Artificial Intelligence- Agents

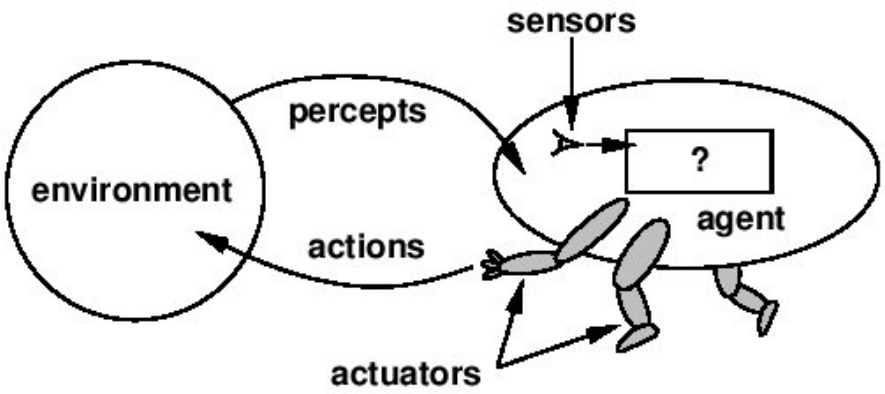
Amir Aavani
Amir@Aavani.net
http://Amir.Aavani.net/CS/AI-86

2. Oktober 2007

Outline

- Agents and Environments
- Rational Behavior
- Different Kinds of Environments
- Different Kinds of Agents

Agent



- Agent: Anything **perceives** its environment by **sensors** & acts upon environment via **effectors**.
- Agent includes Humans, Robots, Programs and etc.

Performance Measure

- How Agents Should Act: Right things
- **How** to evaluate agent's success?
  - **Performance Measure** evaluates **How**.
  - Possible PFs for a "Vacuum Cleaner Agent"
- **When** to evaluate agent's success?
- Importance of Performance Measure

Ideal Rational Agent

- Rational Agent $\neq$ Omniscient Agent
- Factors that effect the rationality:
  - Performance Measure
  - Percept Sequence
  - Knowledge from Environment
  - Possible Actions
- A rational agent chooses an action maximizes the expected value of Peformance Measure.

Structure of Intelligent Agent

- Mapping: Relation between Percepts and Action.
- Job of AI is to design **Agent Program**: a function that implement agent mapping from percepts to action.
- **Architecture** receives percepts from sensors, runs program, act the selected action to effectors.
- Agent= Architecture + Program

Structure of Intelligent Agent, Cont

- PAGE: **P**ercepts, **A**ctions, **G**oal(s) and **E**nvironment.
- PAGE for automated taxi driver agent:
- Percepts:
- Actions:
- Goal:
- Environment:

Structure of Intelligent Agent, Cont

- PAGE: **P**ercepts, **A**ctions, **G**oal(s) and **E**nvironment.
- PAGE for automated taxi driver agent:
- Percepts: Cameras, Speedometer, GPS, ...
- Actions: Steer, accelerate, brake, ...
- Goal: Safety, destination, profits, legality, ...
- Environment: Roads/Street/freeways ...

Environment Types

- Fully vs. Partially observable: An environment is **full observable** when sensors can detect all aspects relevant to action selection.
- Deterministic vs Stochastic: An environment in which the next state is completely determined by **current state** and **action** is Deterministic.

Environment Types, Cont

- Episodic vs. Sequential: In an episodic environment, agents experience can be divided into atomic independent steps where agents perceives and performs a single action.
- Static vs. Dynamic: If environment changes while agent is choosing an action, the environment is dynamic.
  Semi-dynamic if the agents performance changes even when the environment remains the same.

Environment Types, Cont

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

Environment Types, Cont

	Solitaire	Backgammon	Taxi
Observable	FULL	FULL	Partial
Deterministic	Yes	No	No
Episodic	No	No	No
Static	Yes	Yes	No
Discrete	Yes	Yes	No

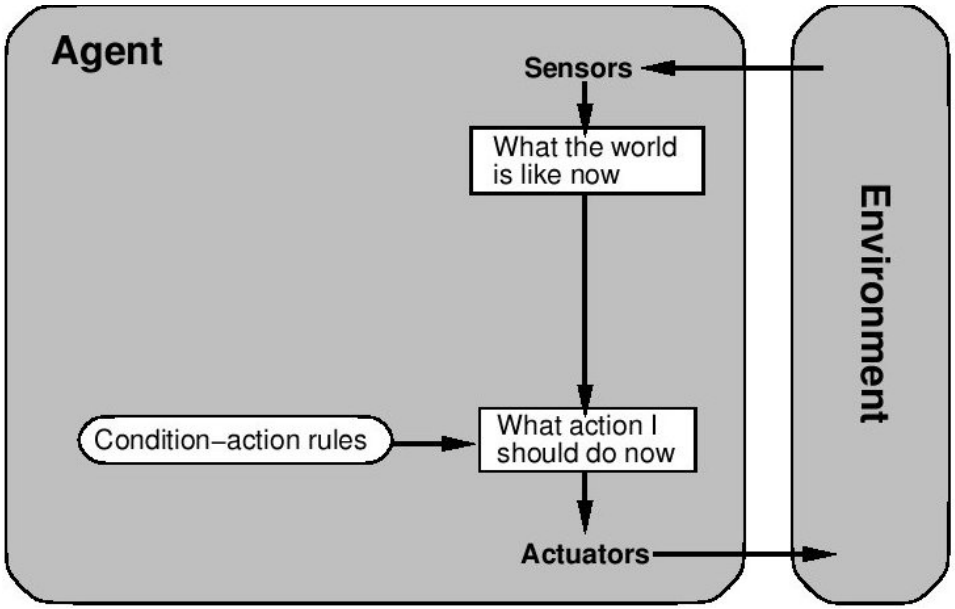
Agent Programs

- Simple reflex agents
- Agent that keeps track of the world
- Goal-based agents
- Utility-based agents

Structure of Intelligent Agent, Cont

```
function SKELETON-AGENT (percept) return action
      static memory, the agent's memmory of the world

      memory= UPDATE-MEMORY (memory, percept)
      action= CHOOSE-BEST-ACTION (memory)
      memory= UPDATE-MEMORY (memory, action)
      return action
```

Simple reflex agents

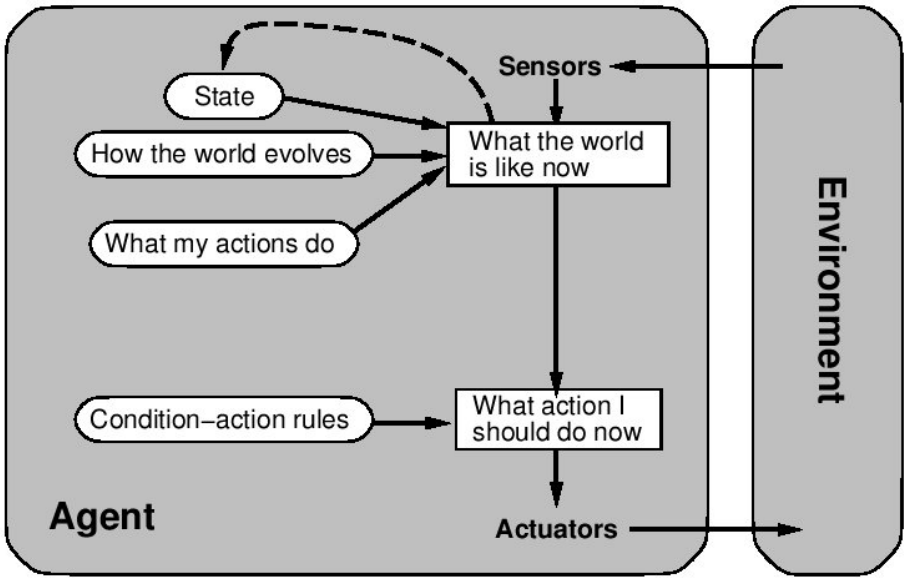


- Has rule-based (some rule) in form of "condition-action"
- Has a component to extract feature
- There is no access to complete state of the world
- Works only if correct decision can be made on basis of current percept.

Simple reflex agents, Cont

```
function SIMPLE-REFLEX-AGENT (percept) return action
     static rule, a set of condition-action rules

     state= INTERPRET-INPUT (percept)
     rule= RULE-MACHE (state, rules)
     action= RULE-ACTION [rule].action
     return action
```

Agents that keep track of world

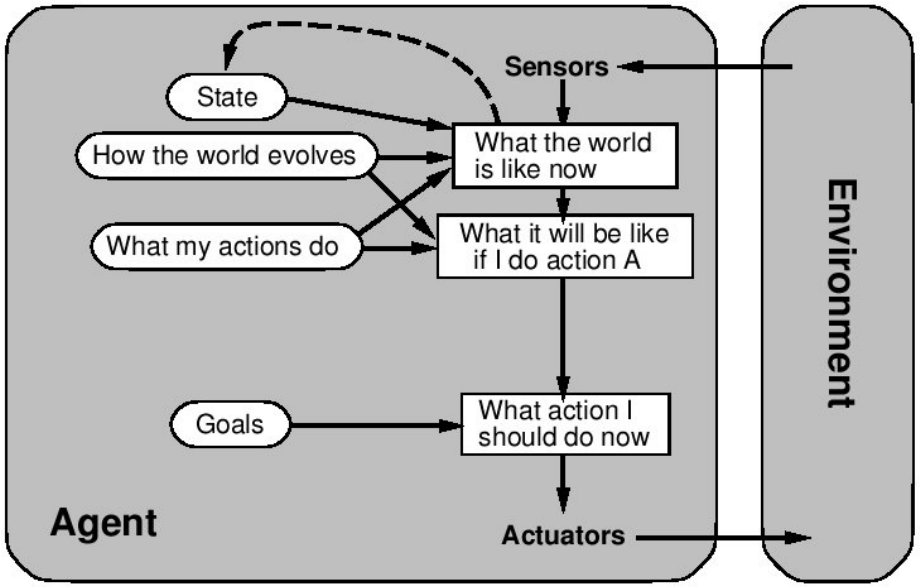


- Complete state can be available if agent knows:
  - How environment evolves independently of agent
  - How agent's acion effect the world
- They are more powerfull than "Simple Reflex Agents"

Agents that keep track of world, Cont

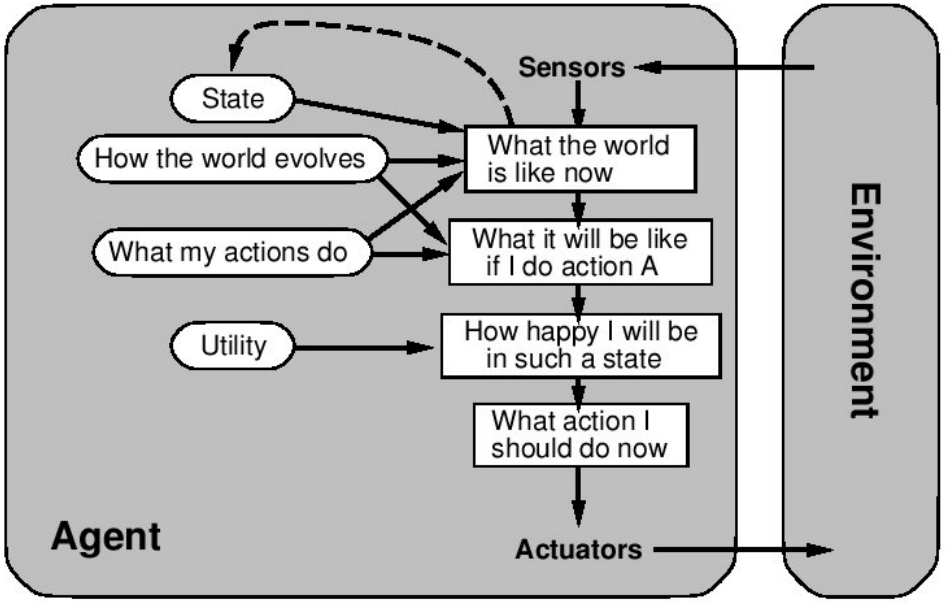
```
function REFLEX-AGENT-With-State (percept) return action
     static rules, a set of condition-action rules
            state, a description of the currect world stat


     state= Update_STATE (state, percept)
     rule= RULE-MACHE (state, rules)
     action= rules [rule].action
     state= UPDATE-STATE (state,action)
     return action
```

Goal-based Agents



- Have information about their goal
- "Reflective Agents" can be used for a fixed goal
- To implement these agents, a "Search" or "Planning" Alg. is needed.
- There can not be "condition-action" rules.
- They are more powerfull than "Reflective Agents"

Utility-based Agents



- The Goal by itself is not enough.
- There may be many way to achieve the goal.
- Utility is a function that maps a state onto a real number, which describes the associated degree of hapiness.
- Any rational agent can be described as possing a utility function.
- They are more powerfull than "Reflective Agents"

Utility Based Agents, Cont

- Exercise Two: Write the functoin based on which a utility function can work
- Deadline: Oct, 9th
- Email Subject: "AI-02"
- TO: Amir@Aavani.net