Robotics and Autonomous Systems

Lecture 16: Agent-based systems

Richard Williams

Department of Computer Science
University of Liverpool

Today

- We will start on the second part of the module.
  - Autonomous agents
- Things you will need for the second assignment.
- We will recap some of the basic ideas about agents from earlier in the module.
- Look at some aspects in more detail.
- Introduce the idea of the intentional stance

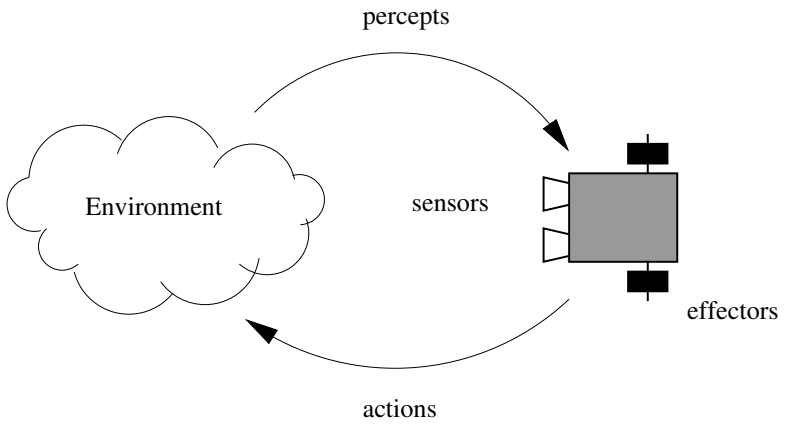
What is an agent?

- As we said before:

  An agent is a computer system that is situated in some environment, and that is capable of autonomous action in that environment in order to meet its delegated objectives.

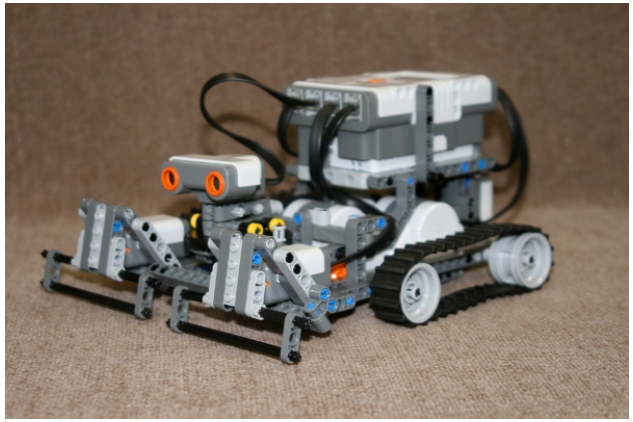
- Key word is “action”.
- It is all about decisions that relate to actions.

What is an agent?



- An agent has to choose what action to perform.
- An agent has to decide when to perform an action.

What is an agent?



- An agent has to choose what action to perform.
- An agent has to decide when to perform an action.

Intelligent agents

- Making good decisions requires the agent to be intelligent.

- Agent has to do the right thing.

Intelligent agents

- An intelligent agent is a computer system capable of flexible autonomous action in some environment.
  By flexible, we mean:
  - reactive;
  - pro-active;
  - social.
- All these properties make it able to respond to what is around it.
  (More on the next few slides).

Reactivity

- If a program's environment is guaranteed to be fixed, the program need never worry about its own success or failure
  - Program just executes blindly.
- Example of fixed environment: compiler.

Reactivity

- The real world is not like that:
  - Things change, information is incomplete.
- Many (most?) interesting environments are dynamic.
- Software is hard to build for dynamic domains: program must take into account possibility of failure
  - Ask itself whether it is worth executing!
- A reactive system is one that maintains an ongoing interaction with its environment, and responds to changes that occur in it . . .
- . . . in time for the response to be useful.

Reactivity

Proactiveness

- Reacting to an environment is easy
  - stimulus → response rules
- But we generally want agents to do things for us.
- Hence goal directed behaviour.
- Pro-activeness = generating and attempting to achieve goals; not driven solely by events; taking the initiative.
- Also: recognising opportunities.

Social Ability

- The real world is a multi-agent environment: we cannot go around attempting to achieve goals without taking others into account.
- Some goals can only be achieved with the cooperation of others.
- Similarly for many computer environments.
- Social ability in agents is the ability to interact with other agents (and possibly humans) via some kind of agent-communication language, and perhaps cooperate with others.

Properties of Environments

- Since agents are in close contact with their environment, the properties of the environment affect agents.
  - Also have a big effect on those of us who build agents.
- Common to categorise environments along some different dimensions.

Fully observable vs partially observable

- A fully observable environment is one in which the agent can obtain complete, accurate, up-to-date information about the environment's state.
- Such an environment is also called accessible.
- Most moderately complex environments (including, for example, the everyday physical world and the Internet) are only partially observable.
- Such environments are also known as non-accessible
- The more observable an environment is, the simpler it is to build agents to operate in it.

Deterministic vs non-deterministic

- A deterministic environment is one in which any action has a single guaranteed effect — there is no uncertainty about the state that will result from performing an action.
- The physical world can to all intents and purposes be regarded as non-deterministic.
- It is common to call environments stochastic if we quantify the non-determinism using probability theory.
- Non-deterministic environments present greater problems for the agent designer.

Episodic vs sequential

- In an episodic environment, the performance of an agent is dependent on a number of discrete episodes, with no link between the performance of an agent in different scenarios.
- An example of an episodic environment would be an assembly line where an agent had to spot defective parts.

Episodic vs sequential

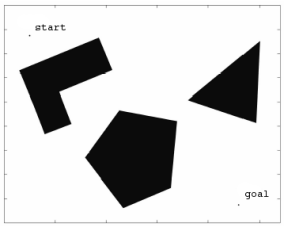
- Episodic environments are simpler from the agent developer's perspective because the agent can decide what action to perform based only on the current episode — it need not reason about the interactions between this and future episodes.
  - Relation to the Markov property.
- Environments that are not episodic are called either non-episodic or sequential. Here the current decision affects future decisions.
- Driving a car is sequential.

Static vs dynamic

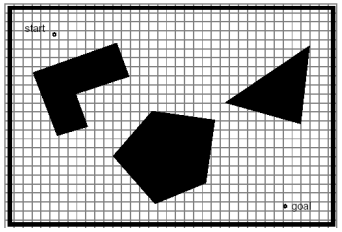
- A static environment is one that can be assumed to remain unchanged except by the performance of actions by the agent.
- A dynamic environment is one that has other processes operating on it, and which hence changes in ways beyond the agent's control.
- The physical world is a highly dynamic environment.
- One reason an environment may be dynamic is the presence of other agents.

Discrete vs continuous

- An environment is discrete if there are a fixed, finite number of actions and percepts in it.
- Continuous otherwise.
- As we have discussed, we often treat a continuous environment as a discrete environment for simplicity.



becomes



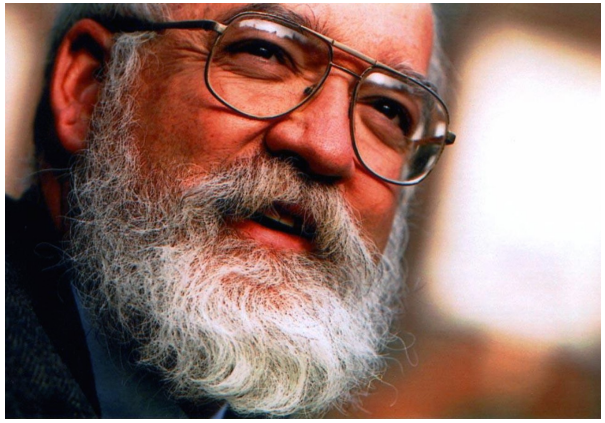
Agents as Intentional Systems

When explaining human activity, it is often useful to make statements such as the following:

1. David promised a green government because he believed it would make him popular.
2. George lowered income tax because he wanted to make his rich friends happy.
3. Nick raised tuition fees because he believed it was what David wanted.

- These statements make use of a folk psychology, by which human behaviour is predicted and explained through the attribution of attitudes.
- Attitudes such as believing and wanting (as in the above examples), hoping, fearing, and so on.
- The attitudes employed in such folk psychological descriptions are called the intentional notions.

Intentional systems



- The philosopher Daniel Dennett coined the term intentional system to describe entities “whose behaviour can be predicted by the method of attributing belief, desires and rational acumen”.

Intentional systems

- Dennett identifies different “grades” of intentional system:

  “A first-order intentional system has beliefs and desires (etc.) but no beliefs and desires about beliefs and desires. . .

  “ A second-order intentional system is more sophisticated; it has beliefs and desires (and no doubt other intentional states) about beliefs and desires (and other intentional states) — both those of others and its own.”

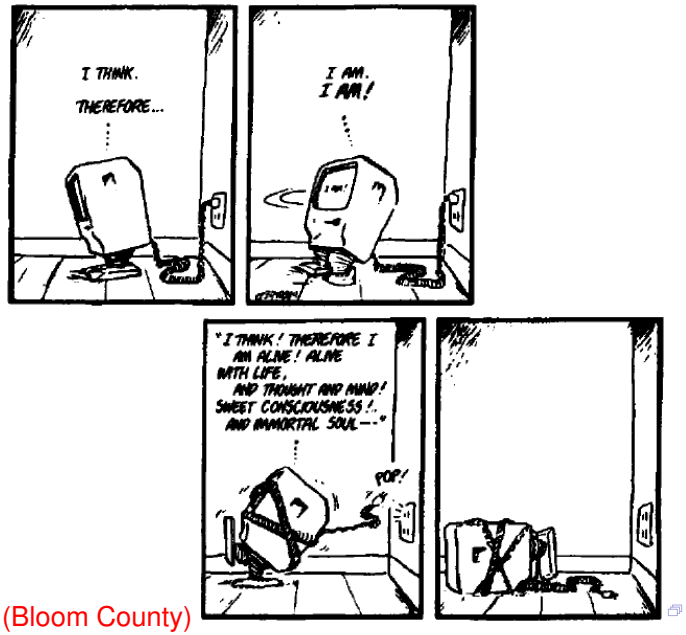
Grades of Intentional System

1. David promised a green government because he believed it would make him popular.
2. Nick raised tuition fees because he believed it was what David wanted.
3. Boris pretended to be an idiot because he believed it would make David believe that he didn’t want to be prime minister.

Intentional systems

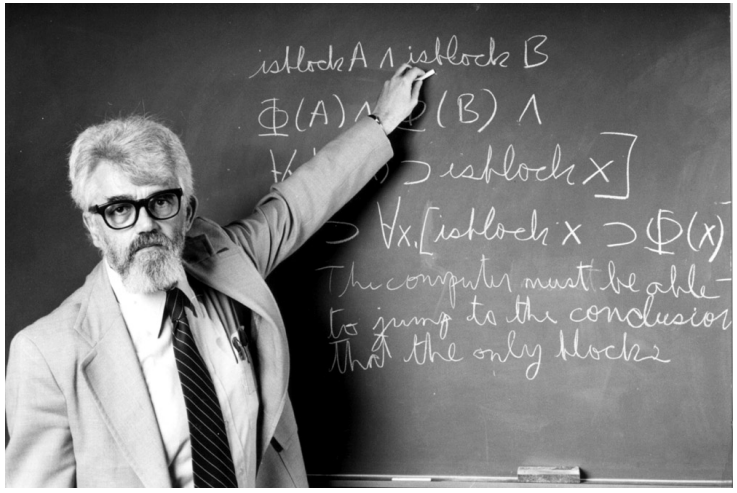
- Is it legitimate or useful to attribute beliefs, desires, and so on, to computer systems?

Intentional systems



(Bloom County)

Intentional systems



- John McCarthy argued that there are occasions when the intentional stance is appropriate.

Intentional systems

"To ascribe beliefs, free will, intentions, consciousness, abilities, or wants to a machine is legitimate when such an ascription expresses the same information about the machine that it expresses about a person. It is useful when the ascription helps us understand the structure of the machine, its past or future behaviour, or how to repair or improve it. It is perhaps never logically required even for humans, but expressing reasonably briefly what is actually known about the state of the machine in a particular situation may require mental qualities or qualities isomorphic to them. Theories of belief, knowledge and wanting can be constructed for machines in a simpler setting than for humans, and later applied to humans. Ascription of mental qualities is most straightforward for machines of known structure such as thermostats and computer operating systems, but is most useful when applied to entities whose structure is incompletely known."

Intentional systems

- What objects can be described by the intentional stance?
- As it turns out, more or less anything can. . . consider a light switch:

  > *"It is perfectly coherent to treat a light switch as a (very cooperative) agent with the capability of transmitting current at will, who invariably transmits current when it believes that we want it transmitted and not otherwise; flicking the switch is simply our way of communicating our desires."*

  (Yoav Shoham)
- But most adults would find such a description absurd!
  Why is this?

Intentional systems

Intentional systems

- The answer seems to be that while the intentional stance description is consistent,

  > *. . . it does not buy us anything, since we essentially understand the mechanism sufficiently to have a simpler, mechanistic description of its behaviour.*

  (Yoav Shoham)

Intentional systems

- Put crudely, the more we know about a system, the less we need to rely on animistic, intentional explanations of its behaviour.
- But with very complex systems, a mechanistic, explanation of its behaviour may not be practicable.
- As computer systems become ever more complex, low level explanations become impractical.
- We need more powerful abstractions and metaphors to explain their operation.
  The intentional stance is such an abstraction.

Intentional systems

- The intentional notions are thus abstraction tools, which provide us with a convenient and familiar way of describing, explaining, and predicting the behaviour of complex systems.

Abstractions

- Remember: most important developments in computing are based on new abstractions.
- Programming has progressed through:
  - machine code;
  - assembly language;
  - machine-independent programming languages;
  - sub-routines;
  - procedures & functions;
  - abstract data types;
  - objects;

  to
  - Agents, as intentional systems, represent a further, and increasingly powerful abstraction.

Abstractions

- Just as moving from machine code to higher level languages brings an efficiency gain, so does moving from objects to agents.
- A 2006 paper:

  > *S. Benfield, Making a Strong Business Case for Multiagent Technology, Invited Talk at AAMAS 2006.*

  claims that developing complex applications using agent-based methods leads to an average saving of 350% in development time (and up to 500% over the use of Java).

Abstractions

- So why not use the intentional stance as an abstraction tool in computing — to explain, understand, and, crucially, program computer systems?
- Three main points in favour of this idea:
  - Characterising agents
  - Nested representations
  - Post declarative systems
- (More on the next few slides)

Characterising Agents

- It provides us with a familiar, non-technical way of understanding and explaining agents.


Nested Representations

- It gives us the potential to specify systems that include representations of other systems.
- It is widely accepted that such nested representations are essential for agents that must cooperate with other agents.
- "If you think that Agent B knows x,
  then move to location L".

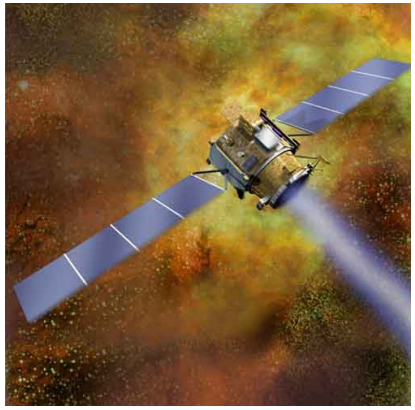
Post-Declarative Systems

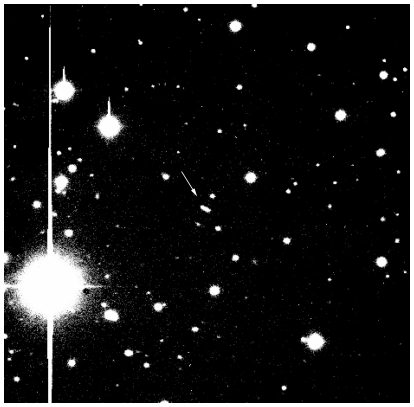
- In procedural programming, we say exactly what a system should do;
- In declarative programming, we state something that we want to achieve, give the system general info about the relationships between objects, and let a built-in control mechanism (e.g., goal-directed theorem proving) figure out what to do;
- With agents, we give a very abstract specification of the system, and let the control mechanism figure out what to do, knowing that it will act in accordance with some built-in theory of agency.

Post-Declarative Systems

- What is this built-in theory?
- Method of combining:
  - What you believe about the world.
  - What you desire to bring about
- Establish a set of intentions
- Then figure out how to make these happen.

BDI in space





DS1 seen 2.3 million miles from Earth

BDI

- How to do this is what we will get to in the next lecture.

Summary

- This lecture recapped the idea of an agent.
- Talked briefly about the environments that agents operate in.
- Introduced the intentional stance.
- Described why this is an important idea.