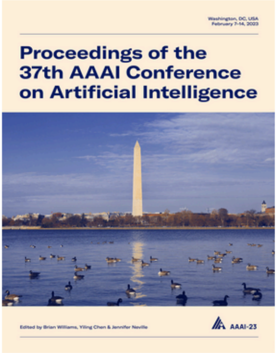
Model-Checking for Ability-Based Logics with Constrained Plans

Stéphane Demri
CNRS, LMF, ENS Paris-Saclay, France

Raul Fervari
Universidad Nacional de Córdoba and CONICET, Argentina GTIIT, China

DOI: https://doi.org/10.1609/aaai.v37i5.25776

Keywords: KRR: Computational Complexity of Reasoning, KRR: Action, Change, and Causality, KRR: Knowledge Representation Languages, KRR: Other Foundations of Knowledge Representation & Reasoning, KRR: Reasoning with Beliefs



Reasoning about Knowledge

- Epistemic logic usually consists of a modal language which can express "an agent knows that φ".   [Hintikka, Book 1962]

- During decades, research on epistemic logics focuses on knowledge expressed by "knowing that".

- "knowing that" is the most common knowledge notion studied in epistemic logics, but many others are interesting such as
  - "knowing how",
  - "knowing why",
  - "knowing what",

  etc.

Knowing How

- Knowing how: epistemic notion related to the abilities of an agent to achieve a goal.

- Examples of goal-directed "knowing how":
  - to know how to prove a theorem,
  - to know how to open the door,
  - to know how to bake a cake, etc.

- In the sequel, "knowing how" encoded by a single modality interpreted as an ability to achieve a goal, e.g. Kh φ.

A Quick Tour on Epistemic Logics of Knowing How

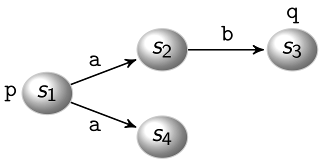
- Introduction of a framework for knowing how logics.
  [Wang, LORI'15; Wang, Synthese 2018]
  - Logic with modality $\mathsf{Kh}(\varphi, \psi)$ (axiomatisation).
  - Logic with modality $\mathsf{Kh}(\varphi, \chi, \psi)$.
  - Extension with announcement-like modality.

- Distinction between factual and epistemic information.
  - Generalisation of Wang's framework by re-introducing the notion of epistemic indistinguishability ($\sim_{\mathfrak{a}}$ on plans).
    [Areces et al, TARK'21]
  - Dynamic epistemic approach of knowing how (*à la* PAL).
    [Areces et al, DaLi'22]

- Plan constraints in [Li, PhD 2017].

Wang's Logic $\mathscr{L}_{kh}$ [Wang, LORI'15]

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



$\mathscr{S} = (S, (R_a)_{a \in Act}, V)$

- Non-empty set of states S.
- Collection of binary relations $(R_a)_{a \in Act}$.
- Labelling function $V : S \to 2^{Prop}$.

- $\mathsf{Kh}(p, q)$: "the agent knows how to achieve the goal q, whenever the initial condition p holds".

- Plan-based "knowing how" (a.k.a. ability-based).

Plans

- In this talk, a **plan** σ is a finite sequence in Act^*.

- $\sigma \in \mathsf{Act}^*$ is **strongly executable** (SE) at $s \in \mathsf{S}$ iff for all $k \in [0, |\sigma| - 1]$ and $t \in \mathsf{R}_{\sigma[0\ldots k]}(s)$, we have $\mathsf{R}_{\sigma(k+1)}(t) \neq \emptyset$.

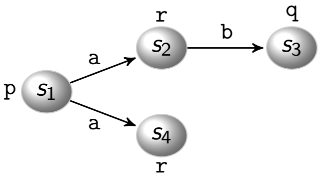
  "Each partial execution of the plan must be completed"

- $\mathsf{SE}(\sigma) \stackrel{\text{def}}{=} \{s \in \mathsf{S} \mid \sigma \text{ is SE at } s\}$.

Wang's Logic: Satisfaction Relation

- $\mathscr{S}, s \Vdash \mathtt{p} \overset{\text{def}}{\Leftrightarrow} \mathtt{p} \in \mathsf{V}(s)$.
- $\mathscr{S}, s \Vdash \mathsf{Kh}(\varphi, \psi) \overset{\text{def}}{\Leftrightarrow}$ there exists $\sigma \in \mathsf{Act}^*$ such that
  (1) σ is SE at every state satisfying φ,

  (2) from every φ-state, executing σ always ends at ψ-states.



$\mathscr{S}, s_1 \Vdash \mathsf{Kh}(\mathtt{p}, \mathtt{r})$
- the plan $\mathtt{a}$ is SE s_1 (the only $\mathtt{p}$-state), and takes the agent from $\mathtt{p}$ to $\mathtt{r}$-states.

$\mathscr{S}, s_1 \nVdash \mathsf{Kh}(\mathtt{p}, \mathtt{q})$
- ε and $\mathtt{a}$: are SE at s_1 ($\mathtt{p}$-state), but do not lead to $\mathtt{q}$;
- $\mathtt{ab}$ is **not** SE at s_1.

Decidability/Complexity Status

- Satisfiability problem is decidable. [Li, PhD 2017]
  (use of model-checking from a small model)

- Hilbert-style axiomatisation. [Wang, Synthese 2018]

- Complexity of satisfiability and model-checking not studied.

Our Motivations

- Knowing how with regularity constraints on plans.
- Knowing how with budget constraints on plans.
  (constrained plans advocated in [Li, PhD 2017])
- Model-checking (instead of satisfiability/validity).
- Connections with
  - formal language theory,
  - vector addition systems with states (VASS).

Model-Checking for Wang's Logic

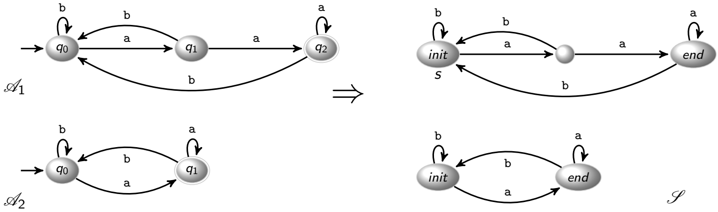
- Model-checking problem for $\mathscr{L}_{kh}$ is PSPACE-complete.
  [Demri & Fervari, AAAI'23]

- PSPACE-hardness from nonemptiness for finite-state automata [Galil, Acta Informatica 1976; Kozen, FOCS'77].

- PSPACE-membership relies on a small plan property.

PSpace-hardness (easy)

Complete finite-state automata $\mathscr{A}_1$, $\mathscr{A}_2$.



$\mathrm{L}(\mathscr{A}_1) \cap \mathrm{L}(\mathscr{A}_2) \neq \emptyset$ if and only if $\mathscr{S}, s \Vdash \mathrm{Kh}(\mathit{init}, \mathit{end})$.

PSpace-Membership

- The set of plans witnessing $\mathscr{S}, s \Vdash \mathsf{Kh}(\varphi, \psi)$ is a regular language.
  - The set of plans σ violating $[\![\varphi]\!]^{\mathscr{S}} \subseteq \mathsf{SE}(\sigma)$ is regular.
  - The set of plans σ violating $\mathsf{R}_\sigma([\![\varphi]\!]^{\mathscr{S}}) \subseteq [\![\psi]\!]^{\mathscr{S}}$ is regular.

- $\mathscr{S}, s \Vdash \mathsf{Kh}(\varphi, \psi)$ iff there is a plan σ of exponential size, witnessing the truth of $\mathsf{Kh}(\varphi, \psi)$.

- PSPACE obtained by a standard labelling algorithm. The case with Kh-formulae requires to build a plan on-the-fly.

Uncertainty-Based Knowing How Logics

[Areces et al., TARK'21]

- To model the situations in which an agent is not aware of the existence of specific plans, is not able to distinguish certain plans from others, does not care about the differences among certain plans, etc.

- Knowing how depends on abilities but also on indistinguishability (uncertainty) between plans.

- Finite set of agents Agt and $\mathsf{Kh}_{\mathfrak{a}}$-formulae with $\mathfrak{a} \in \mathsf{Agt}$.

Uncertainty-Based Knowing How Logics (II)

[Areces et al., TARK'21]

- Models $\mathscr{S} = (\mathsf{S}, (\mathsf{R_a})_{\mathsf{a} \in \mathsf{Act}}, (\sim_{\mathfrak{a}})_{\mathfrak{a} \in \mathsf{Agt}}, \mathsf{V})$ with equivalence relation $\sim_{\mathfrak{a}}$ on $\Pi_{\mathfrak{a}} \subseteq \mathsf{Act}^*$.

  $\sigma \sim_{\mathfrak{a}} \sigma'$ iff σ and σ' cannot be distinguished by the agent $\mathfrak{a}$

- $\mathscr{S}, s \Vdash \mathsf{Kh}_{\mathfrak{a}}(\varphi, \psi) \overset{\text{def}}{\Leftrightarrow}$ there exists an equivalence class Π of $\sim_{\mathfrak{a}}$ such that
  (1) each plan σ in Π is SE at every φ-state, ($[\![\varphi]\!]^{\mathscr{S}} \subseteq \mathsf{SE}(\Pi)$)

  (2) from φ-states, each plan in Π always ends at ψ-states. ($\mathsf{R}_{\Pi}([\![\varphi]\!]^{\mathscr{S}}) \subseteq [\![\psi]\!]^{\mathscr{S}}$)

Model-Checking Problem for $\mathscr{L}_{reg}^{\mathrm{U}}$

- In [Areces et al., TARK'21], each $\Pi_{\mathfrak{a}}$ is finite.
- Herein, each indistinguishability relation $\sim_{\mathfrak{a}}$ is encoded by a finite set of FSA (with pairwise empty intersection).

$$\sim_{\mathfrak{a}} = \{\mathscr{A}_1, \ldots, \mathscr{A}_K\}.$$

- The model-checking problem for $\mathscr{L}_{reg}^{\mathrm{U}}$ is in PTime.

Adding a Budget

- Budget constraints on plans advocated in [Li & Wang, ICLA'17].

- Actions have costs and execution of plans require that the agent stays within the budget.

- In models, introduction of some weight function $wf : \mathsf{S} \times \mathsf{Act} \to \mathbb{Z}^r$ for some $r \geq 0$.

- Adding resource reasoning is a well-known paradigm, see e.g.
  - in energy games, [Chatterjee, Doyen, Henzinger, 2017]
  - in multi-agent systems. [Cao & Naumov, IJCAI'17]
  - in ATL-like logics, [Alechina et al., AI 2017]

Wang's Logic with Budget $\mathscr{L}_{kh}(\star)$

- $\mathsf{Kh}^{\vec{b}}(\varphi,\psi)$: Knowing how to make ψ true, given φ, with budget $\vec{b} \in \mathbb{N}^r$.

- Models of $\mathscr{L}_{kh}(r)$ of the form $\mathscr{S} = (\mathsf{S}, (\mathsf{R_a})_{\mathsf{a}\in\mathsf{Act}}, \mathit{wf}, \mathsf{V})$ with weight function $\mathit{wf} : \mathsf{S} \times \mathsf{Act} \rightarrow \mathbb{Z}^r$.

- $\mathscr{S}, s \Vdash \mathsf{Kh}^{\vec{b}}(\varphi,\psi) \overset{\text{def}}{\Leftrightarrow}$ there is a plan $\sigma \in \mathsf{Act}^*$ such that (1) $[\![\varphi]\!]^{\mathscr{S}} \subseteq \mathsf{SE}(\sigma)$; (2) $\mathsf{R}_\sigma([\![\varphi]\!]^{\mathscr{S}}) \subseteq [\![\psi]\!]^{\mathscr{S}}$, and

  (3) σ is $\vec{b}$-compatible at $[\![\varphi]\!]^{\mathscr{S}}$.
  "No negative value reached executing σ from φ-states with initial budget $\vec{b}$"

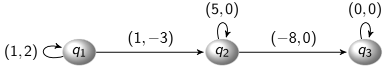
Complexity Inherited from Problems on VASS

- MC($\mathscr{L}_{kh}(\star)$) is $\textsc{ExpSpace}$-hard.
  (reduction from control-state reachability problem for VASS)

- MC($\mathscr{L}_{kh}(\star)$) restricted to LTS with action costs independent of states is $\textsc{ExpSpace}$-complete.
  (reduction to control-state reachability problem for VASS)

MC($\mathscr{L}_{kh}(\star)$) is ExpSpace-hard

- Reduction from the control-state reachability problem for VASS known to be ExpSpace-complete.



- VASS $\mathscr{V} = (Q, r, R)$, $(q_0, \vec{x}_0)$ and q_f.

- $\mathscr{S} = (\mathsf{S}, (\mathsf{R}_\mathtt{a})_{\mathtt{a} \in \mathsf{Act}}, \mathit{wf}, \mathsf{V})$:

$$T = q \xrightarrow{\vec{u}} q' \text{ in } \mathscr{V} \quad \Rightarrow \quad q \xrightarrow{T} q' \quad \text{with } \mathit{wf}(T) = \vec{u} \text{ in } \mathscr{S}$$

  - $\mathsf{S} \stackrel{\text{def}}{=} Q$ and $\mathsf{Act} \stackrel{\text{def}}{=} R$ (set of transitions).
  - For all $T = q \xrightarrow{\vec{u}} q' \in R$, for all $s, s' \in \mathsf{S}$,
    1. $(s, s') \in \mathsf{R}_T \stackrel{\text{def}}{\Leftrightarrow} s = q$ and $s' = q'$,
    2. $\mathit{wf}(s, T) \stackrel{\text{def}}{=} \vec{u}$ if $s = q$.
  - For all $s \in \mathsf{S}$, $\mathtt{init} \in \mathsf{V}(s) \stackrel{\text{def}}{\Leftrightarrow} s = q_0$, and $\mathtt{end} \in \mathsf{V}(s) \stackrel{\text{def}}{\Leftrightarrow} s = q_f$.

MC($\mathscr{L}_{kh}(\star)$) is ExpSpace-hard (II)

- Each action in Act occurs exactly once in $\mathscr{S}$ and therefore wf can be defined so that $\mathit{wf}(s, T)$ does not depend on s.
- Each relation is deterministic.
- $\mathscr{S}, q_0 \Vdash \mathsf{Kh}^{\vec{x}_0}(\texttt{init}, \texttt{end})$ iff there is a finite run from $(q_0, \vec{x}_0)$ to a configuration with state q_f in $\mathscr{V}$.

Concluding Remarks

- Summary
  - Introduction of regularity and budget constraints in logics with knowing how.
  - Complexity results for the model-checking problem, in particular with budgets.
- Future work.
  - More general plans with an operational semantics, see [Li & Wang, AI 2021].
  - How to add constraints to the alternative approach from [Fervari et al., IJCAI'17]?