ASAI Networking Event 2023

Semantic Systems Group

Artificial Intelligence und Cybersecurity
University of Klagenfurt

Wolfgang Faber, Michael Morak

University of Klagenfurt, Austria

Research Area

Overview

ASP is a declarative, rule-based logic programming language. Each answer set represents a solution of the problem.

Answer Set Programming lacks meta-reasoning features: epistemic operators to the rescue. Allow reasoning over *all* answer sets.

Question: What can we use epistemic logic programs for?

Background

(Ground) ASP programs are sets of rules (i.e. implications):

$$a_1 \vee \cdots \vee a_l \leftarrow a_{l+1}, \ldots, a_m, \neg a_{m+1}, \ldots, \neg a_n.$$

Interpretation? *Stable Model Semantics*
Notation: $AS(\Pi)$ denote the set of answer sets of program Π

(Ground) ELPs are sets of rules of the form:

$$a_1 \vee \cdots \vee a_k \leftarrow \ell_1, \ldots, \ell_m, \xi_1, \ldots, \xi_j, \neg\xi_{j+1}, \ldots, \neg\xi_n.$$

ξ_i: an *epistemic literal*, i.e. $\mathbf{not}\,\ell$ for some literal ℓ;

Semantics for ELPs

Several contenders exist, but the general principle is similar. Here is one approach (by Shen & Eiter, 2016):

For an ELP Π, an *epistemic guess* is a set $\Phi \subseteq elit(\Pi)$ of epistemic literals. The epistemic reduct Π^Φ w.r.t. a guess Φ is obtained from Π by replacing all occurrences of epistemic literals ξ in the rules of Π with the following:

- $\top$ if $\xi \in \Phi$, or
- $\neg\ell$ if $\xi \notin \Phi$ and $\xi = \mathbf{not}\,\ell$.

Definition (Candidate World View)

Let Φ be a guess. The set $\mathcal{M} = AS(\Pi^\Phi)$ is a candidate world view (CWV) of the epistemic logic program Π iff

1. $\mathcal{M} \neq \emptyset$,
2. for each epistemic literal $\mathbf{not}\,\ell \in \Phi$, there exists an answer set $M \in \mathcal{M}$ wherein ℓ is false, and
3. for each epistemic literal $\mathbf{not}\,\ell \in elit(\Pi) \setminus \Phi$, it holds that ℓ is true in each answer set $M \in \mathcal{M}$.

Main decision problem: does a candidate world view exist?
Complexity: Σ_3^P-complete in general

Projects

- Reasoning about actions and change
- Ontological reasoning
- Meta-modelling in ontological modelling languages
- ...

Some Topics of Interest

Topic: Strong Equivalence for ELPs

- Two ELPs are *equivalent* iff their CWVs are equivalent.
- Two ELPs Π_1 and Π_2 are *strongly equivalent* iff, for any third program Π, $\Pi \cup \Pi_1$ and $\Pi \cup \Pi_2$ are equivalent.

Questions:

1. What is the complexity of deciding strong equivalence?
2. How can strong equivalence be used?
3. What about *uniform equivalence*?

Topic: Reversibility in Planning via ELPs

- Reversibility of action = a way to "undo" the action's effects
- (Uniform) Reversibility of an action with a polynomial-length reverse plan can be decided in Σ_2^P.

Questions:

1. Can we come up with an intuitive encoding of reversibility via ELPs?
2. How does this encoding behave in practice?

Topic: Solving ELPs via ASP(Q)

- ASP(Q) enhances ASP with quantifiers: e.g. $\exists P_1 \forall P_2 \exists \ldots : C$, with the intuitive meaning that there exists an answer set for P_1, such that for all answer sets of P_2 there exists, ..., such that constraints C are satisfied.
- Deciding ASP(Q) programs is PSPACE-complete in general.

Questions:

1. Is there a way to intuitively rewrite ELPs into ASP(Q)?
2. Compared to ELP solvers, how does such a rewriting-based solving approach perform in practice?

Event: Annual EELP Workshop

- Since 2019, we have been organizing a workshop on epistemic extensions of logic programming (EELP).
- Last took place last year as part of FLoC in Haifa, Israel.

Information:

1. `https://www.semsys.aau.at/events/eelp2022/`

Some Other Work

- Parameterized complexity of solving ELPs
- Reversibility of actions in planning
- Ontological reasoning and metamodelling
- Extensions of ASP with probability distributions
- ...

Contact: wolfgang.faber@aau.at