Complexity Results in Epistemic Planning

Thomas Bolander, DTU Compute, Tech Univ of Denmark
Joint work with: Martin Holm Jensen and Francois Schwarzentruber

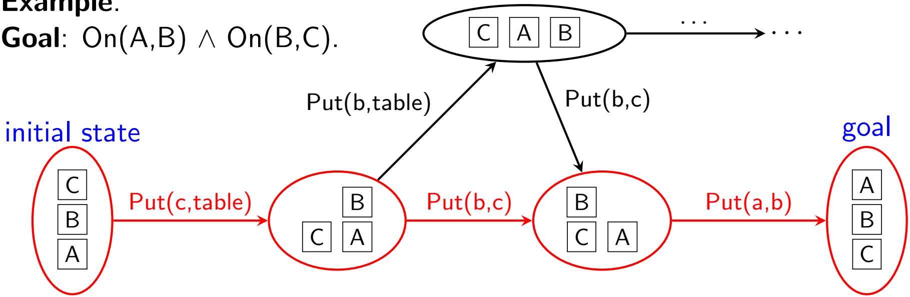
DTU Compute
Department of Applied Mathematics and Computer Science

Automated planning

Automated planning (or, simply, **planning**):

- Given is a **planning task**: **initial state** + **goal formula** + finite set of **actions**.
- Aim is to compute a **solution**: sequence of actions that leads from the initial state to a state satisfying the goal formula.

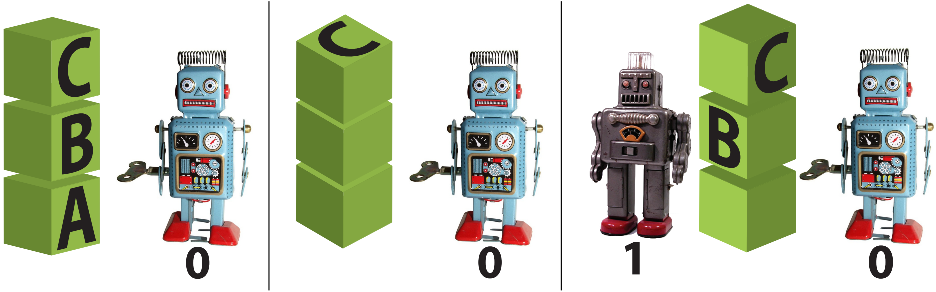
Example.
Goal: On(A,B) $\wedge$ On(B,C).



In automated planning, such a graph is called a **state space** (induced by a planning task).

From classical to epistemic planning

In many scenarios, classical (deterministic, fully observable, single-agent, static) planning is not enough. Restack $\begin{array}{|c|}\hline C\\\hline B\\\hline A\\\hline\end{array}$ as $\begin{array}{|c|}\hline A\\\hline B\\\hline C\\\hline\end{array}$:



Epistemic planning

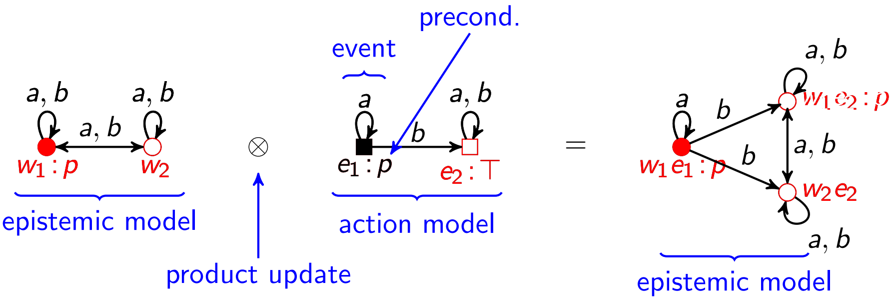
We will here use **epistemic planning** to refer to planning based on Dynamic Epistemic Logic (DEL).

Essentially our framework of epistemic planning is obtained by replacing the propositional logic underlying classical planning by **Dynamic Epistemic Logic** (**DEL**).

	Classical	**DEL-based**
States	models of prop. logic	models of MA epist. logic
Goal formula	formula of prop. logic	formula of MA epist. logic
Actions	induced by action schemas	action models of DEL

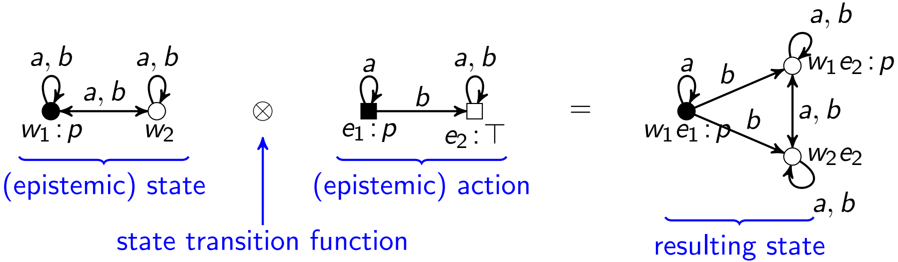
Epistemic planning is a framework for multi-agent planning that allows **(arbitrary levels of) higher-order reasoning**. Higher-order reasoning here means reasoning about the beliefs of yourself and other agents (and nesting of such).

DEL by example: A private announcement



- **Action models**: Only **propositional** preconditions and **no postconditions**. Means: Purely epistemic planning, no change of ontic facts.
- **Action model above**: Private announcement of p to agent a.
- **Actual world/event**: Colored black.
- **Product update**: As in [Baltag et al., 1998].
- **In resulting model**: Agent a knows p ($\Box_a p$ holds), but agent b didn't learn anything.

Planning interpretation of DEL



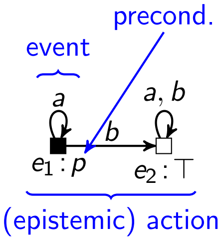
- **Epistemic states**: Pointed, finite epistemic models.
- **Epistemic actions**: Pointed, finite action models.
- **Result of applying an action in a state**: Product update of state with action.

Formal definition of action models and epistemic actions

Definition (Action models and epistemic actions)

An **action model** is $A = (E, Q, pre)$ where:

- E is a finite set of **events**.
- $Q : Ag \to 2^{E \times E}$ assigns an **epistemic (accessibility) relation** to each agent.
- $pre : E \to L_{\text{Prop}}$ assigns a **precondition** of the propositional language to each event.



We write Q_a for $Q(a)$. For $e \in E$, the pair (A, e) is called an *epistemic action* whose *actual event* is e.

Note: Only propositional preconditions and no postconditions.

Applicability: An epistemic action $\alpha = (A, e)$ is said to be **applicable** in an epistemic state s is $s \models pre(e)$.

Planning tasks and plan existence problem

Definition (Planning tasks)

A **planning task** is $T = (s_0, \mathcal{L}, \varphi_g)$, where

- s_0 is the **initial state**: a finite epistemic state.
- $\mathcal{L}$ is the **action library**: a finite set of epistemic actions.
- φ_g is the **goal formula**: a formula of epistemic logic.

Definition (Plans)

A **plan** (or **solution**) for a planning task $T = (s_0, \mathcal{L}, \varphi_g)$ is a sequence of epistemic actions $\alpha_1, \ldots, \alpha_n \in \mathcal{L}$ such that $s_0 \models \langle \alpha_1 \rangle \cdots \langle \alpha_n \rangle \varphi_g$ (where, by definition, $s \models \langle \alpha \rangle \varphi$ iff α is applicable in s and $s \otimes \alpha \models \varphi$).

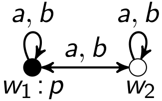
Definition (Plan existence problem)

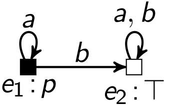
Let X denote a class of planning tasks. The **plan existence problem for** X is the following decision problem "Given an epistemic planning task $T \in X$, does it have a solution?"

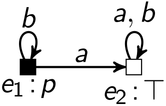
Example

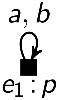
Consider the planning task $\{s_0, \{\alpha_1, \alpha_2, \alpha_3\}, \varphi_g\}$ with

$$\varphi_g = \Box_a p \wedge \Box_b p \wedge \neg \Box_a \Box_b p \wedge \neg \Box_b \Box_a p$$

$s_0 =$ 

$\alpha_1 =$ 

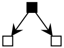
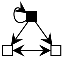
$\alpha_1 =$ 

$\alpha_3 =$ 

α_1: privately announcing p to a; α_2: privately announcing p to b; α_3: publicly announcing p to both agents.

A **solution (plan)** is α_1, α_2, since $s_0 \models \langle \alpha_1 \rangle \langle \alpha_2 \rangle \varphi_g$. Another solution is α_2, α_1. Also $\alpha_1, \alpha_2, \alpha_1$ and $\alpha_1, \alpha_1, \alpha_2$ are solutions, etc.

Summary of complexity results for plan existence

Underlying graphs of actions	Types of epistemic actions		
	Non-factual, propositional preconditions	Factual, propositional preconditions	Factual, epistemic preconditions
SINGLETONS 	NP-complete	PSPACE-hard [Jensen, 2014]	PSPACE-hard [Jensen, 2014]
CHAINS 	NP-complete	? (open question)	? (open question)
TREES 	PSPACE-complete	? (open question)	? (open question)
GRAPHS 	in EXPSPACE	in NON-ELEMENTARY [Yu et al., 2013]	Undecidable [Bolander and Andersen, 2011]

The green results will be covered in this talk. From Bolander, Jensen, Schwarzentruber: Complexity Results in Epistemic Planning (under submission).

Why study very expressively restricted fragments?

Motivation for studying complexity of very restrictive fragments of epistemic planning:

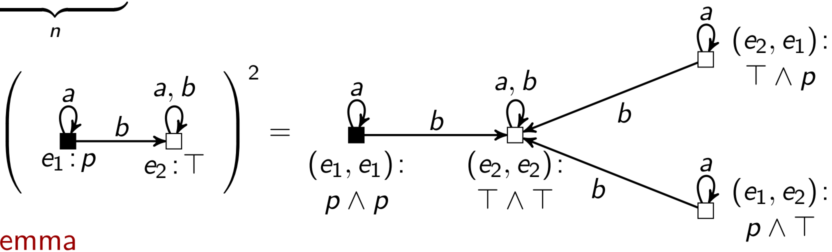
- Where does the complexity come from?
- Constructing search heuristics for planning engines (relaxed problems).
- Subclasses of more general fragments might be translatable into simpler fragments.

Definition (*n*-ary product)

Let $\alpha = (A, e)$ be an epistemic action where $A = (E, Q, pre)$. We denote by $A^n = (E^n, Q^n, pre^n)$ the ***n*-ary product of** A.

1. $E^n = \{(e_1, \dots, e_n) \mid e_i \in E \text{ for all } i = 1, \dots, n\}$,
2. $Q_a^n = \{((e_1, \dots, e_n), (f_1, \dots, f_n)) \mid e_i Q_a f_i \text{ for all } i = 1, \dots, n\}$
3. $pre^n((e_1, \dots, e_n)) = \bigwedge_{i=1,\dots,n} pre(e_i)$.

The ***n*-ary product of** α is defined as $\alpha^n = (A^n, e^n)$, where e^n denotes $\underbrace{(e, e, \dots, e)}_{n}$.



Lemma

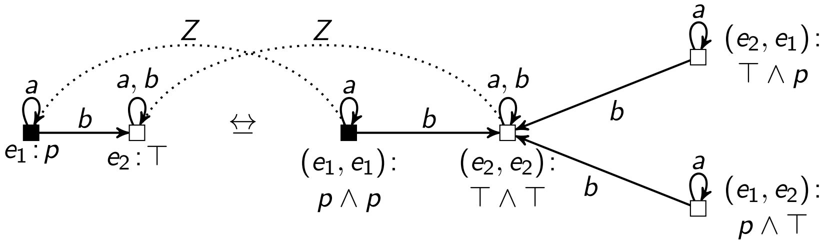
For any epistemic action α *and any* $\varphi \in L_E$ *we have that* $\langle\alpha\rangle^n \varphi \equiv \langle\alpha^n\rangle\varphi$ *(that is,* $\underbrace{\langle\alpha\rangle\langle\alpha\rangle \cdots \langle\alpha\rangle}_{n} \varphi$ *is modally equivalent to* $\langle\alpha^n\rangle\varphi$*).*

Bisimilarity on epistemic actions

Definition (**Bisimilarity**)

Two epistemic actions $\alpha = ((E, Q, pre), e)$ and $\alpha' = ((E', Q', pre'), e')$ are called *bisimilar*, written $\alpha \leftrightarroweq \alpha'$, if there exists a (bisimulation) relation $Z \subseteq E \times E'$ containing (e, e') and satisfying for every $a \in Ag$:

- **[atom]** If $(f, f') \in Z$ then $pre(f) \equiv pre'(f')$.
- **[forth]** If $(f, f') \in Z$ and $fQ_a g$ then $\exists g' \in E'$: $f' Q'_a g'$ and $(g, g') \in Z$.
- **[back]** Other direction.



n-bisimilarity on epistemic actions

Definition (n-**bisimilarity**)

Let $\alpha = ((E, Q, pre), e)$ and $\alpha' = ((E', Q', pre'), e')$ be epistemic actions. They are 0-*bisimilar*, written $\alpha \leftrightarroweq_0 \alpha'$, if $pre(e) \equiv pre'(e')$. For $n > 0$, they are *n-bisimilar*, written $\alpha \leftrightarroweq_n \alpha'$, if for every $a \in Ag$:

- **[atom]** $pre(e) \equiv pre'(e')$.
- **[forth]** If $eQ_a f$ then $\exists f' \in E' : e' Q'_a f'$ and $(A, f) \leftrightarroweq_{n-1} (A', f')$.
- **[back]** Other direction.

Equivalently: $\alpha \leftrightarroweq_n \alpha'$ if for any path of length $m \leq n$ in α:

$$\underset{e}{\blacksquare} \xrightarrow{a_1} \square \xrightarrow{a_2} \square \cdots\cdots \square \xrightarrow{a_m} \underset{f}{\square}$$

there exists a path in α':

$$\underset{e'}{\blacksquare} \xrightarrow{a_1} \square \xrightarrow{a_2} \square \cdots\cdots \square \xrightarrow{a_m} \underset{f'}{\square}$$

with $pre(f) \equiv pre'(f')$; and vice versa.

Stabilisation

$md(\varphi)$ denotes modal depth of φ.

Lemma ([Sadzik, 2006], slightly reformulated)

Let α, α' be two epistemic actions and φ any formula.

1. *If $\alpha \leftrightarroweq \alpha'$, then $\langle\alpha\rangle\varphi \equiv \langle\alpha'\rangle\varphi$.*
2. *If $md(\varphi) \leq n$ and $\alpha \leftrightarroweq_n \alpha'$, then $\langle\alpha\rangle\varphi \equiv \langle\alpha'\rangle\varphi$.*

Definition (**Stabilisation**)

Let α be an epistemic action.

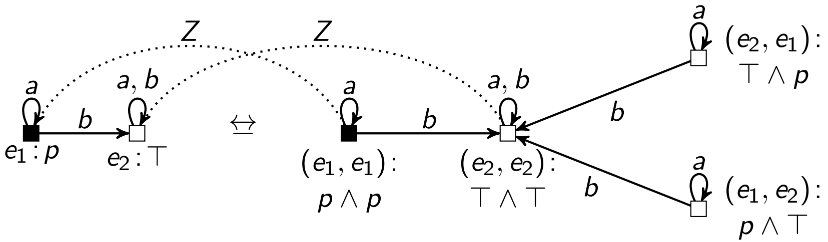
1. α is **stabilising at stage** i if $\alpha^i \leftrightarroweq \alpha^{i+k}$ for all $k \geq 0$.
2. α is n-**stabilising at stage** i if $\alpha^i \leftrightarroweq_n \alpha^{i+k}$ for all $k \geq 0$.

Lemma

If two epistemic actions are n-bisimilar for all n, then they are bisimilar.

Example

Recall $\alpha \leftrightarroweq \alpha^2$ where α is private announcement of p to agent a:



Hence private announcements are stabilising at stage 1. Clearly, so are public announcements.

Epistemic actions (our type) **commute**: $\langle\alpha_1\rangle\langle\alpha_2\rangle\varphi \equiv \langle\alpha_2\rangle\langle\alpha_1\rangle\varphi$ [Löwe et al., 2011].

Consequence: If the action library of a planning task only consists of public and private announcements, then we never have to repeat any action more than once.

Stabilisation and plan-existence problem

Lemma

Let $T = (s_0, \{\alpha_1, \ldots, \alpha_m\}, \varphi_g)$ be a planning task and $B \in \mathbb{N}$. Suppose one of the following holds:

1. *Every α_i is stabilising at stage B, or*
2. *$md(\varphi_g) = n$ and every α_i is n-stabilising at stage B.*

Then T is solvable iff there exists $k_1, \ldots, k_m \leq B$ s.t. $s_0 \models \langle \alpha_1 \rangle^{k_1} \cdots \langle \alpha_m \rangle^{k_m} \varphi_g$.

Non-deterministic algorithm for deciding the plan-existence problem when B satisfes 1 or 2 above:

procedure $\textsc{PlanExists}((s_0, \{\alpha_1, \ldots, \alpha_m\}, \varphi_g), B)$

a) Guess a vector $(k_1, \ldots, k_m) \in \{0, \ldots, B\}^m$.

b) Accept when $s_0 \models \langle \alpha_1 \rangle^{k_1} \cdots \langle \alpha_m \rangle^{k_m} \varphi_g$.

Sadzik's lemma

Lemma ([Sadzik, 2006])

Let $\alpha = ((E, Q, pre), e)$ be an epistemic action and n a natural number. Then α is n-stabilising at stage $|E|^n$.

We will now improve the upper bound on stabilisation...

Stabilisation Lemma

Given event e in action α and $n \in \mathbb{N}$, define:

$\text{mpaths}_n(e) =$ the number of distinct maximal paths of length $\leq n$ rooted at e (ignoring agent labels).

Lemma (Stabilisation Lemma)

Let $\alpha = (A, e_0)$ be an epistemic action and n any natural number. Then α is n-stabilising at stage $\text{mpaths}_n(e_0)$.

Proof sketch.

Let $k = \text{mpaths}_n(e_0)$. We then show:

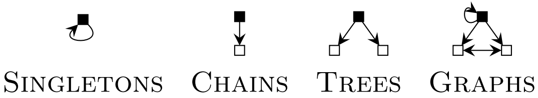
$$(A^{k+1}, \underbrace{(e_0, \ldots, e_0)}_{k+1}) \leftrightarrows_n (A^k, \underbrace{(e_0, \ldots, e_0)}_{k}).$$

$$\begin{array}{c} \overbrace{(e_0,\ e_0,\ e_0, \cdots,\ e_0)}^{k+1} \\ \downarrow\quad\downarrow\quad\downarrow\qquad\quad\downarrow \\ (e_1,\ e_2,\ e_3, \cdots,\ e_{k+1}) \end{array} \qquad \begin{array}{c} \overbrace{(e_0,\ e_0,\ e_0, \cdots,\ e_0)}^{k} \\ \downarrow\quad\downarrow\quad\downarrow\qquad\quad\downarrow \\ (e'_1,\ e'_2,\ e'_3, \cdots,\ e'_k) \end{array}$$

Classes of planning tasks

We define the following classes of planning tasks:

- SINGLETONS: Every action in the action library is a singleton (i.e. public announcement).
- CHAINS: The underlying graph of every action is a unary tree (a chain). Leafs can be reflexive.
- TREES: The underlying graph of every action is a tree. Leafs may be reflexive.
- GRAPHS: Arbitrary actions.



Note that

$$\textsc{Singletons} \subseteq \textsc{Chains} \subseteq \textsc{Trees} \subseteq \textsc{Graphs}$$

and that GRAPHS contain **all** planning tasks where actions have propositional preconditions (and no postconditions).

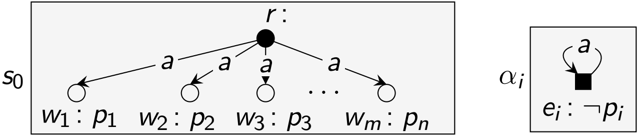
Plan existence problem for Singletons and Chains

Theorem

The plan existence problem for both SINGLETONS *and* CHAINS *is NP-complete.*

Proof sketch of NP-hardness for SINGLETONS .

Polynomial-time reduction from SAT. Given propositional formula $\varphi(p_1, \ldots, p_n)$, we construct planning task $T = (s_0, \{\alpha_1, \ldots, \alpha_m\}, \varphi(\Diamond_a p_1, \ldots, \Diamond_a p_m))$.



We can represent any propositional valuation ν by an epistemic state s satisfying: For all p_i, $\nu \models p_i$ iff $s \models \Diamond p_i$.

$\square$

Recall the plan-existence procedure that decides plan existence when all actions are stabilising at stage $\leq B$ (by previous lemma):

procedure $\textsc{PlanExists}((s_0, \{\alpha_1, \ldots, \alpha_m\}, \varphi_g), B)$

a) Guess a vector $(k_1, \ldots, k_m) \in \{0, \ldots, B\}^m$.

b) Accept when $s_0 \models \langle\alpha_1\rangle^{k_1} \cdots \langle\alpha_m\rangle^{k_m} \varphi_g$.

Proof sketch of NP-membership of $\textsc{Chains}$.

Let $T = (s_0, \{\alpha_1, \ldots, \alpha_m\}, \varphi_g)$ denote a planning task in $\textsc{Chains}$.

1. α_i has at most *one* maximal path of length $\leq n$ for all n.
2. By Stabilisation Lemma, α_i is n-stabilising at stage 1 for any n.
3. α_i is stabilising at stage 1 ($\leftrightarroweq_n$ for all n implies $\leftrightarroweq$).
4. The procedure $\textsc{PlanExists}(T, 1)$ is accepting iff T is solvable.
5. $\textsc{PlanExists}(T, 1)$ runs in non-deterministic polynomial time (non-increasing states).

$\square$

Plan existence problem for Trees

Theorem

The plan existence problem for $\textsc{Trees}$ *is PSPACE-complete.*

PSPACE-hardness is by a polynomial-time reduction from QSAT (satisfiability of quantified boolean formulas).

Proof sketch of PSPACE-membership of $\textsc{Trees}$.

Same overall proof strategy as for $\textsc{Chains}$. Let $T = (s_0, \{\alpha_1, \ldots, \alpha_m\}, \varphi_g)$ denote a planning task in $\textsc{Trees}$.

1. Let $l(\alpha_i)$ denote number of leaves in (underlying graph of) of α_i. Then $\text{mpaths}_n(e) \leq l(\alpha_i)$ for any n.
2. By Stabilisation Lemma, α_i is n-stabilising at stage $l(\alpha_i)$ for any n.
3. α_i is stabilising at stage $l(\alpha_i)$ ($\leftrightarroweq_n$ for all n implies $\leftrightarroweq$).
4. The procedure $\textsc{PlanExists}(T, \max_i l(\alpha_i))$ is accepting iff T is solvable.
5. $\textsc{PlanExists}(T, \max_i l(\alpha_i))$ uses polynomial space in the size of s_0 and $\langle\alpha_1\rangle^{k_1} \cdots \langle\alpha_m\rangle^{k_m}\varphi_g$ where $k_i \leq \max_i l(\alpha_i)$ [Aucher and Schwarzentruber, 2013].

Corollary on plan verification/model checking

Plan verification problem: Given a finite epistemic state s_0 and a formula of the form $\langle\alpha_1\rangle\cdots\langle\alpha_j\rangle\varphi_g$, does $s_0 \models \langle\alpha_1\rangle\cdots\langle\alpha_j\rangle\varphi_g$ hold.

Theorem

The plan verification problem (restricted to propositional action models that are trees) is PSPACE-complete.

This is a generalisation of a result in van de Pol, van Rooij and Szymanik: How difficult is it to Think that you Think that I think that...? (under submission).

We generalise by: only single-pointed models, no postconditions, only propositional preconditions, only tree structured action models.

Plan existence problem for Graphs

Theorem

The plan existence problem for GRAPHS *is in EXPSPACE.*

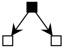
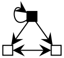
Proof sketch of EXPSPACE-membership of GRAPHS .

Same overall proof strategy as for CHAINS and TREES. Let $T = (s_0, \{\alpha_1, \ldots, \alpha_m\}, \varphi_g)$ denote a planning task in GRAPHS where $\alpha_i = (E_i, Q_i, pre_i), e_i)$ and let $k = md(\varphi_g)$.

1. By Sadzik's Lemma, α_i is k-stabilising at stage $|E_i|^k$.
2. The procedure PLANEXISTS$(T, \max_i |E_i|^k)$ is accepting iff T is solvable. This procedure runs in NEXPSPACE = EXPSPACE.

□

Summary

Underlying graphs of actions	Types of epistemic actions		
	Non-factual, propositional preconditions	Factual, propositional preconditions	Factual, epistemic preconditions
SINGLETONS 	NP-complete	PSPACE-hard [Jensen, 2014]	PSPACE-hard [Jensen, 2014]
CHAINS 	NP-complete	? (open question)	? (open question)
TREES 	PSPACE-complete	? (open question)	? (open question)
GRAPHS 	in EXPSPACE	in NON-ELEMENTARY [Yu et al., 2013]	Undecidable [Bolander and Andersen, 2011]

Appendix: References I

Aucher, G. and Schwarzentruber, F. (2013).
On the Complexity of Dynamic Epistemic Logic.
In TARK.

Jensen, M. (2014).
Epistemic and Doxastic Planning.
PhD thesis, Technical University of Denmark.
DTU Compute PHD-2014.

Löwe, B., Pacuit, E. and Witzel, A. (2011).
DEL planning and some tractable cases.
In LORI 2011, (van Ditmarsch, H., Lang, J. and Ju, S., eds), vol. 6953, of Lecture Notes in Artificial Intelligence pp. 179–192, Springer.

Sadzik, T. (2006).
Exploring the Iterated Update Universe.
ILLC Publications PP-2006-26.

Yu, Q., Wen, X. and Liu, Y. (2013).
Multi-Agent Epistemic Explanatory Diagnosis via Reasoning about Actions.
In IJCAI, (Rossi, F., ed.), IJCAI/AAAI.