Multi-Agent Systems

Dynamic Epistemic Logic

Albert-Ludwigs-Universität Freiburg

Bernhard Nebel, Rolf Bergdoll, and Thorsten Engesser

Winter Term 2019/20

Action Models

So far: Only public announcements.

Now: How to model other ways of knowledge changes, such as private announcements, sensing, or ontic (world-changing) actions that affect knowledge along the way?

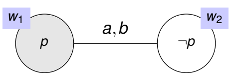
Idea: Action models similar to Kripke models.

Action Models

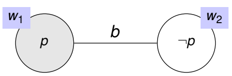
Example

Agents a and b both don't know the value of proposition p. This is common knowledge among them. In fact, p is true. Then agent a receives a letter containing the value of p and reads it. Agent b observes a reading the letter and knows that it is about p, but b does not learn the value of p.

Model Before:



Model After:



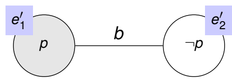
Action Models

Question: How to get from Before to After?

Answer: Action models.

Remark: After, $w_1' \models$
$K_a p \wedge (\neg K_b p \wedge \neg K_b \neg p) \wedge K_b(K_a p \vee K_a \neg p) \wedge K_a(\neg K_b p \wedge \neg K_b \neg p)$
$\rightsquigarrow$ action model needs to achieve exactly that!

Action model Read:



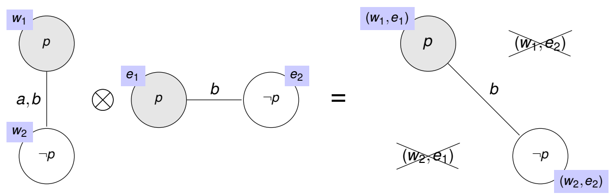
With this action model, After = Before $\otimes$ Read, for an appropriate definition of $\otimes$.

Action Models

Product update, informally

The product update $\otimes$ denotes a restricted modal update with component worlds (w,e) only present if $(M,w) \models pre(e)$.

Model Before $\otimes$ Read:



- $(w_1,e_1) \sim_b (w_2,e_2)$ because $w_1 \sim_b w_2$ and $e_1 \sim_b e_2$.
- (w_1,e_2) and (w_2,e_1) were eliminated because e_2 cannot be applied in w_1 and e_1 cannot be applied in w_2.

Action Models

Action model

Let $\mathscr{L}$ be any logical language for a set of agents $\mathscr{I}$ and a set of atoms P. Then an $S5$ action model A is a structure $(E,\sim,pre)$ such that:

- E is the domain of events,
- $\sim_a$ is an equivalence relation on E for all $a \in \mathscr{I}$, the indistinguishability relation for agent a, and
- $pre : E \to \mathscr{L}$ is the precondition function that assigns a precondition $pre(e) \in \mathscr{L}$ to all $e \in E$.

A pointed action model is such a structure (A,e) with $e \in E$.

Action Models

Example (Action model Read, formally)

Read is the action model $(\{e_1,e_2\},\sim,pre)$ with

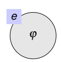
$$\sim_a = \{(e_1,e_1),(e_2,e_2)\} \qquad pre(e_1) = p$$
$$\sim_b = \{(e_1,e_1),(e_1,e_2),(e_2,e_1),(e_2,e_2)\} \qquad pre(e_2) = \neg p.$$

(and with pointed event e_1).

Remark: Public announcements are a special case of action models.

Example (Public announcements)

Action model for the public announcement of φ:



Action Models

Fix agents $\mathscr{I}$ and atomic propositions P.

Example (Skip)

Action skip (or **1**) is the pointed action model $((\{e\},\sim,pre),e)$ with $pre(e) = \top$ and $\sim_a = \{(e,e)\}$ for all $a \in \mathscr{I}$.

Example (Crash)

Action crash (or **0**) is the pointed action model $((\{e\},\sim,pre),e)$ with $pre(e) = \bot$ and $\sim_a = \{(e,e)\}$ for all $a \in \mathscr{I}$.

Action Models

Language

Let P be a countable set of atomic propositions and $\mathscr{I}$ a finite set of agent symbols. Then the language of action model logic is the union of the formulas φ and the actions α defined by the following BNF:

$$\varphi ::= p \mid \neg\varphi \mid (\varphi \wedge \varphi) \mid K_a\varphi \mid C_B\varphi \mid [\alpha]\varphi$$
$$\alpha ::= (A, e) \mid \alpha \cup \alpha$$

where $p \in P$, $a \in \mathscr{I}$, $B \subseteq \mathscr{I}$, and (A, e) is a pointed action model with a finite domain E, and

- for all events $e' \in E$, the precondition $pre(e')$ is a formula that has already been constructed in a previous step of the induction.

Action Models

Intuition:

- $[\alpha]\varphi$: After (every) application of action α, φ is true.

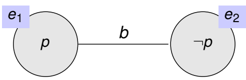
Abbreviations:

- $\langle\alpha\rangle\varphi := \neg[\alpha]\neg\varphi$
  After (some) application of action α, φ is true.
- $A := \bigcup_{e \in E}(A, e)$

Action Models

Deterministic vs. nondeterministic actions:

- $\alpha = (A, e)$: Deterministic action α with unique pointed event e. Example: $\alpha = (\text{Read}, e_1)$.
- $\alpha = \alpha_1 \cup \alpha_2$: Nondeterministic choice, i. e., either α_1 or α_2 happens. Example: $\alpha = (\text{Read}, e_1) \cup (\text{Read}, e_2) = \text{Read}$.
  - Remark 1a: α = Read not properly nondeterministic, since preconditions of e_1 and e_2 are mutually exclusive.
  - Remark 1b: We will see a properly nondeterministic action later (action Mayread).
  - Remark 2: If, for $\alpha = (A_1, e_1) \cup (A_2, e_2)$, we have $A_1 = A_2$, then we can depict α as a multi-pointed model, like $(\text{Read}, e_1) \cup (\text{Read}, e_2)$:



Action Models

Product update

Let $M = (W, \sim, V)$ be an epistemic (i.e., S5) model and let $A = (E, \sim, pre)$ be an action model. Then the product update $M \otimes A$ is the epistemic model $M' = (W', \sim', V')$ with:

- $W' = \{(w, e) \in W \times E \mid M, w \models pre(e)\}$,
- $(w, e) \sim'_a (t, \varepsilon)$ iff $w \sim_a t$ and $e \sim_a \varepsilon$, for $a \in \mathscr{I}$, and
- $(w, e) \in V'_p$ iff $w \in V_p$.

Example

$(\text{Before}, w_1) \otimes (\text{Read}, e_1) = (\text{After}, (w_1, e_1))$

Action Models

Semantics of formulas and actions

Let (M, w) be an epistemic state, φ be a formula and α an action model.

$$M, w \models p,\ \neg\varphi,\ \varphi \wedge \psi,\ K_a\varphi,\ C_B\varphi \ \text{ as usual}$$

$$M, w \models [\alpha]\varphi \quad \text{iff} \quad \text{for all } (M', w') : (M, w)[\![\alpha]\!](M', w') \text{ implies } (M', w') \models \varphi$$

where

- $(M, w)[\![(A, e)]\!](M', w')$ iff $(M, w) \models pre(e)$ and $(M', w') = (M \otimes A, (w, e))$, and
- $[\![\alpha \cup \alpha']\!] = [\![\alpha]\!] \cup [\![\alpha']\!]$.

Action Models

Remarks:

- For $\alpha = (A, e)$, $[\![\alpha]\!]$ is functional, i. e., for each (M, w), there is at most one (M', w') with $(M, w)[\![(A, e)]\!](M', w')$.
- For $\alpha = \alpha_1 \cup \alpha_2$, this is no longer necessarily the case. Careful with duality between $[\alpha]$ and $\langle\alpha\rangle$, then.

Special case $\alpha = (A, e)$: Then $M, w \models [\alpha]\varphi$ iff $M, w \models pre(e)$ implies $(M \otimes A, (w, e)) \models \varphi$.

Dual $\langle\alpha\rangle$, for $\alpha = (A, e)$:

$$M, w \models \langle\alpha\rangle\varphi \quad \text{iff}$$
$$M, w \not\models [\alpha]\neg\varphi \quad \text{iff}$$
$$M, w \models pre(e) \ \text{ does not imply } \ (M \otimes A, (w, e)) \models \neg\varphi \quad \text{iff}$$
$$M, w \models pre(e) \ \text{ and } \ (M \otimes A, (w, e)) \not\models \neg\varphi \quad \text{iff}$$
$$M, w \models pre(e) \ \text{ and } \ (M \otimes A, (w, e)) \models \varphi$$

Action Models

Remark: This is very similar to the semantics of $[\varphi]\psi$ and $\langle\varphi\rangle\psi$ in public announcement logic.

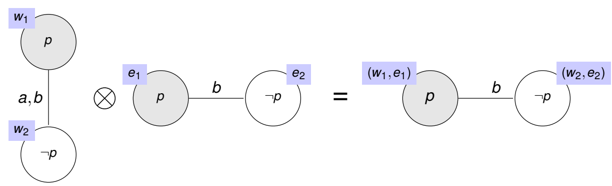
For completeness, dual $\langle\alpha\rangle$, for general α:

$$M, w \models \langle\alpha\rangle\varphi \quad \text{iff}$$
$$M, w \not\models [\alpha]\neg\varphi \quad \text{iff}$$
$$\text{not f. a. } (M', w') : (M, w)[\![\alpha]\!](M', w') \text{ implies } (M', w') \models \neg\varphi \quad \text{iff}$$
$$\text{there ex. } (M', w') : (M, w)[\![\alpha]\!](M', w') \text{ and } (M', w') \not\models \neg\varphi \quad \text{iff}$$
$$\text{there ex. } (M', w') : (M, w)[\![\alpha]\!](M', w') \text{ and } (M', w') \models \varphi$$

Action Models

Example

Model $(\text{Before}, w_1) \otimes (\text{Read}, e_1)$:



Then:

- $\text{Before}, w_1 \models [\text{Read}, e_1]K_a p$
- $\text{Before}, w_1 \models [\text{Read}, e_1]\neg K_b K_a p$
- $\text{Before}, w_1 \models [\text{Read}, e_1]C_{ab}(K_a p \vee K_a \neg p)$

Example

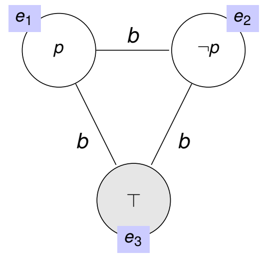
Now, *a* may only read the letter, but does not have to. Agent *b* does not know whether *a* will read it or not. Actually, *a* does not read the letter.

From *b*'s perspective, there are three possibilities:

- *a* reads the letter and learns that *p* is true.
- *a* reads the letter and learns that *p* is false.
- *a* does not read the letter and learns nothing about *p*.

Example (ctd.)

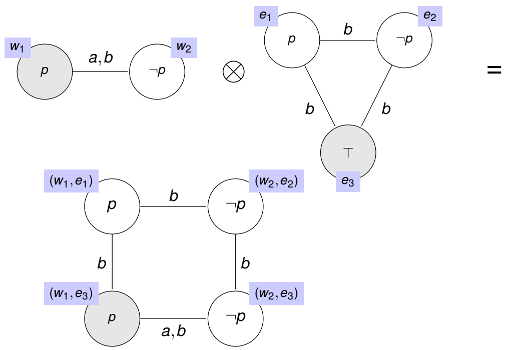
Action model $(\text{Mayread}, e_3)$:



$$\text{Mayread} = (\text{Mayread}, e_1) \cup (\text{Mayread}, e_2) \cup (\text{Mayread}, e_3)$$

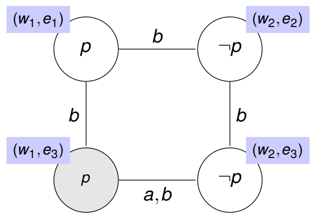
Example (ctd.)

Model $(\text{Before}, w_1) \otimes (\text{Mayread}, e_3)$:



Example (ctd.)

Model $(\text{Before}, w_1) \otimes (\text{Mayread}, e_3)$:



- $\text{Before}, w_1 \models [\text{Mayread}, e_3]\neg(K_a p \vee K_a \neg p) \wedge \hat{K}_b(K_a p \vee K_a \neg p)$
- $\text{Before} \models p \rightarrow$ $(\langle\text{Mayread}\rangle K_a p \wedge \langle\text{Mayread}\rangle \neg K_a p \wedge \neg\langle\text{Mayread}\rangle K_a \neg p)$

Summary

- Action models allow more epistemic change than just public announcements.
- Action models similar to Kripke structures. State update by product update operator.
- Axiomatization similar to public announcement logic. Actions and (common) knowledge slightly trickier.

Literature

- L. S. Moss, Dynamic Epistemic Logic, Chapter 6, In H. van Dithmarschen, J. Y. Halpern, W. van der Hoek, B. Kooi (eds.) **Handbook of Epistemic Logic**, College Publications, 2015.
- Hans P. van Ditmarsch and Wiebe van der Hoek and Barteld Kooi, **Dynamic Epistemic Logic**, Springer, 2007.