Lecture 06
Types of Agent Programs

Artificial Intelligence

Dr. Ahmed Mateen

Agenda

- Types of Agent Programs
  - Simple Reflex Agents
  - Model Based Reflex Agents

Basic Kinds of Agent Programs

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

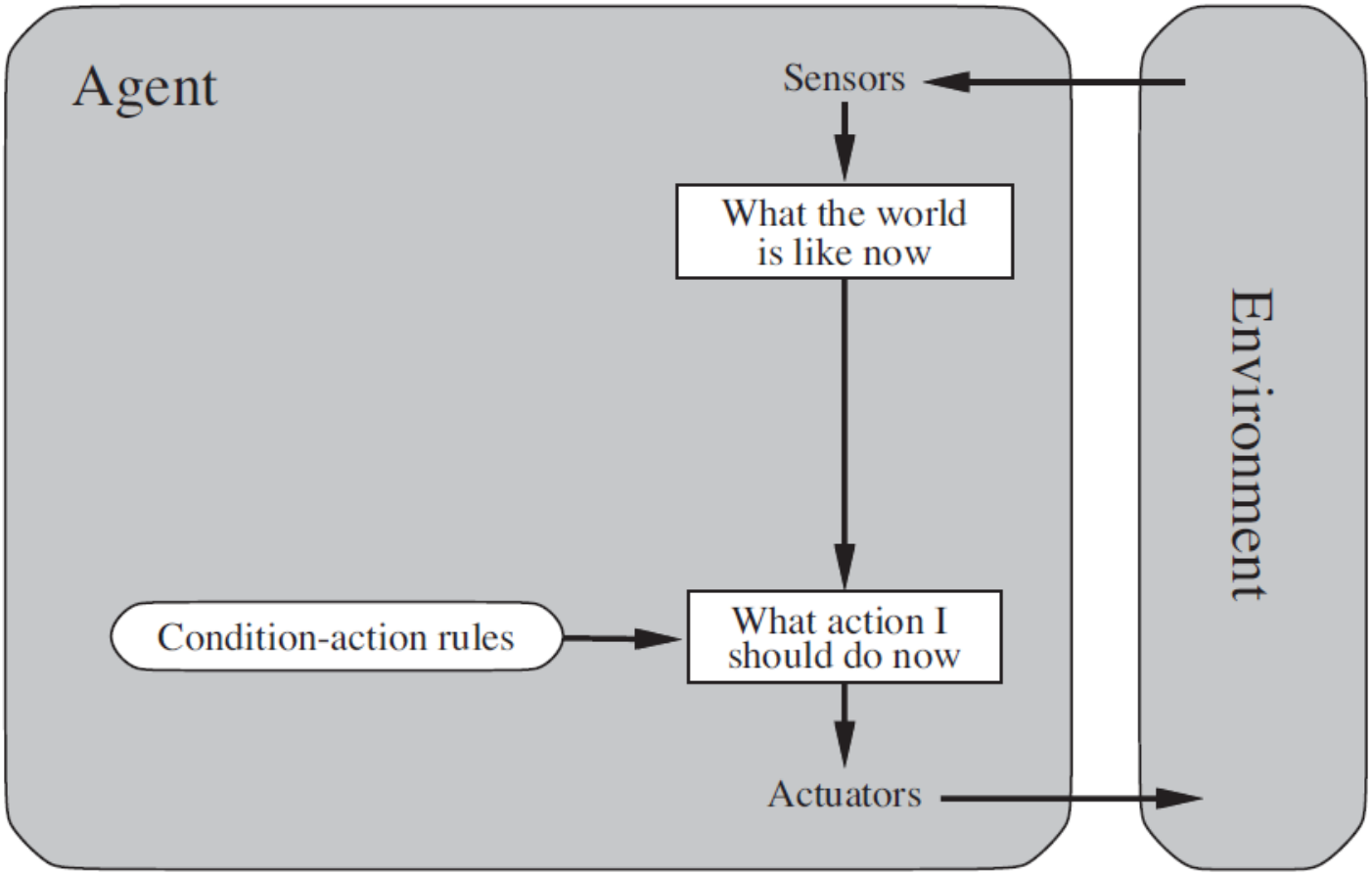
Simple Reflex Agents

- Simplest kind of agent
- Select actions on the basis of the ***current*** percept
- Ignores the rest of the percept history
- Makes use of **condition–action rule.**

Condition–Action rule

- In Condition-Action rule a change to the condition "is" the triggering event.
  - —A CA rule becomes active when its condition becomes true. {rule is **activated**}
  - —A CA rule becomes inactive when its condition becomes false. {rule is **deactivated**}
  - —CA rules do not actually need the event.
- Syntax: **if** *Condition* **then** *Action*
  - Example: **if** *Room-is-dirty* **then** *Clean*.

Simple Reflex Agent



Schematic diagram of a simple reflex agent.

Simple Reflex Agent

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

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

It acts according to a rule whose condition matches the current state, as defined by the percept.

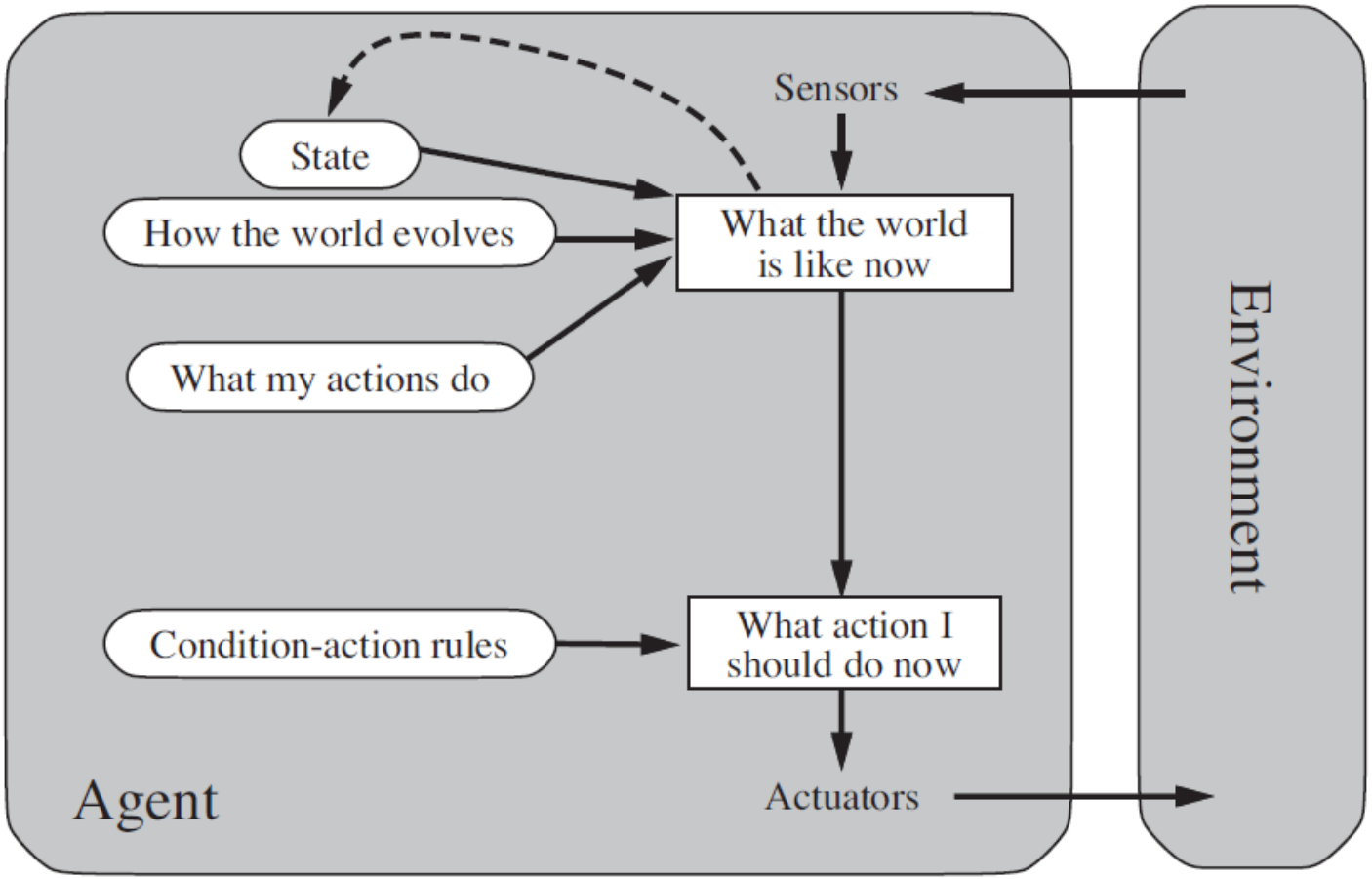
Limitations of Simple Reflex Agents

- Have limited intelligence
- Even a little bit of un-observability can cause serious trouble
- If there occurs any change in the environment, then the collection of rules need to be updated.
- Infinite loops are often unavoidable for simple reflex agents operating in **partially observable environments**.
  - Escape is randomization

Randomized Simple Reflex Agents

- It is a useful trick that helps a simple reflex agent in some situations
- Might outperform simple deterministic reflex agents
- In some multiagent environments, **randomized behavior** is rational
  - it avoids the pitfalls of predictability

Model Based Reflex Agents



A model-based reflex agent.

Model Based Reflex Agents

- Maintains **internal state**
  - the percept history
- Updating internal state information requires:
  - some information about how the world evolves independently of the agent
  - some information about how the agent's own actions affect the world
- This knowledge about "how the world works" is called a **model** of the world
  - An agent that uses such a model is called a **model-based agent**.

Model Based Reflex Agents

function MODEL-BASED-REFLEX-AGENT(*percept*) **returns** an action
  persistent: *state*, the agent's current conception of the world state
    model, a description of how the next state depends on current state and action
    rules, a set of condition–action rules
    action, the most recent action, initially none

  state ← UPDATE-STATE(*state*, *action*, *percept*, *model*)
  rule ← RULE-MATCH(*state*, *rules*)
  action ← *rule*.ACTION
  return *action*

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

Waymo Google self driven car project

- https://www.youtube.com/watch?v=uHbMt6WDhQ8