The dynamic logic of propositional assignments

Andreas Herzig

University of Toulouse, CNRS, IRIT, France

joint work with Philippe Balbiani (Toulouse) and Nicolas Troquard (Trento)

February 2013

Overview

- PDL: abstract actions only

  Propositional dynamic logic "abstracts away from the nature of the domain of computation and studies the pure interaction between programs and propositions" [Harel et al. 2000]

- concrete programs: propositional assignments

  $p{\leftarrow}\varphi$ = "p is assigned the truth value of φ"

Outline

1 The logic of propositional assignments

2 Complexity of satisfiability

DL-PA: language

- $\mathtt{Prp} = \{p, q, \dots\}$ = set of propositional variables
- programs:
  - $p{\leftarrow}\varphi$ = "p is assigned the truth value of φ"
    - N.B.: don't confuse with assignments of object variables $x{\leftarrow}t$ of first-order dynamic logic
  - complex assignment programs: $p{\leftarrow}\top \cup p{\leftarrow}\bot, \dots$
- formulas: ...

DL-PA: language, ctd.

- BNF for *assignment programs* π and *formulas* φ:

$$\begin{array}{rcl} \pi & ::= & p{\leftarrow}\varphi \mid \pi;\pi \mid \pi \cup \pi \mid \pi^* \mid \varphi? \\ \varphi & ::= & p \mid \top \mid \bot \mid \neg\varphi \mid \varphi\vee\varphi \mid [\pi]\varphi \end{array}$$

- just as in PDL:
  - $\texttt{skip} \stackrel{\text{def}}{=} \top?$
  - $\texttt{if } \varphi \texttt{ then } \pi_1 \texttt{ else } \pi_2 \stackrel{\text{def}}{=} \ldots$
  - $\texttt{while } \varphi \texttt{ do } \pi \stackrel{\text{def}}{=} \ldots$

Models

- valuations $V \subseteq \mathtt{Prp}$
- interpretation of a formula = set of valuations
  - $||\varphi|| = \{V_1, V_2, \ldots\}$
- interpretation of a modality = relation on the set of valuations
  - $||\pi|| = \{\langle V_1, V_1' \rangle, \langle V_2, V_2' \rangle, \ldots\}$

Interpretation of formulas

$$
\begin{aligned}
||\top|| &= 2^{\texttt{Prp}} \\
||\bot|| &= \emptyset \\
||p|| &= \{V \ : \ p \in V\} \\
||\neg\varphi|| &= \ldots \\
||\varphi \vee \psi|| &= \ldots \\
||[\pi]\varphi|| &= \{V \ : \ \text{for every } V' \text{ s.th. } V||\pi||V', \ \ V' \in ||\varphi||\}
\end{aligned}
$$

Interpretation of assignment programs

$$
\begin{aligned}
||p\leftarrow\varphi|| &= \Big\{ \langle V, V'\rangle \ : \ V' = V \cup \{p\} \text{ if } V \in ||\varphi||, \text{ and} \\
&\qquad\qquad\qquad\quad V' = V \setminus \{p\} \text{ if } V \notin ||\varphi|| \Big\} \\
||\pi_1 ; \pi_2|| &= ||\pi_1|| \circ ||\pi_2|| \\
||\pi_1 \cup \pi_2|| &= ||\pi_1|| \cup ||\pi_2|| \\
||\pi^*|| &= \big(||\pi||\big)^* \\
||\varphi?|| &= \{ \langle V, V\rangle \ : \ V \in ||\varphi|| \}
\end{aligned}
$$

Example:
$||p\leftarrow\varphi|| = ||(\varphi?; p\leftarrow\top) \cup (\neg\varphi?; p\leftarrow\bot)||$

Example:
$||\langle p\leftarrow\varphi\rangle\top|| = 2^{\mathrm{Prp}}$

Satisfiability and validity

φ satisfiable	iff	$\|\|\varphi\|\| \neq \|\|\bot\|\|$
φ is valid	iff	$\|\|\varphi\|\| = \|\|\top\|\|$

Complexity of DL-PA satisfiability: overview

NP complete	if no complex programs (apply reduction axioms)
PSPACE complete	if star-free (no π^*) [Herzig et al. IJCAI 2011]
EXPTIME complete	for full language (v.i.)
undecidable	if moreover abstract actions à la PDL [Tiomkin and Makowsky 1985]

Outline

1. The logic of propositional assignments

2. Complexity of satisfiability

DL-PA: decidability

key step: eliminate the Kleene star

1. choose some π^* such that π is star-free
2. transform π into

$$(\varphi_1?;\alpha_1) \cup \cdots \cup (\varphi_n?;\alpha_n)$$

   where every α_k is a sequence of assignments
3. make all the assignment sequences α_k assign exactly the same variables:

$$(\varphi_1?;\alpha_1) \cup \cdots \cup (\varphi_n?;\alpha_n) \quad \text{and} \quad \texttt{Prp}_{\alpha_1} = \ldots = \texttt{Prp}_{\alpha_n}$$

4. replace π^* by

$$((\varphi_1?;\alpha_1) \cup \cdots \cup (\varphi_n?;\alpha_n))^{\leq n}$$

(uses that $\texttt{Prp}_{\alpha_k} = \texttt{Prp}_{\alpha_l}$ implies $\alpha_k;\alpha_l = \alpha_l$)

DL-PA: complexity of the star-free fragment

Theorem ([Herzig et al. IJCAI 2011])

Satisfiability checking is PSPACE-complete for the star-free fragment of DL-PA.

- hardness: encode QBF
- membership:
  1. satisfiability is in NPSPACE: guess valuation V; check $V \in ||\varphi||$
  2. NPSPACE $=$ PSPACE [Savitch]

Full DL-PA: complexity

Theorem

Both model checking (MC) and satisfiability checking (SAT) are EXPTIME-complete.

- membership of SAT: translate into PDL
- hardness of MC: encode PEEK-G_5
  - alternative: encode corridor tiling problem, cf. PDL [Harel et al. 2000]
- polynomial transformation from MC into SAT:

$$V \in ||\varphi|| \text{ iff } \varphi \wedge (\bigwedge_{p \in V \cap \mathtt{Prp}_\varphi} p) \wedge (\bigwedge_{p \notin V \cap \mathtt{Prp}_\varphi} \neg p) \text{ satisfiable}$$

Full DL-PA: proof of EXPTIME hardness

1. PEEK-$G_5(\mathtt{Prp}_E, \mathtt{Prp}_A, \Phi, V_0, \tau)$:
   - propositional variables of $\mathtt{Prp}$ partitioned among Abelard and Eloïse
     - $\mathtt{Prp}_A$: propositional variables of Abelard
     - $\mathtt{Prp}_E$: propositional variables of Eloïse
     - $\mathtt{Prp}_A \cup \mathtt{Prp}_E = \mathtt{Prp}$, $\mathtt{Prp}_A \cap \mathtt{Prp}_E = \emptyset$
   - V_0 is the initial valuation
   - A and E alternatively choose one of their variables and change its truth value
   - player $\tau \in \{A, E\}$ begins
   - Eloïse wins if Φ is true
   - "does Eloïse have a winning strategy?"
     - EXPTIME complete [Stockmeyer and Chandra 79]

Full DL-PA: proof of EXPTIME hardness, ctd.

- define valuation V_1:

$$V_1 = \begin{cases} V_0 \cup \{\text{nowinE}, \text{turnE}\} & \text{if } \tau = E \\ V_0 \cup \{\text{nowinE}\} & \text{if } \tau = A \end{cases}$$

- define a 'move' program:

$$\text{moveE} = \text{turnE}?; \bigcup_{x \in \mathtt{Prp}_E} (x{\leftarrow}\bot \cup x{\leftarrow}\top); \text{turnE}{\leftarrow}\bot$$

$$\text{moveA} = \neg\text{turnE}?; \bigcup_{y \in \mathtt{Prp}_A} (y{\leftarrow}\bot \cup y{\leftarrow}\top); \text{turnE}{\leftarrow}\top$$

$$\text{move} = (\text{moveE} \cup \text{moveA}) ; ((\Phi?; \text{nowinE}{\leftarrow}\bot) \cup \neg\Phi?)$$

Lemma

Eloïse has no winning strategy iff

$$V_1 \models [\text{move}^*]\Big(\text{nowinE} \to (\neg\Phi \land (\text{turnE} \to [\text{move}]\text{nowinE}) \land (\neg\text{turnE} \to \langle\text{move}\rangle\text{nowinE}))\Big)$$

Full DL-PA: proof of EXPTIME hardness, ctd.

- define valuation V_1:

$$V_1 = \begin{cases} V_0 \cup \{\text{nowinE}, \text{turnE}\} & \text{if } \tau = E \\ V_0 \cup \{\text{nowinE}\} & \text{if } \tau = A \end{cases}$$

- define a 'move' program:

$$\text{moveE} = \text{turnE}?; \bigcup_{x \in \mathtt{Prp}_E} (x{\leftarrow}\bot \cup x{\leftarrow}\top); \text{turnE}{\leftarrow}\bot$$

$$\text{moveA} = \neg\text{turnE}?; \bigcup_{y \in \mathtt{Prp}_A} (y{\leftarrow}\bot \cup y{\leftarrow}\top); \text{turnE}{\leftarrow}\top$$

$$\text{move} = (\text{moveE} \cup \text{moveA}) ; ((\Phi?; \text{nowinE}{\leftarrow}\bot) \cup \neg\Phi?)$$

Lemma

Eloïse has no winning strategy iff

$$V_1 \models [\text{move}^*]\Big(\text{nowinE} \to (\neg\Phi \wedge (\text{turnE} \to [\text{move}]\text{nowinE}) \wedge (\neg\text{turnE} \to \langle\text{move}\rangle\text{nowinE}))\Big)$$

Conclusions

- DL-PA $=$ PDL with concrete programs
  - full DL-PA: EXPTIME complete
  - star-free DL-PA: PSPACE complete
    - conjecture: limitation of quantifier alternation $\Rightarrow$ complexity classes Σ_2^p, Π_2^p, etc.