Reactive Architectures

Many problems with symbolic reasoning agents.

Brooks put forward three theses:

1. Intelligent behaviour can be generated without explicit representations of the kind that symbolic AI proposes. Intelligent behaviour arises as a result of an agent's interaction with its environment. Also, intelligence is situated in the world, not in disembodied systems such as theorem provers or expert systems.
2. Intelligent behaviour can be generated without explicit abstract reasoning of the kind that symbolic AI proposes. Intelligence is an emergent property of certain complex systems.
3. Intelligent agents do not have an innate, isolated property. Intelligence is in the eye of the beholder; it is not an innate, pre-programmed quality.

The development of reactive architectures.

These problems have led some researchers to question the viability of the whole paradigm, and to rethink the way intelligent systems are designed.

Research:

Brooks put forward three theses:

1. Situatedness and embodiment: 'Real' intelligence is situated in the world, not in disembodied systems.
2. Intelligence and emergence: 'Intelligent' behaviour arises as a result of an agent's interaction with its environment. Also, intelligence is 'in the eye of the beholder'; it is not an innate, isolated property.
3. Intelligent behaviour can be generated without explicit representations of the kind that symbolic AI proposes.

Brooks—behaviour languages.
To illustrate his ideas, Brooks built agents based on his subsumption architecture.

A subsumption architecture is a hierarchy of lower layers representing more primitive kinds of control over the agent. Each behaviour is a simple, rule-like structure. Task- accomplishment behaviours.

http://www.csc.liv.ac.uk/~mjw/pubs/imas/

Chapter 5 An Introduction to Multiagent Systems

Steels’ Marsexplorersystem, using the subsumption architecture, achieves near-optimal cooperation. The objective is to explore a distant planet, and in particular, to collect samples of a precious rock. The location of the samples is not known in advance, but it is known that they tend to be clustered.

For individual (non-cooperative) agents, the lowest-level behaviour, and hence the behaviour with highest “priority” is obstacle avoidance:

if detect an obstacle then
direction.

Any samples carried by agents are dropped back at the mother-ship:

if carryingsamples and at the base then dropsamples

Agents carrying samples will return to the mother-ship:

if carrying samples and not at the base then travel up gradient.

Agents will collect samples they find:

if true then move randomly:

Randomly: if nothing better to do, will explore

An agent with “nothing better to do” will explore

Agents carrying samples are dropped back at the mother-ship:

if carryingsamples and at the base then dropsamples

Agents carrying samples will return to the mother-ship:

if carrying samples and not at the base then travel up gradient.

An agent with “nothing better to do”, will explore randomly.

If a sample is detected, then pick it up.

If any samples are carried, then drop them at the mother-ship.

If an agent is not carrying samples, then move randomly.

The objective is to explore a distant planet, and in particular, to collect samples of a precious rock. The location of the samples is not known in advance, but it is known that they tend to be clustered.

Chapter 5 An Introduction to Multiagent Systems

Steels’ Marsexplorersystem, using the subsumption architecture, achieves near-optimal cooperation. The objective is to explore a distant planet, and in particular, to collect samples of a precious rock. The location of the samples is not known in advance, but it is known that they tend to be clustered.

For individual (non-cooperative) agents, the lowest-level behaviour, and hence the behaviour with highest “priority” is obstacle avoidance:

if detect an obstacle then
direction.

Any samples carried by agents are dropped back at the mother-ship:

if carryingsamples and at the base then dropsamples

Agents carrying samples will return to the mother-ship:

if carryingsamples and not at the base then travel up gradient.

An agent with “nothing better to do”, will explore randomly.

if true then move randomly:

Randomly: if nothing better to do, will explore

An agent with “nothing better to do”, will explore randomly.

if true then move randomly:

Randomly: if nothing better to do, will explore

An agent with “nothing better to do”, will explore randomly.

if true then move randomly:

Randomly: if nothing better to do, will explore

An agent with “nothing better to do”, will explore randomly.
Chapter 5: An Introduction to Multiagent Systems

Situated Automata

- In the situated automata paradigm, an agent is specified in a rule-like (declarative) language, and this specification is then compiled down to a digital machine which satisfies the declarative specification. This digital machine can operate in a provable time bound. The digital machine can thus be simulated on a regular computer.

- This digital machine can be simulated on a regular computer, and thus the specified in a rule-like (declarative) language, and this

Hybrid Architectures

- Many researchers have argued that neither a completely deliberative nor completely reactive approach is suitable for building agents. An obvious approach is to build an agent out of two sub-systems: a deliberative one, containing a symbolic model of the world, and a reactive one, capable of reacting to events without complex reasoning. A reactive one, which is capable of reacting to events as they occur, is proposed by symbolic AI; and another approach is suitable for building agents.

- Often, the reactive component is given some kind of precedence over the deliberative one.

- The theoreticallimitations of the approach are not well understood.
A key problem in such architectures is what kind of control framework to embed the agent's subsystems in, to manage the interactions between the various layers.

- Vertical layering. Layers are each directly connected to the sensory input and action output. In effect, each layer itself acts like an agent, producing suggestions as to what action to perform.

- Horizontal layering. Layers are each directly connected to the sensory input and action output. In effect, each layer itself acts like an agent, producing suggestions as to what action to perform.

The TOURING MACHINES architecture consists of perception and action subsystems, which interface directly with the agent's environment, and three control layers, embedded in a control framework, which mediates between the layers.
Chapter 5: An Introduction to Multiagent Systems

- The reactive layer is implemented as a set of situation-action rules. It is subsumption architecture.

\[ \text{Example:} \]

\[ \text{rule-1: kerb-avoidance} \]

\[ \text{if} \quad \text{is-in-front(Kerb,Observer)} \quad \text{and} \quad \text{speed(Observer)}>0 \quad \text{and} \quad \text{separation(Kerb,Observer)}<\text{KerbThreshold} \]

\[ \text{then} \quad \text{change-orientation(KerbAvoidanceAngle)} \]

- The planning layer constructs plans and selects actions to execute in order to achieve the agent's goals.

\[ \text{Example:} \]

\[ \text{censor-rule-1:} \]

\[ \text{if} \quad \text{entity(obstacle-6) in perception-buffer} \]

\[ \text{then} \quad \text{remove-sensory-record(layer-R, entity(obstacle-6))} \]

- The modeling layer contains symbolic representations of the 'cognitivestate' of other entities in the agent's environment. The three layers communicate with each other and are embedded in a control framework, which use control rules.

\[ \text{Example:} \]

\[ \text{change-orientation( KerbAvoidance) \quad \text{if} \quad \text{separation(Kerb, Observer)} < \text{KerbThreshold} \quad \text{and} \quad \text{speed(Observer)} < 0 \quad \text{and} \quad \text{separation(Kerb, Observer)} < \text{KerbThreshold}}\]

\[ \text{then} \quad \text{change-orientation(KerbAvoidanceAngle)} \]

• The modeling layer contains symbolic representations of the 'cognitivestate' of other entities in the agent's environment.