

- Rationality

What is rational at any given time depends on four things:

The performance measure that defines the criterion of success.

The agent's prior knowledge of the environment.

The actions that the agent can perform.

The agent's percept sequence to date.

1. Task environment: The nature of the task

environment directly affects the appropriate design for the agent program.

2. Four basic kinds of agent programs that embody

the principles underlying almost all intelligent systems:

- Simple reflex agents;
- Model-based reflex agents;
- Goal-based agents; and
- Utility-based agents.

3. Model-based agent

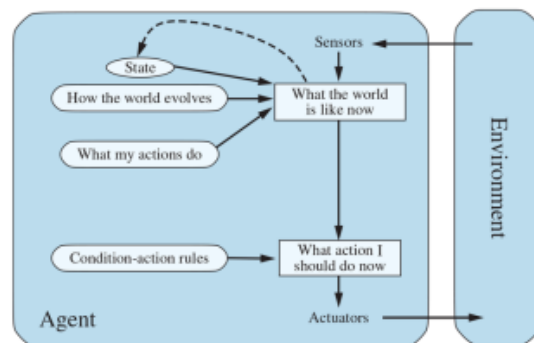
The transition model and sensor model allow an agent to keep track of the state of the world—to the extent possible given the limitations of the agent's sensors. An agent that uses such models is called a model-based agent.

```
function MODEL-BASED-REFLEX-AGENT(percept) returns an action
  persistent: state, the agent's current conception of the world state
               transition_model, a description of how the next state depends on
               the current state and action
               sensor_model, a description of how the current world state is reflected
               in the agent's percepts
               rules, a set of condition-action rules
               action, the most recent action, initially none

  state ← UPDATE-STATE(state, action, percept, transition_model, sensor_model)
  rule ← RULE-MATCH(state, rules)
  action ← rule.ACTION
  return action
```

A model-based reflex agent. It keeps track of the current state of the world, using an internal model. It then chooses an action in the same way as the reflex agent.

Figure 2.11

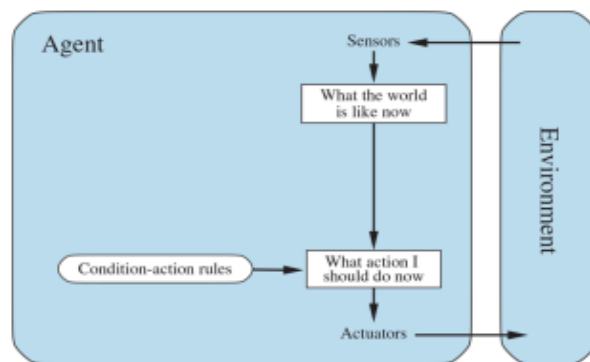


A model-based reflex agent.

Figure 2.12

4. Simple reflex agents

Figure 2.9



Schematic diagram of a simple reflex agent. We use rectangles to denote the current internal state of the agent's decision process, and ovals to represent the background information used in the process.

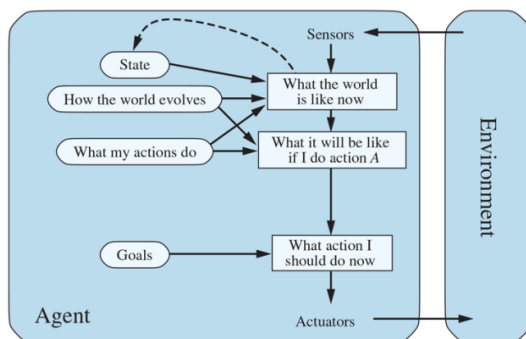
Figure 2.8

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

The agent program for a simple reflex agent in the two-location vacuum environment. This program implements the agent function tabulated in Figure 2.3.

Simple reflex agent

5. Goal-based agents



A model-based, goal-based agent. It keeps track of the world state as well as a set of goals it is trying to achieve, and chooses an action that will (eventually) lead to the achievement of its goals.

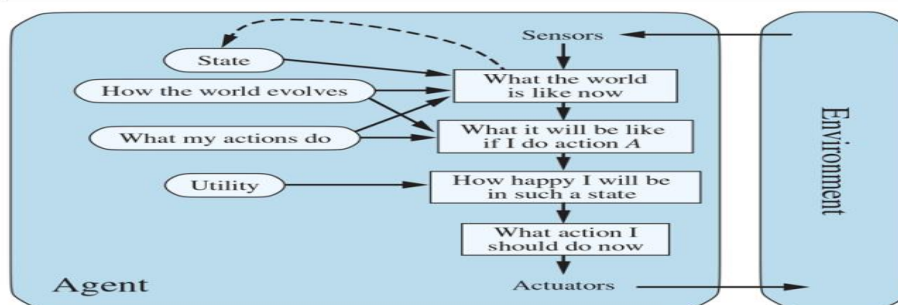
Goal

Depends on Goal

Searching and Planning takes place

6. Utility-based agents

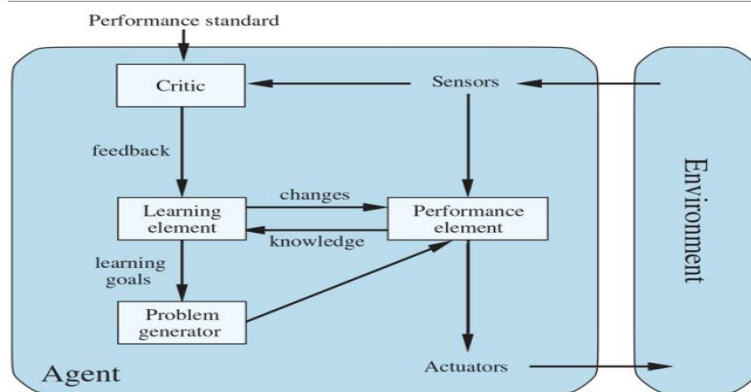
Figure 2.14



A model-based, utility-based agent. It uses a model of the world, along with a utility function that measures its preferences among states of the world. Then it chooses the action that leads to the best expected utility, where expected utility is computed by averaging over all possible outcome states, weighted by the probability of the outcome.

7. Learning agent

Figure 2.1.1



A general learning agent. The "performance element" box represents what we have previously considered to be the whole agent program. Now, the "learning element" box gets to modify that program to improve its performance.

Critic