Rational Agents (Chapter 2)
Agents

• An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators
Example: Vacuum-Agent

- **Percepts:**
  Location and status, e.g., [A,Dirty]

- **Actions:**
  Left, Right, Suck, NoOp

```python
def 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
```
Rational agents

• For each possible percept sequence, a **rational agent** should select an action that is expected to maximize its **performance measure**, *given the evidence provided by the percept sequence and the agent’s built-in knowledge*

• **Performance measure (utility function):** An *objective* criterion for success of an agent's behavior

• **Expected utility:**

\[
EU(\text{action}) = \sum_{\text{outcomes}} P(\text{outcome} \mid \text{action})U(\text{outcome})
\]

• Can a rational agent make mistakes?
Back to Vacuum-Agent

- **Percepts:**
  - Location and status, e.g., [A,Dirty]

- **Actions:**
  - Left, Right, Suck, NoOp

```python
function 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
```

- Is this agent rational?
  - Depends on performance measure, environment properties
Specifying the task environment

- **PEAS**: Performance measure, Environment, Actuators, Sensors
- **P**: a function the agent is maximizing (or minimizing)
  - Assumed given
- **E**: a formal representation for *world states*
  - For concreteness, a tuple \((\text{var}_1=\text{val}_1, \text{var}_2=\text{val}_2, \ldots, \text{var}_n=\text{val}_n)\)
- **A**: actions that change the state according to a *transition model*
  - Given a state and action, what is the successor state (or distribution over successor states)?
- **S**: observations that allow the agent to infer the world state
  - Often come in very different form than the state itself
  - E.g., in tracking, observations may be pixels and state variables 3D coordinates
PEAS Example: Autonomous taxi

- **Performance measure**
  - Safe, fast, legal, comfortable trip, maximize profits
- **Environment**
  - Roads, other traffic, pedestrians, customers
- **Actuators**
  - Steering wheel, accelerator, brake, signal, horn
- **Sensors**
  - Cameras, LIDAR, speedometer, GPS, odometer, engine sensors, keyboard
Another PEAS example: Spam filter

• **Performance measure**
  – Minimizing false positives, false negatives

• **Environment**
  – A user’s email account, email server

• **Actuators**
  – Mark as spam, delete, etc.

• **Sensors**
  – Incoming messages, other information about user’s account
Environment types

- Fully observable vs. partially observable
- Deterministic vs. stochastic
- Episodic vs. sequential
- Static vs. dynamic
- Discrete vs. continuous
- Single agent vs. multi-agent
- Known vs. unknown
Fully observable vs. partially observable

• Do the agent's sensors give it access to the complete state of the environment?
  – For any given world state, are the values of all the variables known to the agent?

Source: L. Zettlemoyer
Deterministic vs. stochastic

• Is the next state of the environment completely determined by the current state and the agent’s action?
  – Is the transition model deterministic (unique successor state given current state and action) or stochastic (distribution over successor states given current state and action)?
  – **Strategic:** the environment is deterministic except for the actions of other agents
Episodic vs. sequential

- Is the agent’s experience divided into unconnected episodes, or is it a coherent sequence of observations and actions?
  - Does each problem instance involve just one action or a series of actions that change the world state according to the transition model?
Static vs. dynamic

- Is the world changing while the agent is thinking?
  - **Semidynamic**: the environment does not change with the passage of time, but the agent's performance score does
Discrete vs. continuous

- Does the environment provide a fixed number of distinct percepts, actions, and environment states?
  - Are the values of the state variables discrete or continuous?
  - Time can also evolve in a discrete or continuous fashion
Single-agent vs. multiagent

- Is an agent operating by itself in the environment?
Known vs. unknown

- Are the rules of the environment (transition model and rewards associated with states) known to the agent?
  - Strictly speaking, not a property of the environment, but of the agent’s state of knowledge
## Examples of different environments

<table>
<thead>
<tr>
<th>Observable</th>
<th>Deterministic</th>
<th>Episodic</th>
<th>Static</th>
<th>Discrete</th>
<th>Single agent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Word jumble solver</td>
<td>Fully</td>
<td>Deterministic</td>
<td>Episodic</td>
<td>Static</td>
<td>Single</td>
</tr>
<tr>
<td>Chess with a clock</td>
<td>Fully</td>
<td>Strategic</td>
<td>Sequential</td>
<td>Semidynamic</td>
<td>Multi</td>
</tr>
<tr>
<td>Scrabble</td>
<td>Partially</td>
<td>Stochastic</td>
<td>Sequential</td>
<td>Static</td>
<td>Multi</td>
</tr>
<tr>
<td>Autonomous driving</td>
<td>Partially</td>
<td>Stochastic</td>
<td>Sequential</td>
<td>Dynamic</td>
<td>Multi</td>
</tr>
</tbody>
</table>

Examples:
- Word jumble solver
- Chess with a clock
- Scrabble
- Autonomous driving
Preview of the course

• **Deterministic environments:** search, constraint satisfaction, logic
  – Can be sequential or episodic

• **Multi-agent, strategic environments:** minimax search, games
  – Can also be stochastic, partially observable

• **Stochastic environments**
  – **Episodic:** Bayesian networks, pattern classifiers
  – **Sequential, known:** Markov decision processes
  – **Sequential, unknown:** reinforcement learning