COMP4418, 2017 – Exercise

Weeks 6, 7, 8, 9

---

1 Answer Set Programming

1.1 Modelling

A *set cover* of a set S of sets $s_1, \ldots, s_n$ is a set of sets $C \subseteq S$ such that $\bigcup_{s \in S} s = \bigcup_{s \in C} s$. A *k-set cover* is a set cover of size k, that is, $|C| = k$.

For instance, for an input $S = \{\{1,2\},\{2,3\},\{4,5\},\{1,2,3\}\}$, there is a 2-set cover $C = \{\{1,2,3\},\{4,5\}\}$ since $\bigcup_{s \in S} s = \{1,2\} \cup \{2,3\} \cup \{4,5\} \cup \{1,2,3\} = \{1,2,3\} \cup \{4,5\} = \bigcup_{s \in C} s$.

Write an ASP program that decides the k-Set-Cover problem:

> Input: a set of sets and a natural number $k \geq 0$.
>
> Problem: decide if there is a k-set cover.

Assume the input parameter $S = \{s_1, \ldots, s_n\}$ is encoded by a binary predicate `s` in the way that $x \in s_i$ iff `s(`i`,`x`)`. The input parameter k is given as constant symbol `k`. Use a unary predicate `c` to represent the output C in the way that $s_i \in C$ iff `c(`i`)`.

1.2 Semantics

Consider the following program P.

$$
\begin{aligned}
&a. \\
&c \;\texttt{:-}\; \text{not}\, b, \text{not}\, d. \\
&d \;\texttt{:-}\; a, \text{not}\, c.
\end{aligned}
$$

Determine the stable models of S.

2 Reasoning about Knowledge

2.1 Cardinality of different sets related to $\mathcal{OL}$

(This question is not relevant for the exam, but a good exercise to think a bit about the logic.)

Is the...

- set of formulas of $\mathcal{OL}_{\text{PL}}$
- set of worlds of $\mathcal{OL}_{\text{PL}}$
- set of epistemic states $\mathcal{OL}_{\text{PL}}$

- set of formulas of $\mathcal{OL}$
- set of worlds of $\mathcal{OL}$
- set of epistemic states $\mathcal{OL}$

...finite, countably infinite, or uncountable?

2.2 Introspection

Prove the following results from Slide 26:

- $\models \exists x\, \mathbf{K}\, \alpha \rightarrow \mathbf{K}\, \exists x\, \alpha$.
- $\not\models \mathbf{K}\, \exists x\, \alpha \rightarrow \exists x\, \mathbf{K}\, \alpha$.

2.3 Only-Knowing

Suppose all you know is

- the father of Sally is Frank or Fred, and
- Sally's father is rich.

Formalise this statement in $\mathcal{OL}$. Show that this statement does entail that Frank or Fred is known to be rich, but it is not known who of them is rich.

2.4 Representation Theorem

Suppose you have a wedding database that tells you who is married to whom.[1]

$$\begin{aligned}&\text{Married}(\text{Mia}, \text{Frank}) \land \\ &\exists x\, \text{Married}(x, \text{Fred}) \land \\ &\text{Married}(\text{motherOf}(\text{Sally}), \text{fatherOf}(\text{Sally}))\end{aligned}$$

where Frank, Fred, Mia, Sally are standard names. Call this sentence KB.

(a) Who is not known to be married to Sally?

1. What is the set of tuples of standard names N such that $n \in N$ iff $\mathbf{O}\text{KB} \models \neg\mathbf{K}\, \text{Married}(\text{Sally}, n)$?
2. Determine $\mathsf{RES}[\text{KB}, \text{Married}(\text{Sally}, x)]$.
3. Determine whether $\mathbf{O}\text{KB} \models \exists x\, \neg\mathbf{K}\, \text{Married}(\text{Sally}, x)$ using the representation theorem (Slide 31), that is, by checking whether $\models \|\exists x\, \neg\mathbf{K}\, \text{Married}(\text{Sally}, x)\|_{\text{KB}}$.

[1] In a realistic scenario, we would add $\forall x \forall y\, (\text{Married}(x, y) \leftrightarrow \text{Married}(y, x))$ to formalise that marriage is a symmetric relation. For the sake of this example, we do not add this symmetry constraint to our knowledge.

(b) Who is known to be married?

1. What is the set of standard names N such that $\mathbf{O}\mathrm{KB} \models \mathbf{K}\,\exists y\,(\mathrm{Married}(n,y) \vee \mathrm{Married}(y,n))$?
2. Determine $\mathsf{RES}[\mathrm{KB}, \exists y\,(\mathrm{Married}(x,y) \vee \mathrm{Married}(y,x))]$. (Note: there is one free variable, x.)
3. Determine whether $\mathbf{O}\mathrm{KB} \models \exists x\,\mathbf{K}\,\exists y\,(\mathrm{Married}(x,y) \vee \mathrm{Married}(y,x))$ using the representation theorem, that is, by checking whether $\models \|\exists x\,\mathbf{K}\,\exists y\,(\mathrm{Married}(x,y) \vee \mathrm{Married}(y,x))\|_{\mathrm{KB}}$.

(c) Who is known to be married to an unknown person?

1. What is the set of tuples of standard names N such that $(n_1, n_2) \in N$ iff $\mathbf{O}\mathrm{KB} \models \mathbf{K}\,\mathrm{Married}(n_1, n_2)$?
2. What is the set of standard names N such that $n \in N$ iff $\mathbf{O}\mathrm{KB} \models \mathbf{K}\,\exists x\,(\mathrm{Married}(x,n) \wedge \neg\mathbf{K}\,\mathrm{Married}(x,n))$?
3. Determine $\mathsf{RES}[\mathrm{KB}, \mathrm{Married}(x,y)]$. (Note: there are two free variables, x and y.)
4. Determine $\mathsf{RES}[\mathrm{KB}, \exists x\,(\mathrm{Married}(x,y) \wedge \neg(x = \mathrm{Mia} \wedge y = \mathrm{Frank}))]$. (Note: there is one free variable, y.)
5. Determine whether $\mathbf{O}\mathrm{KB} \models \exists y\,\mathbf{K}\,\exists x\,(\mathrm{Married}(x,y) \wedge \neg\mathbf{K}\,\mathrm{Married}(x,y))$ using the representation theorem, that is, by checking whether $\models \|\exists y\,\mathbf{K}\,\exists x\,(\mathrm{Married}(x,y) \wedge \neg\mathbf{K}\,\mathrm{Married}(x,y))\|_{\mathrm{KB}}$.

3 Limited Reasoning

3.1 Unit Propagation and Subsumption

Determine $\mathrm{UP}(s)$, $\mathsf{UP}^+(s)$, $\mathsf{UP}^-(s)$, whether s is obviously inconsistent, and whether s is obviously consistent, for…

1. $s = \{\}$
2. $s = \{p, \neg p\}$
3. $s = \{(p \vee q), (\neg q \vee \neg r), r\}$
4. $s = \{(p \vee q), (p \vee \neg q), (\neg p \vee q), (\neg p \vee \neg q)\}$

3.2 Minimal Belief Level

1. Let $s = \{\}$. Find the minimal k such that $s \mathrel{|\!\approx} \mathbf{K}_k\,(p \vee \neg p)$.
2. Let $s = \{p, \neg p\}$. Find the minimal k such that $s \mathrel{|\!\approx} \mathbf{K}_k\,q$.
3. Let $s = \{(p \vee q), (\neg p \vee r)\}$. Find the minimal k such that $s \mathrel{|\!\approx} \mathbf{K}_k\,(q \vee r)$.
4. Let $s = \{(o \vee p \vee r), (o \vee \neg p \vee r), (\neg o \vee q), (\neg o \vee \neg q)\}$. Find the minimal k such that $s \mathrel{|\!\approx} \mathbf{K}_k\,r$.

4 Reasoning about Actions

4.1 Basic Action Theories

- Consider a light switch. Model that the fluent LightOn is toggled by an action switch.
- Consider some object that may contain other objects. Setting the containing object alight also sets alight the objects in the box. Model a $\mathrm{Burning}(x)$ fluent using an action $\mathrm{setAlight}(x)$ and another predicate $\mathrm{In}(x, y)$ that indicates that x is in y.
- You're participating in a drug trial: you're sick; you take a some medication, which may be placebo or not; and you see whether or not you feel better afterwards. Model the Sick fluent, which is "disabled" when you take medication x, represented by action $\mathrm{take}(x)$, provided that x is not placebo, that is, $\neg\mathrm{Placebo}(x)$. Also model the sensing axiom for the feel action, which shall tell you whether you're still sick or not.

4.2 Regression

Consider the following basic action theory, where γ and φ are the right-hand sides of the successor-state axiom of Sick and the axiom for SF from the previous task.

$$\begin{aligned}
\Sigma_0 &= \{\mathrm{Sick} \wedge \neg\mathrm{Placebo}(\#1) \wedge \mathrm{Placebo}(\#2)\} \\
\Sigma_1 &= \{\textsc{true}\} \\
\Sigma_{\mathrm{dyn}} &= \{\Box[a]\mathrm{Sick} \leftrightarrow \gamma, \\
&\qquad \Box[a]\mathrm{Placebo}(x) \leftrightarrow \mathrm{Placebo}(x), \\
&\qquad \Box\mathrm{Poss}(a) \leftrightarrow \textsc{true}, \\
&\qquad \Box\mathrm{SF}(a) \leftrightarrow \varphi\}
\end{aligned}$$

(a) Prove that $\Sigma_0 \wedge \Sigma_{\mathrm{dyn}} \models [\mathrm{take}(\#1)]\neg\mathrm{Sick}$ using regression.[2]

(b) Prove that $\Sigma_0 \wedge \Sigma_{\mathrm{dyn}} \wedge \mathbf{O}(\Sigma_1 \wedge \Sigma_{\mathrm{dyn}}) \models [\mathrm{take}(\#1)]\neg\mathbf{K}\,\neg\mathrm{Sick}$.

(c) Prove that $\Sigma_0 \wedge \Sigma_{\mathrm{dyn}} \wedge \mathbf{O}(\Sigma_1 \wedge \Sigma_{\mathrm{dyn}}) \models [\mathrm{take}(\#1)][\mathrm{feel}]\mathbf{K}\,\neg\mathrm{Sick}$.

4.3 Knowledge after Actions

Prove the theorem from Slide 34, which is crucial for the regression of knowledge:

$$\begin{aligned}
\models \Box[a]\mathbf{K}\,\alpha \leftrightarrow\ & (\mathrm{SF}(a) \rightarrow \mathbf{K}\,(\mathrm{SF}(a) \rightarrow [a]\alpha)) \wedge \\
& (\neg\mathrm{SF}(a) \rightarrow \mathbf{K}\,(\neg\mathrm{SF}(a) \rightarrow [a]\alpha))
\end{aligned}$$

[2] We defined Σ_0 and Σ_{dyn} as sets of sentences. We identify such a set of sentences with the conjunction of its elements. That is, writing $\Sigma_0 \wedge \Sigma_{\mathrm{dyn}} \models \alpha$ stands for $\bigwedge_{\phi\in\Sigma_0} \phi \wedge \bigwedge_{\psi\in\Sigma_{\mathrm{dyn}}} \psi \models \alpha$.