Intelligent Agents

Michal Jakob

Agent Technology Center

A3M33UI Spring 2010 - Lecture 9

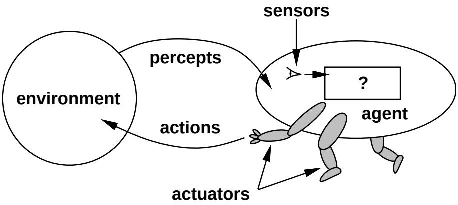
Outline

Outline

Intelligent Agent

- We adopt the view that intelligence is concerned with **rational action**.
- **Intelligent agent** should take the best possible action in a situation, i.e., the action which achieves the *best outcome* or, when there is uncertainty, the *best expected outcome*.
  - Although this is difficult and – in most real cases – impossible, . . .
  - . . . it is a very good design principle.
- We will study the problem of building agents that are intelligent in this sense.

Agents



Definition (Russel & Norvig)

An *agent* is anything that can *perceive* its *environment* (through its *sensors*) and *act* upon that environment (through its *effectors*).

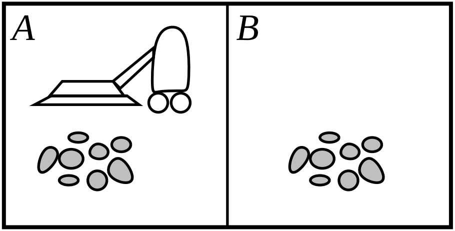
Agents

- Agents include humans, robots, softbots, controllers etc.
- Agent's behavior is described by the **agent function** maps percept sequences to actions

$$f : \mathscr{P} \mapsto \mathscr{A}$$

- The **agent program** runs on a physical architecture to produce f
- Key questions: What is the *right* function? Can it be implemented in a small agent program?

Example: Vacuum Cleaner World



- Percepts: location and contents, e.g. $[A, Dirty]$
- Actions: *Left*, *Right*, *Suck*, *NoOp*

Vacuum Cleaner Agent

Percept sequence	Action
$[A, Clean]$	$Right$
$[A, Dirty]$	$Suck$
$[B, Clean]$	$Left$
$[B, Dirty]$	$Suck$
$[A, Clean], [A, Clean]$	$Right$
$[A, Clean], [A, Dirty]$	$Suck$
$\vdots$	$\vdots$
$[A, Clean], [A, Clean], [A, Clean]$	$Right$
$[A, Clean], [A, Clean], [A, Dirty]$	$Suck$
$\vdots$	$\vdots$

Outline

Rational Behavior

What is the right behavior?

Definition (Rational Agent)

Rational agent chooses whichever action **maximizes the expected value of the performance measure** given the percept sequence to date and whatever bulit-in knowledge the agent has.

Rationality is relative and depends on four aspects:

- performance measure which defines the degree of success
- percept sequence (complete perceptual history)
- agent's knowledge about the environment
- actions available to the agent

Rational $\neq$ omniscient, rational $\neq$ clairvoyant $\Rightarrow$ rational $\neq$ successful

Outline

Specifying Task Environments

To design a rational agent, we must specify the **task environment (PEAS)**

- Performance measure
- Environment
- Actuators
- Sensors

Task environments define problems to which rational agents are the solutions.

PEAS Examples

Agent	Performance measure	Environment	Actuators	Sensors
Taxi driver	safe, fast, legal, comfortable trip, maximize profits	roads, other traffic, pedestrians, customers	steering, accelerator, brake, signal, horn, display	cameras, sonar, speedometer, GPS, engine sensors, keyboard
Part picking robot	percentage of parts in correct bins	conveyor belt with parts, bins	jointed arm and hand	camera, joint angle sensors
Refinery controller	maximize purity, yield, safety	refinery operators	valves, pumps, heaters, displays	temperature, pressure, chemical sensors

Properties of Environments

- **Fully observable vs. partially observable** – can agents obtain complete and correct information about the state of the world?
- **Deterministic vs. stochastic** – Do actions have guaranteed and uniquely defined effects?
- **Episodic vs. sequential** – Can agents decisions be made for different, independent episodes?
- **Static vs. dynamic** – Does the environment change by processes beyond agent control?
- **Discrete vs. continuous** – Is the number of actions and percepts fixed and finite?
- **Single-agent vs. multi-agent** – Does the behavior of one agent depends on the behavior of other agents?

Example Environments

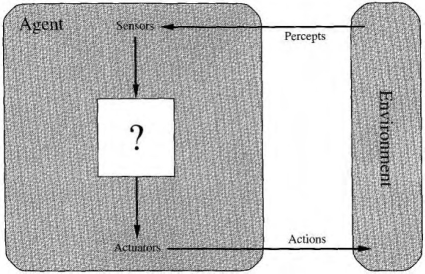
	Solitaire	Backgammon	Internet shopping	Taxi
Observable	Yes	Yes	No	No
Deterministic	Yes	No	Partly	No
Episodic	No	No	No	No
Static	Yes	Semi	Semi	No
Discrete	Yes	Yes	Yes	No
Single-agent	Yes	No	Yes (except auctions)	No

Outline

Implementing the Agent

How should one implement the agent function?

- So that the resulting behavior is (near) rational.
- So that its calculation is computationally tractable.



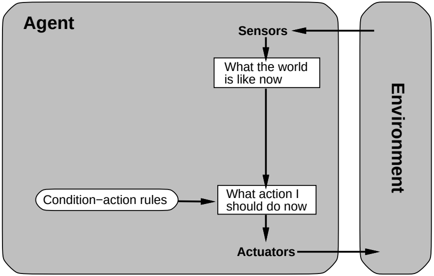
Hierarchy of Agents

Four basic types in order of increasing capability:

- simple reflex agents
- reflex agents with state
- goal-based agents
- utility-based agents

All these can be turned into learning agents.

Simple Reflex Agents



Simple reflex agent chooses the next action on the basis of the current percept

- Condition-action rules provide a way to present common regularities appearing in input/output associations
- Ex.: `if car-in-front-is-braking then initialize-braking`

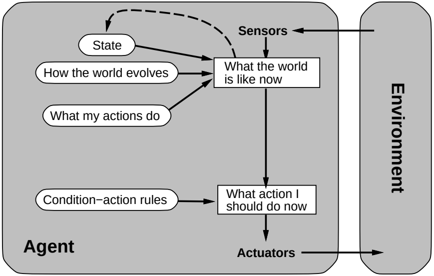
Adding State

Decision making is seldom possible based on the basis of a single percept

- the choice of action may depend on the entire percept history
- sensors do not necessarily provide access to the complete state of the environment

$\Rightarrow$ It can be advantageous to store information about the world in the agent.

Reflex Agents with State



Reflex agent with internal state keeps track of the world by extracting relevant information from percepts and storing it in its memory.

Towards Goal-based Agents

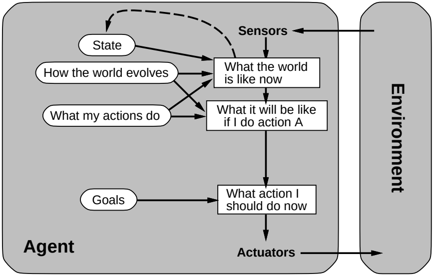
Knowing about the current state of the environment is not necessariliy enough for deciding what to do.

- The agent may need **goals** to distinguish which situations are desirable and which are not.
- Goal information can be combined with the agent's knowledge about the results of possible actions in order to choose an action leading to a goal.

Problem: goals are not necessarily achievable by a single action:

$\rightarrow$ **search and planning** are subfields of AI devoted to finding actions sequences that achieve the agent's goals.

Goal-based Agents



Goal-based agent utilizes goals and planning to determine which action to take.

Towards Utility-based Agents

Goals alone are not sufficient for decision making:

1. there may be multiple ways of achieving them;
2. agents may have several conflicting goals that cannot be achieved simultaneously.

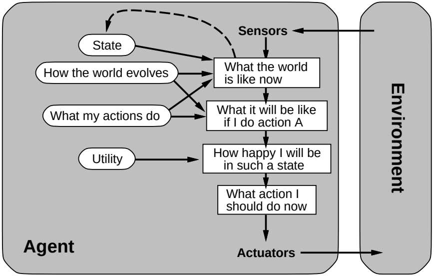
We introduce the concept of **utility**:

- utility is a function that maps a state onto a real number;
- if an agent prefers one world state to another state then the former state has higher utility for the agent.

Utility can be used for:

1. choosing the best plan
2. resolving conflicts among goals
3. estimating the successfulness of an agent if the outcomes of actions are uncertain

Utility-based Agents



Utility-based agent use the utility function to choose the most desirable action/course of actions to take.

Summary

- Agents interact with **environments** through **actuators** and **sensors**.
- The **agent function** describes what the agent does in all circumstances.
- The **performance measure** evaluates the sequence of environmental states.
- A **perfectly rational agent** maximizes expected performance.
- **Agent programs** implement (some) agent functions.
- **PEAS** descriptions define **task environments**.
- Environments are categorized along several dimensions.
- Several basic agent architectures exist.
- See *Russel and Norvig: Artificial Intelligence: A Modern Approach – Chapter 2* for more information.