Multi-Agent Systems

Epistemic Planning

Albert-Ludwigs-Universität Freiburg

Bernhard Nebel, Rolf Bergdoll, and Thorsten Engesser

Winter Term 2019/20

Epistemic Planning

Remark: Epistemic planning can also be based on formalisms other than DEL. We only focus on DEL here, though.

Before we begin: We first want to introduce to extensions to our DEL models:

- Multipointed models
- Action models with ontic effects

Multipointed Models

So far: State and action models only had a unique designated world/event.

- The actual world
- The event that actually takes place

Now: We also allow state and action models with more than one designated world/event.

- The set of worlds that may be the actual world (from some agent's perspective)
- The set of events that may actually take place (nondeterministically)

Multipointed Models

closure under indistinguishability

Let $M = (W, \sim, V)$ be an epistemic model, $W' \subseteq W$, and $a \in \mathscr{I}$. Then W' is closed under indistinguishability of agent a if $w \in W'$ and $w \sim_a w'$ implies $w' \in W'$ for all $w, w' \in W$.

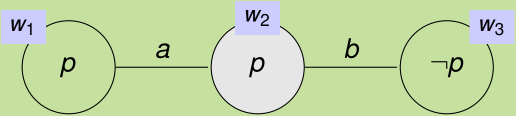
multipointed epistemic model

Let $M = (W, \sim, V)$ be an epistemic model, and $\emptyset \neq W_\mathrm{d} \subseteq W$. Then (M, W_d) is a multipointed model. If $W_\mathrm{d} = \{w\}$, then (M, W_d) is a global state. If W_d is closed under indistinguishability for some agent $a \in \mathscr{I}$, then (M, W_d) is local for agent a. Given a global state $(M, \{w\})$, the associated local state for agent a is the model $(M, \{w\})^a = (M, \{w' \in W \mid w' \sim_a w\})$. Similarly, $(M, W_\mathrm{d})^a = (M, W'_\mathrm{d})$, for $W'_\mathrm{d} = \{w' \in W \mid w' \sim_a w \text{ for some } w \in W_\mathrm{d}\})$.

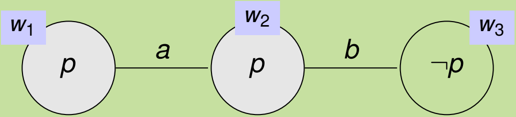
Multipointed Models

Example

Global state $(M, \{w_2\})$:



Associated local state for agent a: $(M, \{w_2\})^a = (M, \{w_1, w_2\})$



Multipointed Models

Truth condition in multipointed models

Given a formula φ and a multipointed model (M, W_d), we define:

$$M, W_\mathrm{d} \models \varphi \quad \text{iff} \quad M, w \models \varphi \text{ for all } w \in W_\mathrm{d}.$$

Note: If (M, W_d) is local for some agent a, then $M, W_\mathrm{d} \models K_a\varphi$ iff $M, W_\mathrm{d} \models \varphi$.

Multipointed Models

multipointed action model

Let $A = (E, \sim, pre)$ be an action model and $\emptyset \neq E_\mathrm{d} \subseteq E$. Then we call (A, E_d) a multipointed action model.

Note: Definitions of closure under indistinguishability, local/global/associated local (action) models similar to those for multipointed epistemic models.

Multipointed Models

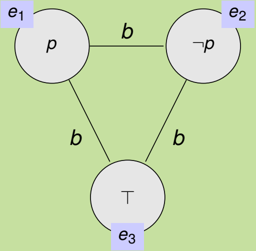
Remark: Multipointed action models show up if

- an action is actually nondeterministic, or
- an action appears nondeterministic from some agent's perspective.

Multipointed Models

Example (Nondeterministic action)

Action model $(\text{Mayread}, \{e_1, e_2, e_3\})$:

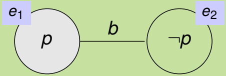


Alice may or may not read the letter, nondeterministically.

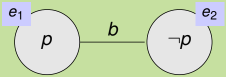
Multipointed Models

Example (Seemingly nondeterministic action)

Action model (Read, e_1):



Associated action model $(\text{Read}, \{e_1, e_2\})$ for agent b:



Although the Read action is deterministic (in every state, only one of the events can possibly take place), it appears nondeterministic to agent *b*, since he does not know which event occurs.

Ontic Effects

So far: Actions only affect knowledge (via announcements, other forms of communication, sensing, . . .).

Now: We also want actions to change ontic facts (opening a door, tossing a coin, toggling a switch, moving from A to B, . . .).

Ontic Effects

Action model with ontic effects

An action model with ontic effects $A = (E, \sim, \mathit{pre}, \mathit{eff})$ is an action model $(E, \sim, \mathit{pre})$ together with a function eff, where for all $e \in E$, $\mathit{eff}(e)$ is a conjunction of atoms and negated atoms from P.

Example

$\mathit{eff}(e) = p \wedge q \wedge \neg r \wedge \neg x$ means that event e makes p and q true and r and x false.

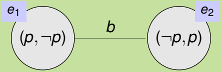
Note: This corresponds to add and delete lists in STRIPS planning.

Ontic Effects

Graphical notation: Label (φ, ψ) means that $pre(e) = \varphi$ and $eff(e) = \psi$.

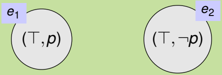
Example (Toggling a switch)

The truth value of p is complemented. Agent a sees p, agent b does not.



Example (Tossing a coin)

A coin is tossed (p means heads, $\neg p$ means tails). The coin toss happens in public.



Ontic Effects

In order to correctly reflect ontic effects in our semantics, we need to make the product update take them into account.

Product update

Let $M = (W, \sim, V)$ be an epistemic state with designated worlds $W_\mathrm{d} \subseteq W$, and let $A = (E, \sim, pre, eff)$ be an action model with designated events $E_\mathrm{d} \subseteq E$. Then the product update $(M, W_\mathrm{d}) \otimes (A, E_\mathrm{d})$ is the epistemic state $M' = (W', \sim', V')$ with with designated worlds $W'_\mathrm{d} \subseteq W'$, where:

- $W' = \{(w,e) \in W \times E \mid M, w \models pre(e)\}$,
- $(w,e) \sim'_a (t, \varepsilon)$ iff $w \sim_a t$ and $e \sim_a \varepsilon$, for $a \in \mathscr{I}$,
- $(w,e) \in V'_p$ iff ($w \in V_p$ and $eff(e) \not\models \neg p$) or $eff(e) \models p$, for all $p \in P$, and
- $(w,e) \in W'_\mathrm{d}$ iff $w \in W_\mathrm{d}$ and $e \in E_\mathrm{d}$.

Ontic Effects

applicability

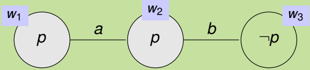
Action (A, E_d) is applicable in local state (M, W_d) iff, for all $w \in W_\mathrm{d}$, there is at least one $e \in E$ with $M, w \models pre(e)$.

Everything else stays more or less the same.

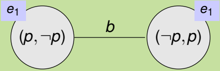
Ontic Effects

Example

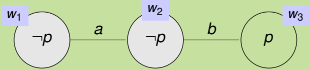
Initially, *a* knows *p* and considers it possible that *b* does not know *p*.



We then apply the toggling action.



Resulting epistemic state: like initially, but with *p* toggled.



Ontic Effects

Recall the funniest joke in the world:

> *Two hunters are out in the woods when one of them collapses. He doesn't seem to be breathing and his eyes are glazed. The other guy whips out his phone and calls the emergency services. He gasps, "My friend is dead! What can I do?" The operator says, "Calm down. I can help. First, let's make sure he's really dead." There is a silence; then a gun shot is heard. Back on the phone, the guy says, "OK, now what?"*

Homework:

- DEL action model for the "epistemic reading" of making sure he's really dead?
- DEL action model for the "ontic reading" of making sure he's really dead?

What is planning?

Planning

“Planning is the art and practice of thinking before acting.”

— Patrik Haslum

- intelligent decision making: What actions to take?
- general-purpose problem representation
- algorithms for solving any problem expressible in the representation

Transition systems

transition system

A transition system is a 5-tuple $\mathscr{T} = (S, L, T, s_0, S_\star)$ where

- S is a finite set of states,
- L is a finite set of (transition) labels,
- $T \subseteq S \times L \times S$ is the transition relation,
- $s_0 \in S$ is the initial state, and
- $S_\star \subseteq S$ is the set of goal states.

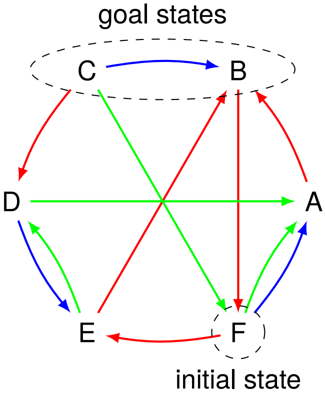
We say that $\mathscr{T}$ has the transition (s, ℓ, s') if $(s, \ell, s') \in T$.

We also write this $s \xrightarrow{\ell} s'$, or $s \rightarrow s'$ when not interested in ℓ.

$\mathscr{T}$ is called deterministic if for all states s and labels ℓ, there is at most one state s' with $s \xrightarrow{\ell} s'$.

Transition systems: example

Transition systems are often depicted as directed arc-labeled graphs with marks to indicate the initial state and goal states.



Transition system terminology

We use common graph theory terms for transition systems:

- s' successor of s if $s \to s'$
- s predecessor of s' if $s \to s'$
- s' reachable from s if there exists a sequence of transitions from s to s'.

Compact representations

- Classical (i. e.„ deterministic) planning is in essence the problem of finding solutions in huge transition systems.
- The transition systems we are usually interested in are too large to explicitly enumerate all states or transitions.
- Hence, the input to a planning algorithm must be given in a more concise form.

State variables

How to represent huge state sets without enumerating them?

- represent different aspects of the world in terms of different state variables

⤳ a state is a valuation of state variables

- *n* state variables with *m* possible values each induce m^n different states

⤳ exponentially more compact than “flat” representations

Boolean state variables

Problem:

- How to succinctly represent transitions and goal states?

Idea: Use propositional logic

- state variables: propositional variables (0 or 1)
- goal states: defined by a propositional formula
- transitions: defined by actions given by
  - precondition: when is the action applicable?
  - effect: how does it change the valuation?

Note: general finite-domain state variables can be compactly encoded as Boolean variables

Operators

Transitions for state sets described by propositions *P* can be concisely represented as operators or actions $o = (pre, eff)$ where

- the precondition *pre* is a propositional formula over *P* describing the set of states in which the transition can be taken (states in which a transition starts), and
- the effect *eff* describes how the resulting successor states are obtained from the state where the transitions is taken (where the transition goes).

Effects (for deterministic operators)

effects

(Deterministic) effects are recursively defined as follows:

- If $p \in P$ is a state variable, then p and $\neg p$ are effects (atomic effect).
- If $eff_1, \ldots, eff_n$ are effects, then $eff_1 \wedge \cdots \wedge eff_n$ is an effect (conjunctive effect).
  The special case with $n = 0$ is the empty effect $\top$.
- If pre is a propositional formula and eff is an effect, then $pre \rhd eff$ is an effect (conditional effect).

Atomic effects p and $\neg p$ are best understood as assignments $p := 1$ and $p := 0$, respectively.

Operator semantics

changes caused by an operator

For each effect *eff* and state s, we define the change set of *eff* in s, written $[\mathit{eff}]_s$, as the following set of literals:

- $[p]_s = \{p\}$ and $[\neg p]_s = \{\neg p\}$ for atomic effects p, $\neg p$
- $[\mathit{eff}_1 \wedge \cdots \wedge \mathit{eff}_n]_s = [\mathit{eff}_1]_s \cup \cdots \cup [\mathit{eff}_n]_s$
- $[\mathit{pre} \rhd \mathit{eff}]_s = [\mathit{eff}]_s$ if $s \models \mathit{pre}$ and $[\mathit{pre} \rhd \mathit{eff}]_s = \emptyset$ otherwise

applicable operators

Operator $(\mathit{pre}, \mathit{eff})$ is applicable in a state s iff $s \models \mathit{pre}$ and $[\mathit{eff}]_s$ is consistent (i. e.„ does not contain two complementary literals).

Operator semantics (ctd.)

successor state

The successor state $app_o(s)$ of s with respect to operator $o = (pre, eff)$ is the state s' with $s' \models [eff]_s$ and $s'(p) = s(p)$ for all state variables p not mentioned in $[eff]_s$.

This is defined only if o is applicable in s.

Deterministic planning tasks

deterministic planning task

A deterministic planning task is a 4-tuple $\Pi = (P, I, \text{Act}, \gamma)$ where

- P is a finite set of state variables (propositions),
- I is a valuation over P called the initial state,
- Act is a finite set of operators over P, and
- γ is a formula over P called the goal.

Mapping planning tasks to transition systems

induced transition system of a planning task

Every planning task $\Pi = (P, I, \mathrm{Act}, \gamma)$ induces a corresponding deterministic transition system $\mathscr{T}(\Pi) = (S, L, T, s_0, S_\star)$:

- S is the set of all valuations of P,
- L is the set of operators Act,
- $T = \{(s, o, s') \mid s \in S,\ o \text{ applicable in } s,\ s' = app_o(s)\}$,
- $s_0 = I$, and
- $S_\star = \{s \in S \mid s \models \gamma\}$

Planning tasks: terminology

- Terminology for transitions systems is also applied to the planning tasks that induce them.
- A sequence of operators that forms a goal path of $\mathscr{T}(\Pi)$ is called a plan of Π.

Planning

By planning, we mean the following two algorithmic problems:

Satisficing planning

Given: a planning task Π

Output: a plan for Π, or **unsolvable** if no plan for Π exists

Optimal planning

Given: a planning task Π

Output: a plan for Π with minimal length among all plans for Π, or **unsolvable** if no plan for Π exists

Nondeterministic operators

Nondeterministic operator

A nondeterministic operator is a pair $o = (pre, Eff)$, where

- pre is a conjunction of atoms (the precondition), and
- $Eff = \{eff_1, \ldots, eff_n\}$ is a finite set of possible effects of o, each eff_i being a conjunction of atomic finite-domain effects.

Nondeterministic operators

Nondeterministic operator application

Let $o = (\mathit{pre}, \mathit{Eff})$ be a nondeterministic operator and s a state.

Applicability of o in s is definied as in the deterministic case, i.e., o is applicable in s iff $s \models \mathit{pre}$ and the change set of each effect $\mathit{eff} \in \mathit{Eff}$ is consistent.

If o is applicable in s, then the application of o in s leads to one of the states in the set $\mathit{app}_o(s) := \{\mathit{app}_{(\mathit{pre},\mathit{eff})}(s) \mid \mathit{eff} \in \mathit{Eff}\}$ nondeterministically.

Nondeterministic planning tasks and transition systems

Nondeterministic planning task: Like a deterministic planning task, but now possibly with nondeterministic actions.

Induced transition system: Like before, but now possibly with nondeterministic transitions.

What is a plan?

In nondeterministic planning, plans are more complicated objects than in the deterministic case:

The best action to take may depend on nondeterministic effects of previous operators.

Nondeterministic plans thus often require branching. Sometimes, they even require looping.

Here: Only consider branching, no looping.

Nondeterministic plans: formal definition

Strategy

Let $\Pi = (P, I, \mathrm{Act}, \gamma)$ be a nondeterministic planning task with state set S and goal states $S_\star$.

A strategy (or policy) for Π is a function $\pi : S_\pi \rightarrow \mathrm{Act}$ for some subset $S_\pi \subseteq S$ such that for all states $s \in S_\pi$ the action $\pi(s)$ is applicable in s.

The set of states reachable in $\mathscr{T}(\Pi)$ starting in state s and following π is denoted by $S_\pi(s)$.

Nondeterministic plans: formal definition

Proper and acyclic strategies

Let $\Pi = (P, I, \mathrm{Act}, \gamma)$ be a nondeterministic planning task with state set S and goal states $S_\star$, and let π be a strategy for Π. Then π is called

- proper iff $S_\pi(s') \cap S_\star \neq \emptyset$ for all $s' \in S_\pi(s_0)$, and
- acyclic iff there is no state $s' \in S_\pi(s_0)$ such that s' is reachable from s' following π in a strictly positive number of steps.

Strongness

Let $\Pi = (P, I, \mathrm{Act}, \gamma)$ be a nondeterministic planning task with state set S and goal states $S_\star$.

A strategy for Π is called a strong plan if it is proper and acyclic.

Strong Planning

strong planning

Given: a nondeterministic planning task Π

Output: a strong plan for Π, or **unsolvable**
if no strong plan for Π exists

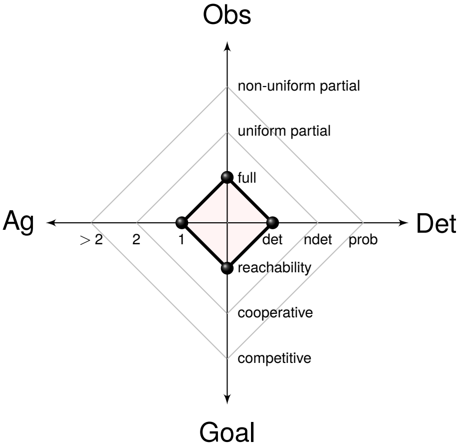
From Classical to Epistemic Planning

Summary: Classical planning on one slide:

- Given:
  - Initial world state
  - Goal description
  - Available actions
- Wanted:
  - Plan leading from initial state to goal state
- Assumptions:
  - Single agent
  - Full observability
  - Deterministic actions
  - Static and discrete environment
  - Reachability goal
  - . . .

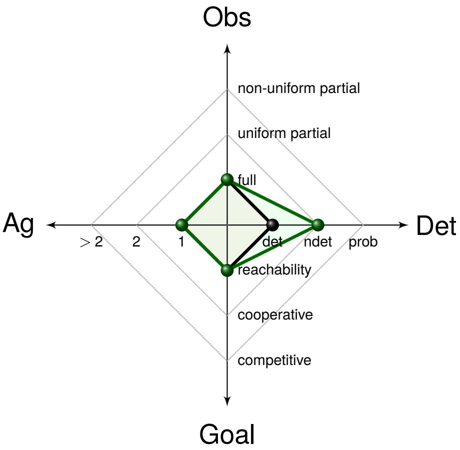
From Classical to Epistemic Planning

Classical



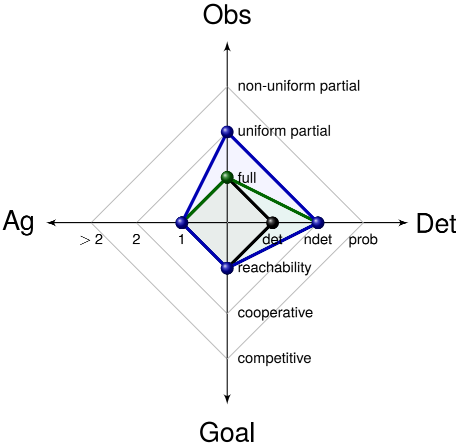
From Classical to Epistemic Planning

Classical, FOND



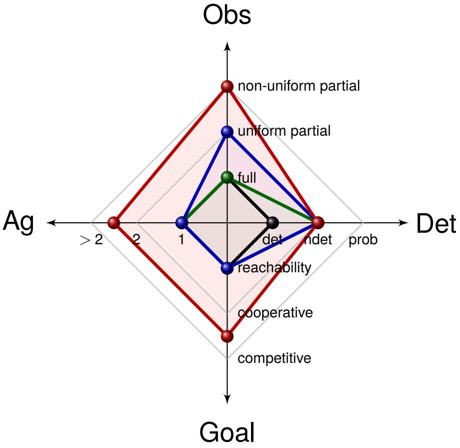
From Classical to Epistemic Planning

Classical, FOND, POND



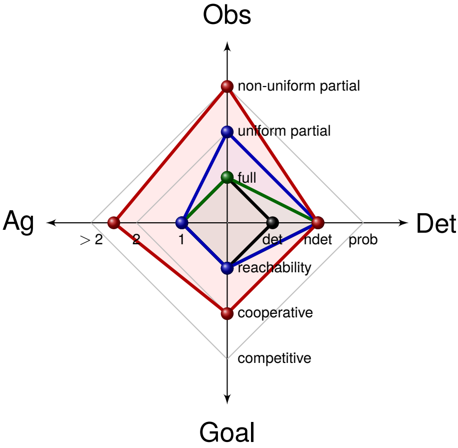
From Classical to Epistemic Planning

Classical, FOND, POND, epistemic planning, . . .

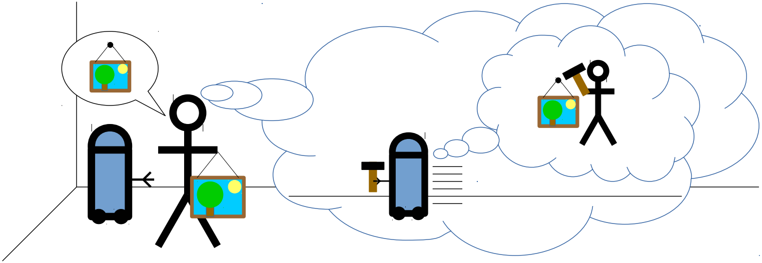


From Classical to Epistemic Planning

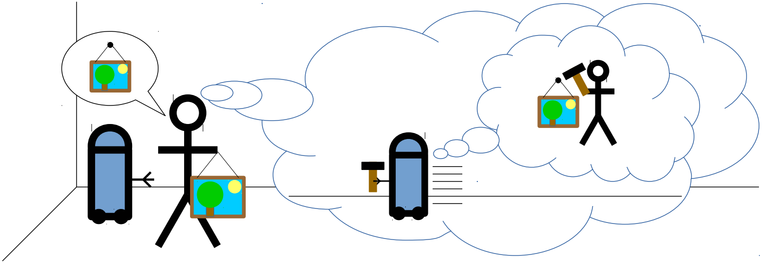
Classical, FOND, POND, epistemic planning, . . .



Example: Robot Collaborating with Human



Example: Robot Collaborating with Human



- Epistemic planning useful if we want the agents to coordinate implicitly

Cooperative Epistemic Planning

Cooperative epistemic planning:

- Task: Collaboratively reach joint goal
- Challenge: Required knowledge and capabilities distributed among agents
- Idea: Communication / coordination as part of the plan

Cooperative Epistemic Planning Tasks

From now on: Multi-pointed models, ontic effects.
Fix a finite set of agents $\mathscr{I}$.

A

cooperative epistemic planning task $\Pi = (P, I, \mathrm{Act}, \gamma, \omega)$ consists of

- a finite set of state variables (atomic propositions) P,
- an initial global epistemic state $I = (M_0, w_0)$ over P,
- a finite set Act of epistemic actions over P,
- a goal formula γ over P, and
- an owner function $\omega : \mathrm{Act} \to \mathscr{I}$, such that each action $\alpha \in \mathrm{Act}$ is local for $\omega(\alpha)$.

Assumption: Act is common knowledge among all agents.

Centralized Sequential Plans

An epistemic model (M, W_{d}) is a goal state iff $(M, W_{\mathrm{d}}) \models \gamma$ iff $(M, w) \models \gamma$ for all $w \in W_{\mathrm{d}}$.

Terminology: In the following, we abbreviate “cooperative epistemic planning task” as “planning task”.

Centralized sequential epistemic plan

A centralized sequential (or linear) epistemic plan for a planning task $\Pi = (P, I, \mathrm{Act}, \gamma, \omega)$ is a sequence of actions from Act, $\pi = \alpha_1, \ldots, \alpha_n$ such that

- for each $i = 1, \ldots, n$, action α_i is applicable in $I \otimes \alpha_1 \otimes \ldots \otimes \alpha_{i-1}$, and
- $I \otimes \alpha_1 \otimes \ldots \otimes \alpha_n \models \gamma$.

Centralized Sequential Plans

In order to simplify and to highlight the simplicity to the definition of implicitly coordinated sequential plans (see below), we give an equivalent definition of centralized sequential epistemic plans:

Proposition

Let $\Pi = (P, I, \mathrm{Act}, \gamma, \omega)$ *be a planning task and* $\pi = \alpha_1, \dots, \alpha_n$ *be a sequence of actions from* Act. *Then* π *is a centralized sequential epistemic plan for* Π *iff*

- $n = 0$ *and* $I \models \gamma$*, or*
- $n > 0$ *and* α_1 *is applicable in* I *and* $\alpha_2, \dots, \alpha_n$ *is a centralized sequential epistemic plan for* $\Pi' = (P, I \otimes \alpha_1, \mathrm{Act}, \gamma, \omega)$. □

Centralized Sequential Plans

For convenience, we add a new modality as an abbreviation:

Modality $(\!(\alpha)\!)$ is defined such that, for all formulas φ, we have

$$(\!(\alpha)\!)\varphi \quad \equiv \quad \langle\alpha\rangle\top \wedge [\alpha]\varphi$$

Truth condition:

$$M, w \models (\!(\alpha)\!)\varphi \quad \text{iff} \quad \alpha \text{ is applicable in } M, w \text{ and } (M, w) \otimes \alpha \models \varphi$$

Centralized Sequential Plans

Proposition

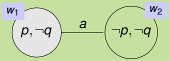
Let $\Pi = (P, I, \mathrm{Act}, \gamma, \omega)$ *be a planning task and* $\pi = \alpha_1, \ldots, \alpha_n$ *a sequence of actions from* Act. *Then* π *is a centralized sequential epistemic plan for* Π *if and only if* $I \models ⦇\alpha_1⦈⦇\alpha_2⦈ \ldots ⦇\alpha_n⦈\gamma$.

Proof by straightforward induction of the length of the plan.

Example

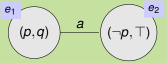
Example

Initial state:

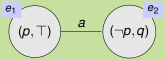


Actions of agent *a*:

α_1 :

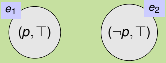


α_2 :



Actions of agent *b*:

β_1 :

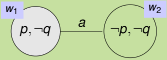


Goal: *q*. Centralized plan:

Example

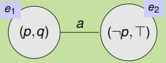
Example

Initial state:

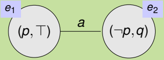


Actions of agent *a*:

α_1 :

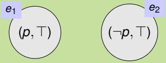


α_2 :



Actions of agent *b*:

β_1 :



Goal: *q*. Centralized plan: $\langle \alpha_1 \rangle$

Plantime vs. Runtime Indistinguishability

Plantime vs. runtime indistinguishability

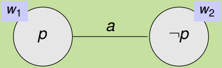
Let $A = (E, \sim, pre, eff)$, $E_\mathrm{d} \subseteq E$, and assume that (A, E_d) is local to some agent $a \in \mathscr{I}$. Let $e_1, e_2 \in E_\mathrm{d}$. Then e_1 and e_2 are called runtime indistinguishable for agent a if $e_1 \sim_a e_2$. Otherwise (if $e_1 \not\sim_a e_2$), they are runtime distinguishable for a, but plantime indistinguishable for a.

Above, we defined plantime and runtime indistinguishability of events. Plantime and runtime indistinguishability of worlds in epistemic states can be defined similarly.

Plantime vs. Runtime Indistinguishability

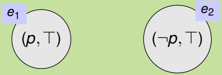
Example (for $\mathscr{I} = \{a\}$)

Model $(\text{Before}, \{w_1, w_2\})$:



Worlds w_1 and w_2 both plantime and runtime indistinguishable to agent a.

Action model $(\text{Read}_a, \{e_1, e_2\})$:

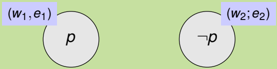


Events e_1 and e_2 plantime indistinguishable, but runtime distinguishable to agent a.

Plantime vs. Runtime Indistinguishability

Example (ctd.)

Model $(\text{After}, W_\mathrm{d}) = (\text{Before}, \{w_1, w_2\}) \otimes (\text{Read}_a, \{e_1, e_2\})$:



Worlds (w_1, e_1) and (w_2, e_2) plantime indistinguishable, but runtime distinguishable to agent *a*.

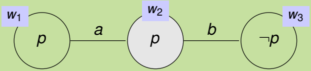
Implicitly Coordinated Sequential Plans

Recall (local perspective of an agent):

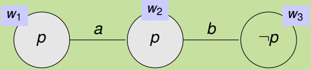
If (M, W_d) is an epistemic state and a is an agent, then $(M, W_\mathrm{d})^a = (M, W'_\mathrm{d})$ is agent a's associated local state, where $W'_\mathrm{d} = \{w' \in W \mid w' \sim_a w \text{ for some } w \in W_\mathrm{d}\}$).

Example

Global state $(M, \{w_2\})$:



Associated local state for agent a: $(M, \{w_2\})^a = (M, \{w_1, w_2\})$



Implicitly Coordinated Sequential Plans

In an implicitly coordinated plan, an agent knows that its chosen action is applicable and makes progress towards the goal.

Implicitly coordinated sequential plan

Let $\Pi = (P, I, \text{Act}, \gamma, \omega)$ be a planning task and $\pi = \alpha_1, \ldots, \alpha_n$ a sequence of actions from Act. Then π is an implicitly coordinated sequential epistemic plan (ICSEP) for Π iff either

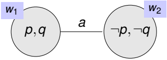
- $n = 0$ and $I \models \gamma$, or
- $n > 0$ and α_1 is applicable in $I^{\omega(\alpha_1)}$ and $\alpha_2, \ldots, \alpha_n$ is a ICSEP for $\Pi' = (P, I^{\omega(\alpha_1)} \otimes \alpha_1, \text{Act}, \gamma, \omega)$.

Implicitly Coordinated Sequential Plans

Recall the previous example:

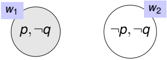
1. (α_1) is not an ICSEP.
2. (β_1, α_1) is an ICSEP.

Ad (1) α_1 is applicable in $I^{\omega(\alpha_1)}$, but $I^{\omega(\alpha_1)} \otimes \alpha_1$:



So, *a* was successful but does not know it!

Ad (2) β_1 is applicable in $I^{\omega(\beta_1)}$ leading to:



From this state, *a* knowing that *p*, α_1 is a ICSEP!

Implicitly Coordinated Sequential Plans

A simple lemma we will need in a moment.

Proposition (knowledge and associated local states)

$(M, W_{\mathrm{d}})^a \models \varphi$ *iff* $M, W_{\mathrm{d}} \models K_a\varphi$.

Proof.

$$\begin{aligned}
(M, W_{\mathrm{d}})^a \models \varphi \quad &\text{iff} \quad (M, \{w' \mid w' \sim_a w \text{ for some } w \in W_{\mathrm{d}}\}) \models \varphi \\
&\text{iff} \quad M, w' \models \varphi \text{ f.a. } w' \text{ s.t. ex. } w \in W_{\mathrm{d}} \text{ s.t. } w' \sim_a w \\
&\text{iff} \quad M, w \models K_a\varphi \text{ for all } w \in W_{\mathrm{d}} \\
&\text{iff} \quad M, W_{\mathrm{d}} \models K_a\varphi
\end{aligned}$$

□

Implicitly Coordinated Sequential Plans

Proposition

Let $\Pi = (P, I, \mathrm{Act}, \gamma, \omega)$ *be a planning task and* $\pi = \alpha_1, \dots, \alpha_n$ *a sequence of actions from* Act*. Then* π *is an implicitly coordinated sequential epistemic plan for* Π *if and only if* $I \models K_{\omega(\alpha_1)}⦇\alpha_1⦈K_{\omega(\alpha_2)}⦇\alpha_2⦈\dots K_{\omega(\alpha_n)}⦇\alpha_n⦈\gamma$.

Proof.

Induction on plan length n.

- Base case ($n = 0$): Then π is an implicitly coordinated sequential epistemic plan iff $I \models \gamma$.
- Inductive case ($n > 0$): [...]

Implicitly Coordinated Sequential Plans

Proof (ctd.)

- Inductive case ($n > 0$): Let $\pi = \alpha_1, \ldots, \alpha_n$.

  Then π is an implicitly coordinated epistemic plan for Π iff (definition)

  α_1 is applicable in $I^{\omega(\alpha_1)}$ and $\alpha_2, \ldots, \alpha_n$ is an implicitly coordinated epistemic plan for $\Pi' = (P, I^{\omega(\alpha_1)} \otimes \alpha_1, \mathrm{Act}, \gamma, \omega)$ iff (induction hypothesis!)

  α_1 is applicable in $I^{\omega(\alpha_1)}$ and
  $I^{\omega(\alpha_1)} \otimes \alpha_1 \models K_{\omega(\alpha_2)}(\!(\alpha_2)\!) \ldots K_{\omega(\alpha_n)}(\!(\alpha_n)\!)\gamma$ iff
  (truth condition of $(\!(\cdot)\!)$)

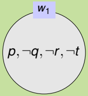
  $I^{\omega(\alpha_1)} \models (\!(\alpha_1)\!)K_{\omega(\alpha_2)}(\!(\alpha_2)\!) \ldots K_{\omega(\alpha_n)}(\!(\alpha_n)\!)\gamma$ iff
  (knowledge and associated local states)

  $I \models K_{\omega(\alpha_1)}(\!(\alpha_1)\!)K_{\omega(\alpha_2)}(\!(\alpha_2)\!) \ldots K_{\omega(\alpha_n)}(\!(\alpha_n)\!)\gamma.$ □

Implicitly Coordinated Sequential Plans

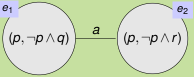
Example

Initial state



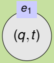
Actions of *a*:

α_1 :

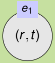


Actions of *b*:

β_1 :



β_2 :

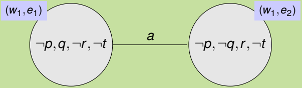


Goal: $\gamma = t$

Implicitly Coordinated Sequential Plans

Example (ctd.)

There is no ICSEP for this planning task. Reason: if there were one, it would have to start with α_1 (nothing else is applicable).

Then, $I^{\omega(\alpha_1)} \otimes \alpha_1 =$  $=: (M^1, W^1_{\mathrm{d}})$.

In (M^1, W^1_{d}), none of the available actions is applicable, and it is not a goal state.

This task would be solvable with a branching or conditional plan: start with α_1, and depending on the outcome, continue with β_1 or β_2. This would even be implicitly coordinated in the sense that at each point in the plan, the agent to move knows that it can move and that this leads to progress.

Branching Plans

Notation: In the following, we use w not only to refer to worlds in epistemic models, but also to (single-pointed) epistemic models themselves. Will be clear from the context.

Policy

Let $\Pi = (P, I, \mathrm{Act}, \gamma, \omega)$ be an epistemic planning task and let W^{gl} be the set of global epistemic states of Π. Then a policy is a mapping $\pi : W^{\mathrm{gl}} \to 2^{\mathrm{Act}}$ such that:

- Applicability (APP): for all $w \in W^{\mathrm{gl}}$ and all $\alpha \in \pi(w)$, α is applicable in w.
- Determinism (DET): for all $w \in W^{\mathrm{gl}}$ and all $\alpha, \alpha' \in \pi(w)$ with $\omega(\alpha) = \omega(\alpha')$, we have $\alpha = \alpha'$.
- Uniformity (UNIF): for all $w, t \in W^{\mathrm{gl}}$ and all $\alpha \in \pi(w)$ with $w^{\omega(\alpha)} = t^{\omega(\alpha)}$, we have $\alpha \in \pi(t)$.

Branching Plans

Note:
APP and UNIF together imply knowledge of preconditions (KOP): for all $w \in W^{\mathrm{gl}}$ and all $\alpha \in \pi(w)$, α is applicable in $w^{\omega(\alpha)}$, i. e., agents supposed to act know that their action is applicable.

Note:
We also need to require that the policy is strong in the sense that one always achieves the goal (which we will not do here).

Summary

- Multipointed models and ontic effects
- Review of classical planning
- Centralied vs. implicitly coordinated plans
- Sequential vs. branching plans

Literature

- Baral et al., **Epistemic Planning** (Dagstuhl Seminar 17231), Dagstuhl Reports, Vol. 7, Issue 6, 2017, `http://drops.dagstuhl.de/opus/volltexte/2017/8285/`
- Bolander, **A Gentle Introduction to Epistemic Planning: The DEL Approach**, 2017, `https://arxiv.org/abs/1703.02192`