

Agent-Based Systems

Michael Rovatsos
mrovatso@inf.ed.ac.uk

Lecture 5 – Reactive and Hybrid Agent Architectures

1 / 19

Symbolic AI: A Critical View

- Recall “Symbol system hypothesis”
 - Is inference on symbols representing the world sufficient to solve real-world problems . . .
 - . . . or are these symbolic representations irrelevant as long as the agent is successful in the physical world?
 - “Elephants don’t play chess” (or do they?)
- Problems with “symbolic AI”:
 - Computational complexity of reasoning in real-world applications
 - The transduction/knowledge acquisition bottleneck
 - Logic-based approaches largely focus on theoretical reasoning
 - In itself, detached from interaction with physical world

3 / 19

Where are we?

Last time . . .

- Practical reasoning agents
- The BDI architecture
- Intentions and commitments
- Planning and means-ends reasoning
- Putting it all together

Today . . .

- **Reactive and Hybrid Agent Architectures**

2 / 19

Types of Agent Architectures

- From this dispute a distinction between **reactive** (, behavioural, situated) and **deliberative** agents evolved
- Alternative view: distinction arises naturally from tension between reactivity and proactiveness as key aspects of intelligent behaviour
- Broad categories:
 - Deliberative Architectures
 - focus on planning and symbolic reasoning
 - Reactive Architectures
 - focus on reactivity based on behavioural rules
 - Hybrid Architectures
 - attempt to balance proactiveness with reactivity

4 / 19

Reactive Architectures

- BDI certainly most widespread model of rational agency, but also criticism as it is based on symbolic AI methods
- Some of the (unsolved/insoluble) problems of symbolic AI have lead to research in **reactive architectures**
- One of the most vocal critics of symbolic AI: Rodney Brooks
- Brooks has put forward three theses:
 - 1 Intelligent behaviour can be generated without explicit representations of the kind that symbolic AI proposes
 - 2 Intelligent behaviour can be generated without explicit abstract reasoning of the kind that symbolic AI proposes
 - 3 Intelligence is an emergent property of certain complex systems

5 / 19

Subsumption architecture

- Formally: *see* as before, *action* function = set of behaviours
 - Set of all behaviours $Beh = \{(c, a) | c \subseteq Per \text{ and } a \in Ac\}$
 - Behaviour will fire in state s iff $see(s) \in c$
 - Agent's set of behaviours $R \subseteq Beh$, **inhibition relation** $\prec \subseteq R \times R$
 - \prec is a strict total ordering (transitive, irreflexive, antisymmetric)
 - If $b_1 \prec b_2$, b_1 will get priority over b_2

- Action selection in the subsumption architecture:

Function: Action Selection in the Subsumption Architecture

```

1. function action( $p : Per$ ) :  $Ac$ 
2. var fired :  $\wp(R)$ , selected :  $A$ 
3. begin
4.   fired  $\leftarrow \{(c, a) | (c, a) \in R \text{ and } p \in c\}$ 
5.   for each  $(c, a) \in \textit{fired}$  do
6.     if  $\neg(\exists(c', a') \in \textit{fired} \text{ such that } (c', a') \prec (c, a))$  then
7.       return  $a$ 
8.   return null
9. end
  
```

7 / 19

Subsumption Architecture

- Brooks' research based on two key ideas:
 - Situatedness/embodiment: Real intelligence is situated in the world, not in disembodied systems such as theorem provers or expert systems
 - Intelligence and emergence: Intelligent behaviour results from agent's interaction with its environment. Also, intelligence is "in the eye of the beholder" (not an innate property)
- **Subsumption architecture** illustrates these principles:
 - Essentially a hierarchy of task-accomplishing **behaviours** (simple rules) competing for control over agent's behaviour
 - Behaviours (simple situation-action rules) can fire simultaneously
 - ➡ need for meta-level control
 - Lower layers correspond to "primitive" behaviours and have precedence over higher (more abstract) ones
 - Extremely simple in computational terms (but sometimes extremely effective)

6 / 19

Example: The Mars explorer system

- Luc Steels' cooperative Mars explorer system
- Domain: a set of robots are attempting to gather rock samples on Mars (location of rocks unknown but they usually come in clusters); there is a radio signal from the mother ship to find way back
- Only five rules (from top (high priority) to bottom (low priority)):
 - 1 If detect an obstacle *then* change direction
 - 2 If carrying samples and at the base *then* drop samples
 - 3 If carrying samples and not at the base *then* travel up gradient
 - 4 If detect a sample *then* pick sample up
 - 5 If true *then* move randomly
- This performs well, but doesn't consider clusters (➡ potential for cooperation)

8 / 19

Example: The Mars explorer system

- When finding a sample, it would be helpful to tell others
- Direct communication is not available
- Inspiration from ants' foraging behaviour
 - Agent will create trail by dropping crumbs of rock on way back to base, other agents will pick these up (making trail fainter)
 - If agents find that trail didn't lead to more samples, they won't reinforce trail
- Modified set of behaviours:
 - 1 If detect an obstacle *then* change direction
 - 2 If carrying samples and at the base *then* drop samples
 - 3 If carrying samples and not at the base *then* **drop 2 crumbs** and travel up gradient
 - 4 If detect a sample *then* pick sample up
 - 5 If **sense crumbs** *then* **pick up 1 crumb** and **travel down gradient**
 - 6 If true *then* move randomly

9 / 19

Hybrid Architectures

- Idea: Neither completely deliberative nor completely reactive architectures are suitable → combine both perspectives in one architecture
- Most obvious approach: Construct an agent that exists of one (or more) reactive and one (or more) deliberative sub-components
- Reactive sub-components would be capable to respond to world changes without any complex reasoning and decision-making
- Deliberative sub-system would be responsible for abstract planning and decision-making using symbolic representations

11 / 19

Discussion

- Reactive architectures achieve tasks that would be considered very impressive using symbolic AI methods
- But also some drawbacks:
 - Agents must be able to map local knowledge to appropriate action
 - Impossible to take non-local (or long-term) information into account
 - If it works, how do we know why it works?
 - departure from “knowledge level” → loss of transparency
 - What if it doesn't work?
 - purely reactive systems typically hard to debug
 - Lack of clear design methodology (although learning control strategy is possible)
 - Design becomes difficult with more than a few rules
 - How about communication with humans?

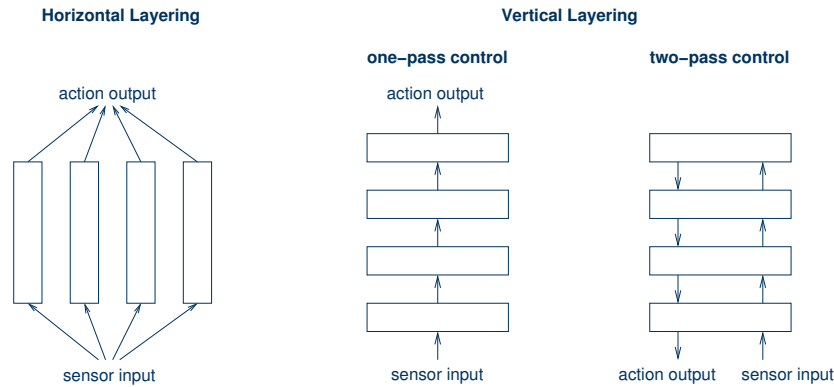
10 / 19

Hybrid Architectures

- Meta-level control of interactions between these components becomes a key issue in hybrid architectures
- Commonly used: **layered** approaches
- Horizontal layering:
 - All layers are connected to sensory input/action output
 - Each layer produces an action, different suggestions have to be reconciled
- Vertical layering:
 - Only one layer connected to sensors/actuators
 - Filtering approach (one-pass control): propagate intermediate decisions from one layer to another
 - Abstraction layer approach (two-pass control): different layers make decisions at different levels of abstraction

12 / 19

Hybrid Architectures



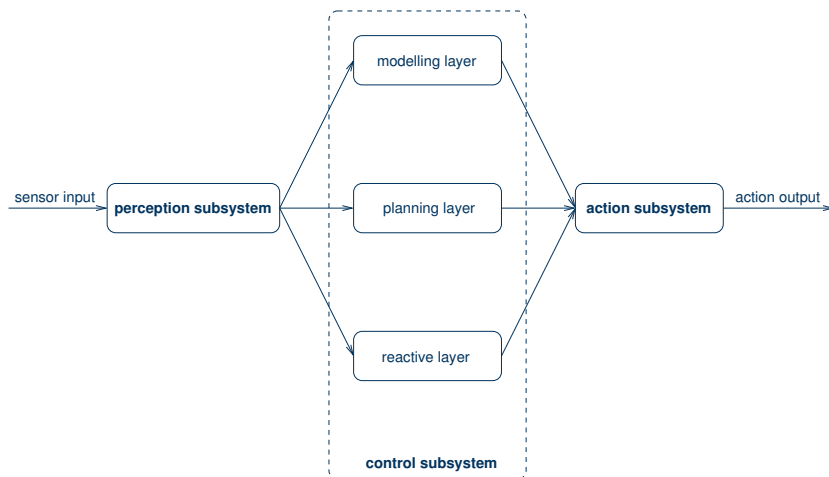
Touring Machines

- Horizontal layering architecture
- Three sub-systems: Perception sub-system, control sub-system and action sub-system
- Control sub-system consists of
 - Reactive layer: situation-action rules
 - Planning layer: construction of plans and action selection
 - Modelling layer: contains symbolic representations of mental states of other agents
- The three layers communicate via explicit **control rules**

13 / 19

14 / 19

Touring Machines



InteRRaP

- InteRRaP: Integration of rational planning and reactive behaviour
- Vertical (two-pass) layering architecture
- Three layers:
 - Behaviour-Based Layer: manages reactive behaviour of agent
 - Local Planning Layer: individual planning capabilities
 - Social Planning Layer: determining interaction/cooperation strategies
- Two-pass control flow:
 - Upward activation: when capabilities of lower layer are exceeded, higher layer obtains control
 - Downward commitment: higher layer uses operation primitives of lower layer to achieve objectives

15 / 19

16 / 19

InteRRaP

- Every layer consists of two modules:
 - situation recognition and goal activation module (SG)
 - decision-making and execution module (DE)
- Every layer contains a specific kind of knowledge base
 - World model
 - Mental model
 - Social model
- Only knowledge bases of lower layers can be utilised by any one layer (nice principle for decomposition of large KB's)
- Very powerful and expressive, but highly complex!

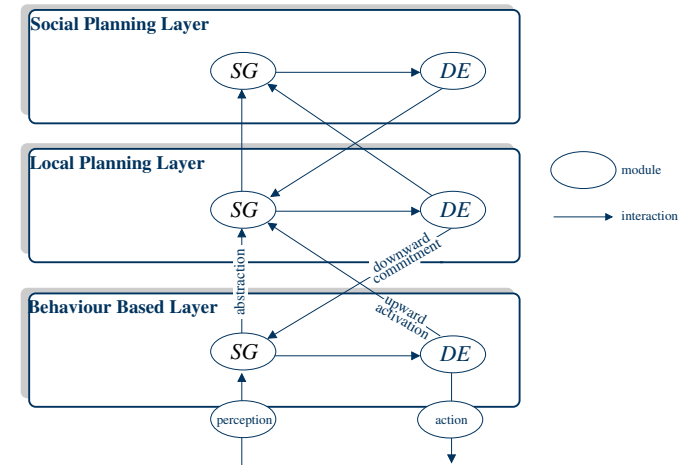
17 / 19

Summary

- Agent architectures: deliberative, reactive and hybrid
- Tension between reactivity and proactiveness
- BDI architecture: “intentional stance”, computationally heavy
- Subsumption architecture: effective, but reasons for success sometimes “obscure” (“black-box” character)
- Hybrid architecture: attempt to balance both aspects, but increased complexity (and lack of conceptual clarity)
- Next time: **Agent Communication**

19 / 19

InteRRaP



18 / 19