Intelligent Agents

Chapter 2

Reminders

Assignment 0 (lisp refresher) due 1/28

Lisp/emacs/AIMA tutorial: 11-1 today and Monday, 271 Soda

Outline

♦ Agents and environments
♦ Rationality
♦ PEAS (Performance measure, Environment, Actuators, Sensors)
♦ Environment types
♦ Agent types

Outline

Vacuum-cleaner world

Percepts: location and contents, e.g., [A, Dirty]

Actions: Left, Right, Suck, NoOp

A vacuum-cleaner agent

<table>
<thead>
<tr>
<th>Percept sequence</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>[A, Clean]</td>
<td>Right</td>
</tr>
<tr>
<td>[A, Dirty]</td>
<td>Suck</td>
</tr>
<tr>
<td>[B, Clean]</td>
<td>Left</td>
</tr>
<tr>
<td>[B, Dirty]</td>
<td>Suck</td>
</tr>
<tr>
<td>[A, Clean], [A, Clean]</td>
<td>Right</td>
</tr>
<tr>
<td>[A, Clean], [A, Dirty]</td>
<td>Suck</td>
</tr>
</tbody>
</table>

function REFLEX-VACUUM-AGENT([location, status]) returns an action
if status = Dirty then return Suck
else if location = A then return Right
else if location = B then return Left

What is the right function?
Can it be implemented in a small agent program?
**Rationality**

Fixed performance measure evaluates the environment sequence
- one point per square cleaned up in time $T$?
- one point per clean square per time step, minus one per move?
- penalize for $k$ dirty squares?

A rational agent chooses whichever action maximizes the expected value of the performance measure given the percept sequence to date

Rational $\neq$ omniscient
- percepts may not supply all relevant information
Rational $\neq$ clairvoyant
- action outcomes may not be as expected
Hence, rational $\neq$ successful
Rational $\Rightarrow$ exploration, learning, autonomy

---

**Internet shopping agent**

Performance measure??
Environment??
Actuators??
Sensors??

---

**PEAS**

To design a rational agent, we must specify the task environment
Consider, e.g., the task of designing an automated taxi:

Performance measure??
Environment??
Actuators??
Sensors??

---

**Internet shopping agent**

Performance measure?? price, quality, appropriateness, efficiency
Environment?? current and future WWW sites, vendors, shippers
Actuators?? display to user, follow URL, fill in form
Sensors?? HTML pages (text, graphics, scripts)

---

**PEAS**

To design a rational agent, we must specify the task environment
Consider, e.g., the task of designing an automated taxi:

Performance measure?? safety, destination, profits, legality, comfort, …
Environment?? US streets/freeways, traffic, pedestrians, weather, …
Actuators?? steering, accelerator, brake, horn, speaker/display, …
Sensors?? video, accelerometers, gauges, engine sensors, keyboard, GPS, …

---

**Environment types**

<table>
<thead>
<tr>
<th>Observable??</th>
<th>Solitaire</th>
<th>Backgammon</th>
<th>Internet shopping</th>
<th>Taxi</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deterministic??</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Episodic??</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Static??</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Discrete??</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Single-agent??</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The environment type largely determines the agent design

The real world is (of course) partially observable, stochastic, sequential, dynamic, continuous, multi-agent
Agent types

Four basic types in order of increasing generality:
- simple reflex agents
- reflex agents with state
- goal-based agents
- utility-based agents

All these can be turned into learning agents

Reflex agents with state

Example

Simple reflex agents

Example

Goal-based agents
Utility-based agents

Agent

Environment

Sensors

State

What the world evolves

What my actions do

Utility

What it will be like if I do action A

How happy I will be in such a state

What action I should do now

Actuators

Chapter 2

Learning agents

Performance standard

Critical

Environment

Sensors

feedback

changes

Performance element

Problem generator

Learning element

learning goals

Agent

Actuators

Chapter 2

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 environment sequence

A perfectly rational agent maximizes expected performance

Agent programs implement (some) agent functions

PEAS descriptions define task environments

Environments are categorized along several dimensions:
- observable?
- deterministic?
- episodic?
- static?
- discrete?
- single-agent?

Several basic agent architectures exist:
- reflex, reflex with state, goal-based, utility-based