Intelligent Agent

Chapter 2

2.3 For each of the following assertions, say whether it is true or false and support your answer with examples or counterexamples where appropriate.

a. An agent that senses only partial information about the state cannot be perfectly rational.

b. There exist task environments in which no pure reflex agent can behave rationally.

c. There exists a task environment in which every agent is rational.

d. The input to an agent program is the same as the input to the agent function.

e. Every agent function is implementable by some program/machine combination.

f. Suppose an agent selects its action uniformly at random from the set of possible actions. There exists a deterministic task environment in which this agent is rational.

g. It is possible for a given agent to be perfectly rational in two distinct task environments.

h. Every agent is rational in an unobservable environment.

i. A perfectly rational poker-playing agent never loses.

a. False. Perfect rationality refers to the ability to make good decisions given the sensor information received.

b. True. A pure reflex agent ignores previous percepts, so cannot obtain an optimal state estimate in a partially observable environment. For example, correspondence chess is played by sending moves; if the other player's move is the current percept, a reflex agent could not keep track of the board state and would have to respond to, say, "a4" in the same way regardless of the position in which it was played.

c. True. For example, in an environment with a single state, such that all actions have the same reward, it doesn't matter which action is taken. More generally, any environment that is reward-invariant under permutation of the actions will satisfy this property.

d. False. The agent function, notionally speaking, takes as input the entire percept sequence up to that point, whereas the agent program takes the current percept only.

e. False. For example, the environment may contain Turing machines and input tapes and the agent's job is to solve the halting problem; there is an agent function that specifies the right answers, but no agent program can implement it. Another example would be an agent function that requires solving intractable problem instances of arbitrary size in constant time.

f. True. This is a special case of (c); if it doesn't matter which action you take, selecting randomly is rational.

g. True. For example, we can arbitrarily modify the parts of the environment that are unreachable by any optimal policy as long as they stay unreachable.

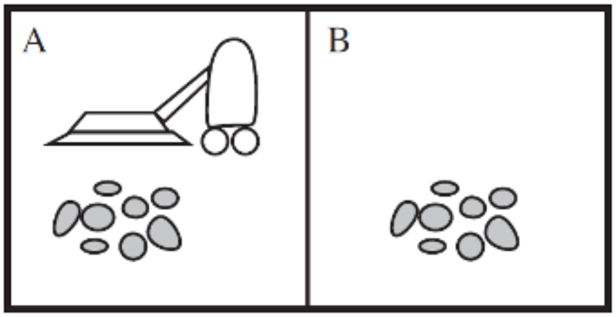
h. False. Some actions are stupid—and the agent may know this if it has a model of the environment—even if one cannot perceive the environment state.

i. False. Unless it draws the perfect hand, the agent can always lose if an opponent has better cards. This can happen for game after game. The correct statement is that the agent's expected winnings are nonnegative.

2.4 For each of the following activities, give a PEAS description of the task environment and characterize it in terms of the properties listed in Section 2.3.2.

- Playing soccer.
- Exploring the subsurface oceans of Titan.
- Shopping for used AI books on the Internet.
- Playing a tennis match.
- Practicing tennis against a wall.
- Performing a high jump.
- Knitting a sweater.
- Bidding on an item at an auction.

Performance Measures, Environment, Actuators, Sensors.
Many Answers are valid.
E.g. Playing Soccer:
P: My team goals minus opponent's team goals
E: Playground, Goals, Ball, Referee, etc
A: Legs, Head, Body, etc
S: Eyes, Ears - If Robot: Camera, Microphone.



2.10 Consider a modified version of the vacuum environment in Exercise 2.8, in which the agent is penalized one point for each movement.

a. Can a simple reflex agent be perfectly rational for this environment? Explain.

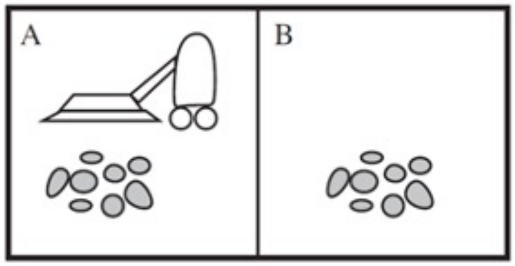
b. What about a reflex agent with state? Design such an agent.

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?

a. No, the agent will either keep moving to clean the environment or will never move, since it cannot remember whether it has already cleaned other locations or not.

b. It will become rational since it will be able to stop as soon as it finishes cleaning everywhere.

c. In this case, a simple reflex agent can be perfectly rational. The agent can consist of a table with eight entries, indexed by percept, that specifies an action to take for each possible state. After the agent acts, the world is updated and the next percept will tell the agent what to do next. For larger environments, constructing a table is infeasible. Instead, the agent could run one of the optimal search algorithms in Chapters 3 and 4 and execute the first step of the solution sequence. Again, no internal state is required, but it would help to be able to store the solution sequence instead of recomputing it for each new percept.



2.11 Consider a modified version of the vacuum environment in Exercise 2.8, in which the geography of the environment—its extent, boundaries, and obstacles—is unknown, as is the initial dirt configuration. (The agent can go *Up* and *Down* as well as *Left* and *Right*.)

a. Can a simple reflex agent be perfectly rational for this environment? Explain.

b. Can a simple reflex agent with a *randomized* agent function outperform a simple reflex agent? Design such an agent and measure its performance on several environments.

c. Can you design an environment in which your randomized agent will perform poorly? Show your results.

d. Can a reflex agent with state outperform a simple reflex agent? Design such an agent and measure its performance on several environments. Can you design a rational agent of this type?

a. Because the agent does not know the geography and perceives only location and local dirt, and cannot remember what just happened, it will get stuck forever against a wall when it tries to move in a direction that is blocked—that is, unless it randomizes.

b. One possible design cleans up dirt and otherwise moves randomly. This is fairly close to what the RoombaTM vacuum cleaner does (although the Roomba has a bump sensor and randomizes only when it hits an obstacle). It works reasonably well in nice, compact environments. In maze-like environments or environments with small connecting passages, it can take a very long time to cover all the squares. (The performance measurement needs the function to be implemented and run on different environments).

c. In maze-like environments or environments with small connecting passages, it can take a very long time to cover all the squares. Draw anything like that.

d. A reflex agent with state can build a map (see Chapter 4 for details). An online depth-first exploration will reach every state in time linear in the size of the environment; therefore, the agent can do much better than the simple reflex agent. The question of rational behavior in unknown environments is a complex one but it is worth encouraging students to think about it. We need to have some notion of the prior probability distribution over the class of environments; call this the initial belief state. Any action yields a new percept that can be used to update this distribution, moving the agent to a new belief state. Once the environment is completely explored, the belief state collapses to a single possible environment. Therefore, the problem of optimal exploration can be viewed as a search for an optimal strategy in the space of possible belief states. This is a well-defined, if horrendously intractable, problem. Chapter 21 discusses some cases where optimal exploration is possible. Another concrete example of exploration is the Minesweeper computer game (see Exercise 7.22). For very small Minesweeper environments, optimal exploration is feasible although the belief state update is nontrivial to explain.

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Crossword puzzle	FO	SA	D	NE	S	Dis
Chess with a clock	F	MA	D	NE	D	Dis
Poker	PO	MA	S	NE	S	Dis
Backgammon	FO	MA	S	NE	S	Dis
Taxi driving	PO	MA	S	NE	D	Cont
Medical diagnosis	PO	MA	S	NE		
Image analysis						
Part-picking robot						
Refinery controller						
Interactive English tutor						

Crossword puzzle: Fully - Single - Deterministic - Sequential - Static - Discrete

Chess with a clock: Fully - Multi - Deterministic - Sequential - Semi - Discrete

Poker: Partially - Multi - Stochastic - Sequential - Static - Discrete

Backgammon: Fully - Multi - Stochastic - Sequential - Static - Discrete

Taxi driving: Partially - Multi - Stochastic - Sequential - Dynamic - Continuous

Medical diagnosis: Partially - Single - Stochastic - Sequential - Dynamic - Continuous

Image analysis: Fully - Single - Deterministic - Episodic - Semi - Continuous

Part-picking robot: Partially - Single - Stochastic - Episodic - Dynamic - Continuous

Refinery controller: Partially - Single - Stochastic - Sequential - Dynamic - Continuous

Interactive English tutor: Partially - Multi - Stochastic - Sequential - Dynamic - Discrete