Lecture 5: Reactive and Hybrid Architectures
An Introduction to Multiagent Systems

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

0.1 Reactive Architectures

There are many unsolved (some would say insoluble) problems associated with symbolic AI. It is not an idle speculation that many researchers have come to doubt the assumptions underpinning representations of the kind that symbolic AI proposes. Abstraction reasoning can be generated without explicit representations of the kind that symbolic AI proposes.

1. Situatedness and embodiment: Real intelligence is situated in the world, not in disembodied systems such as theorem provers or expert systems.

Rodney Brooks has put forward three of these:

Brooks—behaviour languages

He identifies two key ideas that have informed his research:

1. Situatedness and embodiment: Real intelligence is situated in the world, not in disembodied systems such as theorem provers or expert systems.

2. Abstract reasoning can be generated without explicit representations of the kind that symbolic AI proposes.

3. Intelligence is an emergent property of certain complex systems.

In this presentation, we start by reviewing the work of one of the most vocal critics of mainstream AI: Rodney Brooks.

Although united by a belief that the assumptions underpinning reactive architectures are in some sense wrong, reactive agents mainstream AI are in some sense embodied.

Researchers use many different techniques.

In this presentation, we start by reviewing the work of one of the most vocal critics of mainstream AI: Rodney Brooks.

Although united by a belief that the assumptions underpinning reactive architectures are in some sense wrong, reactive agents mainstream AI are in some sense embodied.

In this presentation, we start by reviewing the work of one of the most vocal critics of mainstream AI: Rodney Brooks.

Although united by a belief that the assumptions underpinning reactive architectures are in some sense wrong, reactive agents mainstream AI are in some sense embodied.

In this presentation, we start by reviewing the work of one of the most vocal critics of mainstream AI: Rodney Brooks.

Although united by a belief that the assumptions underpinning reactive architectures are in some sense wrong, reactive agents mainstream AI are in some sense embodied.

In this presentation, we start by reviewing the work of one of the most vocal critics of mainstream AI: Rodney Brooks.

Although united by a belief that the assumptions underpinning reactive architectures are in some sense wrong, reactive agents mainstream AI are in some sense embodied.

In this presentation, we start by reviewing the work of one of the most vocal critics of mainstream AI: Rodney Brooks.

Although united by a belief that the assumptions underpinning reactive architectures are in some sense wrong, reactive agents mainstream AI are in some sense embodied.

In this presentation, we start by reviewing the work of one of the most vocal critics of mainstream AI: Rodney Brooks.

Although united by a belief that the assumptions underpinning reactive architectures are in some sense wrong, reactive agents mainstream AI are in some sense embodied.

In this presentation, we start by reviewing the work of one of the most vocal critics of mainstream AI: Rodney Brooks.

Although united by a belief that the assumptions underpinning reactive architectures are in some sense wrong, reactive agents mainstream AI are in some sense embodied.

In this presentation, we start by reviewing the work of one of the most vocal critics of mainstream AI: Rodney Brooks.

Although united by a belief that the assumptions underpinning reactive architectures are in some sense wrong, reactive agents mainstream AI are in some sense embodied.

In this presentation, we start by reviewing the work of one of the most vocal critics of mainstream AI: Rodney Brooks.

Although united by a belief that the assumptions underpinning reactive architectures are in some sense wrong, reactive agents mainstream AI are in some sense embodied.

In this presentation, we start by reviewing the work of one of the most vocal critics of mainstream AI: Rodney Brooks.

Although united by a belief that the assumptions underpinning reactive architectures are in some sense wrong, reactive agents mainstream AI are in some sense embodied.

In this presentation, we start by reviewing the work of one of the most vocal critics of mainstream AI: Rodney Brooks.

Although united by a belief that the assumptions underpinning reactive architectures are in some sense wrong, reactive agents mainstream AI are in some sense embodied.

In this presentation, we start by reviewing the work of one of the most vocal critics of mainstream AI: Rodney Brooks.

Although united by a belief that the assumptions underpinning reactive architectures are in some sense wrong, reactive agents mainstream AI are in some sense embodied.

In this presentation, we start by reviewing the work of one of the most vocal critics of mainstream AI: Rodney Brooks.

Although united by a belief that the assumptions underpinning reactive architectures are in some sense wrong, reactive agents mainstream AI are in some sense embodied.
To illustrate his ideas, Brooks built some robots based on his subsumption architecture. A subsumption architecture is a hierarchy of task-accomplishing behaviours. Brooks’ robots are simple rule-like structures that ‘compete’ to exercise control over the agents. Lower layers represent more primitive kinds of behaviour (such as obstacle avoidance), and have precedence over layers further up the hierarchy. The resulting systems, in terms of the amount of computation they do, are extremely simple.

Some of the robots do tasks that would be impressive if they were accomplished by symbolic AI systems.

The Mars explorers system, using the subsumption architecture, achieves near-optimal cooperative performance in simulated rock-gathering on Mars. Domain-specific subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieving near-optimal cooperative performance in the subsumption architecture, using the subsumption architecture, achieved near-optimal cooperative performance in the subsumption architecture.
As a sophisticated approach is that of Rosenschein and Kaelbling. In their 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

- a deliberative one, containing a symbolic world model, which
- a reactive one, which is capable of reacting to events without
- symbolic AI; and
- develops plans and makes decisions in the way proposed by

An obvious approach is to build an agent out of two (or more) subsystems:

- marry classical and alternative approaches.
- They have suggested using hybrid systems, which attempt to
- building agents.
- hybrid architectures.

Many researchers have argued that neither a completely deliberative or completely reactive approach is suitable for

Often, the reactive component is given some kind of precedence over the deliberative one. This kind of structuring leads naturally to the idea of a layered architecture, in which the reactive component is given some kind of precedence over the deliberative one. The more expressive the agent specification language, the harder it is to compile it.

• The more expressive the agent specification language, the harder it is to compile it.

There are some deep theoretical results which say that after a certain expressiveness, the compilation simply can't be done.

Theoretically, some deep theoretical results which say that after a certain expressiveness, the compilation simply can't be done.

1. Hybrid Architectures
A key problem in such architectures is what kind control framework to embed the agent's subsystems in, to manage the interactions between the various layers. Sensory input and action output are each dealt with by at most one layer each.

- **Horizontal layering**: Suggestions as to what action to perform. In effect, each layer itself acts like an agent, producing action output.
- **Vertical layering**: Layers are each directly connected to the sensory input and action output. Interactions between the various layers.

**Horizontally layered control**

**Vertically layered control**

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.
The reactivelayer is implemented as a set of situation-action rules, *ala* subsumption architecture.

Example:

\[
\text{rule-1: kerb-avoidance} \quad \text{if} \quad \text{is-in-front(Kerb, Observer)} \quad \text{and} \quad \text{speed(Observer)} > 0 \quad \text{and} \quad \text{separation(Kerb, Observer)} < \text{KerbThreshold} \quad \text{then} \quad \text{change-orientation(KerbAvoidanceAngle)}
\]

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

Example:

\[
\text{censor-rule-1:} \quad \text{if} \quad \text{entity(obstacle-6) in perception-buffer} \quad \text{then} \quad \text{remove-sensory-record(layer-R, entity(obstacle-6))}
\]

The modellinglayer contains symbolic representations of the agent's environment. Cognitive states of other entities in the agent's environment are embedded in a control framework, which use control rules.

Example:

\[
\text{change-orientation(kerb-avoidance)} \quad \text{is} \quad \text{an} \quad \text{method} \quad \text{of} \quad \text{kerb-avoidance\_controller} \quad \text{and} \quad \text{results} \quad \text{in} \quad \text{change}(\text{kerb-avoidance\_angle})
\]