i4013701-4

ARTIFICIAL INTELLIGENCE

LECTURE 2. INTELLIGENT AGENTS

INSTRUCTOR: DR. AMR ABDULLAH MUNSHI

ADMINISTRIVIA

- Course home page: (https://uqu.edu.sa/aaamunshi/) for lectures notes, assignments and labs.
- Form groups for projects
- Any programming language

RECAP

- An **agent** is just something that acts
- A **rational agent** is one that acts so as to achieve the best outcome or, when there is uncertainty, the best expected outcome.

AGENTS AND ENVIRONMENTS

- Now, examining agents, environments, and the coupling between them.
- Some agents behave better than others ⟶ **rational agent**
- The **agent** behaviour depends on the nature of the environment; some environments are more difficult than others.

AGENTS AND ENVIRONMENTS

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

- **Agent:**

  Perceives its environment → **sensors**

  Acts upon its environment → **actuators**

AGENTS AND ENVIRONMENTS

AGENTS AND ENVIRONMENTS

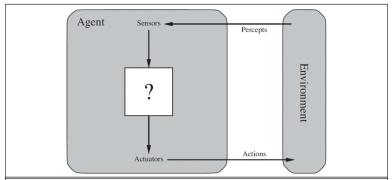


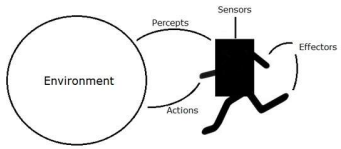
Figure 2.1 Agents interact with environments through sensors and actuators.

AGENTS AND ENVIRONMENTS

- Human agent

  sensors: eyes, ears and other organs

  actuators: hands, legs, vocal tract, etc…



AGENTS AND ENVIRONMENTS

- Robot agent

sensors:

actuators:

AGENTS AND ENVIRONMENTS

- Robot agent

sensors: cameras, infrared range finders

actuators: various motor parts

AGENTS AND ENVIRONMENTS

- Software agent

sensors:

actuators:

AGENTS AND ENVIRONMENTS

- Software agent

sensors: keystrokes, file contents, network packets

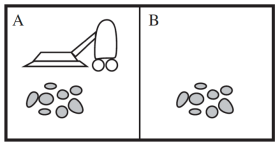
actuators: screen display, writing files, sending network packets

AGENTS AND ENVIRONMENTS

- **Percept Sequence** is the complete history of everything the agent has ever perceived.
- Agent's behavior is described by the **Agent Function.**
- **Agent Function** maps any given percept sequence to an action.
- **Agent Function** is an abstract mathematical description.

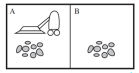
AGENTS AND ENVIRONMENTS

- **Vacuum-cleaner world**



- move left, move right, suck up the dirt

AGENTS AND ENVIRONMENTS



- **Tabulation**

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.

AGENTS AND ENVIRONMENTS

- **Agent Program**

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

Figure 2.8 The agent program for a simple reflex agent in the two-state vacuum environment. This program implements the agent function tabulated in Figure 2.3.

AGENTS AND ENVIRONMENTS

- What is the right way to fill out the table?

  Good or Bad?

  Intelligent or Stupid?

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.

THE CONCEPT OF RATIONALITY

- A **rational agent** is one that does the right thing
- Every entry in the tabulation table for the agent function is filled out correctly
- What does it mean to do the right thing?

THE CONCEPT OF RATIONALITY

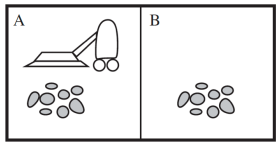
- When an agent is in an environment;
- Generates a sequence of actions according to the percepts it receives.
- The sequence of actions make the environment go through a sequence of states.
- If the sequence is desirable, then the agent has performed well.

THE CONCEPT OF RATIONALITY

- When an agent is in an environment;
- Generates a sequence of actions according to the percepts it receives.
- The sequence of actions make the environment go through a sequence of states.
- If the sequence is desirable, then the agent has performed well.

Performance Measure?

THE CONCEPT OF RATIONALITY



Performance Measure?

THE CONCEPT OF RATIONALITY

"As a general rule, it is better to design performance measures according to what one actually wants in the environment, rather than according to how one thinks the agent should behave."

Performance Measure?

WHAT IS RATIONAL AT ANY GIVEN TIME DEPENDS ON FOUR THINGS

- The performance measure that defines the criterion of success.
- The agent's prior knowledge of the environment.
- The actions that the agent can perform.
- The agent's percept sequence to date.

WHAT IS RATIONAL AT ANY GIVEN TIME DEPENDS ON FOUR THINGS

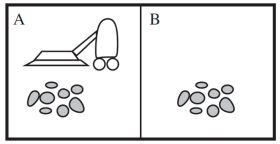
- The performance measure that defines the criterion of success.
- The agent's prior knowledge of the environment.
- The actions that the agent can perform.
- The agent's percept sequence to date.

⬇

Rational Agent

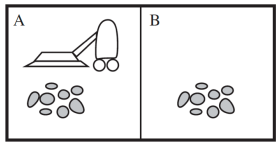
THE CONCEPT OF RATIONALITY

- **Is this a rational agent?**



THE CONCEPT OF RATIONALITY

- **Is this a rational agent?**
  1- What is the performance measure?
  2- What is known about the environment?
  3- What sensors does the agent have?
  4- What actuators does the agent have?



RATIONALITY VS. OMNISCIENCE

- An **Omniscient Agent** knows the actual outcome of its actions and can act accordingly; but omniscience is impossible in reality.
- A **Rational Agent** attempts to maximize the expected performance.

INFORMATION GATHERING

- A **Rational Agent** should perform **Information Gathering** to help maximize the expected performance.
- Doing actions in order to modify future percepts is an important part of rationality.

LEARNING

- A **Rational Agent** should also **learn** as much as possible from what it perceives. The agent's initial configuration could reflect some prior knowledge of the environment, but as the agent gains experience this may be modified and augmented.

THE NATURE OF ENVIRONMENTS

- Now that we have a definition of **Rationality**, we are almost ready to think about building **Rational Agents**.

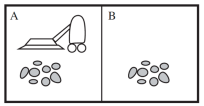
Task Environment = Problems

Rational Agents = Solutions

- **Task Environment** directly affects the appropriate design for the **Agent Program**.

SPECIFYING THE TASK ENVIRONMENT

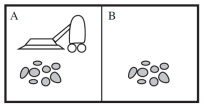
- **Task Environment:**
  1. performance measure
  2. environment
  3. agent's actuators
  4. agent's sensors



vacuum-cleaner agent

SPECIFYING THE TASK ENVIRONMENT

- **Task Environment:**
  1. Performance measure
  2. Environment
  3. agent's Actuators
  4. agent's Sensors



vacuum-cleaner agent

PEAS (Performance, Environment, Actuators, Sensors)

SPECIFYING THE TASK ENVIRONMENT

- **PEAS** description for the taxi's task environment:

SPECIFYING THE TASK ENVIRONMENT

- **PEAS** description for the taxi's task environment:

Performance Measure	Environment	Actuators	Sensors

SPECIFYING THE TASK ENVIRONMENT

- **PEAS** description for the taxi's task environment:

Performance Measure	Environment	Actuators	Sensors
Safe, fast, legal, comfortable, maximize profits	Roads, traffic, pedestrian, customers	Steering, accelerator, brake, signal, horn, display	Camera, speedometer, GPS, odometer, accelerometer, engine sensor, keyboard/mic

PROPERTIES OF TASK ENVIRONMENTS

- Task environment determines the agent design
- No sensors at all then the environment is **Unobservable.**
- Not fully observable = **Uncertain**

Fully observable	Partially observable
- Agent sensors detect all aspects in the environment continuously. - Convenient because the agent need not maintain any internal state to keep track of the world.	- Noisy and inaccurate sensors or because parts of the state are simply missing from the sensor data. - A vacuum agent with only a local dirt sensor cannot tell whether there is dirt in other squares, and an automated taxi cannot see what other drivers are thinking.

PROPERTIES OF TASK ENVIRONMENTS

Single agent	Multiagent
- Agent solving a crossword puzzle by itself.	- Playing chess is in a two agent environment.

Single agent | Multiagent
- Agent solving a crossword puzzle by itself. | - Playing chess is in a two agent environment.

PROPERTIES OF TASK ENVIRONMENTS

- In chess, the opponent entity B is trying to maximize its performance measure, which, by the rules of chess, minimizes agent A's performance measure. Thus, chess is a **competitive multiagent environment**.
- In the taxi-driving environment, on the other hand, avoiding collisions maximizes the performance measure of all agents, so it is a **partially cooperative multiagent environment**.

PROPERTIES OF TASK ENVIRONMENTS

Deterministic	Stochastic
- The next state of the environment is determined by the current state and the action of the agent.	- The next state of the environment is independent.

Deterministic | Stochastic
- The next state of the environment is determined by the current state and the action of the agent. | - The next state of the environment is independent.

PROPERTIES OF TASK ENVIRONMENTS

- The vacuum world described is **deterministic** because the next state is determined by the current state and action.
- Taxi driving is clearly **stochastic** because one can never predict the behavior of traffic exactly; moreover, one's tires blow out and one's engine seizes up without warning.
- **Stochastic** ⟶ **Uncertain** (quantified in probabilities)

PROPERTIES OF TASK ENVIRONMENTS

Episodic	Sequential
- The agent's experience is divided into atomic episodes. - The next episode does not depend on the actions taken in previous episodes. - On an assembly line, the current decision doesn't affect whether the next part is defective.	- The next state of the environment is independent of the current state. - Chess and taxi driving are sequential: in both cases, short-term actions can have long-term consequences.

- **Episodic environments** are much simpler than **sequential environments** because the agent does not need to think ahead.

PROPERTIES OF TASK ENVIRONMENTS

Static	Dynamic
- Static environments the agent needs not keep looking at the world while it is deciding on an action, nor need it worries about the passage of time. - Crossword puzzles are static	- Dynamic environments continuously ask the agent what it wants to do; if it hasn't decided yet, that counts as deciding to do nothing. - Taxi driving is clearly dynamic: the other cars and the taxi itself keep moving while the driving algorithm dithers about what to do next.

- If the environment itself does not change with the passage of time but the agent's performance score does ⟶ **Semi-dynamic**

PROPERTIES OF TASK ENVIRONMENTS

- The discrete/continuous distinction applies to the state of the environment, to the way time is handled, and to the percepts and actions of the agent.

Discrete	Continuous
- The chess environment has a finite number of distinct states, chess also has a discrete set of percepts and actions. - Input from digital cameras is discrete.	- Taxi driving is a continuous-state and continuous-time problem: the speed and location of the taxi and of the other vehicles sweep through a range of continuous values and do so smoothly over time.

PROPERTIES OF TASK ENVIRONMENTS

Known	Unknown
- The outcomes for all actions are given.	- The agent will have to learn how it works in order to make good decisions.

PROPERTIES OF TASK ENVIRONMENTS

Fully observable	Partially observable
Single agent	Multiagent
Deterministic	Stochastic
Episodic	Sequential
Static	Dynamic
Discrete	Continuous
Known	Unknown

TASK ENVIRONMENTS AND THEIR CHARACTERISTICS

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Medical diagnosis						

Can the agent detect all aspects in the environment continuously?

Or

Are parts of the state are simply missing and not observable?

TASK ENVIRONMENTS AND THEIR CHARACTERISTICS

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Medical diagnosis	Partially					

Is it a single agent?

Or

A multiagent task environment?

TASK ENVIRONMENTS AND THEIR CHARACTERISTICS

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Medical diagnosis	Partially	Single				

Is the next state determined by the current state and action?

Or

The behavior can never be predicted exactly (Stochastic)?

TASK ENVIRONMENTS AND THEIR CHARACTERISTICS

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Medical diagnosis	Partially	Single	Stochastic			

Does the next episode depend on the actions taken in previous episodes?

Or

Could the current decision affect all future decisions (Sequential)?

TASK ENVIRONMENTS AND THEIR CHARACTERISTICS

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Medical diagnosis	Partially	Single	Stochastic	Sequential		

The agent needs not keep looking at the world while it is deciding on an action, nor need it worries about the passage of time?

Or

Continuously ask the agent what it wants to do; if it hasn't decided yet, that counts as deciding to do nothing.?

TASK ENVIRONMENTS AND THEIR CHARACTERISTICS

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Medical diagnosis	Partially	Single	Stochastic	Sequential	Dynamic	

Is the state of the environment a finite number of distinct states?

Or

Is it a continuous-time problem?

TASK ENVIRONMENTS AND THEIR CHARACTERISTICS

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Medical diagnosis	Partially	Single	Stochastic	Sequential	Dynamic	Continuous

Are the outcomes all given (Known)?

Or

The agent will have to learn how it works in order to make good decisions (Unknown)?

TASK ENVIRONMENTS AND THEIR CHARACTERISTICS

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Image Analysis						

Can the agent detect all aspects in the environment continuously?

Or

Are parts of the state are simply missing and not observable?

TASK ENVIRONMENTS AND THEIR CHARACTERISTICS

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Image Analysis	Fully					

Is it a single agent?

Or

A multiagent task environment?

TASK ENVIRONMENTS AND THEIR CHARACTERISTICS

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Image Analysis	Fully	Single				

Is the next state determined by the current state and action?

Or

The behavior can never be predicted exactly (Stochastic)?

TASK ENVIRONMENTS AND THEIR CHARACTERISTICS

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Image Analysis	Fully	Single	Deterministic			

Does the next episode depend on the actions taken in previous episodes?

Or

Could the current decision affect all future decisions (Sequential)?

TASK ENVIRONMENTS AND THEIR CHARACTERISTICS

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Image Analysis	Fully	Single	Deterministic	Episodic		

The agent needs not keep looking at the world while it is deciding on an action, nor need it worries about the passage of time?

Or

Continuously ask the agent what it wants to do; if it hasn't decided yet, that counts as deciding to do nothing.?

Or

The environment itself does not change with the passage of time but the agent's performance score does (**Semi-dynamic)?**

TASK ENVIRONMENTS AND THEIR CHARACTERISTICS

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Image Analysis	Fully	Single	Deterministic	Episodic	Semi-Dynamic	

Is the state of the environment a finite number of distinct states?

Or

Is it a continuous-time problem?

TASK ENVIRONMENTS AND THEIR CHARACTERISTICS

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Image Analysis	Fully	Single	Deterministic	Episodic	Semi-Dynamic	Discrete

Are the outcomes all given (Known)?

Or

The agent will have to learn how it works in order to make good decisions (Unknown)?

TASK ENVIRONMENTS AND THEIR CHARACTERISTICS

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Interactive English Tutor						

Can the agent detect all aspects in the environment continuously?

Or

Are parts of the state are simply missing and not observable?

TASK ENVIRONMENTS AND THEIR CHARACTERISTICS

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Interactive English Tutor	Partially					

Is it a single agent?

Or

A multiagent task environment?

TASK ENVIRONMENTS AND THEIR CHARACTERISTICS

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Interactive English Tutor	Partially	Multi				

Is the next state determined by the current state and action?

Or

The behavior can never be predicted exactly (Stochastic)?

TASK ENVIRONMENTS AND THEIR CHARACTERISTICS

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Interactive English Tutor	Partially	Multi	Stochastic			

Does the next episode depend on the actions taken in previous episodes?

Or

Could the current decision affect all future decisions (Sequential)?

TASK ENVIRONMENTS AND THEIR CHARACTERISTICS

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Interactive English Tutor	Partially	Multi	Stochastic	Sequential		

The agent needs not keep looking at the world while it is deciding on an action, nor need it worries about the passage of time?

Or

Continuously ask the agent what it wants to do; if it hasn't decided yet, that counts as deciding to do nothing.?

Or

The environment itself does not change with the passage of time but the agent's performance score does (**Semi-dynamic)?**

TASK ENVIRONMENTS AND THEIR CHARACTERISTICS

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Interactive English Tutor	Partially	Multi	Stochastic	Sequential	Semi-Dynamic	

Is the state of the environment a finite number of distinct states?

Or

Is it a continuous-time problem?

TASK ENVIRONMENTS AND THEIR CHARACTERISTICS

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Interactive English Tutor	Partially	Multi	Stochastic	Sequential	Semi-Dynamic	Continuous

Are the outcomes all given (Known)?

Or

The agent will have to learn how it works in order to make good decisions (Unknown)?

TASK ENVIRONMENTS AND THEIR CHARACTERISTICS

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Crossword puzzle	Fully	Single	Deterministic	Sequential	Static	Discrete
Chess with a clock	Fully	Multi	Deterministic	Sequential	Semi	Discrete
Poker	Partially	Multi	Stochastic	Sequential	Static	Discrete
Backgammon	Fully	Multi	Stochastic	Sequential	Static	Discrete
Taxi driving	Partially	Multi	Stochastic	Sequential	Dynamic	Continuous
Medical diagnosis	Partially	Single	Stochastic	Sequential	Dynamic	Continuous
Image analysis	Fully	Single	Deterministic	Episodic	Semi	Continuous
Part-picking robot	Partially	Single	Stochastic	Episodic	Dynamic	Continuous
Refinery controller	Partially	Single	Stochastic	Sequential	Dynamic	Continuous
Interactive English tutor	Partially	Multi	Stochastic	Sequential	Dynamic	Discrete

Figure 2.6 Examples of task environments and their characteristics.

TASK ENVIRONMENTS AND THEIR CHARACTERISTICS

Answers are not always cut and dried!

Answers depend on how the task environment is defined!

THE STRUCTURE OF AGENTS

- We have talked about **Agents** by describing **behavior** (the action that is performed after any given sequence of percepts).
- The job of AI is to design an **Agent Program** that implements the **Agent Function** (the mapping from percepts to actions).

Agent = Architecture + Program

THE STRUCTURE OF AGENTS

Agent = Architecture + Program

- The program we choose has to be one that is appropriate for the architecture. If the program is going to recommend actions like **Walk**, the **architecture** had better have **legs**.

THE STRUCTURE OF AGENTS

- Agent programs take the current percept as input from the sensors and return an action to the actuators.

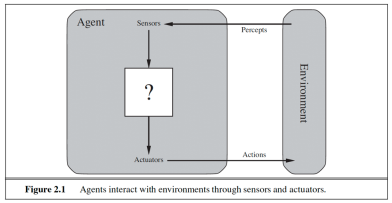


Figure 2.1 Agents interact with environments through sensors and actuators.

THE STRUCTURE OF AGENTS

- **Agent Program**: Takes the current percept as input.
- **Agent Function**: Takes the entire percept history.

THE STRUCTURE OF AGENTS

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

Figure 2.8 The agent program for a simple reflex agent in the two-state vacuum environment. This program implements the agent function tabulated in Figure 2.3.

THE STRUCTURE OF AGENTS

1- Simple Reflex Agents

2- Model-based Reflex Agents

3- Goal-based Agents

4- Utility-based Agents

5- Learning Agents

SIMPLE REFLEX AGENTS

- Agents select actions on the basis of the current percept, ignoring the rest of the percept history.
- **Condition–Action Rule**

  if *car-in-front-is-braking* **then** *initiate-braking*.

SIMPLE REFLEX AGENTS

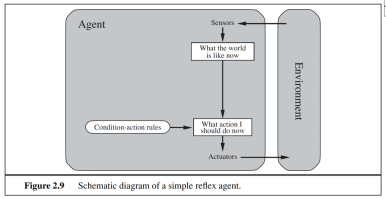


Figure 2.9 Schematic diagram of a simple reflex agent.

```
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
```

Figure 2.10 A simple reflex agent. It acts according to a rule whose condition matches the current state, as defined by the percept.

SIMPLE REFLEX AGENTS

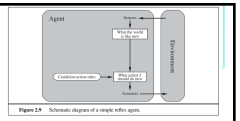
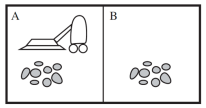


Figure 2.9 Schematic diagram of a simple reflex agent.

- **Simple reflex agents** have the admirable property of being **simple,** but they turn out to be of **limited intelligence.**

SIMPLE REFLEX AGENTS



- Such an agent has just two possible percepts: [Dirty] and [Clean].
- It can Suck in response to [Dirty]; what should it do in response to [Clean]?
- Moving Left fails (forever) if it happens to start in square A
- Moving Right fails (forever) if it happens to start in square B.
- **Infinite loops are often unavoidable for simple reflex agents** operating in partially observable environments.

MODEL-BASED REFLEX AGENTS

- Updating the internal state information as time goes by is called a **model of the world.**
- An agent that uses such a model is called a **Model-Based Agent.**

MODEL-BASED REFLEX AGENTS

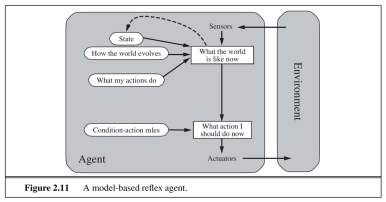
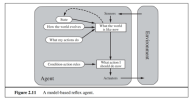


Figure 2.11 A model-based reflex agent.

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
```

Figure 2.12 A model-based reflex agent. 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.

- Is responsible for creating the new internal state description

GOAL-BASED AGENTS

- Knowing something about the current state of the environment is not always enough to decide what to do.
- The agent needs some sort of **Goal Information** that describes situations that are desirable.

GOAL-BASED AGENTS

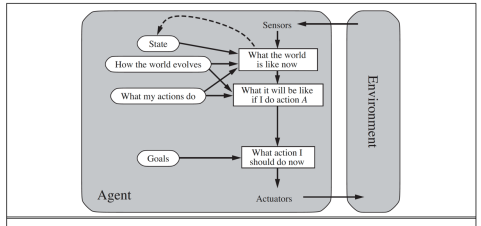


Figure 2.13 A model-based, goal-based agent. It keeps track of the world state as well as a set of goals it is trying to achieve, and chooses an action that will (eventually) lead to the achievement of its goals.

UTILITY-BASED AGENTS

- Goals just provide a crude binary distinction between "happy" and "unhappy" states (**Utility**).
- **Utility func**tion (performance measure).
- Agents that choose actions to maximize their **utility** will be **rational**.

UTILITY-BASED AGENTS

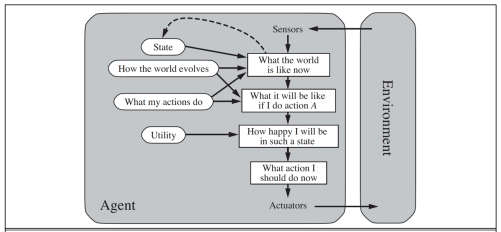


Figure 2.14 A model-based, utility-based agent. It uses a model of the world, along with a utility function that measures its preferences among states of the world. Then it chooses the action that leads to the best expected utility, where expected utility is computed by averaging over all possible outcome states, weighted by the probability of the outcome.

UTILITY-BASED AGENTS

- When there are conflicting goals, only some of which can be achieved (trade-off: speed and safety).
- When there are several goals that the agent aim for, none of which can be achieved with certainty.

LEARNING AGENTS

- **Learning allows the agent to operate in initially unknown environments and to become more competent than its initial knowledge alone might allow.**

LEARNING AGENTS

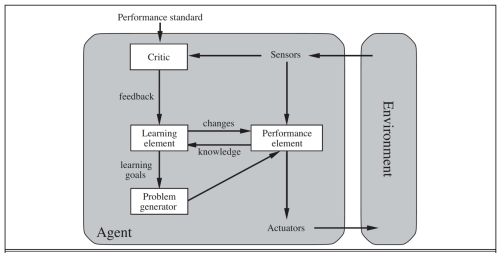


Figure 2.15 A general learning agent.

LEARNING AGENTS

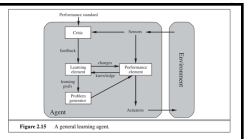


Figure 2.15 A general learning agent.

- **Learning element:** responsible for making improvements.
- **Performance element:** responsible for selecting external actions.
- **Critic:** receives feedback from the **learning element** on how the agent is doing and determines how the performance element should be modified to do better in the future.
- **Problem generator:** responsible for suggesting actions that will lead to new and informative experiences.

THE STRUCTURE OF AGENTS

1- Simple Reflex Agents

2- Model-based Reflex Agents

3- Goal-based Agents

4- Utility-based Agents

5- Learning Agents

SUMMARY

- An **agent** is something that perceives and acts in an environment.
- The **agent function** for an agent specifies the action taken by the agent in response to any percept sequence.
- **Rational agent** act to maximize the expected value of the performance measure.
- Task environments vary along several dimensions: fully or partially observable, single-agent or multiagent, deterministic or stochastic, episodic or sequential, static or dynamic, discrete or continuous, and known or unknown.
- The **agent program** implements the **agent function**.

SUMMARY

- **Simple reflex** agents respond directly to percepts, whereas **model-based reflex** agents maintain internal state to track aspects of the world that are not evident in the current percept. **Goal-based** agents act to achieve their goals, and **utility-based** agents try to maximize their own expected "happiness."
- All agents can improve their performance through **learning.**

READINGS

- 2.1, 2.2, 2.3, 2.4 (2.4.1 – 2.4.6)
- Other interesting readings (optional):

  2.4.7

ASSIGNMENT# 1

Check the course website for Assignment#1