

MODULE-1

COURSE NAME:ARTIFICIAL
INTELLIGENCE-BCS515B





Course objectives:

1. Learn the basic principles and theories underlying artificial intelligence, including machine learning, neural networks, natural language processing, and robotics.
2. Apply AI techniques to solve real-world problems, including search algorithms, optimization, and decision-making processes.
3. Understand the ethical, legal, and societal implications of AI, including topics such as bias, fairness, accountability, and the Impact of AI on the workforce and privacy

MODULE:1

- Introduction
- Intelligent Agents

MODULE:2

- Problem-solving

MODULE:3

- Problem-solving
- Logical Agents

MODULE:4

- First Order Logic
- Inference in First Order Logic

MODULE:5

- Inference in First Order Logic
- Classical Planning



Course outcomes (Course Skill Set)

At the end of the course, the student will be able to:

1. Explain the architecture and components of intelligent agents, including their interaction with the AI environment.
2. Apply problem-solving agents and various search strategies to solve a given problem.
3. Illustrate logical reasoning and knowledge representation using propositional and first-order logic.
4. Demonstrate proficiency in representing knowledge and solving problems using first-order logic.
5. Describe classical planning in the context of artificial intelligence, including its goals, constraints, and applications in problem-solving.

What is AI?

Thinking Humanly

“The exciting new effort to make computers think . . . *machines with minds*, in the full and literal sense.” (Haugeland, 1985)

“[The automation of] activities that we associate with human thinking, activities such as decision-making, problem solving, learning . . .” (Bellman, 1978)

Acting Humanly

“The art of creating machines that perform functions that require intelligence when performed by people.” (Kurzweil, 1990)

“The study of how to make computers do things at which, at the moment, people are better.” (Rich and Knight, 1991)

Thinking Rationally

“The study of mental faculties through the use of computational models.”
(Charniak and McDermott, 1985)

“The study of the computations that make it possible to perceive, reason, and act.”
(Winston, 1992)

Acting Rationally

“Computational Intelligence is the study of the design of intelligent agents.” (Poole *et al.*, 1998)

“AI . . . is concerned with intelligent behavior in artifacts.” (Nilsson, 1998)

Acting humanly: The Turing Test approach

- The Turing Test, proposed by Alan Turing (1950), was designed to provide a satisfactory TURING TEST operational definition of intelligence.
- A computer passes the test if a human interrogator, after posing some written questions, cannot tell whether the written responses come from a person or from a computer.

The computer would need to possess the following capabilities:

- natural language processing
- knowledge representation
- automated reasoning
- machine learning



Total Turing Test – Overview

Original Turing Test avoided physical interaction

Focused on reasoning and intelligence, not physical form.

Total Turing Test includes:

Video signal – to test **perceptual abilities**

Ability to **pass physical objects** through a hatch

To pass the Total Turing Test, AI needs:

Computer Vision – for object perception

Robotics – for manipulating objects and physical movement

Thinking Humanly: The Cognitive Modeling Approach

□ Understanding How Humans Think

To claim a program "thinks like a human", we must understand **human thinking mechanisms**

Three key methods to study the mind:

- **Introspection** – Observing one's own thoughts
- **Psychological experiments** – Watching human behavior
- **Brain imaging** – Monitoring brain activity during tasks

Example: General Problem Solver (GPS)

Created by **Allen Newell & Herbert Simon (1961)**

Goal: Not just correct results, but to **match human reasoning steps**

Compared system output with **human problem-solving traces**

Cognitive Science combines: AI's **computational models**

Psychology's **experimental methods**

Builds **testable theories** of human cognition



Thinking rationally: The “laws of thought” approach

- These laws of thought were supposed to govern the operation of the mind; their study initiated the field called logic.

There are two main obstacles to this approach.

- First, it is not easy to take informal knowledge and state it in the formal terms required by logical notation, particularly when the knowledge is less than 100% certain.
- Second, there is a big difference between solving a problem “in principle” and solving it in practice.

Acting rationally: The rational agent approach

➤ What is an Agent?

An **agent** is anything that **acts** as **computer agent** should:

- Operate **autonomously**
- Perceive** its environment
- Persist** over time
- Adapt** to changes
- Pursue goals**



➤ What is a Rational Agent?

Acts to achieve the **best outcome**, or best **expected outcome** under uncertainty

Rationality can involve **logical reasoning**, but: It's not limited to inference
Sometimes, **reflex actions** are more rational.



THE STATE OF ART

- Robotic vehicles
- Speech recognition
- Autonomous planning and scheduling
- Game playing
- Spam fighting
- Logistics planning
- Robotics
- Machine Translation



➤ Robotic Vehicles

- **Stanley** (Volkswagen Touareg) won the **2005 DARPA Grand Challenge**
Drove 132 miles through desert terrain autonomously
- **Boss** (CMU) won the **2007 Urban Challenge** Navigated city streets, obeyed traffic rules, avoided pedestrians

➤ Speech Recognition

- United Airlines uses **automated speech recognition** to guide flight booking conversations

➤ Autonomous Planning & Scheduling

- **NASA's Remote Agent** (2000): First onboard autonomous system for spacecraft
- **MAPGEN**: Plans daily operations for Mars Rovers
- **MEXAR2**: Managed logistics & science planning for ESA's **Mars Express**

➤ Game Playing

- **IBM Deep Blue** defeated world chess champion **Garry Kasparov** in 1997
 - Described as "a new kind of intelligence"
 - Increased IBM stock by **\$18 billion**

➤ Spam Fighting

- AI filters over a billion spam emails daily
 - Uses **learning algorithms** to adapt to new spamming tactics
 - Saves users from 80–90% junk mail

➤ Logistics Planning

- **DART system** used during the **1991 Gulf War**
 - Managed 50,000+ vehicles, cargo, and personnel
 - AI created complex plans in **hours** instead of weeks

➤ Robotics

- **Roomba**: 2+ million robotic vacuum cleaners sold
- **PackBot**: Deployed in war zones to:
 - Handle hazardous materials
 - Clear explosives
 - Locate snipers

➤ Machine Translation

- Translates Arabic to English using **statistical models**
 - Example: Headline translation without knowing Arabic
 - Built from **2 trillion words** of English data
 - Driven by **machine learning**, not human linguists

Intelligent Agents

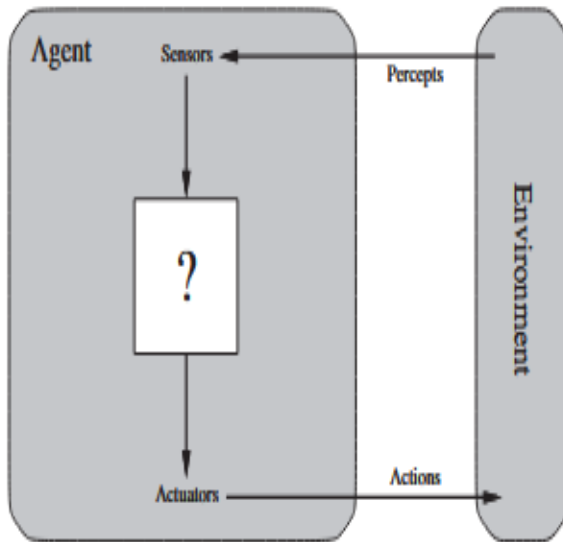


Figure 2.1 Agents interact with environments through sensors and actuators.

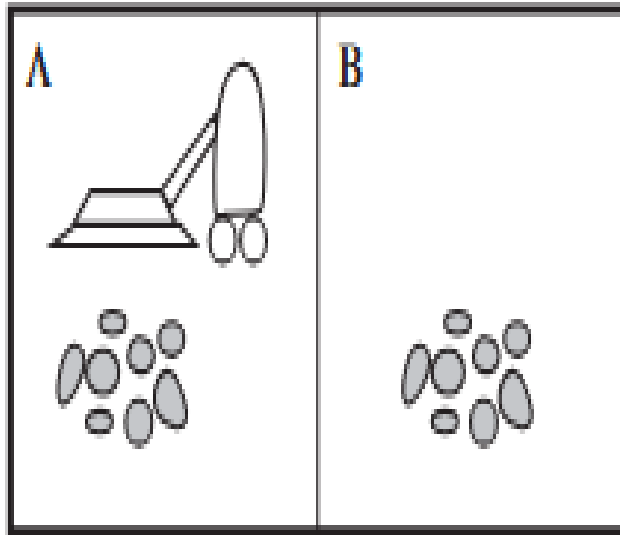
- An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators. This simple idea is illustrated in Figure 2.1. A human agent has eyes, ears, and other organs for sensors and hands, legs, vocal tract, and so on for actuators.
- We use the term percept to refer to the agent's perceptual inputs at any given instant. An agent's percept sequence is the complete history of everything the agent has ever perceived.



Agents and Environment

- Mathematically speaking, we say that an agent's behavior is described by the agent function that maps any given percept sequence to an action.
- We can imagine tabulating the agent function that describes any given agent;
- Given an agent to experiment with, we can, in principle, construct this table by trying out all possible percept sequences and recording which actions the agent does in response. The table is, of course, an external characterization of the agent. Internally, the agent function for an artificial agent will be implemented by an agent program.
- It is important to keep these two ideas distinct. The agent function is an abstract mathematical description; the agent program is a concrete implementation, running within some physical system.

The Vacuum-Cleaner World



This world is so simple that we can describe everything that happens;
it's also a made-up world, so we can invent many variations. This particular world has just two locations: squares A and B. The vacuum agent perceives which square it is in and whether there is dirt in the square. It can choose to move left, move right, suck up the dirt, or do nothing. One very simple agent function is the following: if the current square is dirty, then suck; otherwise, move to the other square.

Program implements the agent function tabulated in
Fig. 2.3

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

⋮

⋮

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

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

Concept of Rationality

- A rational agent is one that does the right thing—conceptually speaking, every entry in the RATIONAL AGENT table for the agent function is filled out correctly
- 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. This leads to a definition of a rational agent

Example of a rational agent

● Performance measure

- Awards one point for each clean square
 - at each time step, over 10000 time steps

● Prior knowledge about the environment

- The geography of the environment
- Only two squares
- The *effect* of the actions

Omniscience

● An omniscient agent

- Knows the *actual* outcome of its actions in advance
- No other possible outcomes
- However, impossible in real world

● An example

- crossing a street but died of the fallen cargo door from 33,000ft → irrational?

Learning

- Does a rational agent depend on only current percept?
 - No, the past percept sequence should also be used
 - This is called learning
 - After experiencing an episode, the agent
 - should adjust its behaviors to perform better for the same job next time.

Autonomy

- If an agent just relies on the prior knowledge of its designer rather than its own percepts then the agent lacks autonomy

A rational agent should be autonomous- it should learn what it can to compensate for partial or incorrect prior knowledge.

- E.g., a clock
 - No input (percepts)
 - Run only but its own algorithm (prior knowledge)
 - No learning, no experience, etc.

Software Agents

- Sometimes, the environment may not be the real world
 - E.g., flight simulator, video games, Internet
 - They are all artificial but very complex environments
 - Those agents working in these environments are called
 - Software agent (softbots)
 - Because all parts of the agent are software



NATURE OF ENVIRONMENT

The rationality of the simple vacuum-cleaner agent, we had to specify the performance measure, the environment, and the agent's actuators and sensors. We group all these under the heading of the task environment. For the acronymically minded, we call this the **PEAS** (Performance, Environment, Actuators, Sensors) description. In designing an PEAS agent, the first step must always be to specify the task environment as fully as possible

Agent Type	Performance Measure	Environment	Actuators	Sensors
Taxi driver	Safe, fast, legal, comfortable trip, maximize profits	Roads, other traffic, pedestrians, customers	Steering, accelerator, brake, signal, horn, display	Cameras, sonar, speedometer, GPS, odometer, accelerometer, engine sensors, keyboard

Figure 2.4 PEAS description of the task environment for an automated taxi.

Properties of task environments

● Fully observable vs. Partially observable

- If an agent's sensors give it access to the complete state of the environment at each point in time then the environment is effectively and fully observable
 - if the sensors detect all aspects
 - That are relevant to the choice of action

- Partially observable
- An environment might be Partially observable because of noisy and inaccurate sensors or because parts of the state are simply missing from the sensor data.
- Example:
 - A local dirt sensor of the cleaner cannot tell
 - Whether other squares are clean or not

Properties of task environments

■ Deterministic vs. stochastic

- next state of the environment Completely determined by the current state and the actions executed by the agent, then the environment is deterministic, otherwise, it is Stochastic.
- Strategic environment: deterministic except for actions of other agents
 - -Cleaner and taxi driver are:
 - Stochastic because of some unobservable aspects □
noise or unknown

Properties of task environments

● Episodic vs. sequential

- An episode = agent's single pair of perception & action
- The quality of the agent's action does not depend on other episodes
 - Every episode is independent of each other
- Episodic environment is simpler
 - The agent does not need to think ahead

● Sequential

- Current action may affect all future decisions
 - -Ex. Taxi driving and chess.

Properties of task environments

● Static vs. dynamic

- A dynamic environment is always changing over time
 - E.g., the number of people in the street
- While static environment
 - E.g., the destination

● Semidynamic

- environment is not changed over time
- but the agent's performance score does

STRUCTURE OF AGENTS

- Agent = architecture + program
 - Architecture = some sort of computing device (sensors + actuators)
 - (Agent) Program = some function that implements the agent mapping = “?”
 - Agent Program = Job of AI

Agent Programs

- Despite of huge size, look up table does what we want.
- The key challenge of AI
 - Find out how to write programs that, to the extent possible, produce rational behavior
 - From a small amount of code
 - Rather than a large amount of table entries
 - E.g., a five-line program of Newton's Method
 - V.s. huge tables of square roots, sine, cosine, ...

Types of agent programs

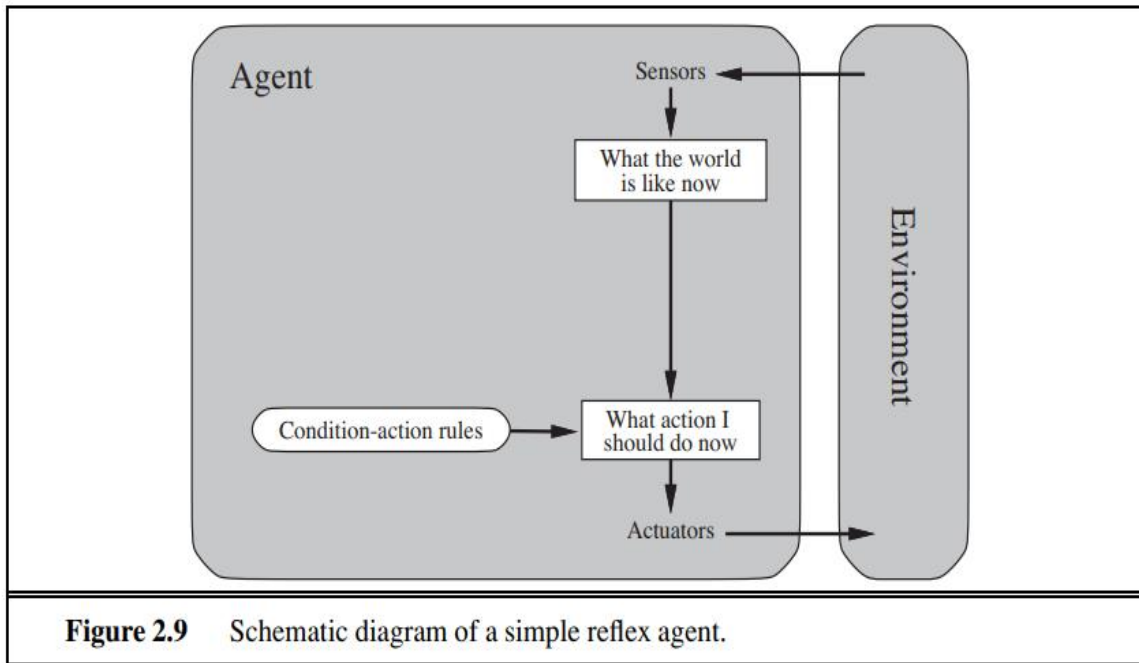
● Four types

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

Simple reflex agents

- It uses just ***condition-action rules***
 - The rules are like the form “if ... then ...”
 - efficient but have narrow range of applicability
 - Because knowledge sometimes cannot be stated explicitly
 - Work only
 - if the environment is fully observable

Simple reflex agents



- The Simple reflex agents are the simplest agents. These agents take decisions on the basis of the current precepts and ignore the rest of the percept history.
- These agents only succeed in the fully observable environment.
- The Simple reflex agent does not consider any part of precepts history during their decision and action process

Model-based Reflex Agents

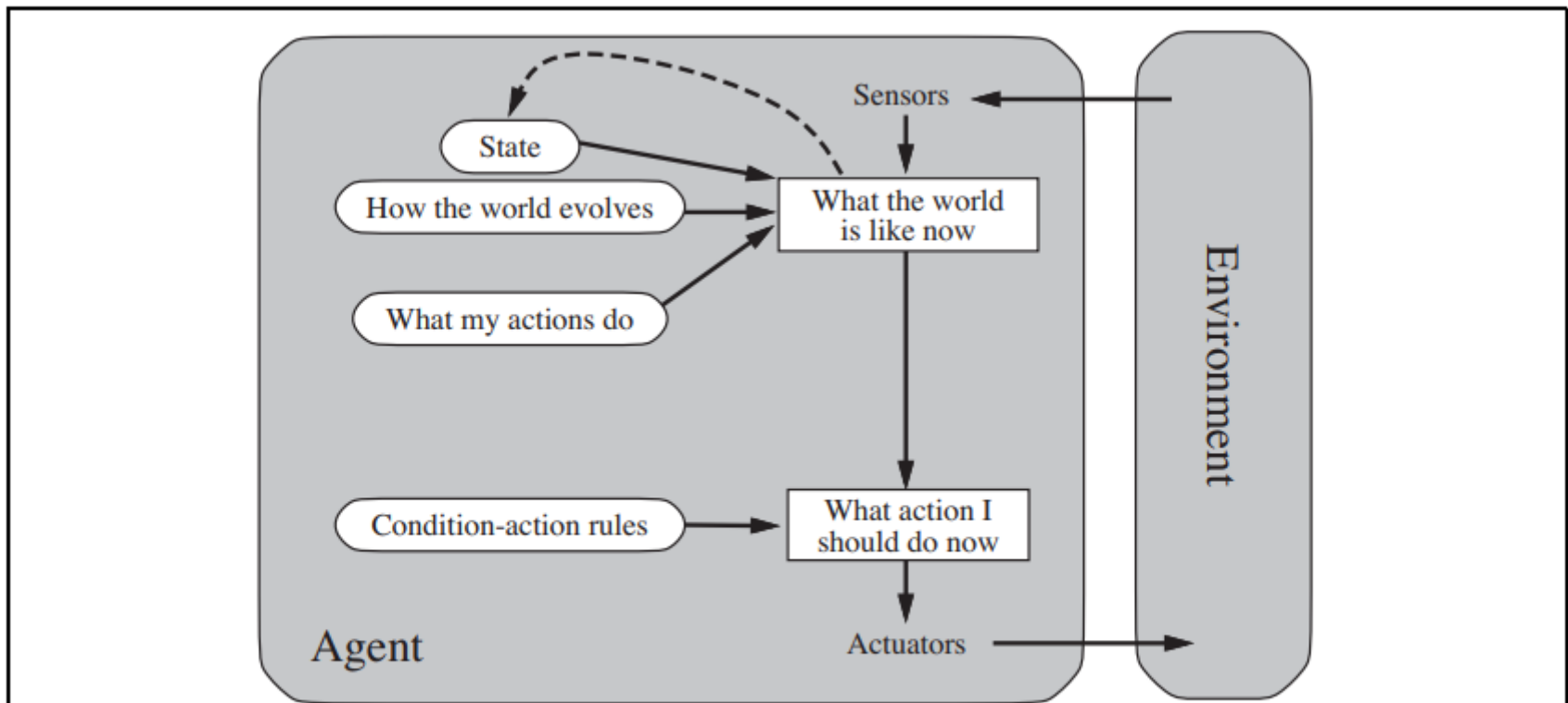


Figure 2.11 A model-based reflex agent.

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.

Goal-based agents

- Current state of the environment is always not enough
- The goal is another issue to achieve
 - Judgment of rationality / correctness
- Actions chosen → goals, based on
 - the current state
 - the current percept

Goal-based agents

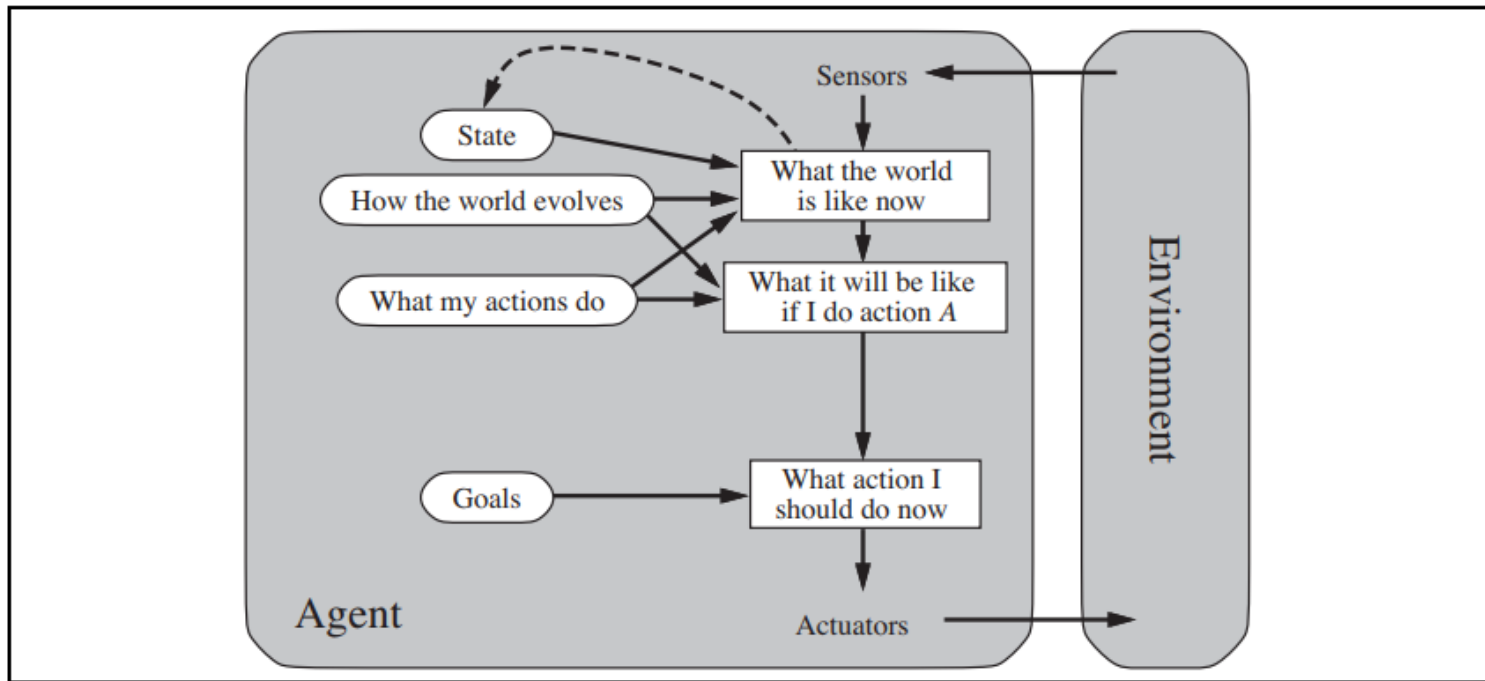


Figure 2.13 A model-based, goal-based agent. It keeps track of the world state as well as a set of goals it is trying to achieve, and chooses an action that will (eventually) lead to the achievement of its goals.

Utility-based agents

- Goals alone are not enough
 - to generate **high-quality** behavior
 - E.g. meals in Canteen, good or not ?
- Many action sequences → the goals
 - some are better and some worse
 - If goal means success,
 - then **utility** means the degree of success (how successful it is)

Utility-based agents

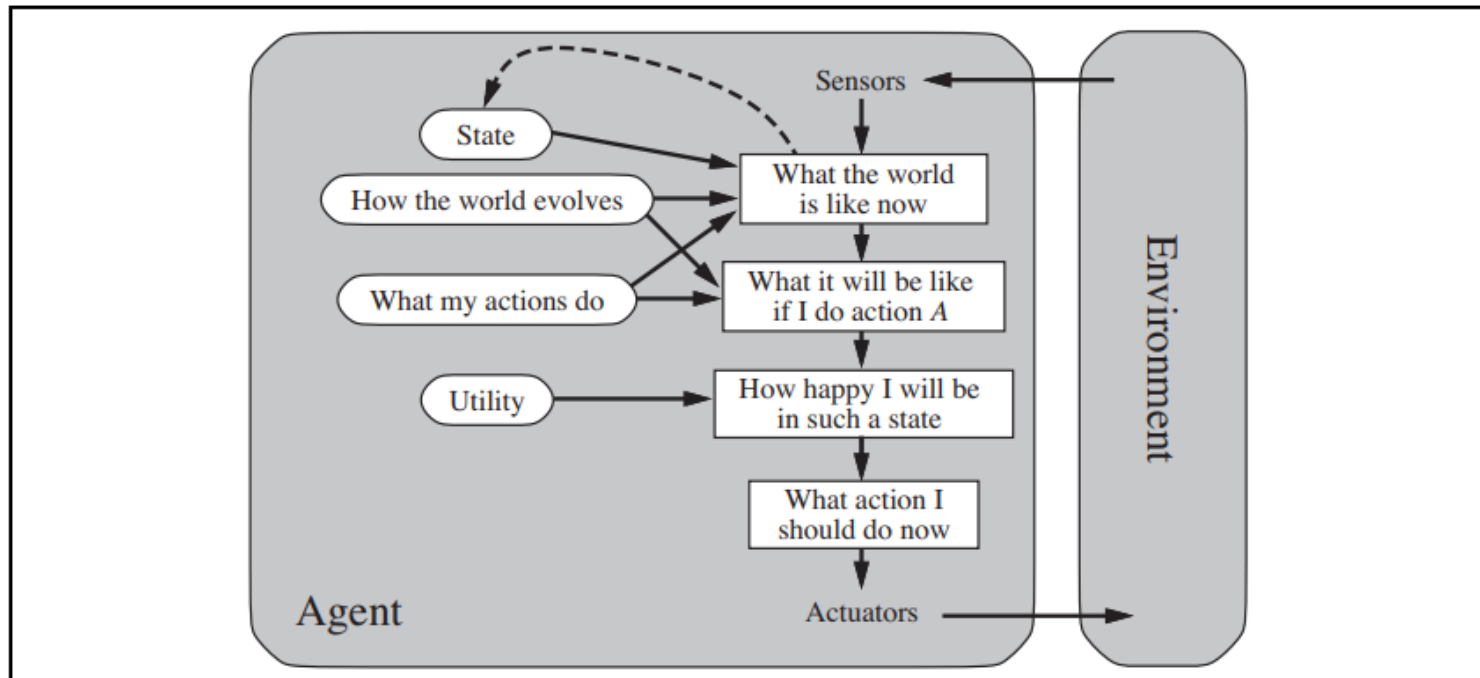
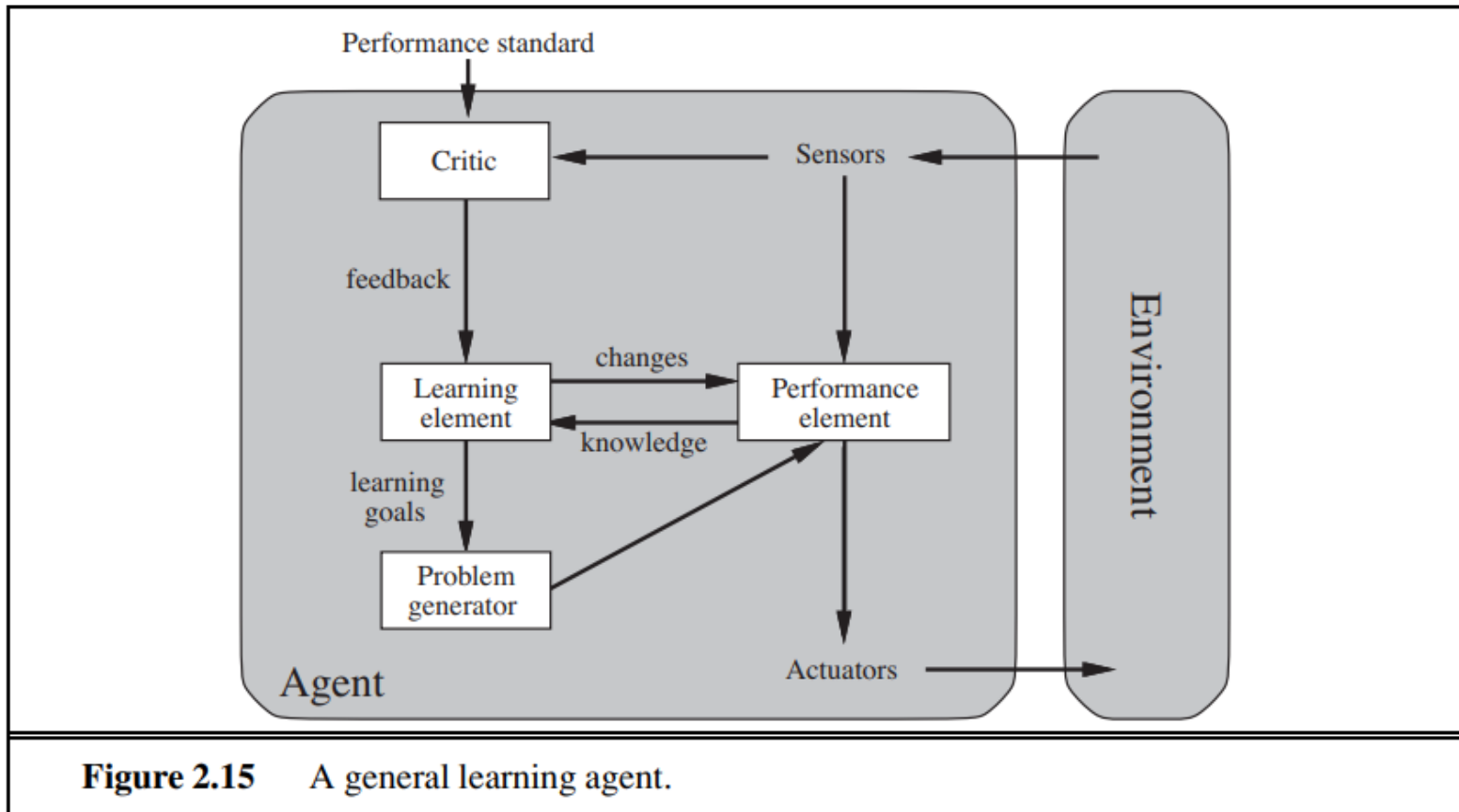
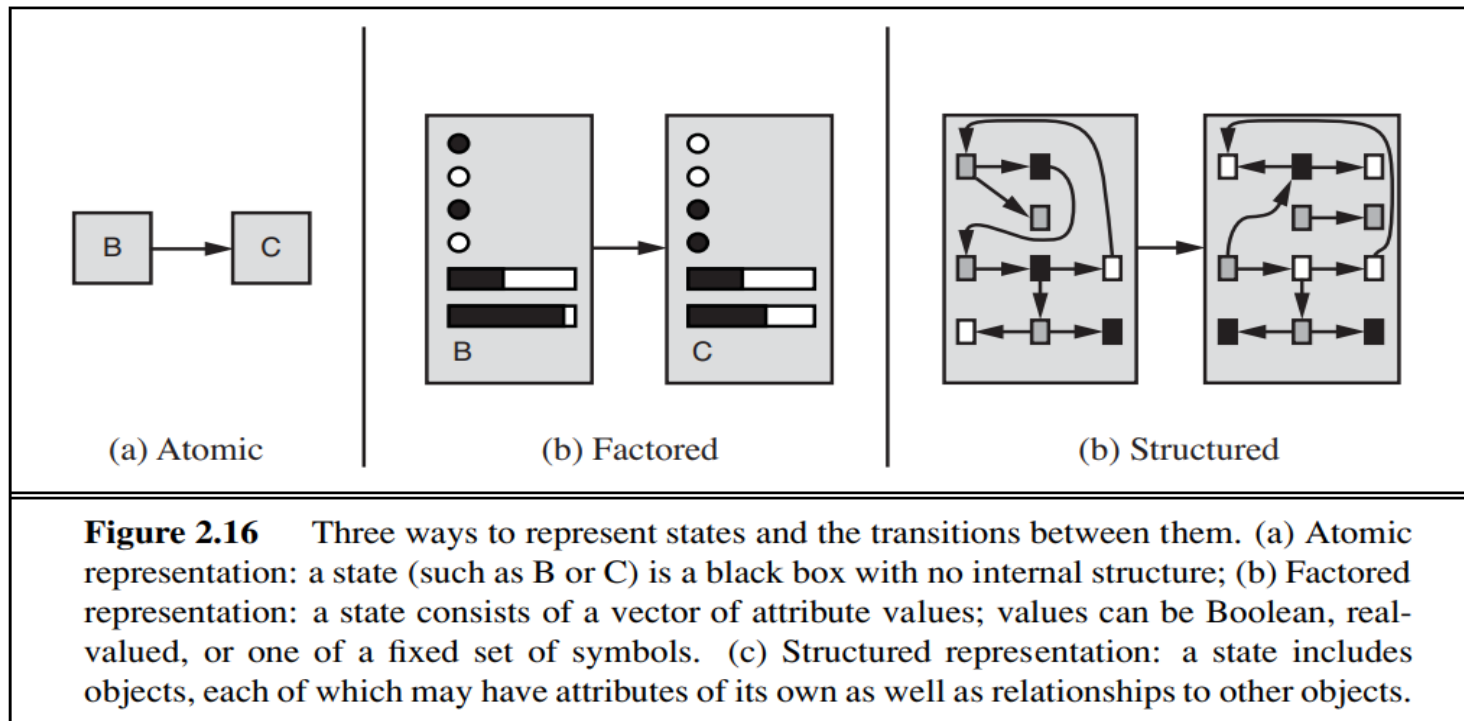


Figure 2.14 A model-based, utility-based agent. It uses a model of the world, along with a utility function that measures its preferences among states of the world. Then it chooses the action that leads to the best expected utility, where expected utility is computed by averaging over all possible outcome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

Learning Agents



How the components of agent programs work





A T M E
College of Engineering

GOLD

QS I-GAUGE
INDIAN COLLEGE RATINGS



THANK YOU