Towards a Rational Agent Programming Language with Prioritized Goals

Shakil M. Khan & Yves Lespérance

Department of Computer Science & Engineering
York University
Toronto, ON, Canada

DALT 2010

May 10, 2010
Motivation

• Recently, much work on BDI APLs with *declarative goals* (e.g. [Hindriks et al. ‘00]); essential for:
  – monitoring goal achievement and performing plan failure recovery
  – modeling rational behavior
• But most of these APLs do not
  – *provide a formal semantics for declarative goals or specify their dynamics*
  – *handle temporally extended goals and prioritized goals*
  – *require consistency between adopted declarative goals and plans*
• One reason for these deficiencies
  – underlying agent theory not expressive enough
A Motivating Example

- Modified blocks world
- Only one action, \( stack(b,a) \)
  - preconditions: possible if both blocks are Clear and OnTable

- Initially
  - has 4 blocks: Blue, Yellow, Green, Red
  - all blocks are Clear and OnTable
A Motivating Example

- Initially has two declarative goals:
  \[ \Diamond \text{Tower}^G_{\neg Y} \quad \Diamond \text{Tower}^B_{\neg R} \]

- Planning rule base with only one rule:
  \[ \Diamond \text{Tower}^Q_{\neg P} : \exists b, b'. \text{OnTable}(b) \land \text{OnTable}(b') \land \text{Clear}(b) \land \text{Clear}(b') \land Q(b) \land \neg P(b') \leftrightarrow \text{stack}(b, b') \]

- If has goal and believes that then do
A Motivating Example

- One trace in the absence of consistency check
  - adopt plan $stack(G,B)$ w.r.t. goal $\Diamond \text{Tower}_{\neg Y}^G$
  - execute $stack(G,B)$

<table>
<thead>
<tr>
<th>Decl. Goals</th>
<th>Plans</th>
<th>World State</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\langle \neg Y, \neg R \rangle$</td>
<td>${ } { } { }$</td>
<td>$\langle \text{Blue}, \text{Yellow}, \text{Green}, \text{Red} \rangle$</td>
</tr>
<tr>
<td>$\langle \neg Y, \neg R \rangle$</td>
<td>${ \text{stack}(G,B) } \langle \text{Blue}, \text{Yellow}, \text{Green}, \text{Red} \rangle$</td>
<td></td>
</tr>
<tr>
<td>$\langle \neg R \rangle$</td>
<td>${ } \langle \text{Green}, \text{Yellow}, \text{Red} \rangle$</td>
<td></td>
</tr>
</tbody>
</table>

No Transitions Possible
A Motivating Example

- One trace with consistency check

<table>
<thead>
<tr>
<th>Decl. Goals</th>
<th>Plans</th>
<th>World State</th>
</tr>
</thead>
<tbody>
<tr>
<td>¬Y ¬R</td>
<td>{}</td>
<td></td>
</tr>
<tr>
<td>¬Y ¬R</td>
<td>stack(G,R)</td>
<td></td>
</tr>
<tr>
<td>¬R</td>
<td>{}</td>
<td></td>
</tr>
<tr>
<td>¬R</td>
<td>stack(B,Y)</td>
<td></td>
</tr>
<tr>
<td>{}</td>
<td>{}</td>
<td></td>
</tr>
</tbody>
</table>
Contributions

• Defined Simple Rational BDI APL (SR-APL)
  – combines elements from BDI APLs e.g. AgentSpeak [Rao ‘01] and the situation calculus-based ConGolog APL [De Giacomo et al. ‘00]
  – maintains consistency between chosen declarative goals and adopted plans

• Based on theory of prioritized goals and their dynamics adopted from [Khan&Lespérance AAMAS ‘10 – Session 5]
  – grounded on a formal action theory (i.e. the situation calculus)
  – handles temporally extended goals
  – formalizes prioritized goals, chosen goals, and their dynamics semantically
  – models the dependencies between subgoals and their parent goals
Contributions

• Proven that SR-APL satisfies some key rationality requirements
• Key issues
  – how to ensure consistency?
  – what does it mean to be committed to execute a plan next?
SR-APL : Components

- Theory specifying actions, knowledge, and goals
  - achievement and procedural goals only
- Planning Rule-Base $\Pi$ with rules of the form $(\phi : \psi \leftarrow \sigma)$
- Plan language for $\sigma$
  - primitive actions $a$
  - wait / test actions $\Phi$?
  - sequence of actions $(\delta_1; \delta_2)$
  - special action for adopting subgoal $\Diamond \Phi$ relative to program $\sigma$
    
    $\textit{adopt}(\Diamond \Phi, \sigma)$
  - Procedural intention base $\Gamma$
SR-APL: Configurations & Semantics

- **Configurations**
  - program configuration $\langle \sigma, s \rangle$
  - agent configuration $\langle \Gamma, s \rangle$
    - theory + situation implicitly specify agent’s knowledge and goals
  - initial agent configuration $\langle \emptyset, S_0 \rangle$

- **Operational semantics provided using a two-tier transition system**
  - program-level transitions $\langle \sigma, s \rangle \to \langle \sigma', s' \rangle$ – as in ConGolog
  - agent-level transitions $\langle \Gamma, s \rangle \Rightarrow \langle \Gamma', s' \rangle$
Why Procedural Intention Base?

- Want to model commitment to execute a plan $\delta$ next
- First attempt: agent has goal that $\exists s. \text{Do}(\delta, now, s)$
  - but does not allow concurrency/interleaving – too strong
- Second attempt: goal $\exists s. \text{DoAtleast}(\delta, now, s)$ – execute $\delta$ possibly with any other actions
  - but this allows unnecessary actions – too weak
- Solution: define procedural intention base $\Gamma$ – list of all plans the agent is committed to
  - require actions the agent performs only come from $\Gamma$
SR-APL: Operational Semantics

- Rule $A_{sel}$ for selecting and adopting a plan using rule-base $\Pi$

$$\text{Member}(\Diamond \Phi : \Psi \leftarrow \sigma, \Pi), D \models \text{RPGoal}(\Diamond \Phi, n, s),$$

$$D \models \neg \text{Handled}(\Diamond \Phi, s) \land \text{Know}(\Psi', s), \text{mgu}(\Psi, \Psi') = \theta,$$

$$D \models \neg \text{CGoal}(\neg \exists s'. \text{Do}(\text{adopt}(\text{DoAtleast}(\sigma \theta), \Diamond \Phi), \text{now}, s'), s)$$

$$\langle \Gamma, s \rangle \Rightarrow \langle \text{Cons}(\sigma \theta, \Gamma), \text{do}(\text{adopt}(\text{DoAtleast}(\sigma \theta), \Diamond \Phi), s) \rangle$$

- if head of a rule $(\Diamond \Phi : \psi \leftarrow \sigma)$ in $\Pi$ matches with an unhandled realistic p-goal $\Diamond \Phi$ and belief-condition $\psi$ of that rule also follows from agent’s knowledge and agent does not intend not to adopt the plan $\text{DoAtleast}(\sigma \theta)$

- then can do transition by adopting $\text{DoAtleast}(\sigma \theta)$ as a subgoal of $\Diamond \Phi$ adding $\text{DoAtleast}(\sigma \theta)$ to $\Gamma$
Simple Rational APL

SR-APL : Operational Semantics

• Rule $A_{\text{step}}$ for single stepping the agent program by executing action from $\Gamma$

\[
\text{Member}(\sigma, \Gamma), D \models \text{RPGoal}(\text{DoAtleast}(\sigma), n, s),
\]
\[
D \models \langle \sigma, s \rangle \rightarrow \langle \sigma', do(a, s) \rangle \land \neg \text{CGoal}(\exists s'.Do(a, \text{now}, s'), s)
\]
\[
\langle \Gamma, s \rangle \Rightarrow \langle \text{Replace}(\sigma, \sigma', \Gamma), do(a, s) \rangle
\]

– if $\sigma$ is in $\Gamma$ and
\[
\text{DoAtleast}(\sigma) \text{ is a realistic p-goal at some priority level and}
\]
\[
\sigma \text{ has a program level transition with action } a \text{ and}
\]
\[
\text{executing } a \text{ next is consistent with the agent's intentions}
\]

– then can do transition by executing action $a$
\[
\text{updating } \Gamma \text{ accordingly}
\]
Weak Notion of Consistency

- In $A_{sel}$ and $A_{step}$, we only do partial consistency check; we require that:
  - e.g. agent does not intend not to adopt the plan
    \[ \text{DoAtleast}(\sigma) \]
  - Why?
    - plans might be abstract, i.e. include unexpanded subgoals
    - thus currently non-executable without introducing additional actions
  - Consequences
    - more efficient
    - but agent could get stuck due to wrong choice of actions, and may need to repair the plan
      - however this does not imply that agent could perform some actions that make other goals impossible
SR-APL : Operational Semantics

- Rule $A_{\text{exo}}$ for accommodating exogenous actions
  - only if the action performed is possible

- Rule $A_{\text{clean}}$ for synchronizing procedural goal-base and declarative counter-part; our theory of goals automatically drop
  - impossible goals
  - goals that has become inconsistent with other higher priority goals

- Rule $A_{\text{repair}}$ for repairing plans when agent gets stuck due to
  - occurrence of exogenous events
  - partial consistency check
Traces

• **Labeled execution trace**: possibly infinite sequence of configurations

\[ l_0 \quad l_1 \quad l_2 \quad l_3 \]

\[ \langle \Gamma_0, s_0 \rangle \Rightarrow \langle \Gamma_1, s_1 \rangle \Rightarrow \langle \Gamma_2, s_2 \rangle \Rightarrow \langle \Gamma_3, s_3 \rangle \Rightarrow \ldots \]

such that \( \Gamma_0 = \{ \} \) and \( s_0 = S_0 \)

and for all \( \langle \Gamma_i, s_i \rangle \) the agent level transition rule \( l_i \) can be used to obtain

\[ \langle \Gamma_{i+1}, s_{i+1} \rangle \]

• **Complete trace**: a finite labeled execution trace such that \( \langle \Gamma_i, s_i \rangle \neq > \)

or \( \langle \Gamma_i, s_i \rangle \) is final
Blocks World Revisited

• Proposition – in the absence of exogenous actions

for all complete traces $\langle \Gamma_0, s_0 \rangle \Rightarrow \ldots \Rightarrow \langle \Gamma_n, s_n \rangle$ without exo. actions

we have: $D_{BW} \models \text{Tower}_{nY}^G(s_n) \land \text{Tower}_{nR}^B(s_n)$

and $\langle \Gamma_n, s_n \rangle$ is final
Rationality of SR-APL Agents

• Consistency of knowledge & intentions
  - holds for all configurations
    \[ D \models \forall s. \neg \text{Know}(false, s) \land \neg \text{CGoal}(false, s) \]

• Consistency of declarative and procedural goals

  Given trace \[ T = \langle \Gamma_0, s_0 \rangle \Rightarrow \cdots \Rightarrow \langle \Gamma_n, s_n \rangle \] without exogenous actions for all \( i \) s.t. \( 0 < i \leq n \)
  if \( \langle \Gamma_i, s_i \rangle \) is a configuration in a trace, then for all \( \sigma \in \Gamma_i \)
  we have: \[ D \models \text{CGoal}(\text{DoAtleast}(\sigma), s_i) \]
Rationality of SR-APL Agents

- **Rationality of actions in a trace**

  Given trace $T = \langle \Gamma_0, s_0 \rangle \Rightarrow \ldots \Rightarrow \langle \Gamma_n, s_n \rangle$ without exogenous actions

  for all $i$ s.t. $0 < i \leq n$ and $s_i = do(a, s_{i-1})$

  1. $D \models \neg CGoal(\neg Do(a), s_{i-1})$
  2. if $l_i = A_{step}$ then $D \models CGoal(DoAtleast(a), s_{i-1})$
  3. if $a = adopt(\psi, \phi)$ then $D \models \neg CGoal(\neg O\psi, s_{i-1})$
Related Work and Future Work

• Related work
  – most APL with declarative goals only handle achievement goals
  – most assume all goals have same priority
  – [Hindriks et al. ‘09] – temporally extended goals
  – to the best of our knowledge, none maintains consistency

• Future work
  – investigate restricted versions of SR-APL to improve efficiency/tractability
  – incorporate other types of temporally extended goals
  – incorporate full look-ahead (hierarchical decomposition) over plans for consistency check before adopting/executing them
Appendix: Weak Consistency Check

- Domain with 3 actions
  - $a$, $b$, and $r$
  - execution of $a$ makes the preconditions of $b$ false
  - execution of $r$ restores the preconditions of $b$
- Only possible execution of $\{a || b\}$ is $(b; a)$

- Agent has
  - declarative intentions $\text{DoAtleast}(a)$ and $\text{DoAtleast}(b)$
  - procedural intention base $= \{a, b\}$

- Rule $A_{\text{step}}$ allows agent to execute action $p$ if it is consistent with its (declarative) intentions, i.e. with $[\text{DoAtleast}(a) || \text{DoAtleast}(b)]$
  - $p$ can be action $a$, since the execution $(a; r; b)$ is a possible execution of the above