Knowledge Engineering

Semester 2, 2004-05

Michael Rovatsos
mrovatso@inf.ed.ac.uk

Lecture 11 – Agent Architectures
18th February 2005

Where are we?

Last time ...

- Introduction to agents and multiagent systems
- Discussed key properties of agents (autonomy, rationality, social ability)
- Looked at different kinds of interaction (coordination, communication, collaboration etc.)
- Discussion of key research topics in agents

Today ...

- **Agent Architectures**

Symbolic AI: A Critical View

- Recall first lecture: symbol system vs. physical grounding 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?)
- Also problems with “symbolic AI”:
  - Computational complexity of reasoning in real-world applications
  - The knowledge acquisition bottleneck
  - Largely focuses on theoretical reasoning about the world

Types of Agent Architectures

- From this dispute a distinction between **reactive** (often called **behaviour-based**) and **deliberative** agents evolved
- Alternative view: distinction arises naturally from tension between reactivity and proactiveness (see previous lecture)
- Broad categories:
  - Deliberative Architectures
    - focus on planning and symbolic reasoning
  - Reactive Architectures
    - focus on reactivity based on behavioural rules
  - Hybrid Architectures
    - attempting to balance proactiveness with reactivity

The BDI Architecture

- BDI: Beliefs, Desires, Intentions
- Based on work on human **practical reasoning**, i.e. everyday reasoning about "what to do"

  Practical reasoning is a matter of weighing conflicting considerations for and against competing options, where the relevant considerations are provided by what the agent desires/values/cares about and what the agent believes. (Michael Bratman)
- Theoretical reasoning is rather directed towards beliefs and knowledge and usually involves no activity

Practical Reasoning

- Practical reasoning consists of two main activities:
  1. Deliberation
  2. Means-ends reasoning

  Combining these appropriately is the foundation of deliberative agency
- **Deliberation** is concerned with determining what one wants to achieve (considering one's preferences, choosing goals to pursue, etc.)
- Deliberation generates **intentions**
- **Means-ends reasoning** is used to determine how the goals are to be achieved (thinking about suitable actions, resources and how to "organise" activity)
- Means-ends reasoning generates **plans**

Intentions

- Bratman's model suggests the following properties:
  - Intentions pose problems for agents, who need to determine ways of achieving them
  - Intentions provide a 'filter' for adopting other intentions, which must not conflict
  - Agents track the success of their intentions, and are inclined to try again if their attempts fail
  - Agents believe their intentions are possible
  - Agents do not believe they will not bring about their intentions
  - Under certain circumstances, agents believe they will bring about their intentions
  - Agents need not intend all the expected side effects of their intentions

Intentions

- Cohen-Levesque theory of intentions based on notion of **persistent goal**
- An agent has a persistent goal of ϕ iff:
  1. It has a goal that ϕ eventually becomes true, and believes that ϕ is not currently true
  2. Before it drops the goal ϕ, one of the following conditions must hold:
     - the agent believes ϕ has been satisfied
     - the agent believes ϕ will never be satisfied
- Definition of intention (consistent with Bratman's list):

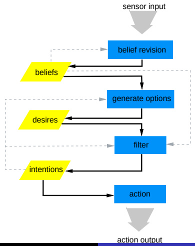
  An agent intends to do action α iff it has a persistent goal to have brought about a state wherein it believed it was about to do α, and then did α.

Desires

- Desires describe the states of affairs that are considered for achievement, i.e. basic preferences of the agent
- Desires are much weaker than intentions, they are not directly related to activity:

  My desire to play basketball this afternoon is merely a potential influencer of my conduct this afternoon. It must vie with my other relevant desires [. . .] before it is settled what I will do. In contrast, once I intend to play basketball this afternoon, the matter is settled: I normally need not continue to weigh the pros and cons. When the afternoon arrives, I will normally just proceed to execute my intentions. (Bratman, 1990)

The BDI Architecture



The BDI Architecture

Sub-components of overall BDI control flow:

- Belief revision function
  - Update beliefs with sensory input and previous belief
- Generate options
  - Use beliefs and existing intentions to generate a set of alternatives/options (=desires)
- Filtering function
  - Choose between competing alternatives and commit to their achievement
- Planning function
  - Given current belief and intentions generate a plan for action
- Action generation: iteratively execute actions in plan sequence (in very simple model)

Issues

- Different commitment strategies:
  - Blind/fanatical commitment: maintain intention until it has been achieved
  - Single-minded commitment: maintain intention until achieved or proves impossible
- Commitment both to ends (intention) and means (plan), particular commitment strategy may lead to overcommitment
- **Re-planning**: include a test for viability of plan after every action (and plan again)
- **Intention reconsideration**
  - Stop to think whether intentions are already fulfilled/impossible to achieve
  - Trade-off: intention reconsideration is costly but necessary
    ➡ meta-level control might be useful
  - Reconsideration always successful if agent *would* have changed intentions *had* he deliberated again

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

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 result 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
  - Lower layers correspond to "primitive" behaviours and have precedence over higher (more abstract) ones
  - Extremely simple in computational terms (but sometimes extremely effective)

Example

- 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 bottom (high priority) to top (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 signal gradient
  4. *If* detect a sample *then* pick sample up
  5. *If* true *then* move randomly
- Near-optimal behaviour!

Discussion

- Reactive architectures achieve tasks that would be considered very impressive using symbolic AI methods
- But also some drawbacks:
  - If it works, how do we know why it works?
    ⇒ departure from "knowledge level" implies 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)
  - How about communication with humans?
- One final remark: don't confuse deliberative/reactive with symbolic/sub-symbolic (e.g. neural networks/genetic algorithms/numerical AI)

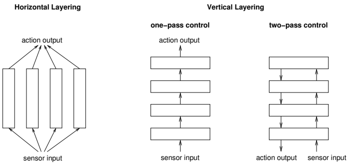
Hybrid Architectures

- Idea: Neither completely deliberateve 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

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/effectors
  - 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

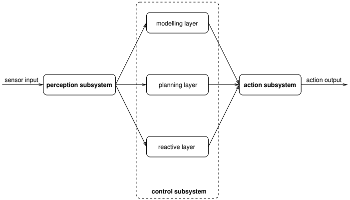
Hybrid Architectures



Touring Machines

- Horizonal 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 state of other agents
- The three layers communicate via explicit **control rules**

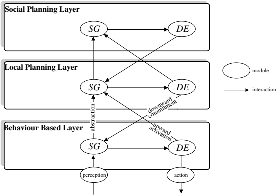
Touring Machines



InteRRaP

- Vertical (two-pass) layering architecture
- InteRRaP: Integration of rational planning and reactive behaviour
- 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

InteRRaP



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
- Very powerful and expressive, but highly complex!

Summary

- Agent architectures: deliberative, reactive and hybrid
- Tension between reactivity and proactiveness
- BDI architecture: "intentional stance", computationally heavy
- Subsumption architecture: effective, but success sometimes "obscure"
- Hybrid architecture: attempt to balance both aspects, but increased complexity
- Next time: **Agent interaction & communication**