Intelligent Agents

Chapter 2

Agent

- An AI system is composed of an agent and its environment.
- The agents act in their environment.

Agents

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

- An agent runs in the **cycle** of **perceiving**, **thinking**, and **acting**. An agent can be:

1. Human-Agent: A human agent has eyes, ears, and other organs which work for sensors and hand, legs, mouth, vocal tract work and other body parts for actuators.

Agents

2. Robotic Agent: A robotic agent can have cameras, infrared range finder, NLP for sensors and various motors for actuators.

- Cog-MIT, Aibo-Sony

3. Software Agent: Software agent can have keystrokes, file contents as sensory input and act on those inputs and display output on the screen.

- Hence the world around us is full of agents such as thermostat, cellphone, camera, and even we are also agents.

Agent Terminology

- **Performance Measure of Agent −** It is the criteria, which determines how successful an agent is.
- **Percept −** It is agent's perceptual inputs at a given instance.
- **Percept Sequence −** It is the history of all that an agent has perceived till date.
- **Behavior of Agent −** It is the action that agent performs after any given sequence of percepts.
- **Agent Function −** It is a map from the precept sequence to an action.

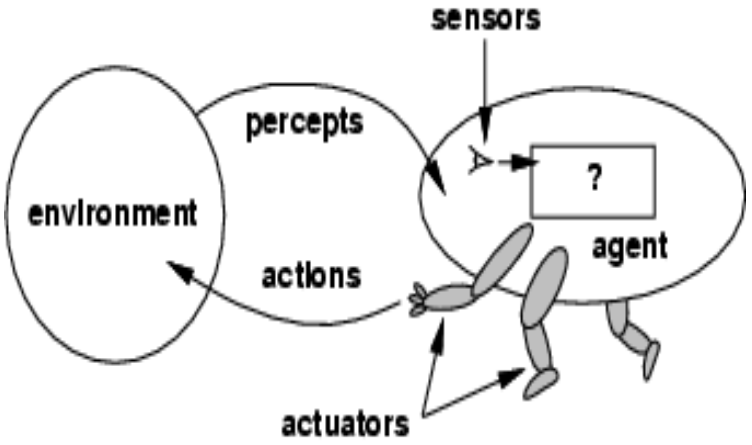
Agents and Environments

Sensor: Sensor is a device which detects the change in the environment and sends the information to other electronic devices. An agent observes its environment through sensors.

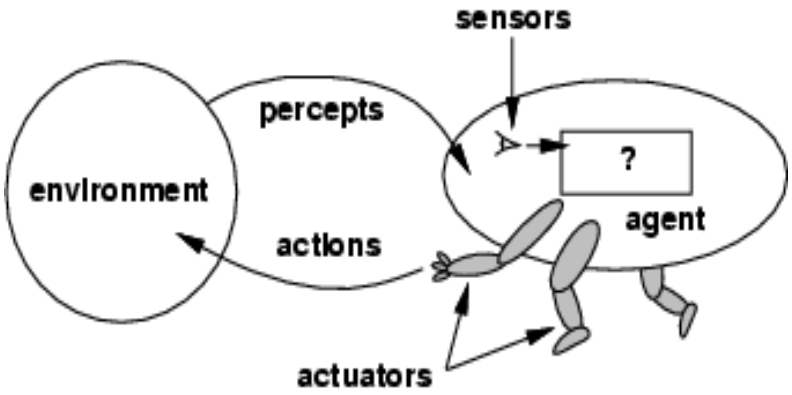
Actuators: Actuators are the component of machines that converts energy into motion. The actuators are only responsible for moving and controlling a system. An actuator can be an electric motor, gears, rails, etc.

Effectors: Effectors are the devices which affect the environment. Effectors can be legs, wheels, arms, fingers, wings, fins, and display screen.

A robot's effector is under the control of the robot. Effectors can range from legs and wheels to arms and fingers.
An actuator is the actual mechanism that enables the effector to execute an action. Actuators typically include electric motors, hydraulic or pneumatic cylinders, etc.



Agents and Environments

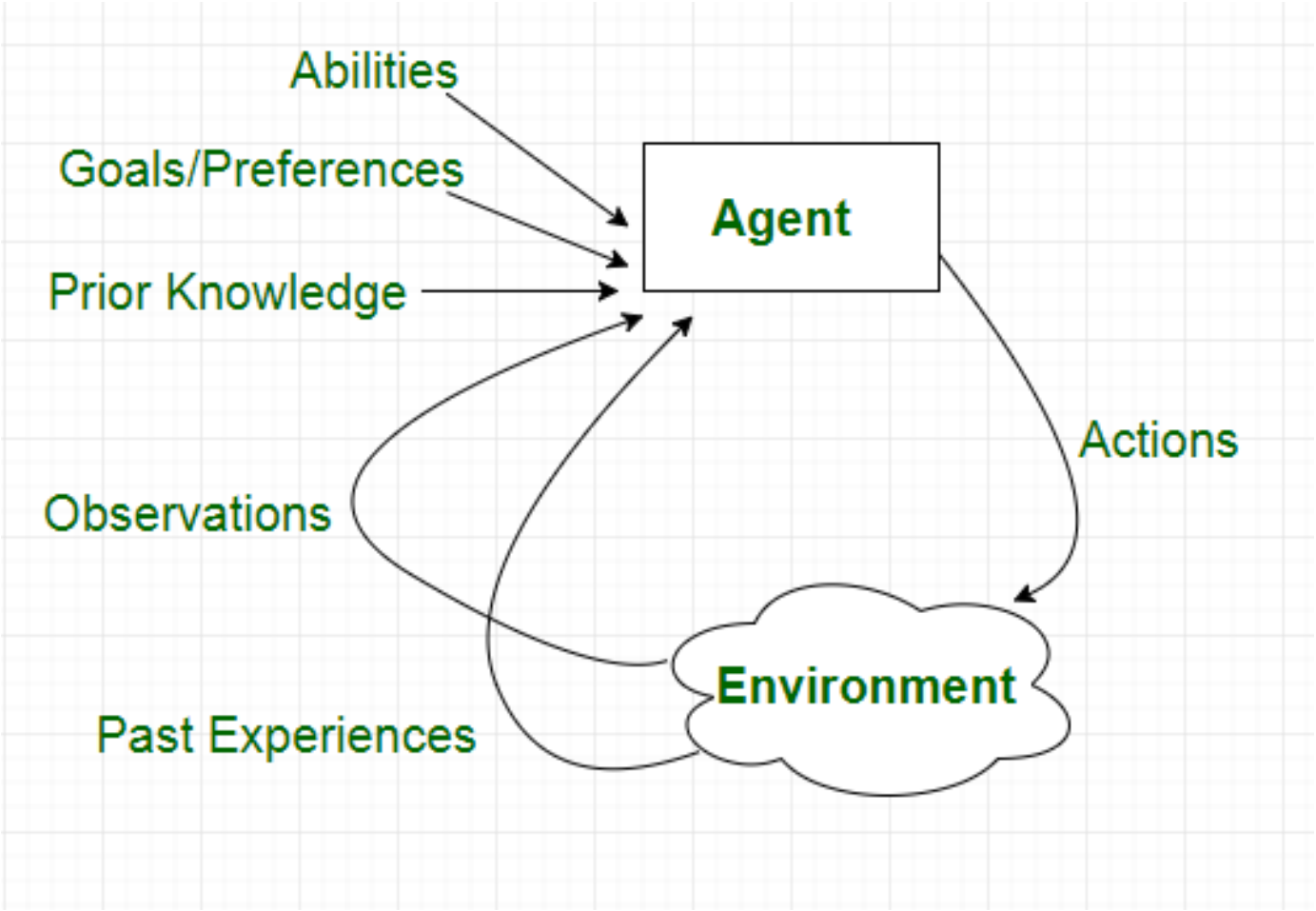


- The agent function maps from percept histories to actions:

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

- The agent program runs on the physical architecture to produce the function (*f*)
- Agent = architecture + program

Agent


Abilities
Goals/Preferences
Prior Knowledge
Agent
Actions
Observations
Environment
Past Experiences

Intelligent Agent

- An intelligent agent is an autonomous entity which act upon an environment using sensors and actuators for achieving goals.
- An intelligent agent may learn from the environment to achieve their goals. A thermostat is an example of an intelligent agent.
- **Four rules** for an AI agent:
- **Rule 1:** An AI agent must have the ability to perceive the environment.
- **Rule 2:** The observation must be used to make decisions.
- **Rule 3:** Decision should result in an action.
- **Rule 4:** The action taken by an AI agent must be a rational action.

Rational Agent

- A rational agent is an agent which has clear preference, models uncertainty, and acts in a way to maximize its performance measure with all possible actions.

- A rational agent is said to perform the right things. AI is about creating rational agents to use for game theory and decision theory for various real-world scenarios.

- For an AI agent, the rational action is most important because in AI reinforcement learning algorithm, for each best possible action, agent gets the positive reward and for each wrong action, an agent gets a negative reward.

Rationality

- The rationality of an agent is measured by its performance measure.
- Rationality can be judged on the basis of following points:
  - Performance measure which defines the success criterion.
  - Agent prior knowledge of its environment.
  - Best possible actions that an agent can perform.
  - The sequence of percepts.

Agent Environment in AI

- An environment is everything in the world which surrounds the agent, but it is not a part of an agent itself.
- An environment can be described as a situation in which an agent is present.
- The environment is where agent lives, operate and provide the agent with something to sense and act upon it.

Features of Environment

- As per Russell and Norvig, an environment can have various features from the point of view of an agent:

- Fully Observable vs Partially Observable
- Deterministic vs Stochastic
- Competitive vs Collaborative
- Single-agent vs Multi-agent
- Static vs Dynamic
- Discrete vs Continuous
- Episodic vs Sequential
- Known vs Unknown

Fully observable vs Partially Observable

- When an agent sensor is capable to sense or access the complete state of an agent at each point in time, it is said to be a fully observable environment else it is partially observable.
- Maintaining a fully observable environment is easy as there is no need to keep track of the history of the surrounding.
- An environment is called **unobservable** when the agent has no sensors in all environments.
- **Examples:**
  - **Chess –** the board is fully observable, and so are the opponent's moves.
  - **Driving –** the environment is partially observable because what's around the corner is not known.

Deterministic vs Stochastic

- If an agent's current state and selected action can completely determine the next state of the environment, then such environment is called a deterministic environment.
- A stochastic environment is random in nature and cannot be determined completely by an agent.
- **Examples:**
  - **Chess –** there would be only a few possible moves for a coin at the current state and these moves can be determined.
  - **Self-Driving Cars-** the actions of a self-driving car are not unique, it varies time to time.

Competitive vs Collaborative

- An agent is said to be in a competitive environment when it competes against another agent to optimize the output.
  - The game of chess is competitive as the agents compete with each other to win the game which is the output.

- An agent is said to be in a collaborative environment when multiple agents cooperate to produce the desired output.
  - When multiple self-driving cars are found on the roads, they cooperate with each other to avoid collisions and reach their destination which is the desired output.

Single-agent vs Multi-agent

- An environment consisting of only one agent is said to be a single-agent environment.
  - A person left alone in a maze is an example of the single-agent system.

- An environment involving more than one agent is a multi-agent environment.
  - The game of football is multi-agent as it involves 11 players in each team.

Dynamic vs Static

- An environment that keeps constantly changing itself when the agent is up with some action is said to be dynamic.
  - Taxi driving is an example of a dynamic environment
  - A roller coaster ride is dynamic as it is set in motion and the environment keeps changing every instant.

- An idle environment with no change in its state is called a static environment.
  - An empty house is static as there's no change in the surroundings when an agent enters.
  - Crossword puzzles are an example of a static environment.

Discrete vs Continuous

- If an environment consists of a finite number of actions that can be deliberated in the environment to obtain the output, it is said to be a discrete environment.
  - The game of chess is discrete as it has only a finite number of moves. The number of moves might vary with every game, but still, it's finite.

- The environment in which the actions are performed cannot be numbered i.e. is not discrete, is said to be continuous.
  - Self-driving cars are an example of continuous environments as their actions are driving, parking, etc. which cannot be numbered.

Episodic vs Sequential

- In **an Episodic task environment**, each of the agent's actions is divided into atomic incidents or episodes.
- There is no dependency between current and previous incidents.
- In each incident, an agent receives input from the environment and then performs the corresponding action.
  - **Example:** Consider an example of **Pick and Place robot**, which is used to detect defective parts from the conveyor belts. Here, every time robot (agent) will make the decision on the current part i.e. there is no dependency between current and previous decisions.

- In a **Sequential environment**, the previous decisions can affect all future decisions. The next action of the agent depends on what action he has taken previously and what action he is supposed to take in the future.
  - **Example:**
    - **Checkers game-** Where the previous move can affect all the following moves.

Known vs Unknown Accessible vs Inaccessible

- Known and unknown are not actually a feature of an environment, but it is an agent's state of knowledge to perform an action.
- In **a known environment**, the results for all actions are known to the agent. While in **unknown environment**, agent needs to learn how it works in order to perform an action.
- In a known environment, the output for all probable actions is given. Obviously, in case of unknown environment, for an agent to make a decision, it has to gain knowledge about how the environment works.
- It is quite possible that a known environment to be partially observable and an Unknown environment to be fully observable.
- If an agent can obtain complete and accurate information about the state's environment, then such an environment is called an Accessible environment else it is called inaccessible.
- An empty room whose state can be defined by its temperature is an example of an accessible environment.
- Information about an event on earth is an example of inaccessible environment.

Types of Agent

- Agents can be grouped into five classes based on their degree of perceived intelligence and capability.
- All these agents can improve their performance and generate better action over the time. These are given below:
  - Simple Reflex Agent
  - Model-based Reflex Agent
  - Goal-based Agents
  - Utility-based Agent
  - Learning Agent

Simple Reflex Agent

- These agents take decisions on the basis of the current percepts and ignore the rest of the percept history.
- These agents only succeed in the fully observable environment.
- It does not consider any part of percepts history during their decision and action process.
- It works on condition-action rule, which means it maps the current state to action. Such as a **Room Cleaner agent**, it works only if there is dirt in the room.
- Problems for the simple reflex agent design approach:
  - They have very limited intelligence
  - They do not have knowledge of non-perceptual parts of the current state
  - Mostly too big to generate and to store.
  - Not adaptive to changes in the environment.

Model-based Reflex Agent

- Can work in a partially observable environment, and track the situation.
- A model-based agent has **two i**mportant **factors**:
  - **Model:** It is knowledge about "how things happen in the world," so it is called a Model-based agent.
  - **Internal State:** It is a representation of the current state based on percept history.
- These agents have the model, "which is knowledge of the world" and based on the model they perform actions.
- Updating the agent state requires information about:
  - How the world evolves.
  - How the agent's action affects the world.

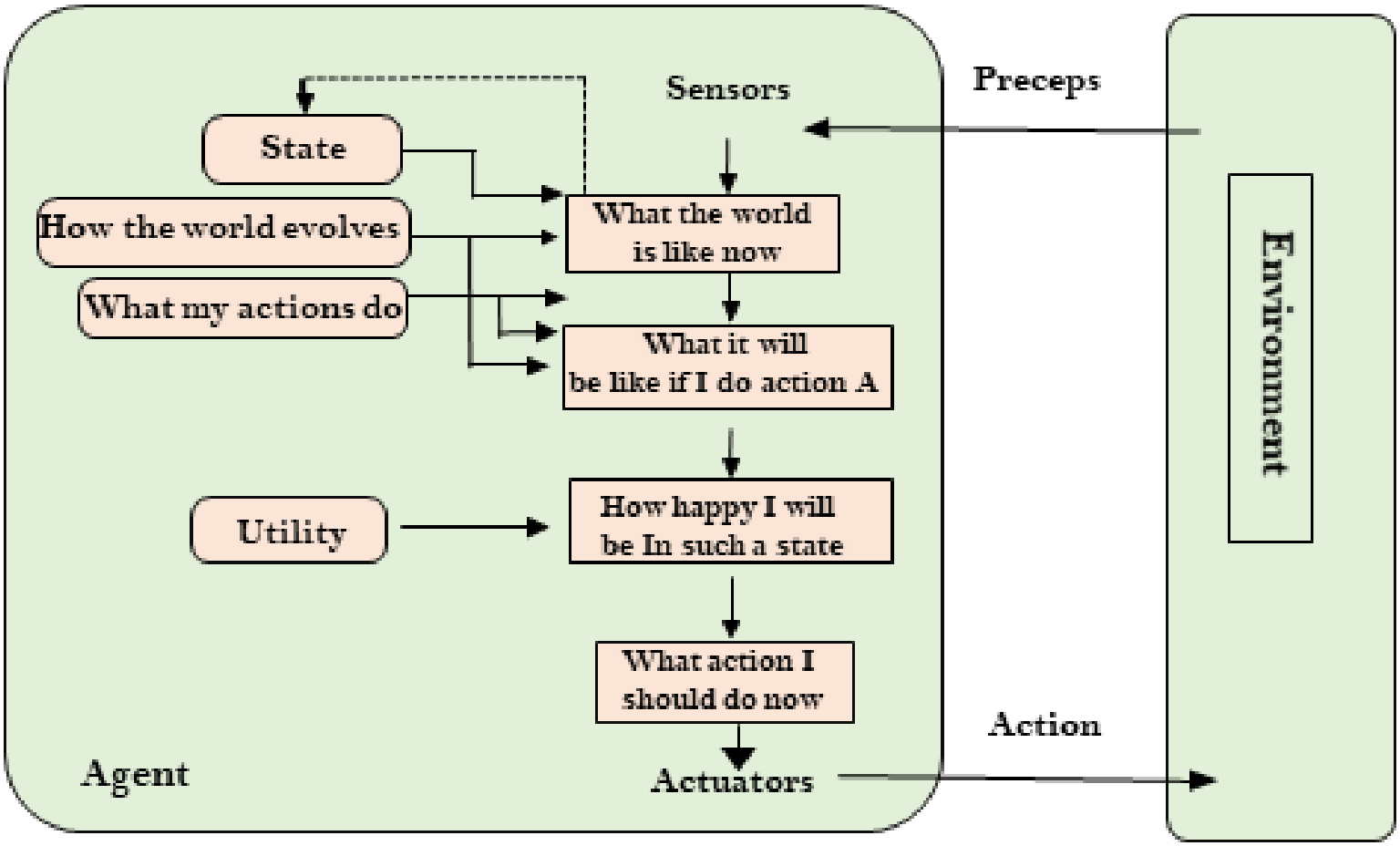
Goal-based Agents

- The knowledge of the current state environment is not always sufficient to decide for an agent to what to do.
- The agent needs to know its goal which describes desirable situations.
- It expand the capabilities of the model-based agent by having the "goal" information.
- They choose an action, so that they can achieve the goal.
- These agents may have to consider a long sequence of possible actions before deciding whether the goal is achieved or not.
- Such considerations of different scenario are called searching and planning, which makes an agent proactive.

Utility-based Agents

- Are similar to the goal-based agent but provide an extra component of utility measurement which makes them different by providing a measure of success at a given state.
- It act based on not only goals but also the best way to achieve that goal.
- Useful when there are multiple possible alternatives, and an agent has to choose in order to perform the best action.
- The utility function maps each state to a real number to check how efficiently each action achieves the goals.

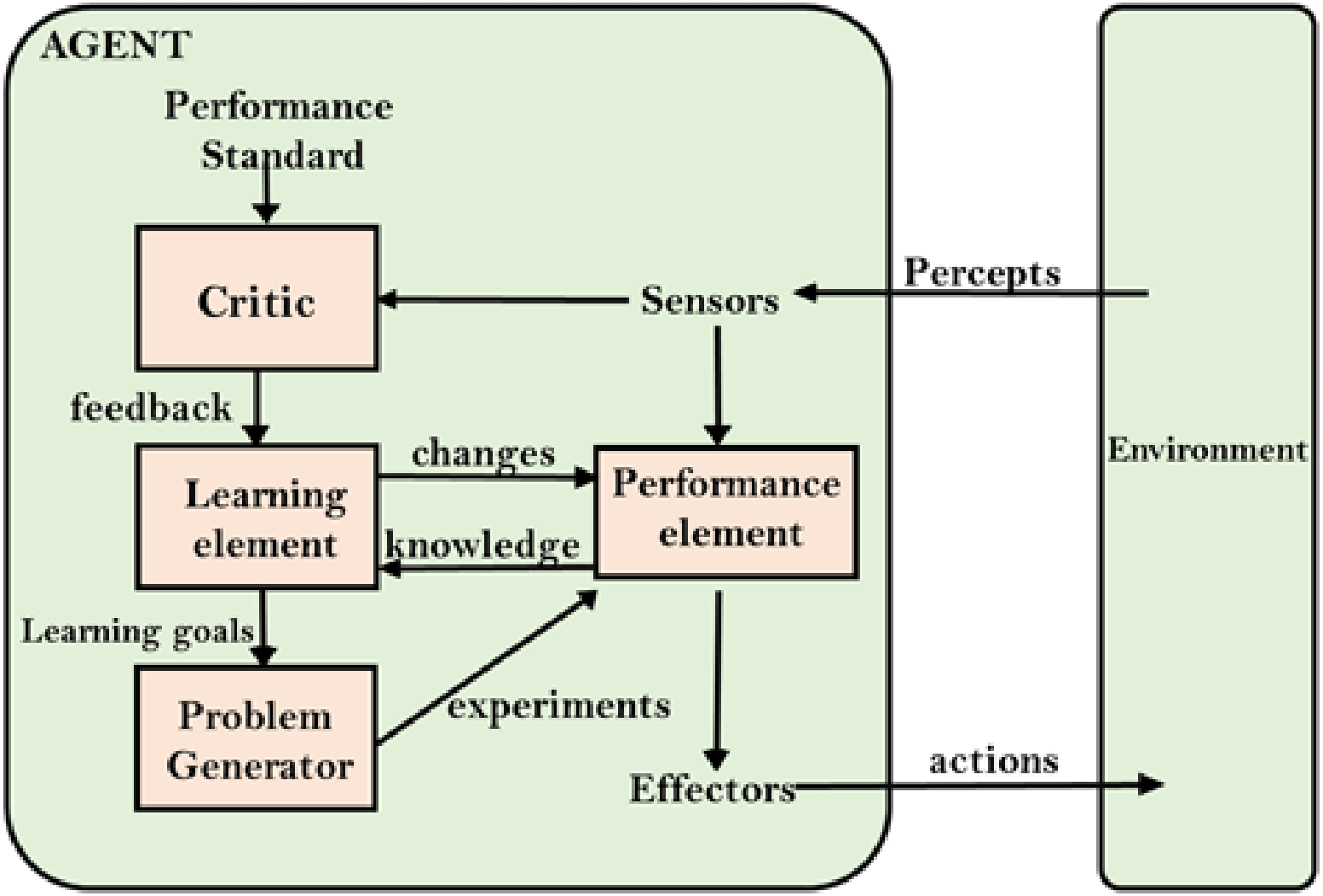
Utility-based Agents



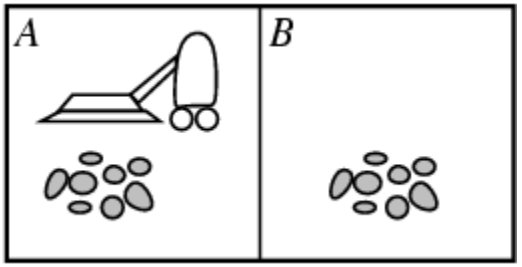
Learning Agents

- It can learn from its past experiences, or it has learning capabilities.
- It starts to act with basic knowledge and then able to act and adapt automatically through learning.
- Has mainly four conceptual components:
  - **Learning Element:** It is responsible for making improvements by learning from environment.
  - **Critic:** Learning element takes feedback from critic which describes that how well the agent is doing with respect to a fixed performance standard.
  - **Performance Element:** It is responsible for selecting external action.
  - **Problem Generator:** Responsible for suggesting actions that will lead to new and informative experiences.
- Hence, its able to learn, analyze performance, and look for new ways to improve the performance.

Learning Agents



Vacuum-cleaner World



- Percepts: location and contents, e.g., [A,Dirty]
- Actions: *Left*, *Right*, *Suck*, *NoOp*
- One very simple agent function is: if the current square is dirty, then suck; otherwise, move to the other square.

Vacuum-cleaner World

Percept sequence	Action
$[A, Clean]$	$Right$
$[A, Dirty]$	$Suck$
$[B, Clean]$	$Left$
$[B, Dirty]$	$Suck$
$[A, Clean], [A, Clean]$	$Right$
$[A, Clean], [A, Dirty]$	$Suck$
⋮	⋮
$[A, Clean], [A, Clean], [A, Clean]$	$Right$
$[A, Clean], [A, Clean], [A, Dirty]$	$Suck$
⋮	⋮

Figure 2.3 Partial tabulation of a simple agent function for the vacuum-cleaner world shown in Figure 2.2.

Vacuum-cleaner World

- Its seen that various vacuum-world agents can be defined simply by filling in the right-hand column in various ways.
- The obvious question, then, is this: *What is the right way to fill out the table?*
- In other words, what makes an agent good or bad, intelligent or stupid?

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 (!= perfect 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:
  1. amount of dirt cleaned up
  2. amount of time taken
  3. amount of electricity consumed
  4. amount of noise generated, etc.

PEAS of Rational Agent

- 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

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

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, findings, patient's answers)

PEAS

Agent	Performance measure	Environment	Actuators	Sensors
1. Medical Diagnose	◦ Healthy patient ◦ Minimized cost	◦ Patient ◦ Hospital ◦ Staff	◦ Tests ◦ Treatments	Keyboard (Entry of symptoms)
2. Vacuum Cleaner	◦ Cleanness ◦ Efficiency ◦ Battery life ◦ Security	◦ Room ◦ Table ◦ Wood floor ◦ Carpet ◦ Various obstacles	◦ Wheels ◦ Brushes ◦ Vacuum Extractor	◦ Camera ◦ Dirt detection sensor ◦ Cliff sensor ◦ Bump Sensor ◦ Infrared Wall Sensor
3. Part -picking Robot	◦ Percentage of parts in correct bins.	◦ Conveyor belt with parts, ◦ Bins	◦ Jointed Arms ◦ Hand	◦ Camera ◦ Joint angle sensors.

Thank You