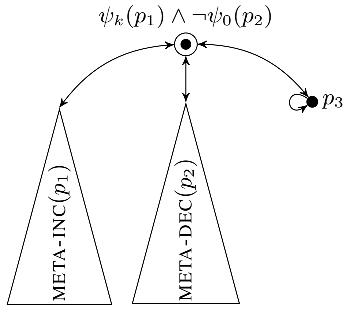


Fig. 19: The action $e_M(k,l,m)$ when $I_k = \mathsf{jzdec}(0,j), l > 0$. Case $I_k = \mathsf{jzdec}(1,j), m > 0$ is by replacing p_2 with p_3.

As a direct corollary of Theorem 3, we have that single-agent epistemic planning for $\mathsf{K4}$ is undecidable as well. Despite all these negative results, there is still room for decidability in the single-agent case if we assume that knowledge or belief are negatively introspective:

Theorem 4. PLANEX($\mathsf{K45}$, 1) *and* PLANEX($\mathsf{S5}$, 1) *are decidable.*

Proof. Any formula of $\mathsf{K45}$ (and hence also of $\mathsf{S5}$) is provably equivalent to a normal form formula of degree 1 [Meyer and van der Hoek, 1995], *i.e.*, a conjunction of disjunctions of the form $\phi \vee \Diamond_i \phi_0 \vee \Box_i \phi_1 \vee \ldots \vee \Box_i \phi_n$. Therefore, any epistemic planning task can be reduced equivalently to a classical planning task whose states are epistemic models of height at most 1. So, because there is a finite number of epistemic models of height at most 1 (for a finite set of propositional atoms), the state space of the classical planning problem is *finite*. Hence, we immediately get decidability. □

6 Multi-agent Epistemic Planning

In the multi-agent setting, we prove a strong result, namely that multi-agent epistemic planning is undecidable for any logic between K and $\mathsf{S5}$. The proof of this undecidability result generalizes the proof for single-agent K given in Section 5.1. Like for the proof of undecidability of the previous section for $\mathsf{S4}$, the idea underlying the proof is to replace worlds with meta-worlds.

6.1 Encoding of Configurations

We encode configurations as epistemic states of two-agent $\mathsf{S5}$. The worlds in CHAIN(p,n) of Figure 2 are replaced with the epistemic models META-WORLD$'(p)$ of Figure 20. The way meta-worlds are connected to each other to form a META-CHAIN$'(p,n)$ is shown in Figure 21. Then, for each configuration $(k,l,m) \in \mathbb{N}^3$, we define an epistemic state META-S$'_{(k,l,m)}$ by replacing in Figure 3 CHAIN with META-CHAIN$'$ and by labeling the accessibility relations originating from the designated world with agent 1. Note that as we are in $\mathsf{S5}$, all relations are equivalence relations, but the reflexive, symmetric and transitive closure is left implicit in figures.

6.2 Encoding of the Computation Function

Similarly to the case of single-agent K and $\mathsf{S4}$, we define path formulas.

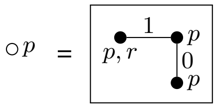


Fig. 20: META-WORLD$'(p)$

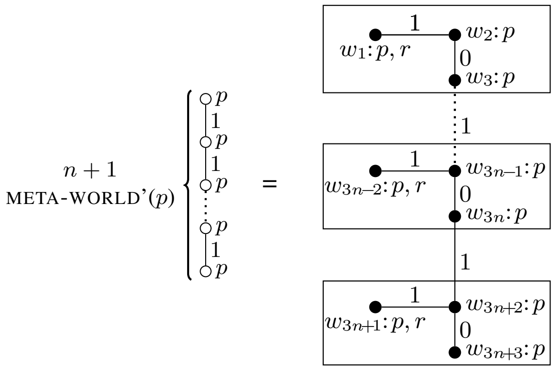


Fig. 21: META-CHAIN$'(p, n)$

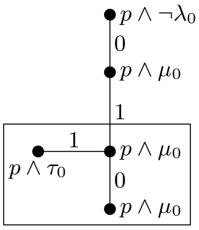


Fig. 22: META-INC$'(p)$

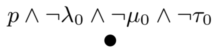


Fig. 23: META-DEC$'(p)$

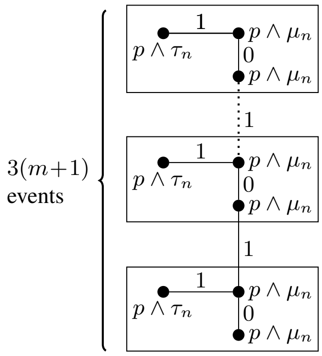


Fig. 24: META-REPL$'(p, n, m)$

Definition 16 (Meta-path' formulas). For all $p \in P$ and $n \in \mathbb{N}$, we define the formulas $\lambda_n(p)$, $\mu_n(p)$ and $\tau_n(p)$ inductively by:

$$\begin{array}{rcl}
\lambda_0(p) & := & p \wedge \Box_1 \neg r \\
\mu_0(p) & := & p \wedge \Diamond_0 \lambda_0(p) \wedge \neg\lambda_0(p) \\
\tau_0(p) & := & p \wedge r \wedge \Diamond_1 \mu_0(p) \\
\lambda_{n+1}(p) & := & p \wedge \Diamond_1 \mu_n(p) \wedge \neg\mu_n(p) \wedge \neg r \\
\mu_{n+1}(p) & := & p \wedge \Diamond_0 \lambda_{n+1}(p) \wedge \neg\lambda_{n+1}(p) \\
\tau_{n+1}(p) & := & p \wedge r \wedge \Diamond_1 \mu_{n+1}(p).
\end{array}$$

We then obtain a counterpart of Lemma 1:

Lemma 6. *For all $p \in P$, $n \in \mathbb{N}$, $0 \leq i \leq n$, $1 \leq j \leq 3n+3$:*

$$\begin{array}{lll}
\text{META-CHAIN}'(p, n), w_j \models \lambda_i(p) & \textit{iff} & j = 3n+3-3i \\
\text{META-CHAIN}'(p, n), w_j \models \mu_i(p) & \textit{iff} & j = 3n+2-3i \\
\text{META-CHAIN}'(p, n), w_j \models \tau_i(p) & \textit{iff} & j = 3n+1-3i.
\end{array}$$

In other words, λ_i holds in the bottom world of the $(i{+}1)$th to last meta-world of META-CHAIN$'(p, n)$, μ_i in the top right world of the same meta-world and τ_i in the top left world of the same meta-world. Now we define META-INC$'(p)$, META-DEC$'(p)$ and META-REPL$'(p, n, m)$ as in Figures 22–24.

Let $M = (I_0, \ldots, I_T)$ be a two-counter machine. For all $k < T$ and all $l, m \in \mathbb{N}$, we define similarly to the epistemic actions of Figures 16–19 an epistemic action $e'_M(k, l, m)$ from the epistemic actions of Figures 7–10 by:

1. replacing INC, DEC and REPL with META-INC$'$, META-DEC$'$ and META-REPL$'$ respectively;
2. labeling the accessibility relations originating from the designated worlds with agent 1;
3. replacing ϕ_k with $\Diamond_1 \mu_k(p_1)$;
4. replacing $\Diamond(p_2 \wedge \gamma_1)$ with $\Diamond_1 \mu_0(p_2)$.

6.3 Encoding of the Halting Problem

If CHAIN, INC, DEC, a_M and $s_{(k,l,m)}$ are replaced in Lemmata 2 and 3 with META-CHAIN$'$, META-INC$'$, META-DEC$'$, e'_M and META-S$'_{(k,l,m)}$ respectively, then these Lemmata still hold. Therefore, Lemma 4 and Theorem 2 also generalize to this two-agent S5 setting, and we finally obtain that:

Theorem 5. PLANEX$(S5, n)$ *is undecidable for any* $n \geq 2$*.*

7 DEL Model Checking

The *DEL language* $\mathcal{L}^*_{DEL}$ is defined by the following BNF [van Ditmarsch et al., 2007]:

$$\begin{aligned} \phi &::= p \mid \neg\phi \mid (\phi \wedge \phi) \mid \Box_i \phi \mid [\pi]\phi \\ \pi &::= (\mathcal{E}, e) \mid (\pi \cup \pi) \mid (\pi ; \pi) \mid \pi^* \end{aligned}$$

where $p \in P$, $i \in \mathcal{A}$ and $(\mathcal{E}, e)$ is any epistemic action. In DEL, one assumes that $|\mathcal{A}| > 1$. The truth conditions for the programs π are defined as follows:

$$\begin{array}{lll} \mathcal{M}, w \models [\mathcal{E}, e]\phi & \text{iff} & \mathcal{M}, w \models pre(e) \text{ implies} \\ & & (\mathcal{M}, w) \otimes (\mathcal{E}, e) \models \phi \\ \mathcal{M}, w \models [\pi \cup \gamma]\phi & \text{iff} & \mathcal{M}, w \models [\pi]\phi \text{ and } \mathcal{M}, w \models [\gamma]\phi \\ \mathcal{M}, w \models [\pi ; \gamma]\phi & \text{iff} & \mathcal{M}, w \models [\pi][\gamma]\phi \\ \mathcal{M}, w \models [\pi^*]\phi & \text{iff} & \text{for all } \textit{finite} \text{ sequences } \pi; \ldots ; \pi, \\ & & \mathcal{M}, w \models [\pi; \ldots ; \pi]\phi \end{array}$$

The formula $[\mathcal{E}, e]\phi$ reads as "after the execution of the epistemic action $(\mathcal{E}, e)$, it holds that ϕ". The *model checking problem* is the following: "Given an epistemic state $(\mathcal{M}, w)$, a formula $\phi \in \mathcal{L}^*_{DEL}$, is it the case that $\mathcal{M}, w \models \phi$?". As an immediate corollary of our results, we have the following theorem. It complements the result of [Miller and Moss, 2005] stating that the *satisfiability* problem of DEL is undecidable.

Theorem 6. *The model checking problem of the language* $\mathcal{L}^*_{DEL}$ *is undecidable.*

Proof. PLANEX$(\mathsf{S5}, n)$ is reducible to the model checking problem of the language $\mathcal{L}^*_{DEL}$: an epistemic planning task $\mathcal{T} = (s_0, A, \phi_g)$ has a solution iff $s_0 \models \neg[A^*]\neg\phi_g$ holds. $\square$

	Single-agent planning	Multi-agent planning
K	UD	UD
KT	UD	UD
K4	UD	UD
K45	D	UD
S4	UD	UD
S5	D	UD

Fig. 25: Summary of results (D=Decidable, UD=UnDecidable)

8 Conclusion

8.1 Related Work

Alternatives to the DEL-based approach to multi-agent planning with ToM abilities can be found both in the literature on temporal epistemic logics [van der Hoek and Wooldridge, 2002] and in the literature on POMDP-based planning [Gmytrasiewicz and Doshi, 2005]. However, these alternative formalisms express planning tasks in terms of an explicitly given state space, and hence do not address how to express actions in a compact and convenient formalism (and how to possibly avoid building the entire state space when solving planning tasks). In the DEL-based formalism the state space is induced by the action descriptions as in classical planning. Note that our assumptions in DEL-based planning correspond to the infinite horizon case of planning based on POMDPs, in which already ordinary, single-agent planning is undecidable [Madani et al., 1999].

8.2 Concluding Remarks

Our results are summarized in the table of Figure 25 (we recall that they hold only for $|P| \geq 2$). From this table, we notice that in the single-agent setting, the property of Euclidicity (defined by Axiom 5: $\neg\Box_i\phi \rightarrow \Box_i\neg\Box_i\phi$) draws the borderline between decidability and undecidability: if 5 is added to K4 or S4, we immediately obtain decidability.

Given these results, an important quest of course becomes to find fragments of the formalism in which interesting problems can still be formulated, but where the complexity is comparable to the complexity of other standard planning formalisms (varying from PSPACE-completeness for classical planning [Bylander, 1994] up to 2-EXP-completeness for planning under nondeterminism and partial observability [Rintanen, 2004]). We leave the quest for decidable fragments to future work. Initial results in this direction can be found in [Löwe et al., 2011].

References

[Andersen et al., 2012] Andersen, M. B., Bolander, T., and Jensen, M. H. (2012). Conditional epistemic planning. In del Cerro, L. F., Herzig, A., and Mengin, J., editors, *JELIA*, volume 7519 of *Lecture Notes in Computer Science*, pages 94–106. Springer.

[Aucher, 2010] Aucher, G. (2010). An internal version of epistemic logic. *Studia Logica*, 1:1–22.

[Aucher, 2012] Aucher, G. (2012). DEL-sequents for regression and epistemic planning. *Journal of Applied Non-Classical Logics*, 22(4):337–367.

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

[Baltag et al., 1998] Baltag, A., Moss, L. S., and Solecki, S. (1998). The logic of public announcements and common knowledge and private suspicions. In *TARK*, pages 43–56.

[Baron-Cohen, 1997] Baron-Cohen, S. (1997). *Mindblindness: An essay on autism and theory of mind*. MIT press.

[Bolander and Andersen, 2011] Bolander, T. and Andersen, M. B. (2011). Epistemic planning for single- and multi-agent systems. *Journal of Applied Non-Classical Logics*, 21(1):9–34.

[Bylander, 1994] Bylander, T. (1994). The computational complexity of propositional STRIPS planning. *Artificial Intelligence*, 69(1–2):165–204.

[Erol et al., 1995] Erol, K., Nau, D., and Subrahmanian, V. (1995). Complexity, decidability and undecidability results for domain-independent planning. *Artificial Intelligence*, 76(1):75–88.

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

[Ghallab et al., 2004] Ghallab, M., Nau, D. S., and Traverso, P. (2004). *Automated Planning: Theory and Practice*. Morgan Kaufmann.

[Gmytrasiewicz and Doshi, 2005] Gmytrasiewicz, P. and Doshi, P. (2005). A framework for sequential planning in multiagent settings. *Journal of Artificial Intelligence Research*, 24(1):49–79.

[Hampson and Kurucz, 2012] Hampson, C. and Kurucz, A. (2012). On modal products with the logic of 'elsewhere'. In *Advances in Modal Logic*, volume 9.

[Hintikka, 1962] Hintikka, J. (1962). *Knowledge and Belief, An Introduction to the Logic of the Two Notions*. Cornell University Press, Ithaca and London.

[Löwe et al., 2011] Löwe, B., Pacuit, E., and Witzel, A. (2011). DEL planning and some tractable cases. In *LORI*, pages 179–192.

[Madani et al., 1999] Madani, O., Hanks, S., and Condon, A. (1999). On the undecidability of probabilistic planning and infinite-horizon partially observable markov decision problems. In *Proceedings of the National Conference on Artificial Intelligence*, pages 541–548. John Wiley & Sons Ltd.

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

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

[Minsky, 1967] Minsky, M. (1967). *Computation*. Prentice-Hall.

[Pardo and Sadrzadeh, 2012] Pardo, P. and Sadrzadeh, M. (2012). Planning in the logics of communication and change. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems - Volume 3*, AAMAS 2012, pages 1231–1232. International Foundation for Autonomous Agents and Multiagent Systems.

[Premack and Woodruff, 1978] Premack, D. and Woodruff, G. (1978). Does the chimpanzee have a theory of mind? *Behavioral and Brain Sciences*, 1(4):515–526.

[Rintanen, 2004] Rintanen, J. (2004). Complexity of planning with partial observability. In Zilberstein, S., Koehler, J., and Koenig, S., editors, *ICAPS*, pages 345–354. AAAI.

[van Benthem, 2011] van Benthem, J. (2011). *Logical Dynamics of Information and Interaction*. Cambridge University Press.

[van Benthem et al., 2006] van Benthem, J., van Eijck, J., and Kooi, B. (2006). Logics of communication and change. *Information and Computation*, 204(11):1620–1662.

[van der Hoek and Wooldridge, 2002] van der Hoek, W. and Wooldridge, M. (2002). Tractable multiagent planning for epistemic goals. In *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-2002)*, pages 1167–1174. ACM Press.

[van Ditmarsch et al., 2005] van Ditmarsch, H., van der Hoek, W., and Kooi, B. (2005). Dynamic epistemic logic with assignment. In Dignum, F., Dignum, V., Koenig, S., Kraus, S., Singh, M. P., and Wooldridge, M., editors, *Autonomous Agents and Multi-agent Systems (AAMAS 2005)*, pages 141–148. ACM.

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

RESEARCH CENTRE
RENNES – BRETAGNE ATLANTIQUE

Campus universitaire de Beaulieu
35042 Rennes Cedex

Publisher
Inria
Domaine de Voluceau - Rocquencourt
BP 105 - 78153 Le Chesnay Cedex
inria.fr

ISSN 0249-6399