Dynamic Logic of Propositional Assignments and its applications to update, revision and planning

Andreas Herzig
University of Toulouse, IRIT-CNRS, France

Workshop on Planning, Logic and Social Intelligence, Copenhagen, April 4, 2014

Aim of talk

- propaganda for Dynamic Logic of Propositional Assignments [Herzig et al., IJCAI 2011; Balbiani et al., LICS 2013]
  - simpler than PDL
  - *the* basic logic to for dynamic systems
  - aim: canonical problems between NP and PSPACE
    - nicer than QBF
- specifically:
  - Forbus's update operation [Forbus 1989]
  - Dalal's revision operation [Dalal 1988]
  - planning tasks and their modification [Smith, ICAPS 2004; Göbelbecker et al., ICAPS 2010]

Outline

Dynamic Logic of Propositional Assignments DL-PA with converse

- instantiates good old Propositional Dynamic Logic PDL
  - atomic programs: assignments of propositional variables

    $p{\leftarrow}\top$ = “make p true”

    $p{\leftarrow}\bot$ = “make p false”
  - complex programs and formulas: as usual in PDL

$$\pi^m \stackrel{\text{def}}{=} \begin{cases} \text{skip} & \text{if } m = 0 \\ \pi; \pi^{m-1} & \text{otherwise} \end{cases}$$

$$\pi^{\leq m} \stackrel{\text{def}}{=} \begin{cases} \text{skip} & \text{if } m = 0 \\ (\text{skip} \cup \pi); \pi^{m-1} & \text{otherwise} \end{cases}$$

$$p{\leftarrow}\varphi \stackrel{\text{def}}{=} (\varphi?; p{\leftarrow}\top) \cup (\neg\varphi?; p{\leftarrow}\bot)$$

Dynamic Logic of Propositional Assignments DL-PA with converse

- language, for $p \in$ PVar:

$$\pi \quad ::= \quad p{\leftarrow}\top \mid p{\leftarrow}\bot \mid \varphi? \mid \pi;\pi \mid \pi \cup \pi \mid \pi^* \mid \pi^{-1}$$
$$\varphi \quad ::= \quad p \mid \top \mid \bot \mid \neg\varphi \mid \varphi \vee \varphi \mid \langle\pi\rangle\varphi$$

- propositional assignments are an old idea:
  - [Meyer & Winklmann, Symp. Theory of Computing, 1980]
  - [Tiomkin & Makowsky, TCS, 1985] (global and local)
  - [Wilm, TCS 1991]
  - [van Eijck, Studia Logica 2000] (no Kleene star)
  - [Guelev et al., Information Security 2004] (deterministic programs)
- but the above language was not considered before [Balbiani et al., LICS 2013]

Semantics of DL-PA: formulas

- *valuation* = subset of PVar
  - set of all valuations is $2^{\mathsf{PVar}} = \{v, v', \ldots\}$
  - $p \in v$: p true in v
  - $p \notin v$: p false in v
- interpretation of formula φ = set of valuations $||\varphi|| \subseteq 2^{\mathsf{PVar}}$

$$
\begin{aligned}
||p|| &= \{v \ : \ p \in v\} \\
||\top|| &= 2^{\mathsf{PVar}} \\
||\bot|| &= \emptyset \\
||\neg\varphi|| &= 2^{\mathsf{PVar}} \setminus ||\varphi|| \\
||\varphi \vee \psi|| &= ||\varphi|| \cup ||\psi|| \\
||\langle\pi\rangle\varphi|| &= \Big\{v \ : \ \text{there is } v', (v, v') \in ||\pi|| \text{ and } v' \in ||\varphi||\Big\}
\end{aligned}
$$

Semantics of DL-PA: programs

- interpretation of program π = relation $||\pi|| \subseteq 2^{\mathsf{PVar}} \times 2^{\mathsf{PVar}}$

$$
\begin{aligned}
||p{\leftarrow}\top|| &= \{(v, v') \ : \ v' = v \cup \{p\}\} \\
||p{\leftarrow}\bot|| &= \{(v, v') \ : \ v' = v \setminus \{p\}\} \\
||\varphi?|| &= \{(v, v) \ : \ v \in ||\varphi||\} \\
||\pi; \pi'|| &= ||\pi|| \circ ||\pi'|| \\
||\pi \cup \pi'|| &= ||\pi|| \cup ||\pi'|| \\
||\pi^*|| &= (||\pi||)^* \ = \ \bigcup_{k \in \mathbb{N}_0} (||\pi||)^k \\
||\pi^{-1}|| &= (||\pi||)^{-1}
\end{aligned}
$$

DL-PA: eliminating the program operators

- eliminate converse operator:

$$||p{\leftarrow}\top^{-1}|| = ||p?; (\text{skip} \cup p{\leftarrow}\bot)||$$
$$||p{\leftarrow}\bot^{-1}|| = \ldots$$

- eliminate Kleene star:

$$||\pi^*|| = ||\pi^{\leq 2^{\text{card}(\text{PVar}(\pi))}}||$$

- eliminate the other program operators: as in star-free PDL

Proposition
([Herzig et al., IJCAI 2011; Balbiani et al., LICS 2013])

For every DL-PA *formula* φ *there is an equivalent formula* φ' *such that no program operators occur in* φ'*.*

DL-PA: eliminating the dynamic operators

- eliminate atomic programs:
  - atomic programs $\langle p{\leftarrow}\top\rangle$ and $\langle p{\leftarrow}\bot\rangle$ distribute over $\wedge$, $\vee$, $\neg$
  - can be eliminated when facing atomic formulas:

$$\langle p{\leftarrow}\top\rangle q \leftrightarrow \begin{cases} \top & \text{if } q = p \\ q & \text{otherwise} \end{cases} \qquad \langle p{\leftarrow}\bot\rangle q \leftrightarrow \begin{cases} \ldots \\ \ldots \end{cases}$$

Proposition
([Herzig et al., IJCAI 2011; Balbiani et al., LICS 2013])

For every DL-PA *formula there is an equivalent boolean formula.*

DL-PA: eliminating the dynamic operators

Example

$$
\begin{aligned}
\langle p{\leftarrow}\bot^{-1}\rangle(p\wedge q) &\leftrightarrow \langle \neg p?\,;\,(\mathrm{skip}\cup p{\leftarrow}\top)\rangle(p\wedge q)\\
&\leftrightarrow \langle \neg p?\rangle\langle(\mathrm{skip}\cup p{\leftarrow}\top)\rangle(p\wedge q)\\
&\leftrightarrow \neg p \wedge \langle(\mathrm{skip}\cup p{\leftarrow}\top)\rangle(p\wedge q)\\
&\leftrightarrow \neg p \wedge \big(\langle \mathrm{skip}\rangle(p\wedge q) \vee \langle p{\leftarrow}\top)\rangle(p\wedge q)\big)\\
&\leftrightarrow \neg p \wedge \big(\langle \mathrm{skip}\rangle(p\wedge q) \vee (\langle p{\leftarrow}\top)\rangle p \wedge \langle p{\leftarrow}\top)\rangle q)\big)\\
&\leftrightarrow \neg p \wedge \big((p\wedge q) \vee (\top \wedge q)\big)\\
&\leftrightarrow \neg p \wedge \big((p\wedge q) \vee q\big)\\
&\leftrightarrow \neg p \wedge q
\end{aligned}
$$

Properties and applications of DL-PA

- properties
  - no nondeterministic composition: NP complete
  - star-free: PSPACE complete [Herzig et al., IJCAI 2011]
  - full language: PSPACE complete [Balbiani et al., ongoing]
    - flawed EXPTIME hardness proof in [Balbiani et al., LICS 2013]
- applications
  - ...
  - ...
  - ...
  - here:
    - various update and revision operations [Herzig, KR 2014]
    - plan existence, planning task modification

Outline

1. DL-PA: dynamic logic of propositional assignments
2. Update and revision via DL-PA programs
3. Forbus update
4. Dalal revision
5. Planning tasks and their modification

Update and revision operations

- $B \circ A$ = modification of belief base B accomodating input A
  - B, A boolean formulas
  - object language operat*ors* (vs. metalanguage operat*ions*)
- two kinds of change:
  - update = "world changes" [Katsuno&Mendelzon 1992]
  - revision = "knowledge about world changes" [Alchourrón et al., JSL 1985]
- two different accounts:
  1. parametrised operations
     - built from arbitrary orderings or distances
     - characterised by postulates
  2. concrete operations
     - based on particular distances between valuations
     - mainly studied from semantic perspective
       ⇒ $B \circ A$ = a set of valuations of classical propositional logic
       ⇒ syntactical representation?
       ⇒ "disjunction of formulas describing the models of $B \circ A$"

Embedding into DL-PA

- idea:
  - update by atomic formula ≡ atomic assignment

    update by p ≡ $p{\leftarrow}\top$

    update by $\neg p$ ≡ $p{\leftarrow}\bot$
  - update by complex formula A ≈ complex assignment $\pi(A)$
  - depends on belief change operation: $\pi^{\text{wss}}(A) \neq \pi^{\text{forbus}}(A)$, etc.

$$\pi^{\text{wss}}(\neg p \vee \neg q) = p{\leftarrow}\bot \cup q{\leftarrow}\bot \cup (p{\leftarrow}\bot; q{\leftarrow}\bot)$$

$$\pi^{\text{forbus}}(\neg p \vee \neg q) = \ldots$$

- aim: find polynomial embeddings into DL-PA
  1. syntactical construction of the new base via reduction to boolean formulas
  2. general framework for belief change
- to be proved for each change operation $\circ^\omega$:

$$\boxed{B \circ^\omega A \; = \; \left\langle (\pi^\omega(A))^{-1} \right\rangle B}$$

  - here: based on Hamming distance [Forbus 1989; Dalal 1988]
  - other operations: see [Herzig, KR 2014]

Some useful DL-PA programs

- nondeterministically assign truth values to $p_1, \ldots, p_n$:

$$\mathtt{vary}\big(\{p_1, \ldots, p_n\}\big) = (p_1{\leftarrow}\top \cup p_1{\leftarrow}\bot) \, ; \, \cdots \, ; \, (p_n{\leftarrow}\top \cup p_n{\leftarrow}\bot)$$

- nondeterministically flip one of $p_1, \ldots, p_n$:

$$\mathtt{flip1}\big(\{p_1, \ldots, p_n\}\big) = p_1{\leftarrow}\neg p_1 \cup \cdots \cup p_n{\leftarrow}\neg p_n$$

Outline

Forbus's update operation [Forbus89]

- Hamming distance between valuations v and v'

  $$\mathrm{h}(\{p,q\},\{q,r,s\}) = \mathrm{card}(\{p,r,s\}) = 3$$

- valuation-wise update:
  1. for each model $v \in \|B\|$:
     select A-valuations closest w.r.t. the Hamming distance
  2. collect the resulting valuations

$$v \diamond^{\text{forbus}} A = \Big\{ v' \;:\; v' \in \|A\| \text{ and there is no } v'' \text{ such that } \mathrm{h}(v,v'') < \mathrm{h}(v,v') \Big\}$$

Example

$$\neg p \land \neg q \diamond^{\text{forbus}} p \lor q = \|p \oplus q\| \qquad \text{(exclusive } \lor\text{)}$$

$$\neg p \land \neg q \land \neg r \diamond^{\text{forbus}} (p \land q) \lor r = \|\neg p \land \neg q \land r\|$$

Expressing Forbus's operation in DL-PA

Proposition

Let A, B be propositional formulas. Let

$$\mathtt{H}(A, \geq m) = \begin{cases} \top & \text{if } m = 0 \\ \neg\big\langle \mathtt{flip}^{\leq m-1}\big(\mathsf{PVar}(A)\big)\big\rangle A & \text{if } m \geq 1 \end{cases}$$

Let $\pi^{\mathsf{forbus}}(A)$ be the DL-PA *program*

$$\left(\bigcup_{0 \leq m \leq \mathsf{card}(\mathsf{PVar}(A))} \mathtt{H}(A, \geq m)?; \mathtt{flip}^{m}\big(\mathsf{PVar}(A)\big) \right); A?$$

Then $B \diamond^{\mathsf{forbus}} A = ||\langle(\pi^{\mathsf{forbus}}(A))^{-1}\rangle B||$.

- program length cubic in length of A

Outline

Dalal's revision operation

- based on minimisation of the Hamming distance between valuations
- revise B by A = "go to the A-valuations that are closest w.r.t. Hamming distance to the B-valuations"
  - if B is satisfiable:

$$B *^{\text{dalal}} A = \Big\{ v_A \in ||A|| \ : \ \text{there is } v_B \in ||B|| \text{ s.t. there are no } v'_A, v'_B \text{ with } \mathrm{h}(v'_A, v'_B) < \mathrm{h}(v_A, v_B) \Big\}$$

  - if B is unsatisfiable:

$$B *^{\text{dalal}} A = ||A||$$

- revision operation: satisfies AGM preservation postulate

$$\text{if } ||B \wedge A|| \neq \emptyset \text{ then } B * A = ||B \wedge A||$$

Example

$\neg p \vee \neg q *^{\text{dalal}} p = ||p \wedge \neg q||$

- different from update: $\neg p \vee \neg q \diamond^{\text{forbus}} p = ||p||$

Expressing Dalal's operation in DL-PA

Proposition

Let $\pi^{\mathsf{dalal}}(A, B)$ *be the* DL-PA *program*

$$\mathtt{vary}\Big(\mathsf{PVar}(B)\Big)\ ;\ B?\ ;$$

$$\left(\bigcup_{0 \leq m \leq \mathsf{card}(\mathsf{PVar}(A))} \Big([\mathtt{vary}\Big(\mathsf{PVar}(B)\Big)\ ;\ B?]\mathrm{H}(A, \geq m)\Big)?\ ;\ \mathtt{flip}^m\Big(\mathsf{PVar}(A)\Big)\right); A?$$

Then for satisfiable B: $B *^{\mathsf{dalal}} A = \|\langle(\pi^{\mathsf{dalal}}(A, B))^{-1}\rangle B\|$.

- program depends on the input A and on the base B
- program length cubic in length of A + length of B

Outline

Classical planning tasks

- action (*planning operator*): $a = (\text{pre}_a, \text{add}_a, \text{del}_a)$ with:
  - $\text{pre}_a \in \text{Fml}_{\text{bool}}$ (precondition)
  - $\text{add}_a, \text{del}_a \subseteq \text{PVar}$ (finite add list + delete list)
- semantics:

$$||a|| = \Big\{(v, v') \ : \ v \in ||\text{pre}_a|| \text{ and } v' = \big(v \setminus \text{del}_a\big) \cup \text{add}_a\Big\}$$

$$\boxed{||a|| = ||\text{pre}_a?; q_1{\leftarrow}\bot; \cdots; q_n{\leftarrow}\bot; p_1{\leftarrow}\top; \cdots; p_m{\leftarrow}\top||}$$

for $\text{add}_a = \{p_1, \ldots, p_m\}$ and $\text{del}_a = \{q_1, \ldots, q_n\}$

Classical planning tasks

- valuation v is *reachable* from valuation s_0 via a set of actions $\mathtt{A}$ iff there is a a sequence of actions $(a_1, \ldots, a_n)$ and a sequence of valuations $(v_0, \ldots, v_n)$ such that
  - $v_0 = s_0$,
  - $v_n = v$, and
  - $(v_{k-1}, v_k) \in ||a_k||$ for every k such that $1 \leq k \leq n$.
- *classical planning task*: $(\mathsf{PVar}, \mathtt{A}, s_0, S_g)$ where:
  - $\mathtt{A}$ is a set of actions
  - $s_0 \subseteq \mathsf{PVar}$ (initial state)
  - $S_g \subseteq 2^{\mathsf{PVar}}$ (goal)
- *solvable* iff there is $v \in S_g$ reachable from s_0 via $\mathtt{A}$
- hypothesis: PVar and $\mathtt{A} = \{a_1, \ldots, a_n\}$ finite
  - $\mathtt{iterate}_{\mathtt{A}} = (a_1 \cup \cdots \cup a_n)^*$

$(\mathsf{PVar}, \mathtt{A}, s_0, S_g)$ solvable iff $\mathtt{Fml}(s_0) \to \langle \mathtt{iterate}_{\mathtt{A}} \rangle \mathtt{Fml}(S_g)$ DL-PA valid

Planning task modification

- suppose $(\mathsf{PVar}, \mathtt{A}, \mathsf{s}_0, S_g)$ has no solution
- task modification [Göbelbecker et al., ICAPS 2010]:
  1. increase or decrease the set of objects of the domain
  2. augment the set of actions $\mathtt{A}$
  3. change the initial state s_0
  4. change the goal description S_g ('over-subscription planning')
- here: 3 and 4

Changing the initial state

- set of candidate initial states:

$$S'_0 = \{s'_0 \;:\; \text{there is } s_g \in S_g \text{ such that } s_g \text{ is reachable from } s'_0 \text{ via } \mathtt{A}\}$$
$$= \| \langle \mathtt{iterate}_{\mathtt{A}} \rangle \mathtt{Fml}(S_g) \|$$

- *set of initial states closest to* s_0 *from which* S_g *is reachable*:

$$s_0 \diamond^{\mathrm{forbus}} \mathtt{Fml}(S'_0)$$

alternatively: $s_0 \diamond^{\mathrm{pma}} \mathtt{Fml}(S'_0)$ [Göbelbecker et al., ICAPS 2010]

$$s_0 \diamond^{\mathrm{forbus}} \| \langle \mathtt{iterate}_{\mathtt{A}} \rangle \mathtt{Fml}(S_g) \| = \left\| \left\langle \left(\pi^{\mathrm{forbus}}(\langle \mathtt{iterate}_{\mathtt{A}} \rangle \mathtt{Fml}(S_g)) \right)^{-1} \right\rangle \mathtt{Fml}(s_0) \right\|$$

Changing the goal

- given: planning task $(\text{PVar}, \mathtt{A}, s_0, S_g)$
- set of candidate goal states:

$$S'_g = \{s'_g \ : \ s'_g \text{ is reachable from } s_0 \text{ via } \mathtt{A}\} = \left\|\langle \mathtt{iterate}_{\mathtt{A}}^{-1} \rangle \mathtt{Fml}(s_0)\right\|$$

- *set of goal states closest to S_g that are reachable from s_0:*

$$S_g *^{\text{dalal}} \mathtt{Fml}(S'_g)$$

$$S_g *^{\text{dalal}} \left\|\langle \mathtt{iterate}_{\mathtt{A}}^{-1} \rangle \mathtt{Fml}(s_0)\right\| = \left\|\left\langle \left(\pi^{\text{dalal}}(\langle \mathtt{iterate}_{\mathtt{A}}{}^{-1} \rangle \mathtt{Fml}(s_0), \mathtt{Fml}(S_g))\right)^{-1} \right\rangle \mathtt{Fml}(S_g)\right\|$$

Conclusion

- embedding of prominent belief change operations into DL-PA
  - Forbus's operation operation
  - Winslett's update operations WSS and PMA
  - Dalal's revision operation
  - Satoh's revision operation
- embedding of planning tasks and their modifications
  - involves update/revision by counterfactuals
- allows for the syntactical construction of the result
- range of applicability of DL-PA:
  - coalition logic of propositional control [Herzig et al, IJCAI 2011]
  - normative systems [Herzig et al, CLIMA 2011]
  - deontic action logic [Herzig et al, DEON 2012]
  - update of ASP programs [Fariñas et al., LPNMR 2013]
  - belief merging [Herzig et al., FOIKS 2014]
  - modification of abstract argumentation frameworks *à la* Dung [Doutre et al., KR 2014]