Artificial Intelligence

Chap 2 – Intelligent Agents

EECS 26' 劉祐廷 111060013

1. For each of the following agents, develop a PEAS description of the task environment:

(a) Autonomous Mars rover

Performance	Environment
1. Successfully navigates the Martian surface without collisions 2. Successfully finishes tasks that were assigned 3. Successfully transmits data and images back to Earth	1. Extreme temperature 2. Extreme radiation 3. Rock terrain and dust 4. Low gravity
Actuators	**Sensors**
1. Wheels for moving 2. Robotic arms for collecting and analyzing rock samples 3. Drill for collecting rock samples 4. Antennas for communication with people on Earth 5. Cameras for taking pictures 6. Balancer for avoiding rollover	1. Cameras for detecting the terrain or other information of space 2. Lidar for obstacles detection 3. Thermometer for detecting the temperature 4. Gyroscope for detecting rotation to avoid rollover

(b) Smart Traffic Control System (managing traffic lights for optimal flow)

Performance	Environment
1. Average speed 2. Numbers of vehicles in a certain area 3. Traffic accident rate	1. Rush hours
Actuators	**Sensors**
1. Traffic light	1. Camera for capturing visualized data to analyze condition of traffic flow

2. For each of the above agent, characterize the environment according to the properties given in Section 2.3.2 (slides pp. 18–23), and select a suitable agent design.

Properties	Autonomous Mars rover	Smart Traffic Control System
Observable	Partially	Partially
Deterministic	Stochastic	Stochastic
Episodic	Sequential	Sequential
Static	Dynamic	Dynamic
Discrete	Continuous	Continuous
Single-agent	Single	Single
Agent Design	Goal-based	Utility-based

Autonomous Mars rover

There must be some conditions that are never considered. Thus, it is partially observable. In navigation tasks, to optimize route in every task, it should take actions that depend on some previous data such as map and terrain, which means that it is stochastic and sequential. The environment changes while making decisions, so it is dynamic. In the time aspect, it is no doubt that it is a continuous time agent because everything that happens in real world is in continuous time. Its goal is to complete the tasks which are assigned by people, and it is not affected by other agents. Therefore, it is a single-agent system and a goal-based agent design.

Smart Traffic Control System

This system is partially observable, stochastic and sequential because it doesn't need only current vision data to analyze the traffic flow in the current moment but other information such as historical traffic flow data to help it make decisions. The environment changes while the agent makes decisions and things happen in continuous time. Thus, it is dynamic and continuous time. Regarding a set of traffic lights in a certain area as a whole agent, it is a single agent system because two systems that are far away don't need to affect each other. Its goal is to maximize average speed in a certain area; therefore, it is utility-based.

3. Consider the vacuum cleaner, for which the agent is penalized one point for each movement.
   (a) Can a simple reflex agent be perfectly rational for this environment? Explain.
   If it is not penalized by one point for each movement, it is perfectly rational. However, if it is penalized for each movement and only considers current percept, it is hard to reach an optimal solution without route planning. Therefore, a simple reflex agent can't be perfectly rational for this environment.

(b) What about a reflex agent with the state? Design such an agent.

The agent can percept whether the current square is dirty or not, knows its location (left, right, forward, backward) and can remember the status of the squares that it has passed. But it doesn't have any prior knowledge to plan the route. It might go to a lot of squares that are unnecessarily visited. Therefore, due to the penalty, it is not perfectly rational.

```
Initialize vacuum with all squares unvisited
A square has two statues that store in the memory of a vacuum:
unvisited, visited

while (there is at least a square is unvisited) {
      if (current_square is dirty) vacuum->suck();
      vacuum->visit(current_square);
      if (forward_square is unvisited) go_forward();
      if (backward_square is unvisited) go_backward();
      if (left_square is unvisited) go_left();
      if (right_square is unvisited) go_right();
}
```

(c) How do your answers to (a) and (b) change if the agent's percepts give it the clean/dirty status of every square in the environment?

Both the agents in (a) and (b) can create a model that calculates the shortest path for cleaning all dirty squares, and the vacuum can move follow the shortest path, which makes these agents perfectly rational.