Môn học:

Trí tuệ nhân tạo

Chương 2: Intelligent Agents

Khoa Công nghệ Thông tin

Nội dung

I. Giới thiệu

- Agent
- Rational agent
- Ideal rational agent

II. Môi trường tác vụ

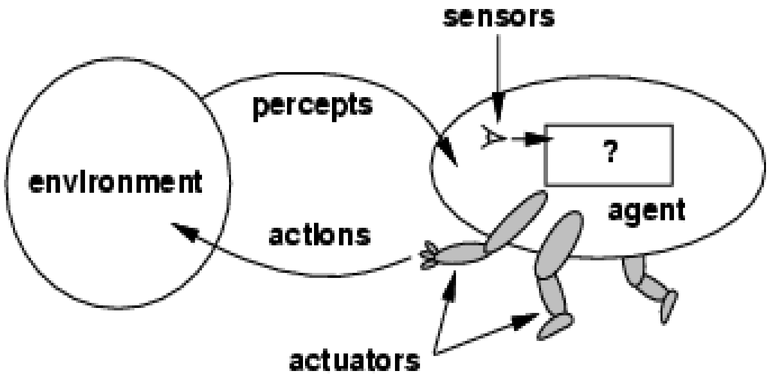
- PEAS
- Ví dụ

III. Tính chất của môi trường tác vụ

IV. Cấu trúc agent

Agents

- Một agent
  - Cảm nhận môi trường (environment) thông qua các cảm biến (sensors)
  - Hoạt động trong môi trường thông qua các bộ truyền động (actuators)



Ví dụ:

- Human agent:
  - Cảm biến: Mắt, tai
  - Bộ truyền động: Tay, chân, miệng.
- Robotic agent:
  - Cảm biến: Camera, bộ dò hồng ngoại
  - Bộ truyền động: Các động cơ

Rational agents (Agent xử lý thông minh/có lý trí)

- Một rational agent là dạng agent làm đúng việc cần làm (do the right thing) dựa trên những gì mà nó có thể cảm nhận (perceive) và các hành động mà nó có thể thực hiện.
- Lưu ý: rational agent đưa ra action dựa trên instant đã cảm nhận được, hoặc lịch sử toàn bộ instant đã từng cảm nhận, chứ không phải trên những instant chưa xuất hiện hay đã xuất hiện nhưng agent chưa cảm nhận được

Rational agents (Agent xử lý thông minh/có lý trí)

- Sự thông minh/có lý trí tại một thời điểm của agent phụ thuộc và 4 yếu tố:
  - Performance measure:
  - Percept sequence to date (sensors)
  - Prior environment knowledge,
  - Actions

Ideal Rational agents

- Một Ideal Rational Agent là với mỗi chuỗi cảm quan, nó lựa chọn một hành động mà mong đợi sẽ cực đại hóa được độ đo hiệu năng (performance measure), dựa trên:
  - Minh chứng quan sát được (instant perceived)
  - Kiến thức của nó (kể cả tự xây dựng hoặc thu thập được)

Nội dung

I. Giới thiệu

- Agent
- Rational agent
- Ideal rational agent

II. **Môi trường tác vụ**

- PEAS
- Ví dụ

III. Tính chất của môi trường tác vụ

IV. Cấu trúc agent

Xác định môi trường tác vụ

- Để có thể thiết kế một rational agent, chúng ta cần xác định rõ môi trường tác vụ (task environment) của agent đó.
- Gồm 4 yếu tố, viết tắt là PEAS:
  - **Performance measure**: Goals/desires the agent should try to achieve
  - **Environment**: in which the agent exists
  - **Actuators**: Actions which may react the environment
  - **Sensors**: Percepts/observations of the environment

Xác định môi trường tác vụ

Performance measure: Goals/desires the agent should try to achieve

- Độ đo hiệu năng được định nghĩa dựa trên trạng thái của enviroment chứ không phải dựa trên trạng thái của agent (people as “sour grapes”)
- Độ đo hiệu năng nên được định nghĩa theo mục tiêu muốn đạt được, không nên định nghĩa theo cách thức agent hoạt động (vacuum machine)

Ví dụ PEAS

- Agent: **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
- Agent: **Intenet shopping**
  - Performance measure: price, quality, appropriateness, ..
  - Environment: current and future WWW sites, vendors, shippers
  - Actuators: display to user, follow URL, fill in form
  - Sensors: HTML pages (text, graphics, scripts)

Ví dụ PEAS

- Agent: **Medical diagnosis system**
  - Performance measure: Healthy patient, minimize costs, lawsuits.
  - Environment: Patient, hospital, staff.
  - Actuators: Screen display (questions, tests, diagnoses, treatments, referrals)
  - Sensors: Keyboard (entry of symptoms, patient's answers)
- Agent: **Part-picking robot**
  - Performance measure: Percentage of parts in correct bins
  - Environment: Conveyor belt with parts, bins.
  - Actuators: Jointed arm and hand.
  - Sensors: Camera, joint angle sensors.

Ví dụ PEAS

- Agent: **Interactive English tutor**
  - Performance measure: Maximize student's score on test
  - Environment: Set of students
  - Actuators: Screen display (exercises, suggestions, corrections)
  - Sensors: Keyboard (typed words)
- Agent: **Chess program**
  - Performance measure: Win game
  - Environment: Opponent, Game board
  - Actuators: Next move
  - Sensors: Current board state

Nội dung

I. Giới thiệu

- Agent
- Rational agent
- Ideal rational agent

II. Môi trường tác vụ

- PEAS
- Ví dụ

III. **Tính chất của môi trường tác vụ**

IV. Cấu trúc agent

III.Tính chất của môi trường tác vụ

- **Fully observable** (vs. **partially observable**)
  - An agent's sensors give it access to the complete state of the environment at any point in time.
  - Noisy and inaccurate sensors can result in partially observable environments.
- **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
  - Randomness and chance are common causes non-deterministic environments.

III.Tính chất của môi trường tác vụ

- **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
    - – the choice of action in each episode does not depend on the actions in prior episodes.
  - – Games are often sequential requiring one to think ahead.
- **Static** (vs. **dynamic**):
  - – The environment is unchanged while an agent is deliberating (between the time of perceiving and acting).
  - – The environment is semidynamic if the environment itself does not change with the passage of time but the agent's performance score does
  - – Time is an important factor in dynamic environments, since perceptions can become "stale".

III.Tính chất của môi trường tác vụ

- **Discrete** (vs. **continuous**):
  - An environment is discrete if there are a limited number of distinct, clearly-defined states of the world which limits range of possible percepts and actions.

- **Single agent** (vs. **multiagent**):
  - An environment is multiagent if more than one agents effect the each other's performance.
  - Multiagent environments can be competitive and/or cooperative.

III.Tính chất của môi trường tác vụ

Fully vs. partially observable: an environment is full observable when the sensors can detect all aspects that are relevant to the choice of action.

	Solitaire	Image-Analysis system	Intenet shopping	Taxi
Observable??	FULL	FULL	PARTIAL	PARTIAL
Deterministic??				
Episodic??				
Static??				
Discrete??				
Single-agent??				

III.Tính chất của môi trường tác vụ

Deterministic vs. stochastic: if the next environment state is completely determined by the current state and the executed action then the environment is deterministic.

	Solitaire	Image-Analysis system	Intenet shopping	Taxi
Observable??	FULL	FULL	PARTIAL	PARTIAL
Deterministic??	YES	YES	YES	NO
Episodic??				
Static??				
Discrete??				
Single-agent??				

III.Tính chất của môi trường tác vụ

Episodic vs. sequential: In an episodic environment the agent’s experience can be divided into atomic steps where the agents perceives and then performs a single action. The choice of action depends only on the episode itself

	Solitaire	Image-Analysis system	Intenet shopping	Taxi
Observable??	FULL	FULL	PARTIAL	PARTIAL
Deterministic??	YES	YES	YES	NO
Episodic??	NO	YES	NO	NO
Static??				
Discrete??				
Single-agent??				

III.Tính chất của môi trường tác vụ

Static vs. dynamic: If the environment can change while the agent is choosing an action, the environment is dynamic. Semi-dynamic if the agent's performance changes even when the environment remains the same.

	Solitaire	Image-Analysis system	Intenet shopping	Taxi
Observable??	FULL	FULL	PARTIAL	PARTIAL
Deterministic??	YES	YES	YES	NO
Episodic??	NO	YES	NO	NO
Static??	YES	SEMI	SEMI	NO
Discrete??				
Single-agent??				

III.Tính chất của môi trường tác vụ

Discrete vs. continuous: This distinction can be applied to the state of the environment, the way time is handled and to the percepts/actions of the agent. .

	Solitaire	Image-Analysis system	Intenet shopping	Taxi
Observable??	FULL	FULL	PARTIAL	PARTIAL
Deterministic??	YES	YES	YES	NO
Episodic??	NO	YES	NO	NO
Static??	YES	SEMI	SEMI	NO
Discrete??	YES	NO	YES	NO
Single-agent??				

III.Tính chất của môi trường tác vụ

Single vs. multi-agent: Does the environment contain other agents who are also maximizing some performance measure that depends on the current agent’s actions?

	Solitaire	Image-Analysis system	Intenet shopping	Taxi
Observable??	FULL	FULL	PARTIAL	PARTIAL
Deterministic??	YES	YES	YES	NO
Episodic??	NO	YES	NO	NO
Static??	YES	SEMI	SEMI	NO
Discrete??	YES	NO	YES	NO
Single-agent??	YES	NO	NO	NO

III.Tính chất của môi trường tác vụ

- Môi trường đơn giản nhất là
  - Quan sát đầy đủ (Fully observable), xác định (deterministic), phân chia từng phần (episodic), tĩnh (static), rời rạc (discrete) và đơn agent (single-agent).
- Môi trường thực tế thường là:
  - Quan sát một phần (Partially observable), ngẫu nhiên (stochastic), không chia từng phần(sequential), động (dynamic), liên tục (continuous), và đa agent (multi-agent).

Nội dung

I. Giới thiệu

- Agent
- Rational agent
- Ideal rational agent

II. Môi trường tác vụ

- PEAS
- Ví dụ

III. Tính chất của môi trường tác vụ

IV. **Cấu trúc agent**

IV. Cấu trúc Agent

- Có thể chia thành 4 nhóm hệ thống thông minh sau
  - Simple reflex agents
  - Model-based reflex agents
  - Goal-based agents
  - Utility-based agents
- Tất cả các loại agent này có thể được chuyển đổi (convert) thành dạng learning agents.

Agent types: Simple Table-Based Reflex

- use a table lookup where each percept is matched to an action

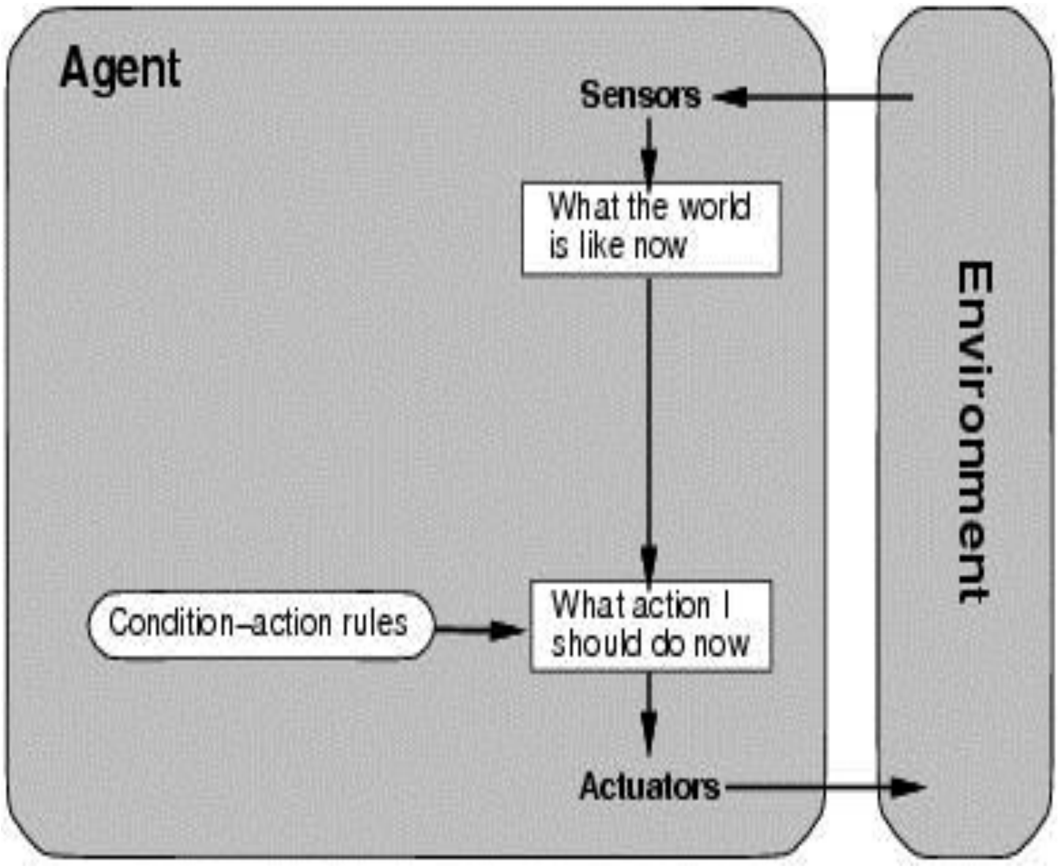
- Problems/Limitations?
  - table may be too big to generate and store
  - not adaptive to changes in the environment; instead table must be updated
  - can't make actions conditional
  - reacts only to current percept; no history kept

Table-lookup agent

```
Function TABLE-DRIVEN_AGENT(percept) returns an action
        static: percepts, a sequence initially empty
                table, a table of actions, indexed by percept sequence

        append percept to the end of percepts
        action ← LOOKUP(percepts, table)
        return action
```

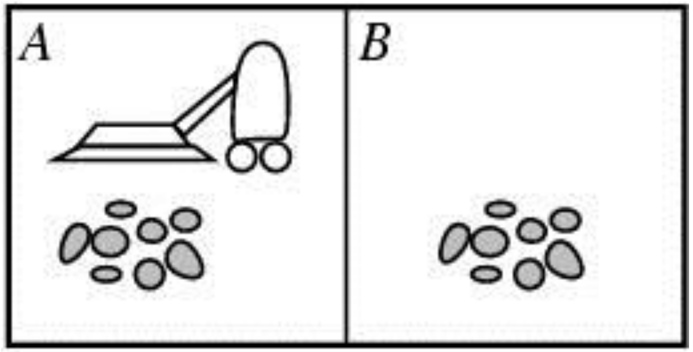
- Drawbacks:
  - Huge table
  - Take a long time to build the table
  - No autonomy
  - Even with learning, need a long time to learn the table entries

Agent types: Simple Rule-Based Reflex



- Select action on the basis of *only the current* percept.
- No need to consider all percepts
- Implemented through *condition-action rules*
  - **If** dirty **then** suck
- Can adapt to changes in the environment by adding rules
- Problems/Limitations?
  - reacts only to current percept; no knowledge of non-perceptual parts of the current state

The vacuum-cleaner world



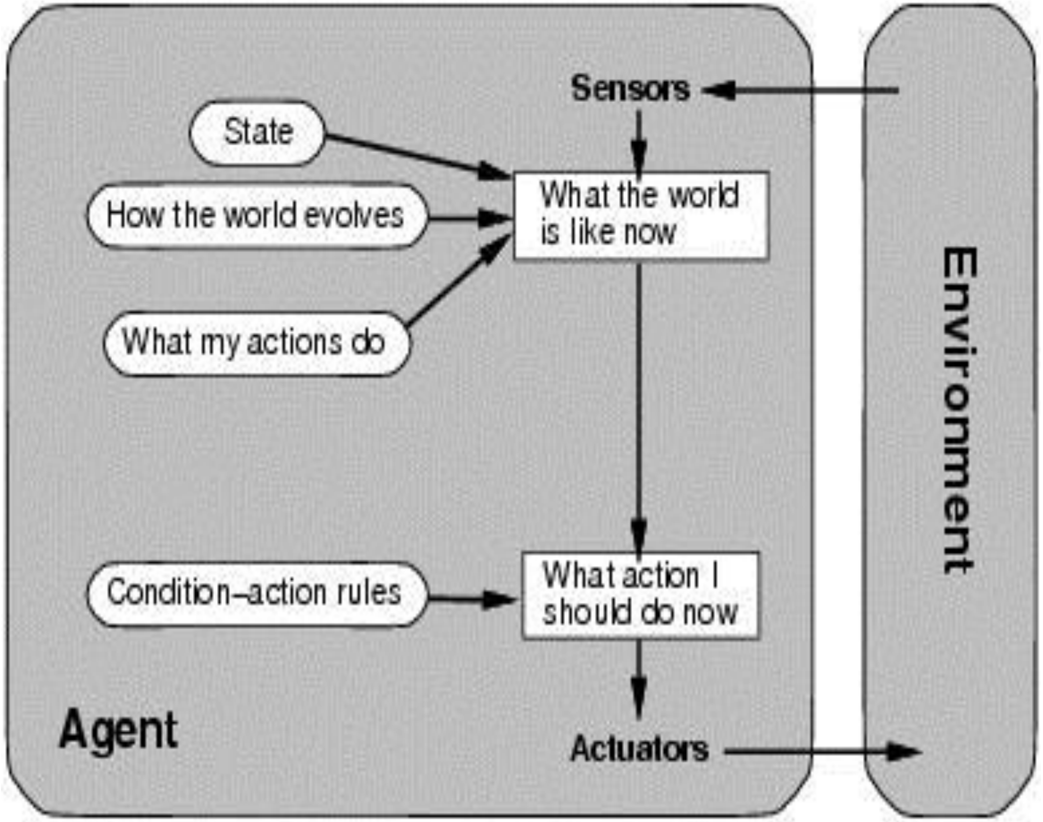
```
function REFLEX-VACUUM-AGENT ([location, status]) return an action
    if status == Dirty then return Suck
    else if location == A then return Right
    else if location == B then return Left
```

Agent types; simple reflex

```
function SIMPLE-REFLEX-AGENT(percept) returns an action
    static: rules, a set of condition-action rules

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

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

Agent types; reflex and state



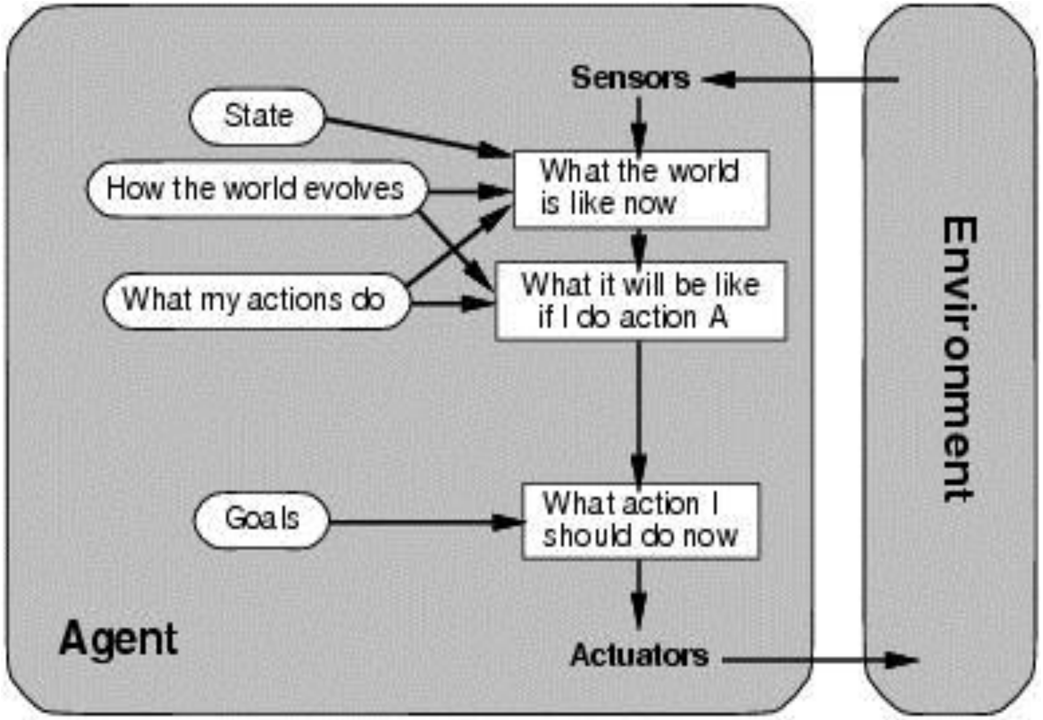
- To tackle partially observable environments.
  - Maintain internal state
- Over time update state using world knowledge
  - How does the world change.
  - How do actions affect world.

  ⇒Model of World
- Problems/Limitations?
  - not deliberative, agent types so far are reactive

Agent types; reflex and state

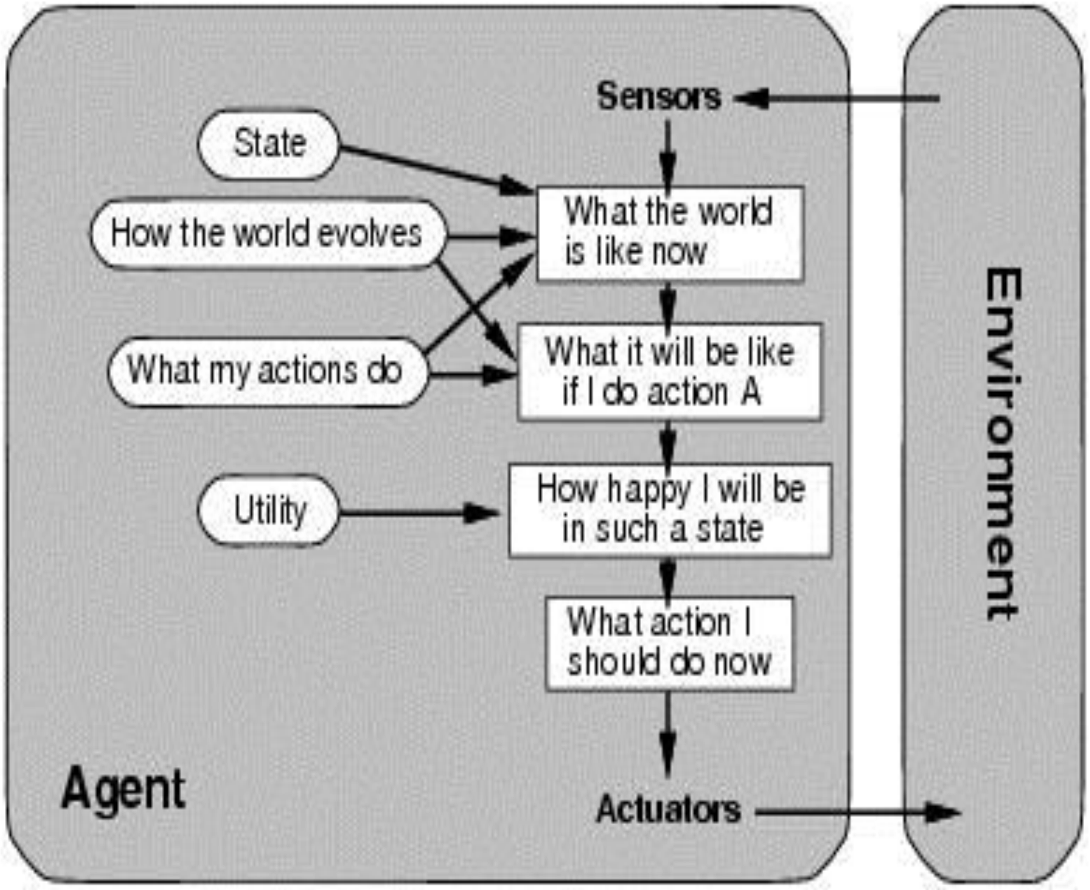
```
function REFLEX-AGENT-WITH-STATE(percept) returns an action

    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
```

Agent types; goal-based


Sensors
State
How the world evolves
What my actions do
What the world is like now
What it will be like if I do action A
Goals
What action I should do now
Agent
Actuators
Environment

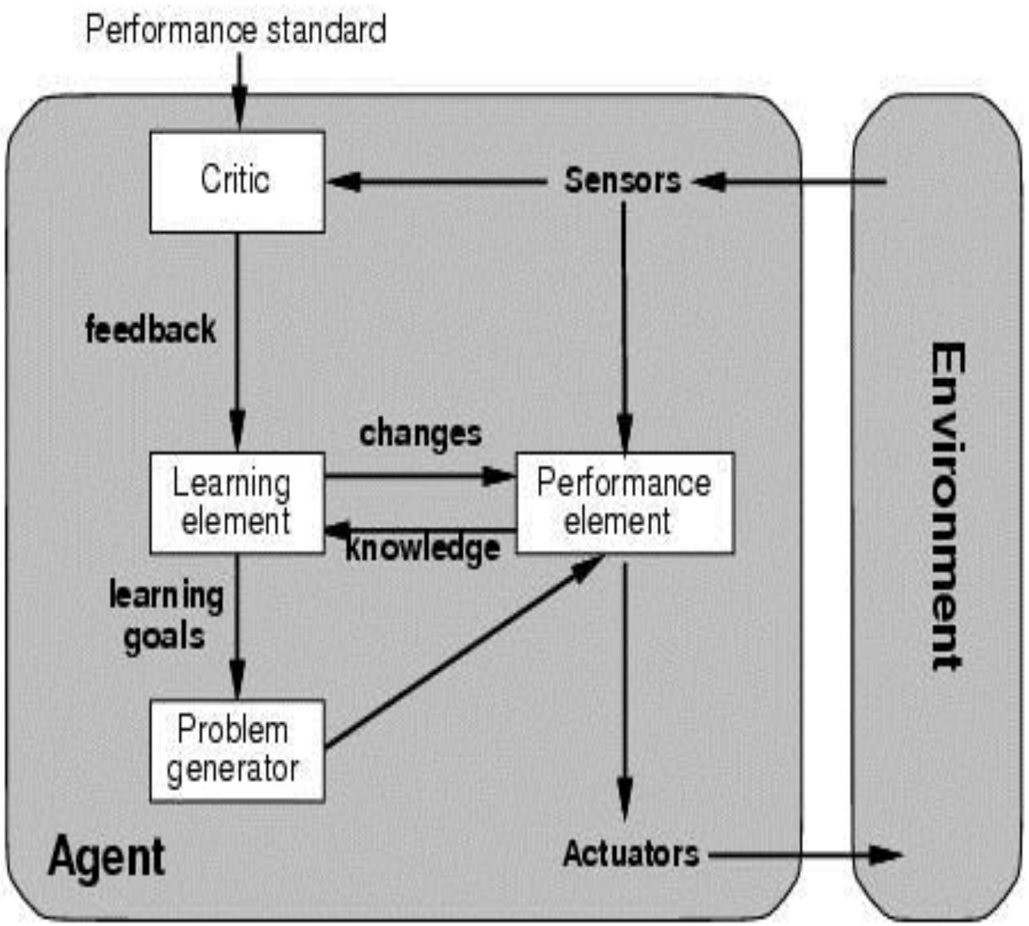
- Chose actions to achieve a desired goal
  - Search or planning often used
- Problems/Limitations?
  - May have to consider long sequences of possible actions before goal is achieved
  - Involves consideration of the future, “What will happen if I do...?”
  - How are competing goals treated?
  - What about degrees of success?

Agent types; utility-based


Sensors
State
How the world evolves
What my actions do
Utility
What the world is like now
What it will be like if I do action A
How happy I will be in such a state
What action I should do now
Agent
Actuators
Environment

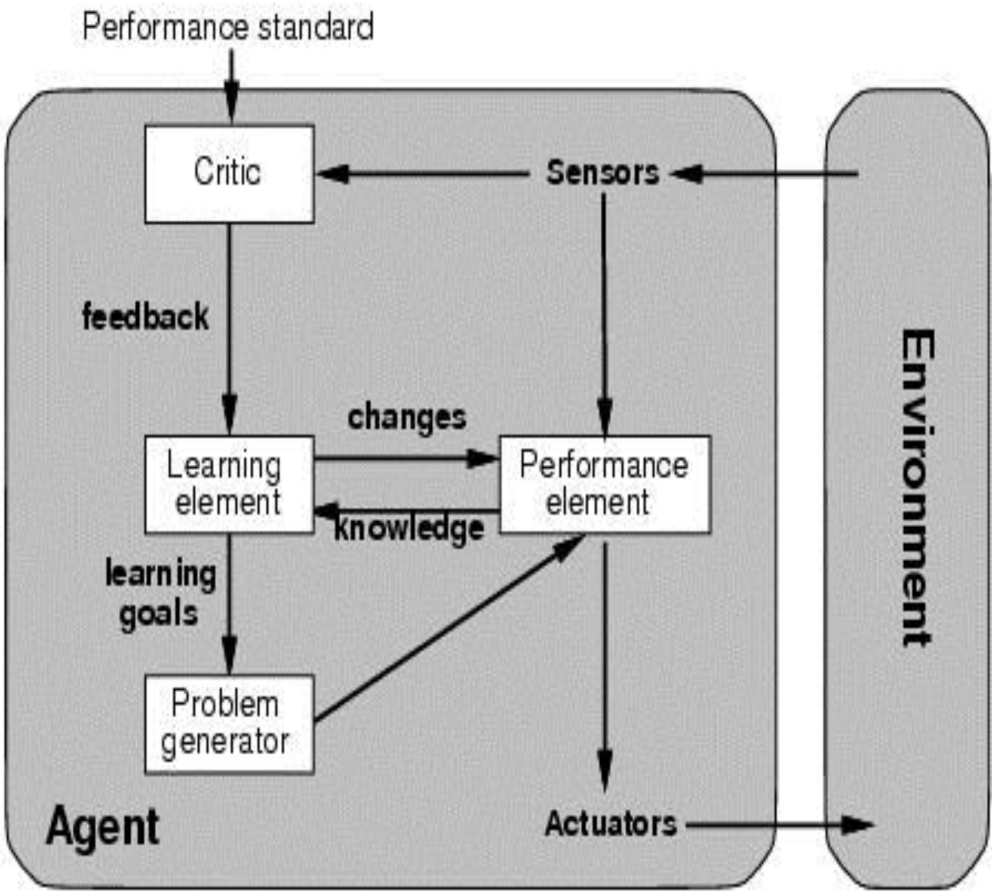
- Achieve goals while trying to maximize some utility value
  - Utility value gives a measure of success or "happiness" for a given situation
- Allows decisions comparing choice between
  - Conflicting goals
  - Likelihood of success and importance of goal

Agent types; learning



- 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; learning



- Learning element:
  - Introduce improvements in performance element.
  - Critic provides feedback on agents performance based on fixed performance standard.
- Performance element:
  - Selecting actions based on percepts.
- Problem generator:
  - Suggests actions that will lead to new and informative experiences.
  - Exploration vs. exploitation