Intelligent Agents

Artificial Intelligence

School of Computing
Universiti Teknologi Malaysia

Outline

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

Agents

An agent is an entity that perceives and acts

An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators

Human agent:

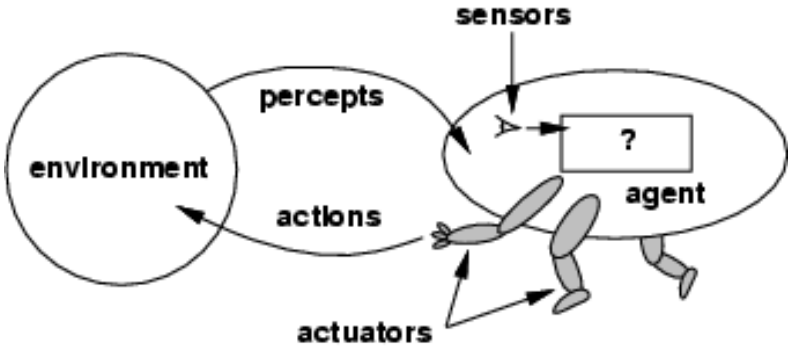
eyes, ears, and other organs for sensors;
hands,legs, mouth, and other body parts for actuators

Robotic agent:

cameras and infrared range finders for sensors;
various motors for actuators

- Percept refers to the agent's perceptual input at a given instant
- Percept Sequence is a complete history of percepts.

Agents and environments



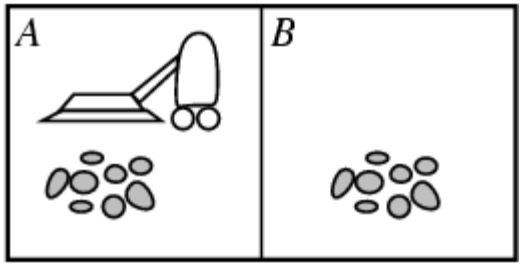
- The agent function maps from percept histories to actions:

$$[f: \mathcal{P}^{\star} \rightarrow \mathcal{A}]$$

- The agent program runs on the physical architecture to produce *f*

- agent = architecture + program

Vacuum-cleaner world



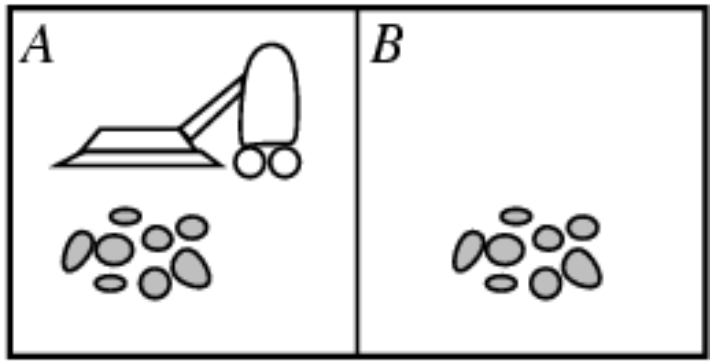
Percepts: location and contents, e.g., [A,Dirty]

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

A vacuum-cleaner function

Percept Sequence	Action
[A, Clean]	Right
[A, Dirty]	Suck
[B, Clean]	Left
[B, Dirty]	Suck
[A, Clean], [A, Clean]	Right
[A, Clean], [A, Dirty]	Suck
…	

The Vacuum-cleaner world



function REFLEX-VACUUM-AGENT ([*location, status*]) return an action

if *status == Dirty* then return *Suck*

else if *location == A* then return *Right*

else if *location == B* then return *Left*

Rational agents

An agent should strive to "do the right thing", based on what it can perceive and the actions it can perform. The right action is the one that will cause the agent to be most successful

Performance measure: An objective criterion for success of an agent's behavior

E.g., performance measure of a vacuum-cleaner agent could be amount of dirt cleaned up, amount of time taken, amount of electricity consumed, amount of noise generated, etc.

Rational agents

Rational Agent: 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 whatever built-in knowledge the agent has.

Rational agents

Rationality is distinct from omniscience (all-knowing with infinite knowledge)

Agents can perform actions in order to modify future percepts so as to obtain useful information (information gathering, exploration)

An agent is autonomous if its behavior is determined by its own experience (with ability to learn and adapt)

Rationality

The proposed definition requires:

- Information gathering/exploration
  - To maximize future rewards
- Learn from percepts
  - Extending prior knowledge
- Agent autonomy
  - Compensate for incorrect prior knowledge

The nature of environments: PEAS

PEAS: Performance measure, Environment, Actuators, Sensors

Must first specify the setting for intelligent agent design

Consider, e.g., the task of designing an automated taxi driver:

- Performance measure
- Environment
- Actuators
- Sensors

The nature of environments: PEAS

Must first specify the setting for intelligent agent design

Consider, e.g., the task of designing an automated taxi driver:

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

Environment types

- Fully observable (vs. partially observable): An agent's sensors give it access to the complete state of the environment at each point in time.
- Deterministic (vs. stochastic): The next state of the environment is completely determined by the current state and the action executed by the agent. (If the environment is deterministic except for the actions of other agents, then the environment is strategic)
- Episodic (vs. sequential): The agent's experience is divided into atomic "episodes" (each episode consists of the agent perceiving and then performing a single action), and the choice of action in each episode depends only on the episode itself.

Environment types

- Static (vs. dynamic): The environment is unchanged while an agent is deliberating. (The environment is semidynamic if the environment itself does not change with the passage of time but the agent's performance score does)

- Discrete (vs. continuous): A limited number of distinct, clearly defined percepts and actions.

- Single agent (vs. multiagent): An agent operating by itself in an environment.

Environment types

	Chess with a clock	Chess without a clock	Taxi driving
Fully observable	Yes	Yes	No
Deterministic	Strategic	Strategic	No
Episodic	No	No	No
Static	Semi	Yes	No
Discrete	Yes	Yes	No
Single agent	No	No	No

- The environment type largely determines the agent design
- The real world is (of course) partially observable, stochastic, sequential, dynamic, continuous, multi-agent

Environment types

- The simplest environment is
  - Fully observable, deterministic, episodic, static, discrete and single-agent.
- Most real situations are:
  - Partially observable, stochastic, sequential, dynamic, continuous and multi-agent.

Environment Types

Give 1 example of AI application that characterized as an agent for the following environment types:

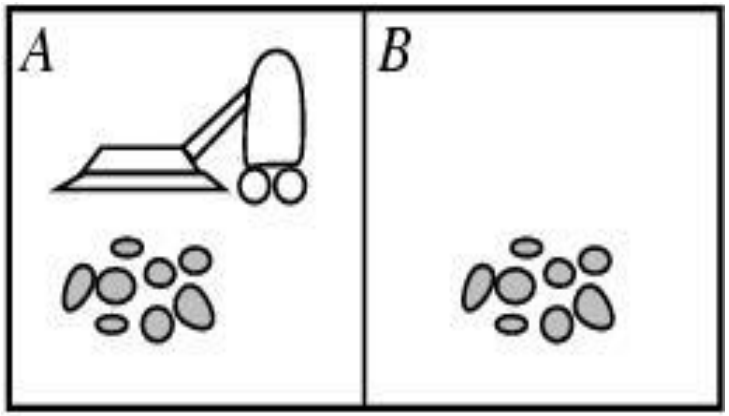
No	Environment type	Example of AI application
1.	Fully observable	
2.	Deterministic	
3.	Episodic	
4.	Static	
5.	Discrete	
6.	Single-agent	

Agent types

Four basic types in order of increasing generality:

- Simple reflex agents
- Model-based reflex agents
- Goal-based agents
- Utility-based agents

The vacuum-cleaner world



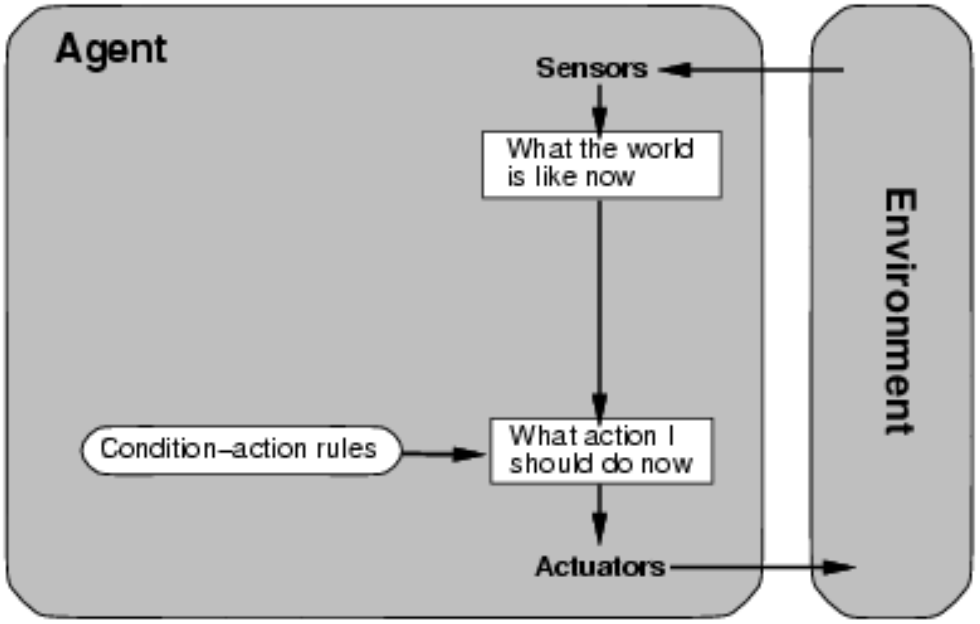
function REFLEX-VACUUM-AGENT ([*location, status*]) return an action

if *status* == *Dirty* then return *Suck*

else if *location* == *A* then return *Right*

else if *location* == *B* then return *Left*

Simple reflex agents



- Select action on the basis of *only the current* percept.
  - E.g. the vacuum-agent
- Large reduction in possible percept/action situations(next page).
- Implemented through *condition-action rules*
  - If dirty then suck

Simple reflex agent

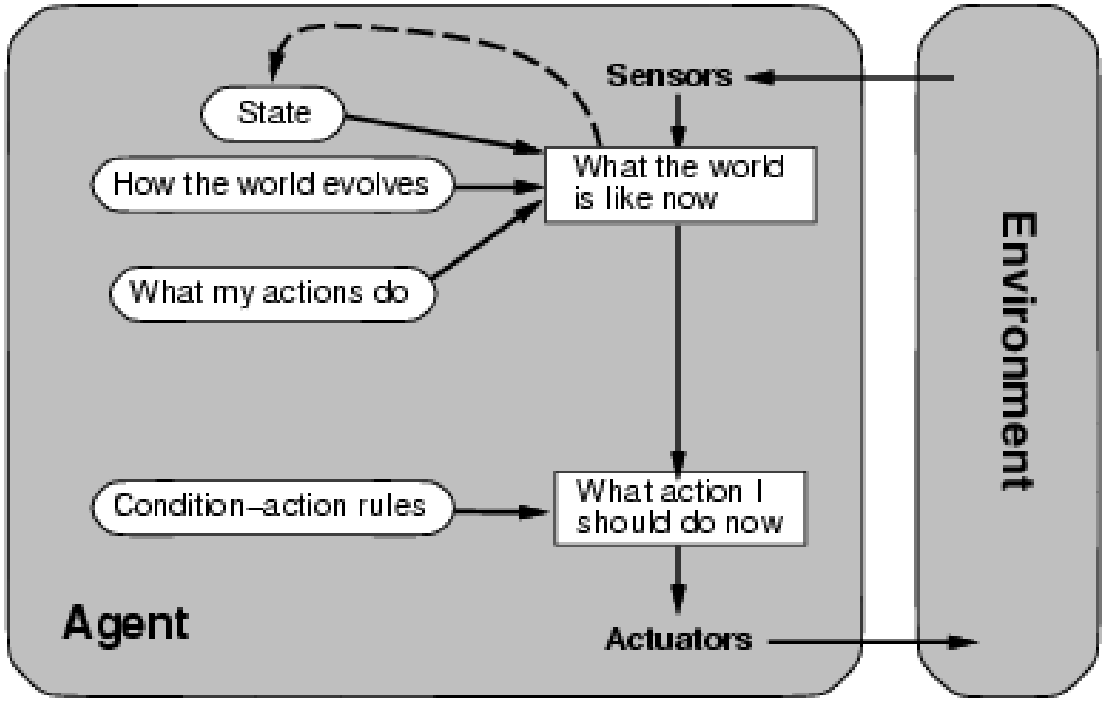
function SIMPLE-REFLEX-AGENT(*percept*) **returns** an action

static: *rules*, a set of condition-action rules

state ← INTERPRET-INPUT(*percept*)
rule ← RULE-MATCH(*state*, *rules*)
action ← RULE-ACTION[*rule*]
return *action*

Will only work if the environment is fully observable otherwise infinite loops may occur.

Model-based reflex agents



- To tackle *partially observable* environments.
  - Maintain internal state
- Over time update state using world knowledge
  - How does the world change.
  - How do actions affect world.

  ⇒ *Model of World*

Model-based reflex agents

function REFLEX-AGENT-WITH-STATE(*percept*) **returns** an action

static: *rules*, a set of condition-action rules

state, a description of the current world state

action, the most recent action.

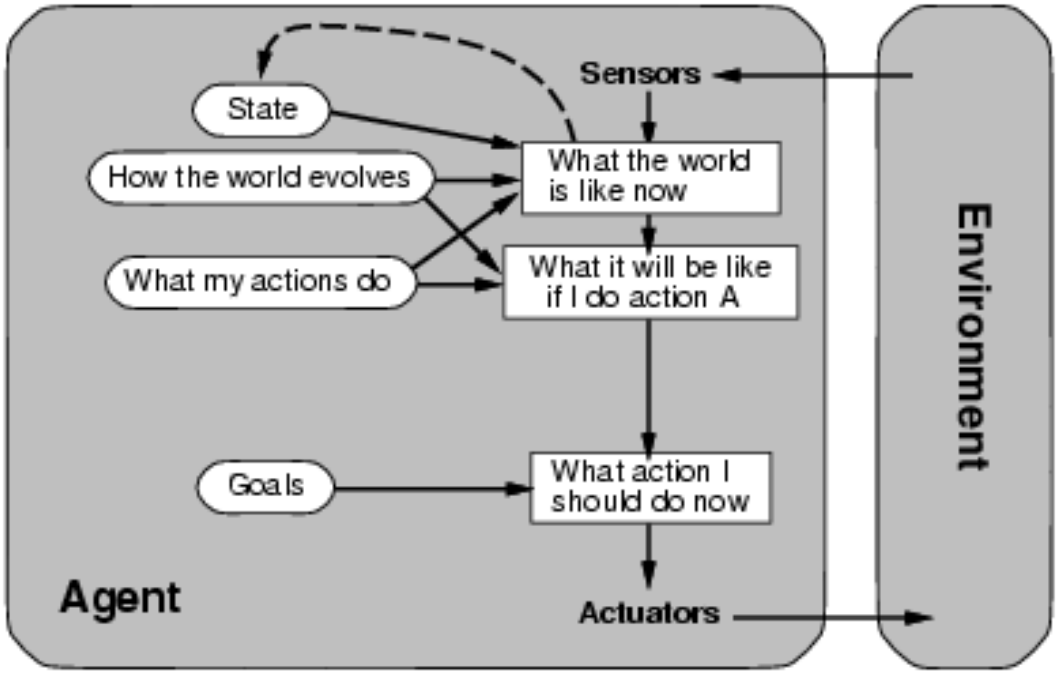
state ← UPDATE-STATE(*state*, *action*, *percept*)

rule ← RULE-MATCH(*state*, *rule*)

action ← RULE-ACTION[*rule*]

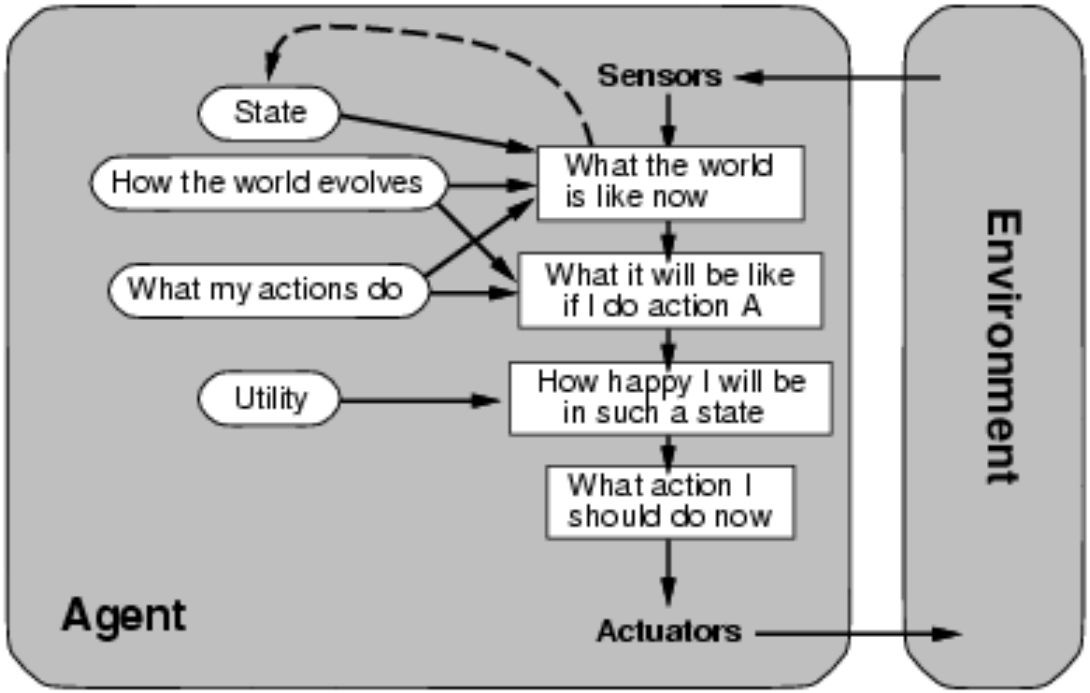
return *action*

Goal-based agents



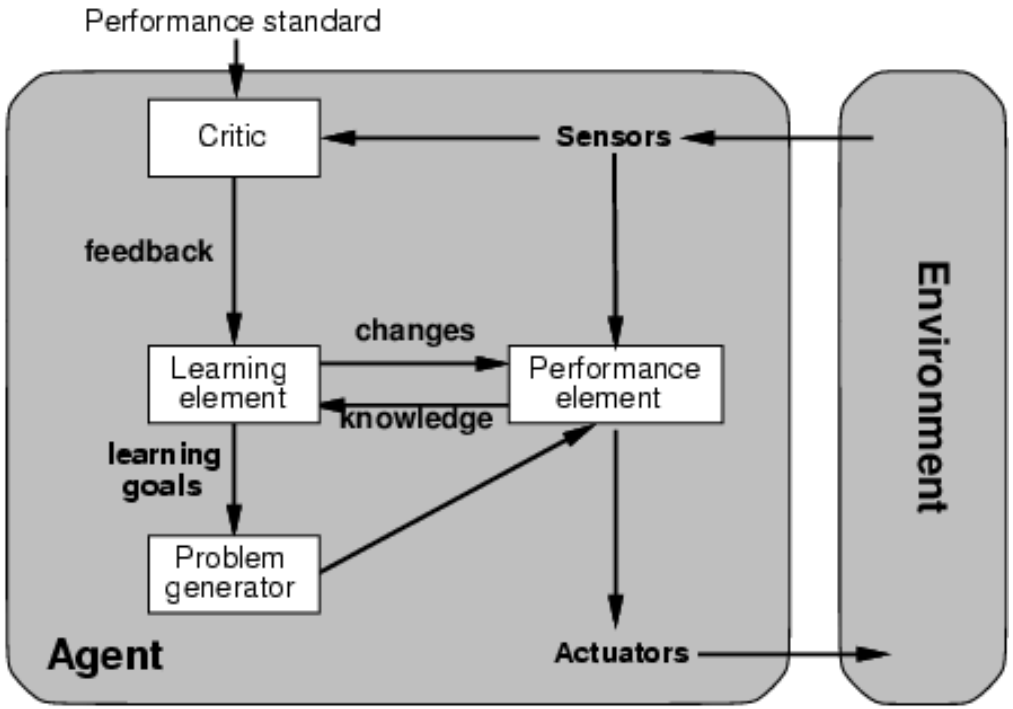
- The agent needs a goal to know which situations are *desirable*.
  - Things become difficult when long sequences of actions are required to find the goal.
- Typically investigated in **search** and **planning** research.
- Major difference: future is taken into account
- Is more flexible since knowledge is represented explicitly and can be manipulated.

Utility-based agents



- Certain goals can be reached in different ways.
  - Some are better, have a higher utility.
- Utility function maps a (sequence of) state(s) onto a real number.
- Improves on goals:
  - Selecting between conflicting goals
  - Select appropriately between several goals based on likelihood of success.

Learning agents



- All previous agent-programs describe methods for selecting *actions*.
  - Yet it does not explain the origin of these programs.
  - Learning mechanisms can be used to perform this task.
  - Teach them instead of instructing them.
  - Advantage is the robustness of the program toward initially unknown environments.

Agent Types

Give 1 example of AI application that characterized the following agent types:

No	Agent type	Example of AI application
1.	Model-based	
2.	Learning-based	
3.	Utility-based	
4.	Learning agent	

Summary: Intelligent Agents

- An **agent** perceives and acts in an environment, has an architecture, and is implemented by an agent program.
- Task environment – **PEAS (P**erformance**, E**nvironment, **A**ctuators, **S**ensors**)**
- The most challenging environments are inaccessible, nondeterministic, dynamic, and continuous.
- An **ideal agent** always chooses the action which maximizes its expected performance, given its percept sequence so far.
- An **agent program** maps from percept history to action and updates internal state.
  - **Reflex agents** respond immediately to percepts.
    - simple reflex agents
    - model-based reflex agents
  - **Goal-based agents** act in order to achieve their goal(s).
  - **Utility-based agents** maximize their own utility function.
- Agents can improve their performance through **learning.**