Updating Knowing How

Raul Fervari

Logics, Interaction and Intelligent Systems Group (LIIS),

FAMAF-UNC / CONICET, Argentina

(Joint work with C. Areces, A. Saravia, F. Velázquez-Quesada)

FM&AI Working Group - LMF - France - 2023

The concept(s) of knowledge

- Usually, epistemic logic is about "knowing that" (Hintikka 1962):
  - *John knows that it is raining in Paris*,
  - *the robot knows that it is standing next to a wall*...

The concept(s) of knowledge

- Usually, epistemic logic is about "knowing that" (Hintikka 1962):
  - *John knows that it is raining in Paris*,
  - *the robot knows that it is standing next to a wall*...
  - Typically, modal formulas $K_i\varphi$ expressing *"agent i knows that φ"*.

The concept(s) of knowledge

- Usually, epistemic logic is about "knowing that" (Hintikka 1962):
  - *John knows that it is raining in Paris*,
  - *the robot knows that it is standing next to a wall*...
  - Typically, modal formulas $K_i\varphi$ expressing *"agent i knows that φ"*.
- Study other patterns of reasoning:
  - knowing why,
  - knowing whether,
  - knowing who,
  - knowing how.

Knowing How

- Autonomous agent: intelligent entities operating in a given environment (perception, decision making, etc).
- Related to the abilities of the agents to achieve a certain goal.
- Inspired by AI planning.
- Interpreted as: *there exists a proper course of action (sequence of actions) that the agent can take to achieve the goal.*

Knowing How

- Autonomous agent: intelligent entities operating in a given environment (perception, decision making, etc).
- Related to the abilities of the agents to achieve a certain goal.
- Inspired by AI planning.
- Interpreted as: *there exists a proper course of action (sequence of actions) that the agent can take to achieve the goal.*
  - What "proper" means?
  - Different costs?
  - How to update the agents' knowledge (how)?

History of Knowing How Approaches

- Knowing that + Abilities (Lespérance et al. 2000), (Herzig & Troquard 2006), etc.
- A single binary modality for knowing how $\mathsf{Kh}(\psi, \varphi)$ and variants: (Wang 2015).
- Knowing how + knowing that (Fervari et al. 2017).
- Semantics based on indistinguishability/uncertainty between plans (Areces et al. 2021).

History of Knowing How Approaches

- Knowing that + Abilities (Lespérance et al. 2000), (Herzig & Troquard 2006), etc.
- A single binary modality for knowing how $\mathsf{Kh}(\psi, \varphi)$ and variants: (Wang 2015).
- Knowing how + knowing that (Fervari et al. 2017).
- Semantics based on indistinguishability/uncertainty between plans (Areces et al. 2021).

We argue the latter enables us to develop a full theory of epistemic logic of knowing how.

Models: Labelled Transition Systems (LTSs)

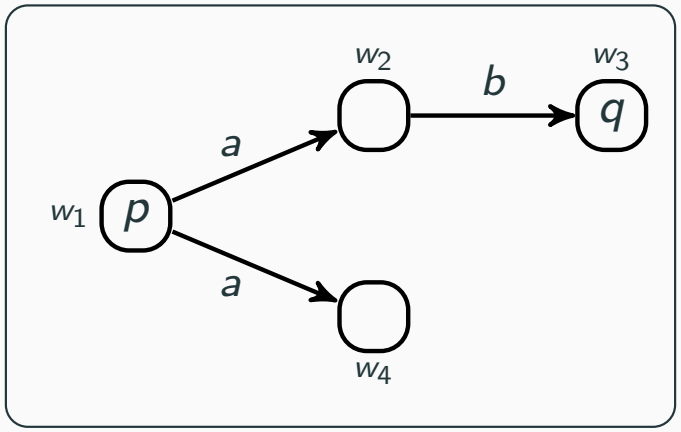
An LTS is a tuple $\mathcal{S} = \langle \mathsf{S}, \{\mathsf{R}_a\}_{a \in \mathsf{Act}}, \mathsf{V} \rangle$ where:

- S is a countable set of states
- $\mathsf{R}_a \subseteq \mathsf{S} \times \mathsf{S}$, for each $a \in \mathsf{Act}$.
- $\mathsf{V} : \mathsf{Prop} \to 2^{\mathsf{S}}$

Models: Labelled Transition Systems (LTSs)

An LTS is a tuple $\mathcal{S} = \langle \mathsf{S}, \{\mathsf{R}_a\}_{a \in \mathsf{Act}}, \mathsf{V} \rangle$ where:

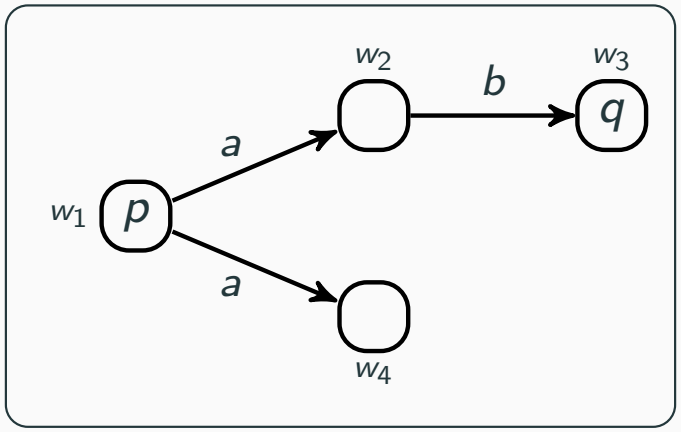
- S is a countable set of states
- $\mathsf{V} : \mathsf{Prop} \to 2^{\mathsf{S}}$
- $\mathsf{R}_a \subseteq \mathsf{S} \times \mathsf{S}$, for each $a \in \mathsf{Act}$.



Models: Labelled Transition Systems (LTSs)

An LTS is a tuple $\mathcal{S} = \langle \mathsf{S}, \{\mathsf{R}_a\}_{a \in \mathsf{Act}}, \mathsf{V} \rangle$ where:

- S is a countable set of states
- $\mathsf{V} : \mathsf{Prop} \to 2^{\mathsf{S}}$
- $\mathsf{R}_a \subseteq \mathsf{S} \times \mathsf{S}$, for each $a \in \mathsf{Act}$.

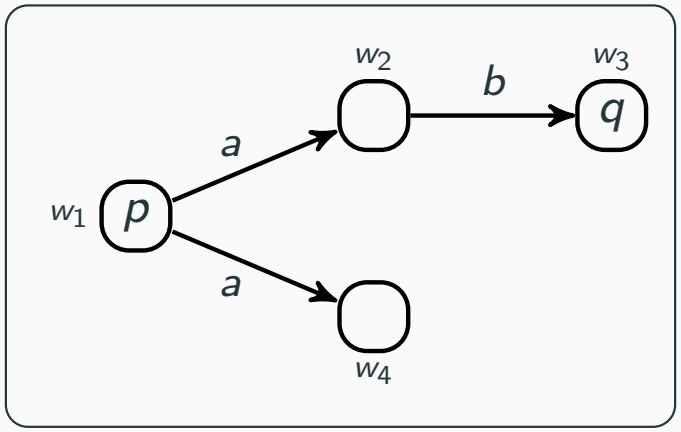


A transition a from w_1 to w_2 is read as "after executing action a at state w_1, the agent reaches state w_2".

Models: Labelled Transition Systems (LTSs)

An LTS is a tuple $\mathcal{S} = \langle \mathsf{S}, \{\mathsf{R}_a\}_{a \in \mathsf{Act}}, \mathsf{V} \rangle$ where:

- S is a countable set of states
- $\mathsf{V} : \mathsf{Prop} \to 2^{\mathsf{S}}$
- $\mathsf{R}_a \subseteq \mathsf{S} \times \mathsf{S}$, for each $a \in \mathsf{Act}$.

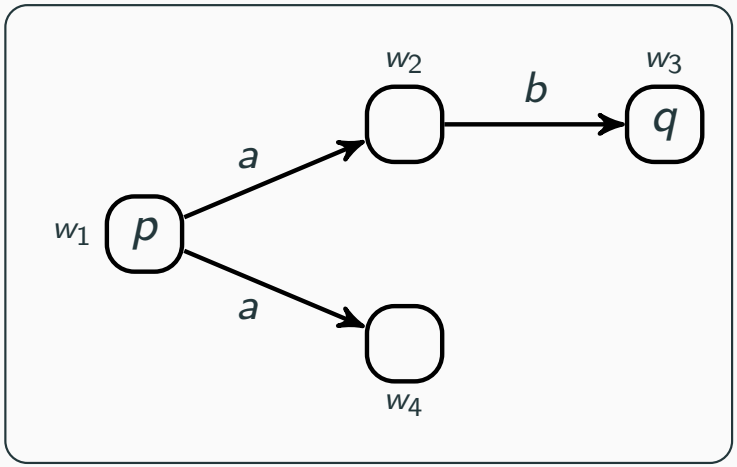


A transition a from w_1 to w_2 is read as "after executing action a at state w_1, the agent reaches state w_2".

For a set of actions Act, a plan σ is an element from Act^* (finite sequences of symbols from Act, such as a, ab and the empty plan ϵ).

Strong executability

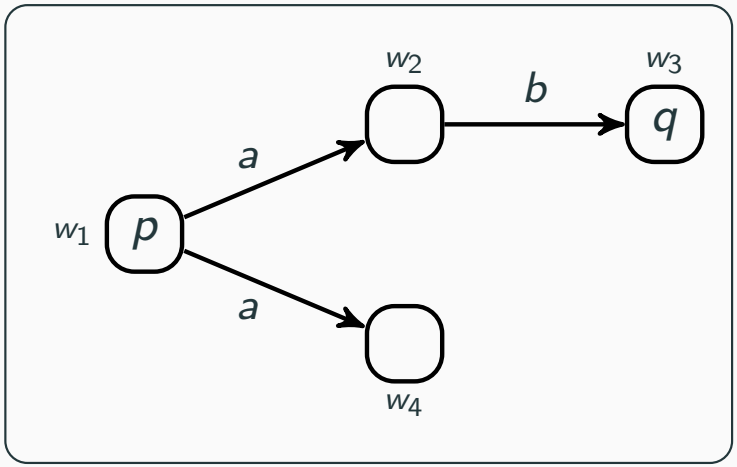
A plan must be fail-proof: each partial execution must be completed.



ab is not SE at w_1

Strong executability

A plan must be fail-proof: each partial execution must be completed.



ab is not SE at w_1

Definition:

A plan σ is strongly executable (SE) at $u \in S$ iff for all partial execution of σ from u, such an execution can be completed.

Towards an epistemic logic of knowing how

There are many reasons to **not** knowing how. What if...

- The agent is not aware of the existence of certain plans?

Towards an epistemic logic of knowing how

There are many reasons to **not** knowing how. What if...

- The agent is not aware of the existence of certain plans?
- The agent is not able to distinguish certain plan from another?

Towards an epistemic logic of knowing how

There are many reasons to **not** knowing how. What if...

- The agent is not aware of the existence of certain plans?
- The agent is not able to distinguish certain plan from another?
- The agent does not care about the difference among certain plans?

Towards an epistemic logic of knowing how

There are many reasons to **not** knowing how. What if...

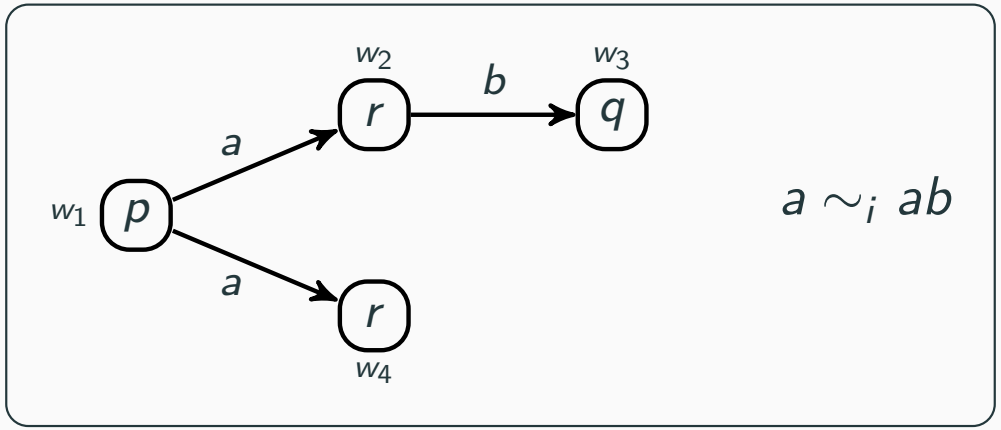
- The agent is not aware of the existence of certain plans?
- The agent is not able to distinguish certain plan from another?
- The agent does not care about the difference among certain plans?

We introduced the notion of epistemic indistinguishability at the level of plans, to fix these issues (Areces et al. 2021).

Uncertainty-based LTS ($\mathrm{LTS}^{\mathrm{U}}$)

An $\mathrm{LTS}^{\mathrm{U}}$ is a tuple $\mathcal{S} = \langle \mathrm{S}, \{\mathrm{R}_a\}_{a \in \mathrm{Act}}, \{\sim_i\}_{i \in \mathrm{Agt}}, \mathrm{V} \rangle$ where:

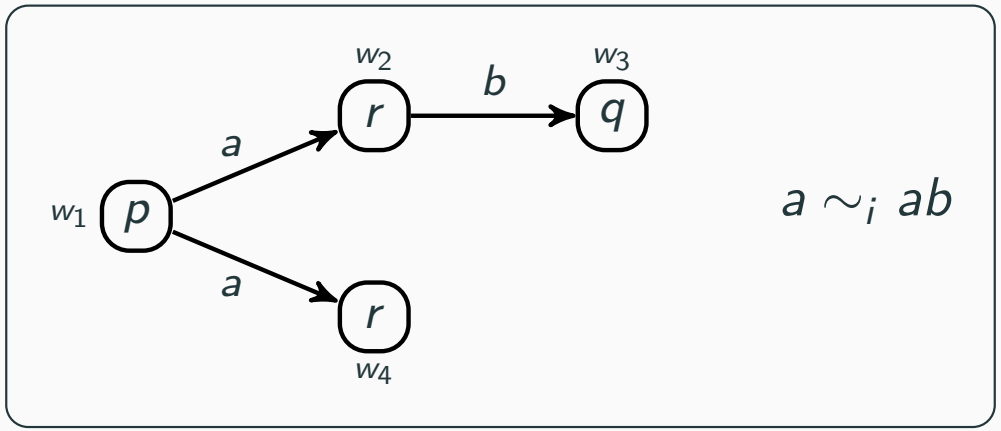
- $\langle \mathrm{S}, \{\mathrm{R}_a\}_{a \in \mathrm{Act}}, \mathrm{V} \rangle$ is an LTS,
- $\sim_i$ is an equivalence relation over a non-empty set of plans, for each $i \in \mathrm{Agt}$ (a set of agent symbols).



Uncertainty-based LTS ($\mathsf{LTS}^{\mathsf{U}}$)

An $\mathsf{LTS}^{\mathsf{U}}$ is a tuple $\mathcal{S} = \langle \mathsf{S}, \{\mathsf{R}_a\}_{a \in \mathsf{Act}}, \{\sim_i\}_{i \in \mathsf{Agt}}, \mathsf{V} \rangle$ where:

- $\langle \mathsf{S}, \{\mathsf{R}_a\}_{a \in \mathsf{Act}}, \mathsf{V} \rangle$ is an LTS,
- $\sim_i$ is an equivalence relation over a non-empty set of plans, for each $i \in \mathsf{Agt}$ (a set of agent symbols).



We call U_i the set of equivalence classes (over plans) by $\sim_i$.

Semantics over $\mathbf{LTS^U}$

Definition ($\mathsf{L}_{\mathsf{Kh}_i}$ over $\mathbf{LTS^U}$).

Formulas of the language $\mathsf{L}_{\mathsf{Kh}_i}$ are given by

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid \mathsf{Kh}_i(\varphi, \varphi),$$

with $p \in \mathsf{Prop}$, and $i \in \mathsf{Agt}$ and $a \in \mathsf{Act}$.

Semantics over $\mathbf{LTS^U}$

Definition ($\mathsf{L}_{\mathsf{Kh}_i}$ over $\mathbf{LTS^U}$).

Formulas of the language $\mathsf{L}_{\mathsf{Kh}_i}$ are given by

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid \mathsf{Kh}_i(\varphi, \varphi),$$

with $p \in \mathsf{Prop}$, and $i \in \mathsf{Agt}$ and $a \in \mathsf{Act}$. $\mathsf{Kh}_i(\psi, \varphi)$ is read as "agent i knows how to achieve φ given ψ".

Semantics over $\mathsf{LTS}^{\mathsf{U}}$

Definition ($\mathsf{L}_{\mathsf{Kh}_i}$ over $\mathsf{LTS}^{\mathsf{U}}$).

Formulas of the language $\mathsf{L}_{\mathsf{Kh}_i}$ are given by

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid \mathsf{Kh}_i(\varphi, \varphi),$$

with $p \in \mathsf{Prop}$, and $i \in \mathsf{Agt}$ and $a \in \mathsf{Act}$. $\mathsf{Kh}_i(\psi, \varphi)$ is read as "agent i knows how to achieve φ given ψ".

$\mathcal{S}, w \models \mathsf{Kh}_i(\psi, \varphi)$ iff there exists a set of plans $\pi \in \mathsf{U}_i$ such that:

1. *each plan in π is SE at every ψ-state; and*
2. *from ψ-states, each plan in π always ends at φ-states.*

Semantics over $\mathsf{LTS}^{\mathsf{U}}$

Definition ($\mathsf{L}_{\mathsf{Kh}_i}$ over $\mathsf{LTS}^{\mathsf{U}}$).

Formulas of the language $\mathsf{L}_{\mathsf{Kh}_i}$ are given by

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid \mathsf{Kh}_i(\varphi, \varphi),$$

with $p \in \mathsf{Prop}$, and $i \in \mathsf{Agt}$ and $a \in \mathsf{Act}$. $\mathsf{Kh}_i(\psi, \varphi)$ is read as "agent i knows how to achieve φ given ψ".

$\mathcal{S}, w \models \mathsf{Kh}_i(\psi, \varphi)$ iff there exists a set of plans $\pi \in \mathsf{U}_i$ such that:

1. *each plan in π is SE at every ψ-state; and*
2. *from ψ-states, each plan in π always ends at φ-states.*

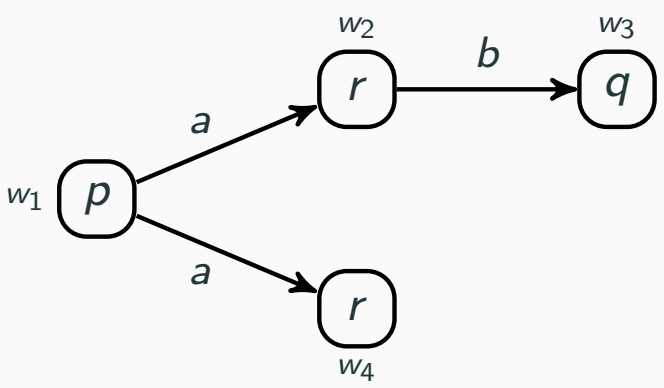
Property:

Define $\mathsf{A}\varphi := \bigvee_{i \in \mathsf{Agt}} \mathsf{Kh}_i(\neg\varphi, \bot)$, we have:

$$\mathcal{S}, w \models \mathsf{A}\varphi \text{ iff for all } v,\ \mathcal{S}, v \models \varphi;$$

i.e., A is the standard universal modality (and its dual: $\mathsf{E}\varphi := \neg\mathsf{A}\neg\varphi$).

Example

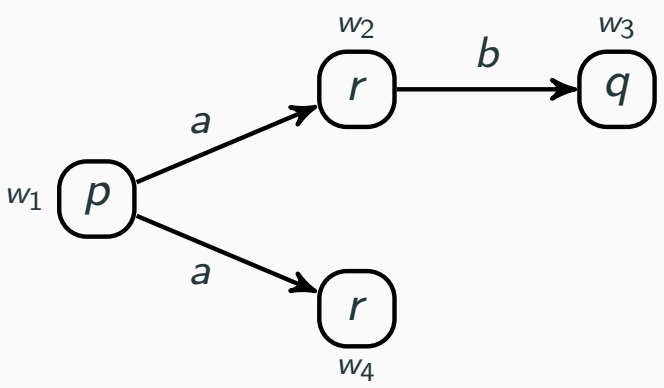


$a \sim_i ab$ ($\mathsf{U}_i = \{\{a, ab\}\}$)

$a \sim_j a,\ ab \sim_j ab$ ($\mathsf{U}_j = \{\{a\}, \{ab\}\}$)

$\mathcal{S}, w_1 \models \neg \mathsf{Kh}_i(p, r)$

Example



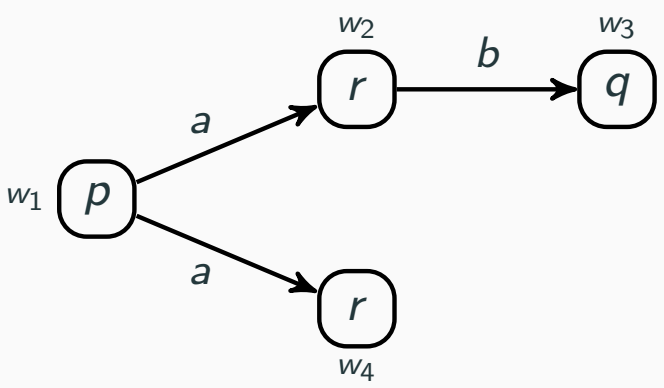
$a \sim_i ab$ ($\mathsf{U}_i = \{\{a, ab\}\}$)

$a \sim_j a,\ ab \sim_j ab$ ($\mathsf{U}_j = \{\{a\}, \{ab\}\}$)

$\mathcal{S}, w_1 \models \neg \mathsf{Kh}_i(p, r)$

take $\pi = \{a, ab\}$:

Example



$a \sim_i ab$ ($\mathsf{U}_i = \{\{a, ab\}\}$)

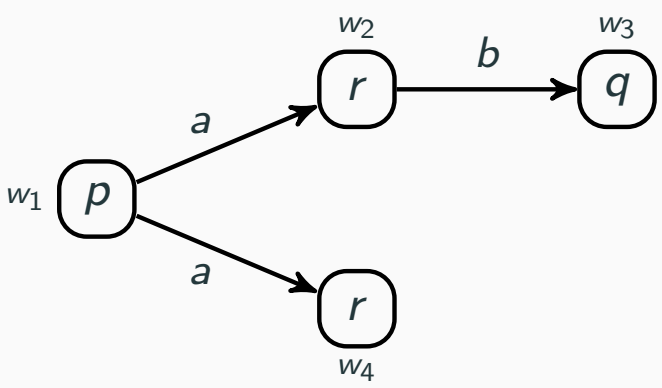
$a \sim_j a,\ ab \sim_j ab$ ($\mathsf{U}_j = \{\{a\}, \{ab\}\}$)

$\mathcal{S}, w_1 \models \neg \mathsf{Kh}_i(p, r)$

take $\pi = \{a, ab\}$:

- a is SE at w_1 (p-state), and takes from p-states to r-states.

Example



$a \sim_i ab$ ($\mathsf{U}_i = \{\{a, ab\}\}$)

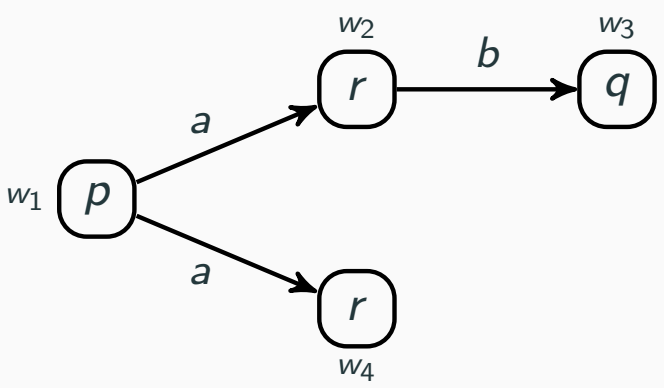
$a \sim_j a,\ ab \sim_j ab$ ($\mathsf{U}_j = \{\{a\}, \{ab\}\}$)

$\mathcal{S}, w_1 \models \neg \mathsf{Kh}_i(p, r)$

take $\pi = \{a, ab\}$:

- a is SE at w_1 (p-state), and takes from p-states to r-states.
- ab is not SE at w_1 (p-state).

Example



$a \sim_i ab$ ($\mathsf{U}_i = \{\{a, ab\}\}$)

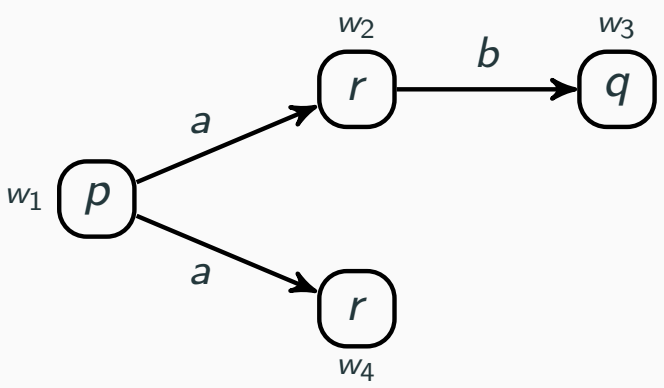
$a \sim_j a,\ ab \sim_j ab$ ($\mathsf{U}_j = \{\{a\}, \{ab\}\}$)

$\mathcal{S}, w_1 \models \neg \mathsf{Kh}_i(p, r)$

take $\pi = \{a, ab\}$:

- a is SE at w_1 (p-state), and takes from p-states to r-states.
- ab is not SE at w_1 (p-state).
- thus, $\pi = \{a, ab\}$ is not SE at w_1.

Example



$a \sim_i ab$ ($\mathsf{U}_i = \{\{a, ab\}\}$)

$a \sim_j a,\ ab \sim_j ab$ ($\mathsf{U}_j = \{\{a\}, \{ab\}\}$)

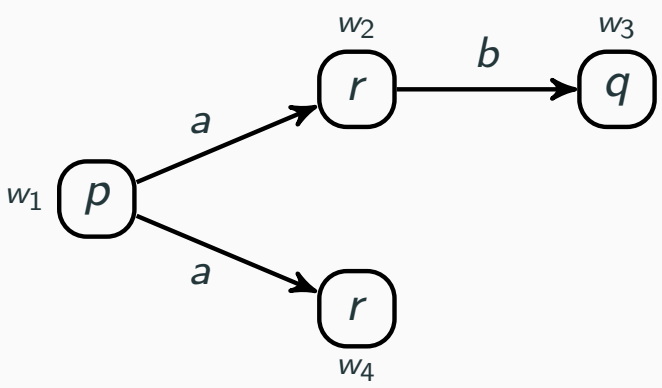
$\mathcal{S}, w_1 \models \neg \mathsf{Kh}_i(p, r)$

take $\pi = \{a, ab\}$:

- a is SE at w_1 (p-state), and takes from p-states to r-states.
- ab is not SE at w_1 (p-state).
- thus, $\pi = \{a, ab\}$ is not SE at w_1.

$\mathcal{S}, w_1 \models \mathsf{Kh}_j(p, r)$

Example



$a \sim_i ab$ ($\mathsf{U}_i = \{\{a, ab\}\}$)

$a \sim_j a,\ ab \sim_j ab$ ($\mathsf{U}_j = \{\{a\}, \{ab\}\}$)

$\mathcal{S}, w_1 \models \neg \mathsf{Kh}_i(p, r)$

take $\pi = \{a, ab\}$:

- a is SE at w_1 (p-state), and takes from p-states to r-states.
- ab is not SE at w_1 (p-state).
- thus, $\pi = \{a, ab\}$ is not SE at w_1.

$\mathcal{S}, w_1 \models \mathsf{Kh}_j(p, r)$

take $\pi' = \{a\}$:

- a is SE at w_1 (p-state), and takes from p-states to r-states.
- thus, $\pi' = \{a\}$ works as a witness.

Ontic Updates vs. Epistemic Updates

- Updating the LTS = update what an agent can do.
- Updating the relation $\sim_i$ (or the set U_i) = epistemic updates (affecting the "knowing how").

Ontic Updates vs. Epistemic Updates

- Updating the LTS = update what an agent can do.
- Updating the relation $\sim_i$ (or the set U_i) = epistemic updates (affecting the "knowing how").
- **Proposal:** refining the indistinguishability between plans, i.e., making plans distinguishable for the agent.

Epistemic updates: Refinement ($\mathsf{L}_{\mathsf{Ref}}$)

Definition ($\mathsf{L}_{\mathsf{Ref}}$ formulas)

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid \mathsf{Kh}_i(\varphi, \varphi) \mid \langle \sigma_1 \not\sim \sigma_2 \rangle \varphi$$

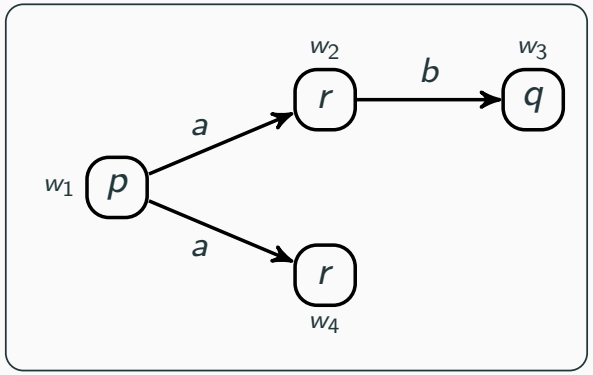
Epistemic updates: Refinement ($\mathsf{L}_{\mathsf{Ref}}$)

Definition ($\mathsf{L}_{\mathsf{Ref}}$ formulas)

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid \mathsf{Kh}_i(\varphi, \varphi) \mid \langle \sigma_1 \not\sim \sigma_2 \rangle \varphi$$

$\langle \sigma \not\sim \sigma_2 \rangle \varphi$: *"After it is stated that plans σ_1 and σ_2 are distinguishable, φ holds."*

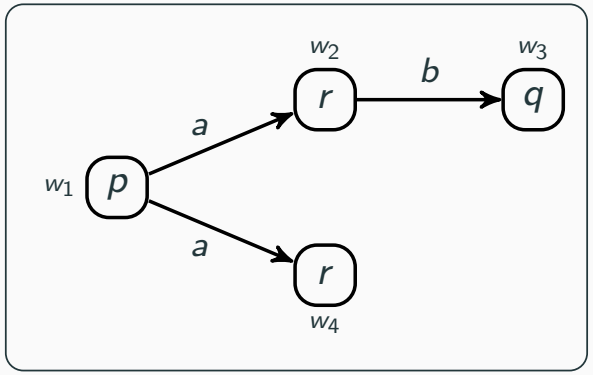
Epistemic updates: Refinement ($\mathsf{L}_{\mathsf{Ref}}$) (cont.)



$$\mathsf{U}_i = \{\{a\}, \{b\}\}, \qquad \mathsf{U}_j = \{\{a, b\}\}$$

- $\mathcal{S}, w \models \mathsf{Kh}_i(p, r)$ and $\mathcal{S}, w \models \langle a \not\sim b \rangle \mathsf{Kh}_i(p, r)$.

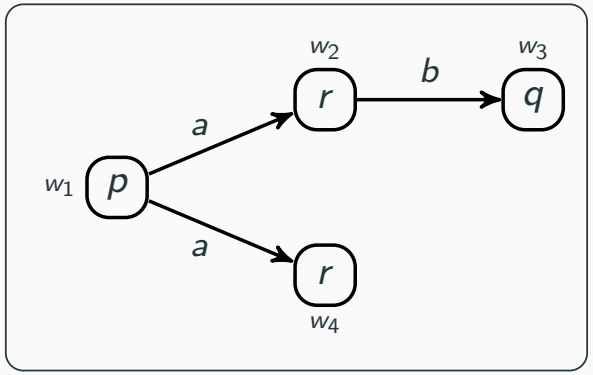
Epistemic updates: Refinement ($\mathsf{L}_{\mathsf{Ref}}$) (cont.)



$$\mathsf{U}_i = \{\{a\}, \{b\}\}, \qquad \mathsf{U}_j = \{\{a, b\}\}$$

- $\mathcal{S}, w \models \mathsf{Kh}_i(p, r)$ and $\mathcal{S}, w \models \langle a \not\sim b \rangle \mathsf{Kh}_i(p, r)$.
  - **preserves knowledge**

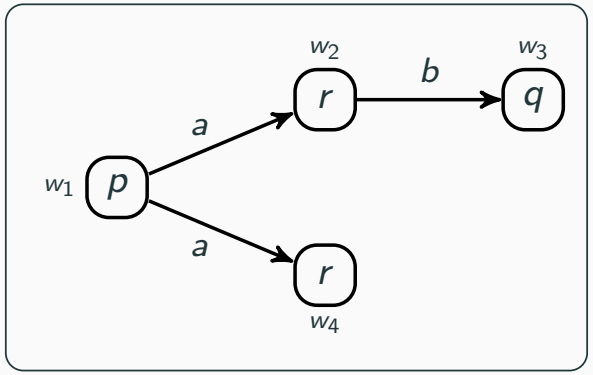
Epistemic updates: Refinement ($\mathsf{L}_{\mathsf{Ref}}$) (cont.)



$$\mathsf{U}_i = \{\{a\}, \{b\}\}, \qquad \mathsf{U}_j = \{\{a, b\}\}$$

- $\mathcal{S}, w \models \mathsf{Kh}_i(p, r)$ and $\mathcal{S}, w \models \langle a \not\sim b \rangle \mathsf{Kh}_i(p, r)$.
  - **preserves knowledge**
- $\mathcal{S}, w \not\models \mathsf{Kh}_j(p, r)$

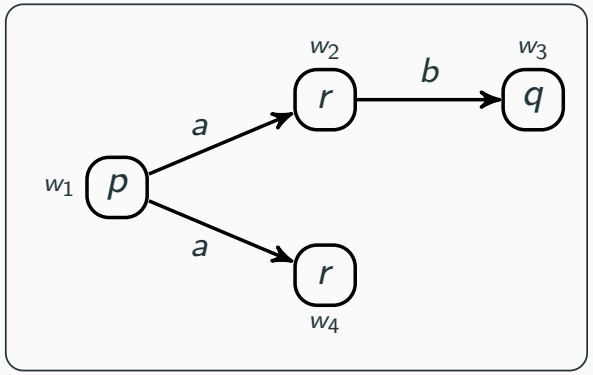
Epistemic updates: Refinement ($\mathsf{L}_{\mathsf{Ref}}$) (cont.)



$$\mathsf{U}_i = \{\{a\}, \{b\}\}, \qquad \mathsf{U}_j = \{\{a\}, \{b\}\}$$

- $\mathcal{S}, w \models \mathsf{Kh}_i(p, r)$ and $\mathcal{S}, w \models \langle a \not\sim b \rangle \mathsf{Kh}_i(p, r)$.
  - **preserves knowledge**
- $\mathcal{S}, w \not\models \mathsf{Kh}_j(p, r)$ but $\mathcal{S}, w \models \langle a \not\sim b \rangle \mathsf{Kh}_j(p, r)$;

Epistemic updates: Refinement ($\mathsf{L}_{\mathsf{Ref}}$) (cont.)



$$\mathsf{U}_i = \{\{a\}, \{b\}\}, \qquad \mathsf{U}_j = \{\{a\}, \{b\}\}$$

- $\mathcal{S}, w \models \mathsf{Kh}_i(p, r)$ and $\mathcal{S}, w \models \langle a \not\sim b \rangle \mathsf{Kh}_i(p, r)$.
  - **preserves knowledge**
- $\mathcal{S}, w \not\models \mathsf{Kh}_j(p, r)$ but $\mathcal{S}, w \models \langle a \not\sim b \rangle \mathsf{Kh}_j(p, r)$;
  - **generates new knowledge**

Expressivity

Property:

L_{Ref} is more expressive than L_{Kh_i}.

Expressivity

Property:

$\mathsf{L}_{\mathsf{Ref}}$ is more expressive than $\mathsf{L}_{\mathsf{Kh}_i}$.

Proof: Let $\mathcal{S}$ and $\mathcal{S}'$ be the LTSs below, with $\mathsf{U}_i := \{\{a\}\}$ and $\mathsf{U}'_i := \{\{a, b\}\}$:

$\mathcal{S}, w \models \neg\langle a \not\sim b\rangle \mathsf{Kh}_i(p, q)$ while $\mathcal{S}', w \models \langle a \not\sim b\rangle \mathsf{Kh}_i(p, q)$.

Uniform substitution

Uniform substitution is an standard property in axiomatizing the logic.

If φ is valid, then $\varphi[p/\psi]$ is also valid.

Uniform substitution

Uniform substitution is an standard property in axiomatizing the logic.

If φ is valid, then $\varphi[p/\psi]$ is also valid.

Property:

Uniform substitution fails in $\mathsf{L}_{\mathsf{Ref}}$.

Uniform substitution

Uniform substitution is an standard property in axiomatizing the logic.

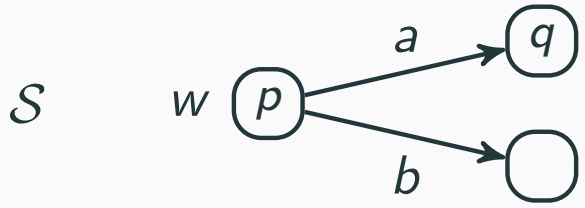
If φ is valid, then $\varphi[p/\psi]$ is also valid.

Property:

Uniform substitution fails in $\mathsf{L}_{\mathsf{Ref}}$.

Proof:

Take the formula $\varphi = p \to \langle a \not\sim b \rangle p$, and the $\mathsf{LTS}^{\mathsf{U}}$ $\mathcal{S}$ below, with $\mathsf{U}_i = \{\{a, b\}\}$:



Uniform substitution

Uniform substitution is an standard property in axiomatizing the logic.

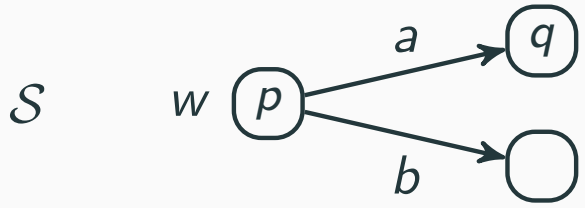
If φ is valid, then $\varphi[p/\psi]$ is also valid.

Property:

Uniform substitution fails in $\mathsf{L}_{\mathsf{Ref}}$.

Proof:

Take the formula $\varphi = p \to \langle a \not\sim b \rangle p$, and the LTS^{U} $\mathcal{S}$ below, with $\mathsf{U}_i = \{\{a, b\}\}$:



Replace p by $\neg\mathsf{Kh}_i(p, q)$: $\neg\mathsf{Kh}_i(p, q) \to \langle a \not\sim b \rangle \neg\mathsf{Kh}_i(p, q)$ is not valid.

Challenges

- This kind of updates increase the expressive power:
  - Failure of uniform substitution.
  - No reduction axioms.
- Quite challenging to obtain axiomatizations.

Challenges

- This kind of updates increase the expressive power:
  - Failure of uniform substitution.
  - No reduction axioms.
- Quite challenging to obtain axiomatizations.
- (A) solution: extend the expressivity of the underlying static language.

Proposal: A Knowing How Logic with Explicit Actions

Definition ($\mathsf{L}_{\mathsf{Kh}_i,\Box}$)

Formulas of the language $\mathsf{L}_{\mathsf{Kh}_i,\Box}$ are given by

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid \mathsf{Kh}_i(\varphi, \varphi) \mid [a]\varphi,$$

with $p \in \mathsf{Prop}$, $i \in \mathsf{Agt}$ and $a \in \mathsf{Act}$. Define: $\langle a \rangle \varphi := \neg[a]\neg\varphi$.

Definition

$\mathcal{S}, w \models [a]\varphi$ iff for all v s.t. $(w, v) \in \mathsf{R}_a$, $\mathcal{S}, v \models \varphi$.

Action Refinement

Definition ($\mathsf{L}_{\mathsf{Kh}_i,\Box,[!\_]}$)

$\mathcal{S}, w \models [!a]\varphi$ iff $\mathcal{S}^a, w \models \varphi$,

where $a \in \mathsf{Act}$ and $\mathcal{S}^a$ is as $\mathcal{S}$, except that for all $i \in \mathsf{Agt}$ we have:

- $\mathsf{U}_i^a = (\mathsf{U}_i \setminus \pi) \cup \{\{a\}\}$, if $a \in \pi$;
- $\mathsf{U}_i^a = \mathsf{U}_i \cup \{\{a\}\}$, otherwise.

Action Refinement

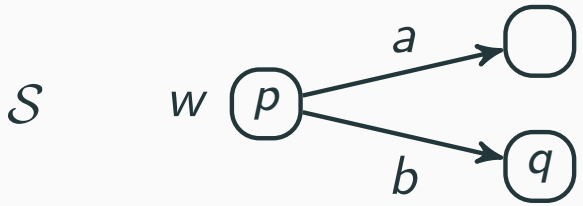
Definition ($\mathsf{L}_{\mathsf{Kh}_i,\Box,[!\_]}$)

$\mathcal{S}, w \models [!a]\varphi$ iff $\mathcal{S}^a, w \models \varphi$,

where $a \in \mathsf{Act}$ and $\mathcal{S}^a$ is as $\mathcal{S}$, except that for all $i \in \mathsf{Agt}$ we have:

- $\mathsf{U}_i^a = (\mathsf{U}_i \setminus \pi) \cup \{\{a\}\}$, if $a \in \pi$;
- $\mathsf{U}_i^a = \mathsf{U}_i \cup \{\{a\}\}$, otherwise.

Let $\mathcal{S}$ be such that $\mathsf{U}_i := \{\{a, b\}\}$.



$\mathcal{S}, w \not\models \mathsf{Kh}_i(p, q)$ and $\mathcal{S}, w \models [!b]\mathsf{Kh}_i(p, q)$.

Reduction Axioms

1. $[!a]p \leftrightarrow p$
2. $[!a]\neg\varphi_1 \leftrightarrow \neg[!a]\varphi_1$
3. $[!a](\varphi_1 \vee \varphi_2) \leftrightarrow ([!a]\varphi_1 \vee [!a]\varphi_2)$
4. $[!a][a]\varphi_1 \leftrightarrow [a][!a]\varphi_1$

Reduction Axioms

1. $[!a]p \leftrightarrow p$
2. $[!a]\neg\varphi_1 \leftrightarrow \neg[!a]\varphi_1$
3. $[!a](\varphi_1 \vee \varphi_2) \leftrightarrow ([!a]\varphi_1 \vee [!a]\varphi_2)$
4. $[!a][a]\varphi_1 \leftrightarrow [a][!a]\varphi_1$
5. $[!a]\mathrm{Kh}_i(\varphi_1, \varphi_2) \leftrightarrow (\mathrm{Kh}_i([!a]\varphi_1, [!a]\varphi_2) \vee$
$\mathsf{A}([!a]\varphi_1 \rightarrow (\langle a\rangle\top \wedge [a][!a]\varphi_2)))$

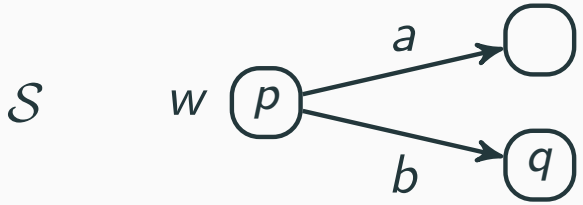
Reduction Axioms

1. $[!a]p \leftrightarrow p$
2. $[!a]\neg\varphi_1 \leftrightarrow \neg[!a]\varphi_1$
3. $[!a](\varphi_1 \vee \varphi_2) \leftrightarrow ([!a]\varphi_1 \vee [!a]\varphi_2)$
4. $[!a][a]\varphi_1 \leftrightarrow [a][!a]\varphi_1$
5. $[!a]\mathrm{Kh}_i(\varphi_1, \varphi_2) \leftrightarrow (\mathrm{Kh}_i([!a]\varphi_1, [!a]\varphi_2) \vee$
$\mathsf{A}([!a]\varphi_1 \rightarrow (\langle a\rangle\top \wedge [a][!a]\varphi_2)))$

Via reduction axioms, we can eliminate all the occurrences of a [!a] modality (i.e., embed $\mathrm{L}_{\mathrm{Kh}_i,\Box,[!\_]}$ into $\mathrm{L}_{\mathrm{Kh}_i,\Box}$).

Example

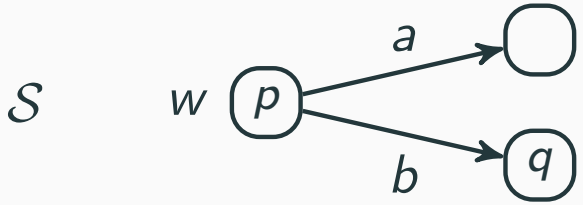
Let $\mathcal{S}$ be s.t. $\mathrm{U}_i := \{\{a, b\}\}$, $\mathcal{S}, w \not\models \mathrm{Kh}_i(p, q)$ and $\mathcal{S}, w \models [!b]\mathrm{Kh}_i(p, q)$.



$$\mathcal{S}, w \models [!b]\mathrm{Kh}_i(p, q) \tag{5}$$

Example

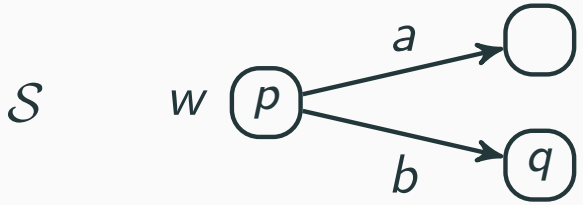
Let $\mathcal{S}$ be s.t. $\mathsf{U}_i := \{\{a, b\}\}$, $\mathcal{S}, w \not\models \mathsf{Kh}_i(p, q)$ and $\mathcal{S}, w \models [!b]\mathsf{Kh}_i(p, q)$.



$$\begin{aligned} & \mathcal{S}, w \models [!b]\mathsf{Kh}_i(p, q) && \textbf{(5)} \\ \text{iff} \quad & \mathcal{S}, w \models \mathsf{Kh}_i([!b]p, [!b]q) \vee \mathsf{A}([!b]p \to (\langle b \rangle \top \wedge [b][!b]q)) && \textbf{(1)} \end{aligned}$$

Example

Let $\mathcal{S}$ be s.t. $\mathsf{U}_i := \{\{a, b\}\}$, $\mathcal{S}, w \not\models \mathsf{Kh}_i(p, q)$ and $\mathcal{S}, w \models [!b]\mathsf{Kh}_i(p, q)$.



$$
\begin{array}{lll}
 & \mathcal{S}, w \models [!b]\mathsf{Kh}_i(p, q) & \textbf{(5)} \\
\text{iff} & \mathcal{S}, w \models \mathsf{Kh}_i([!b]p, [!b]q) \vee \mathsf{A}([!b]p \to (\langle b \rangle \top \wedge [b][!b]q)) & \textbf{(1)} \\
\text{iff} & \mathcal{S}, w \models \mathsf{Kh}_i(p, q) \vee \mathsf{A}(p \to (\langle b \rangle \top \wedge [b]q)) &
\end{array}
$$

Axiomatization

Axioms	Taut	$\vdash \varphi$ for φ a propositional tautology
	DistA	$\vdash \mathsf{A}(\varphi \to \psi) \to (\mathsf{A}\varphi \to \mathsf{A}\psi)$
	TA	$\vdash \mathsf{A}\varphi \to \varphi$
	Dist$\Box$	$\vdash [a](\varphi \to \psi) \to ([a]\varphi \to [a]\psi)$
	A$\Box$	$\vdash \mathsf{A}\varphi \to [a]\varphi$
	4KhA	$\vdash \mathsf{Kh}_i(\psi, \varphi) \to \mathsf{AKh}_i(\psi, \varphi)$
	5KhA	$\vdash \neg\mathsf{Kh}_i(\psi, \varphi) \to \mathsf{A}\neg\mathsf{Kh}_i(\psi, \varphi)$
	KhE	$\vdash (\mathsf{E}\psi \wedge \mathsf{Kh}_i(\psi, \varphi)) \to \mathsf{E}\varphi$
	KhA	$\vdash (\mathsf{A}(\chi \to \psi) \wedge \mathsf{Kh}_i(\psi, \varphi) \wedge \mathsf{A}(\varphi \to \theta)) \to \mathsf{Kh}_i(\chi, \theta)$
Rules	MP	From $\vdash \varphi$ and $\vdash \varphi \to \psi$ infer $\vdash \psi$
	NecA	From $\vdash \varphi$ infer $\vdash \mathsf{A}\varphi$

Theorem

1. *The axiom system above is sound and complete for* $\mathsf{L}_{\mathsf{Kh}_i,\Box}$.

Axiomatization

Axioms	Taut	$\vdash \varphi$ for φ a propositional tautology
	DistA	$\vdash \mathsf{A}(\varphi \to \psi) \to (\mathsf{A}\varphi \to \mathsf{A}\psi)$
	TA	$\vdash \mathsf{A}\varphi \to \varphi$
	Dist$\Box$	$\vdash [a](\varphi \to \psi) \to ([a]\varphi \to [a]\psi)$
	A$\Box$	$\vdash \mathsf{A}\varphi \to [a]\varphi$
	4KhA	$\vdash \mathsf{Kh}_i(\psi, \varphi) \to \mathsf{AKh}_i(\psi, \varphi)$
	5KhA	$\vdash \neg\mathsf{Kh}_i(\psi, \varphi) \to \mathsf{A}\neg\mathsf{Kh}_i(\psi, \varphi)$
	KhE	$\vdash (\mathsf{E}\psi \wedge \mathsf{Kh}_i(\psi, \varphi)) \to \mathsf{E}\varphi$
	KhA	$\vdash (\mathsf{A}(\chi \to \psi) \wedge \mathsf{Kh}_i(\psi, \varphi) \wedge \mathsf{A}(\varphi \to \theta)) \to \mathsf{Kh}_i(\chi, \theta)$
Rules	MP	From $\vdash \varphi$ and $\vdash \varphi \to \psi$ infer $\vdash \psi$
	NecA	From $\vdash \varphi$ infer $\vdash \mathsf{A}\varphi$

Theorem

1. *The axiom system above is sound and complete for* $\mathsf{L}_{\mathsf{Kh}_i,\Box}$.
2. *The axiom system above + the reduction axioms is sound and complete for* $\mathsf{L}_{\mathsf{Kh}_i,\Box,[!\_]}$.

Final remarks

- Preliminary results have been included in (Areces et al.; DALI 2022).
- We extended the expressive power of uncertainty-based knowing how.
- We defined a dynamic modality to update the knowledge, and obtained reduction axioms.
- This idea can be adapted to arbitrary plans, not only actions (i.e., $[!\sigma]\varphi$, with $\sigma \in \mathsf{Act}^*$).
- Reduction axioms for the general case.
- Future work: complexity of the logic, other updates.