Cognitive agents with non-monotonic reasoning

(dissertation research overview)

Peter Novák

Clausthal University of Technology, Germany

May 11th, 2008
DMP @ AAMAS’08, Estoril, Portugal
Cognitive agents

Knowledge intensive/cognitive agents

- **knowledge** - state of environment, attitudes $\leadsto$ mental state
- **body** - sensors/effectors $\leadsto$ environment
- **system dynamics** - performing actions $\leadsto$ behaviours

Niche:

- logic-based KR for modelling knowledge $\leadsto$ NMR/ASP
- dynamic & unstructured environments $\leadsto$ DyLP

State of the art (BDI):

- fixed KR technology $\leadsto$ simple Prolog-based
- complex semantics bound to KR $\leadsto$ engineering? larger case-studies?
Cognitive agents

Knowledge intensive/cognitive agents

- knowledge - state of environment, attitudes $\leadsto$ mental state
- body - sensors/effectors $\leadsto$ environment
- system dynamics - performing actions $\leadsto$ behaviours

Niche:

- logic-based KR for modelling knowledge $\leadsto$ NMR/ASP
- dynamic & unstructured environments $\leadsto$ DyLP

State of the art (BDI):

- fixed KR technology $\leadsto$ simple Prolog-based
- complex semantics bound to KR
  $\leadsto$ engineering? larger case-studies?
Cognitive agents

Knowledge intensive/cognitive agents

- **knowledge** - state of environment, attitudes $\rightsquigarrow$ mental state
- **body** - sensors/effectors $\rightsquigarrow$ environment
- **system dynamics** - performing actions $\rightsquigarrow$ behaviours

Niche:

- logic-based KR for modelling knowledge $\rightsquigarrow$ NMR/ASP
- dynamic & unstructured environments $\rightsquigarrow$ DyLP

State of the art (BDI):

- fixed KR technology $\rightsquigarrow$ simple Prolog-based
- complex semantics bound to KR
  $\rightsquigarrow$ engineering? larger case-studies?
Motivation

Problem statement

Thesis: scope & outline

Driving question

Can non-monotonic reasoning be practically used as a KR technology in non-trivial cognitive agent systems?

1 theoretical basis: agent programming language
   - heterogeneous KRvs. behaviours \(\leadsto\) hybrid architectures

2 evaluation: case studies \(\leadsto\) single agent, non-critical
   - videogames & virtual spaces
   - entertainment robotics

3 methodology guidelines
Driving question

Can non-monotonic reasoning be practically used as a KR technology in non-trivial cognitive agent systems?

1 **theoretical basis**: agent programming language
   - heterogeneous KRs vs. behaviours $\leadsto$ hybrid architectures

2 **evaluation**: case studies $\leadsto$ single agent, non-critical
   - videogames & virtual spaces
   - entertainment robotics

3 **methodology guidelines**
Behavourial State Machines/Jazzyk

**core concept:** KR module $\mathcal{M} = (\mathcal{L}, \mathcal{Q}, \mathcal{U})$

- $\mathcal{L}$ - a KR language,
- $\mathcal{Q}$ - a set of query operators $\models: S \times \mathcal{L} \rightarrow \{\top, \bot\}$,
- $\mathcal{U}$ - set of update operators $\oplus: S \times \mathcal{L} \rightarrow S$. 

**BDI agent system**

- **capabilities**
- **interaction rules**
  - $Q_B \rightarrow \mathcal{U}_D \rightarrow Q_I \rightarrow \mathcal{U}_C$
  - **interpreter**
- **Environment**
  - **events**
  - **actions**
Case-studies: Jazzbot

1. Jazzbot - softbot in a simulated 3D world
2. Agent Contest 2009 - small MAS/coordination
   \(\leadsto\) inter-agent communication
3. simulated robots?

Agent program:

```c
when believes goals(Obj) [[find(Obj)]] and
   believes brain(Obj) [[see(Obj)]] and
   query map(Object, Dist) [[Dist=dist_distance_of(Obj)]]
then {
   act body(Dist) [[move forward Dist]],
   update brain(Obj) [[keeps(Obj)]]
}
```

Jazzyk interpreter

Belief base

- Q: U
- brain
- ASP solver (Smodels/Ipars)
- Ruby interpreter

Goal base

- Q: U
- goals
- ASP solver (Smodels/Ipars)

Environment

- Q: U
- body
- Nexuiz client
- Net
- Game server
Using BSM & Jazzyk

Goal-Oriented Behaviours:

- semi-formal design specification
  - higher level syntactic/semantic constructs
  - code templates: perceptions, goals, interruption handlers, re-usable behaviours, modules, ...
  - logic for BSM
    - annotations (FOL, the language of beliefs)

... towards design guidelines.

Jazzyk BSM = an intermediate/assembly language
- enforces explicit control cycle \(\implies\) comparison platform
- compiling BDI languages to BSM (GOAL)
Using BSM & Jazzyk

Goal-Oriented Behaviours:

- semi-formal design specification
  - higher level syntactic/semantic constructs
  - code templates: perceptions, goals, interruption handlers, re-usable behaviours, modules, ...
- logic for BSM
  - annotations (FOL, the language of beliefs)

... towards design guidelines.

- Jazzyk BSM = an intermediate/assembly language
  - enforces explicit control cycle \(\rightarrow\) comparison platform
  - compiling BDI languages to BSM (GOAL)
Conclusion

Summary & contributions

Thesis

A more abstract computational model is needed to enable a practical use of heterogeneous KRs (NMR/ASP) in non-trivial cognitive agents.

⇝ BSM is a suitable model for this task!

- Behavioural State Machines (AAMAS’06, ProMAS’07, AAAI-SS08/AlTA)
- Jazzyk (ProMAS’08, http://jazzyk.sourceforge.net/)
- Jazzbot original application of ASP (ProMAS’08)
- further case studies
- BSM as an intermediate language (submitted)
- design guidelines/methodology (first steps - submitted)
Summary & contributions

Thesis
A more abstract computational model is needed to enable a practical use of heterogeneous KRs (NMR/ASP) in non-trivial cognitive agents.  
⇝ BSM is a suitable model for this task!

- **Behavioural State Machines** (AAMAS’06, ProMAS’07, AAAI-SS08/AlTA)
- **Jazzyk** (ProMAS’08, http://jazzyk.sourceforge.net/)
- **Jazzbot** original application of ASP (ProMAS’08)
- further case studies
- **BSM** as an intermediate language (submitted)
- design guidelines/methodology (first steps - submitted)
Thank you for your attention.

Questions

?