Contents

1 Designing intelligent agents

- an *agent* operates in a task environment:
  - *task*: the goal(s) the agent is trying to achieve
  - *environment*: that part of the real world or a computational system inhabited by the agent
- agent obtains information about the environment in the form of percepts
- agent changes the environment by performing actions to achieve its goals

2 From agent functions to agent programs

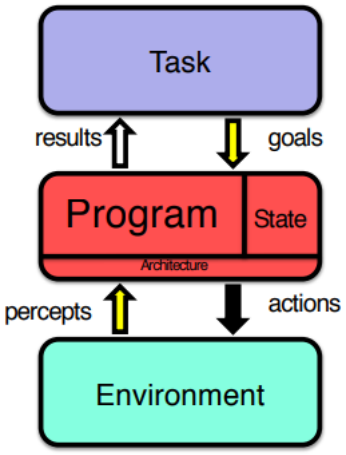
- the **behaviour** of an agent is described by an *agent function* (action selection function) which maps a goal and sequence of percepts to an action (and possibly results)
- agent programming is conventionally conceived as the problem of synthesising an agent function
- Difficult to design, because of unknown environments.

3 Agent architectures

- one way of making the agent programming problem more tractable is make use of the notion of an agent architecture
- the notion of an agent architecture is ubiquitous in the agent literature but is not well analysed
- often discussed in the context of an agent programming language or platform
- architecture is a blueprint for software agents and intelligent control systems, depicting the arrangement of components

3.1 Russell and Norvig view of an agent

- **program**: implements the agent function mapping from goals and percepts to actions (and results)
- **state**: includes all the internal representations on which the agent program operates
- **architecture**: computing device with sensors and actuators that runs the agent program

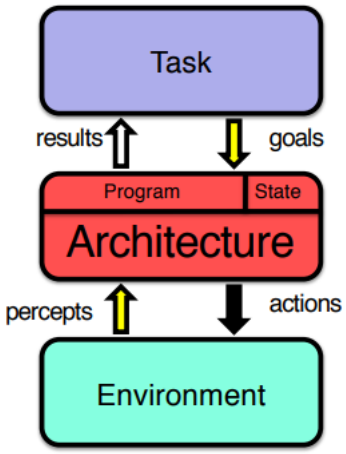


3.2 Architecture as a virutal machine

- the architecture defines a (real or virtual) machine which runs the agent program
- defines the atomic operations of the agent program and implicitly determines the components of the agent
- determines which operations happen automatically, without the agent program having to do anything
- e.g., the interaction between memory, learning and reasoning
- an architecture constrains kinds of agent programs we can write (easily)

3.3 Architectural view of an agent

- **program**: a function mapping from goals and percepts to actions (and results) expressed in terms of virtual machine operations
- **state**: the virtual machine representations on which the agent program operates
- **architecture**: a virtual machine that runs the agent program and updates the agent state



3.4 Hierarchies of virtual machines

In many agents we have a whole hierarchy of virtual machines, used without qualification, agent architecture means the most abstract architecture or the highest level virtual machine

4 Cognitive architecture

- agent architecture is also related to the notion of a cognitive architecture as used in artificial intelligence and cognitive science
- a *cognitive architecture* is an integrated system capable of supporting intelligence
- often used to denote models of **human reasoning**, e.g., ACT-R, SOAR
- in other cases no claims about psychological plausibility are made
- in this latter sense, cognitive architecture is more or less synonymous with agent architecture as used here

4.1 Properties of the architecture

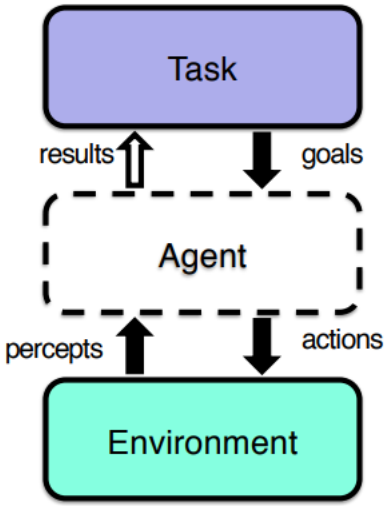
- an agent architecture can be seen as defining a class of agent programs
- just as individual agent programs have properties that make them more or less successful in a given task environment
- architectures (classes of programs) have higher-level properties that determine their suitability for a task environment
- choosing an appropriate architecture can make it much easier to develop an agent program for a particular task environment
- to program an agent which is successful in a given task environment, we must choose an architecture which is **appropriate** for that task environment

5 Task environments and architectures

To choose an architecture which is appropriate for a given task environment we must be able to characterise both task environments and architectures

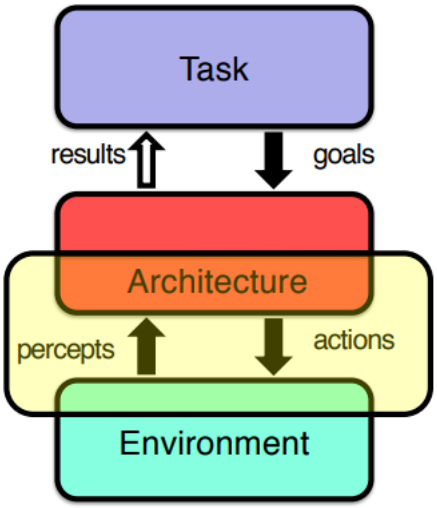
5.1 The task environment defines the problem

- we can sometimes **manipulate** the task and/or environment to make things easier
- e.g., increasing the contrast of objects to make things easier for a robots cameras
- however the task environment is usually given



5.2 Specifying task and environment

- the task specifies the **goals** the agent must achieve (and any results required)
- the agent (architecture) and environment **jointly determine**:
  - the information the agent can obtain (percepts)
  - the actions the agent can perform
- decomposition into task and environment is not always obvious



5.2.1 Specifying the task

- some tasks may come from the **agent itself** (autonomy)
- an agent may generate its own (top-level) **goals**
- e.g., may generate a goal to keep its battery charged
- or it may make use of its own results, e.g., in learning

5.2.2 Specifying the environment

- similarly, an agent may be **part of its own environment**
- e.g., if it has sensors monitoring its battery level
- agent may also form part of the environment of other agents
- e.g., in a multi-agent system

6 Task environment classification

- Will classify the agents **task environment** based on:
  - **task**: properties of the goals that must be achieved
  - **environment**: **properties** of the percepts and **actions** possible in the environment
- the following properties are illustrative onlyfor particular types of task environments other properties may be more useful
- we assume that it is possible to view the agent as an *intentional system*, i.e., that we can **ascribe a set of goals** to it which characterise its behaviour

7 Properties of the task

- **type of goal**: a goal to achieve a particular state in the environment is termed an **achievement goal**; a goal to maintain or preserve a state in the environment is termed a **maintenance goal**
- **number of goals**: if the agent must achieve multiple goals in parallel, we say the agent has multiple goals, otherwise it has a single goal
- **commitment to goals**: if a goal is only abandoned when it is achieved we say the agent is **strongly committed** to its goal(s); otherwise it is only **weakly committed**
- **utilities of goals**: if the reward for achieving each goal is the same, we say the agents goals have equal utility; otherwise its goals have differing utilities
- **constraints on how goals are achieved**: e.g., deadlines, resource bounds

8 Properties of the environment (percepts)

- **discrete / continuous**: if there are a limited number of distinct, clearly defined, states of the environment, the environment is discrete; otherwise it is continuous.
  - Analysing email is a continuous environment, where as a game of chess or checkers where there are a set number of moves is descrete.
- **observable / partially observable**: if it is possible to determine the complete state of the environment at each time point from the percepts it is observable; otherwise it is only partially observable.
  - Checker board is a complete environment because the agent has complete knowledge of the board, where as a game of cards is partially observable as oponent cards can't be seen
- **static / dynamics**: if the environment only changes as a result of the agents actions, it is static; otherwise it is dynamic
  - Empty office with no moving objects is a static environment, where as physical world would be dynamic
- **deterministic / nondeterministic**: if the future state of the environment can be predicted in principle given the current state and the set of actions which can be performed, it is deterministic; otherwise it is nondeterministic
  - Tic Tac Toe game is deterministic, where as a robot on mars is nondeterministic
- **single agent / multiple agents**: the environment may contain other agents which may be of the same kind as the agent, or of different kinds
  - A conveyor belt would have multiple agents, where as sewing machine robot would be a single agent environment.

9 Properties of the environment (actions)

- **fallibility of actions**: an action is infallible if it is guaranteed to produce its intended effects when executed in an environment which satisfies the preconditions of the action; otherwise it is fallible
- **utility of actions**: the utility of an action is the utility of the state which results from the actionthe action with maximum utility is correct
- **costs of actions**: the resource cost of performing the action an action is optimal if it is correct and there is no other correct action with lower cost
- **communicating actions**: an agent can be said to communicate with other agents in a meaningful way if it interacts with them via some kind of agent communication language

10 Architecture

10.1 Tasks and architecture

- if the agent has at least one *maintenance goal*, then the agents lifetime is potentially **unbounded**
- if the agent must pursue *multiple goals* in parallel, it needs some way of choosing which goal it should be working on at the moment
- if the agent is only *weakly committed* to its goals, it needs to be able to be able to decide when it should give up trying to pursue a goal
- if the *utilities* of the agents goals differ, then its commitment to a goal may change and/or the agent needs to be able to determine which goal it should be pursuing at any given time

10.2 Percepts and architecture

- discrete environments are usually **easier to represent** than continuous ones
- if the environment is only partially observable, *internal state* is required to keep track of the current state of the world
- if the environment is dynamic, then *the time it takes the agent to choose* which action to perform becomes important
- if the environment is nondeterministic, then any representation or reasoning capabilities will probably require a **notion of uncertainty**

10.3 Actions and architecture

- if actions are infallible, the agent **does not need to monitor** the environment to tell whether an action has succeeded
- if actions have varying costs and/or utilities and the agent wants to **minimise cost or maximise utility**, it needs to be able to choose between alternative courses of action
- if the agent can communicate with other agents, it must **decide** what to communicate and when, and what to do with information it receives from other agents

Reference section

agent function
: The agent function is a mathematical function that maps a sequence of perceptions into action

agent
: In artificial intelligence, an intelligent agent (IA) is an autonomous **entity**, which observes through sensors and acts upon an environment using actuators and directs its activity towards achieving goal