ARTIFICIAL INTELLIGENCE

UNIT – I Introduction

- ✓ **What is AI,**
- ✓**History of AI,**
- ✓**Intelligent Agents: Agents**
- ✓**Environments,**
- ✓ **Good Behavior: The Concept of Rationality**
- ✓ **The Nature of** Environments,
- ✓ The Structure of Agents.
- ✓Problem solving agents
- ✓Problem formulation

What is ARTIFICIAL INTELLIGENCE

ARTIFICIAL INTELLIGENCE ---The term was coined by McCarthy in 1956.

There are two terms

- Artificial
- Intelligence

❖ **Artificial :- Not Natural -> MAN MADE**

ARTIFICIAL INTELLIGENCE means : **Man made systems are building intelligence into them.**

Or

Artificial Intelligence is concerned with the design of intelligence in an artificial device.

❖ **INTELLIGENCE ? :**

.

❖INTELLEGENCE:

if we take human beings to be intelligent

i.e AI means having behavior which is like a humans

We consider it in two ways:

1. **To have a machine / System that behaves like a human.**
2. **AI concern with intelligence which is the ideal or the best behavior or <u>the rational behavior</u>**

about the behavior :-What type of behavior ?

1. **Thinking :**
   - Estimating
   - reasoning
   - Decision Making

   in order to come up with a solution.

2. **Action** : how the system actually acts/ behaves .

- **performance (Human like)** **Vs** **Ideal Performance (Rational)**

- **Thinking** **Vs** **(ACTION)**

What is Artificial Intelligence ?

	HUMAN	RATIONAL
Thinking	Systems that think like humans	Systems that think rationally
ACTIONS	Systems that acts like human	Systems that act rationally

What is AI? Four Approches

Human-centered approaches (Empirical Science) that involves : Hypothesis and Experimental confirmation

Acting Humanly: The Turing Test
Thinking Humanly: Cognitive Science

Rationalist approach that involves: Combination of Mathematics and Engineering

Thinking Rationally: Laws of Thought
Acting Rationally: The Rational Agent

What is AI?

(Some Definitions of AI, Organized into 4 Categories)

Systems that think like human (cognitive science)	**Systems that think rationally** (laws of thoughts)
•"The exciting new effort to make computers thinks … *machine with minds,* in the full and literal sense" (Haugeland 1985) •"The automation of activities that we associate with human thinking, activities: decision-making, problem-solving, learning…." (Bellman 1978)	•"The study of mental faculties through the use of computational models" (Charniak et al. 1985) •"The study of the computations that make it possible to perceive,reason, and act." (Winston 1992)
Systems that act like human (Turing Test)	**Systems that act rationally** (RATIONAL AGENT)
• "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&Knight 1991)	• A field of study that seeks to explain and emulate intelligent behavior in terms of computational processes" (Schalkol, 1990) • "AI ….. Is concerned with intelligent behavior in artifacts." (Nilsson 1998)

- ✓ In computer science, Artificial Intelligence **(AI)**, sometimes called **machine intelligence**, is intelligence demonstrated by machines, in contrast to the natural intelligence displayed by humans.

- ✓ It also defined as the field as the study of **“Intelligent agents”:** any device that perceives its environment and takes actions that maximize its chance of successfully achieving its goals.

Foundations of AI…

i. Philosophy
ii. Mathematics
iii. Economics
iv. Neuroscience
v. Psychology
vi. Computer engineering
vii. Control theory and Cybernetics
viii. Linguistics

History of Artificial Intelligence(AI)

Prehistory of AI

- 4th cent. B.C : **Aristotle studied mind & thought, defined formal logic**

  (the study of arguments.).

- 14th–16th cent : **New thought built on the idea that all natural or artificial processes could be mathematically analyzed and understood.**

- 18th cent : **distinction between mind & brain.**

The Modern Birth of AI

- **19th cent** : **Babbage's Analytical Engine proposed a general-purpose, programmable computing machine - metaphor for the brain. (Incomplete)**
- **19th-20th cent** : **Many advances in logic formalisms, including Boole's algebra, predicate calculus**
- **20th cent** : **Turing wrote seminar paper on thinking machines (1950). Marvin Minsky & John McCarthy organized the Dartmouth Conference in 1956**

Introduction

- AI is an evolving discipline with a rich historical background.
- The contributions of other fields, e.g. Mathematics, Philosophy, Psychology, Neuroscience, etc. to the development of AI have been so significant that its history is sometimes recounted from the time of Aristotle (350 B.C.)We will focus the period from 1943-present.

Overview of Chronology

A. **The gestation of AI (1943-1955)**
B. **The birth of AI (1956)**
C. **Early enthusiasm (1952-1969)**
D. **A dose of reality (1966-1973)**
E. **Knowledge-based systems (1969-1979)**
F. **AI becomes an industry (1980-present)**
G. **The return of Neural Networks (1986-present)**
H. **AI becomes a science (1987-present)**
I. **The emergence of Intelligent Agents (1995-present)**

Gestation of AI

- Warren McCulloch and Walter Pitts (1943) gave the concept of artificial neural networks. They suggested that suitably defined networks could learn.
- Alan Turing was the first to put forward a complete vision of AI in his 1950 article "Computing Machinery and Intelligence." Therein, he introduced the Turing test, machine learning. genetic algorithms, and reinforcement learning.
- Two graduate students in the Princeton mathematics department, Marvin Minsky and Dean Edmonds, built the first neural network computer in 1951 called SNARC. (**SNARC** :Stochastic Neural Analog Reinforcement Calculator)

The birth of AI (1956)

- U.S. researchers interested in automata theory, neural nets, and the study of intelligence were brought together in a workshop at Dartmouth in the summer of 1956 where John McCarthy proposed the name for the field as "**Artificial Intelligence.**"

Early enthusiasm (1952-1969)

- Starting in 1952. Arthur Samuel wrote a series of programs for checkers (draughts) that eventually learned to play at a strong amateur level.
- McCarthy in 1958 defined a high level language LISP, a dominant AI programming language
- At IBM, Nathaniel Rochester and his colleagues produced some of the first AI programs. Geometry Theorem Prover.
- Newell and Simon developed Logic Theorist (1963)
- James Slagle's SAINT program (1963) solved integration problems.
- Daniel Bobrow's STUDENT program (1967) solved algebra problems.

.A Dose of Reality

- In 1966 the failure of machine translation project brought an end to the US government's funding of the project.
- Minsky and Papert's book: *'Perceptrons'* (1969) proved that. although perceptrons (a simple form of neural network) could be shown to learn anything they were capable of representing, they could represent very little.
- In 1973 Lighthill report entailed cutting of British funding to AI research in all but two universities in the Great Britain.

Knowledge-based Systems (1969-1979)

- MYCIN was developed in mid 1970s at Stanford that diagnosed blood infections. MYCIN was able to perform as well as some experts, and considerably better than junior doctors.

AI becomes an industry **(1980-present)**

- The first successful commercial expert system R1 began operation at the Digital Equipment Corporation (McDermott, 1982).
- Nearly every major U.S. corporation had its own AI group and was either using or investigating expert systems.
- In 1981, the Japanese announced the "Fifth Generation" project, a 10-year plan to build intelligent computers running Prolog.
- United States formed the Microelectronics and Computer Technology Corporation (MCC) as a research consortium.

- Alvey report reinstated the funding that was cut by the Lighthill report.
- The AI industry boomed from a few million dollars in 1980 to billions of dollars in 1988. Soon after that came a period called the "AI Winter." in which many companies suffered as they failed to deliver on extravagant promises.
- Judea Pearl's (1988) *Probabilistic Reasoning in Intelligent Systems* led to a new acceptance of probability and decision theory in AI.
- In terms of methodology, AI has finally came firmly under the scientific method. To be accepted, hypotheses must be subjected to rigorous empirical experiments, and the results must be analyzed statistically for their importance (Cohen. 1995).
- A better understanding of the problems and their complexity properties, combined with increased mathematical sophistication, has led to workable research agendas and robust methods.

The emergence of Intelligent Agents (1995-present)

- The work of Allen Newell, John Laird, and Paul Rosenbloom on SOAR (Newell. 1990: Laird *el al.,* 1987) is the best-known example of a complete agent architecture.
- AI technologies underlie many Internet tools, such as search engines, recommender systems, and web site construction systems.
- A hundred million miles from Earth, NASA's Remote Agent program became the first on-board autonomous planning program to control the scheduling of operations for a .spacecraft (Jonsson *el at.,* 2000).

Agents in Artificial Intelligence

An AI system can be defined as the study of the rational agent and its environment. The agents sense the environment through sensors and act on their environment through actuators. An AI agent can have mental properties such as knowledge, belief, intention, etc.

What is an Agent?

An agent can be anything that perceive its environment through sensors and act upon that environment through actuators. An Agent runs in the cycle of **perceiving**, **thinking**, and **acting**. An agent can be:

- **Human-Agent:** A human agent has eyes, ears, and other organs which work for sensors and hand, legs, vocal tract work for actuators.
- **Robotic Agent:** A robotic agent can have cameras, infrared range finder, NLP for sensors and various motors for actuators.
- **Software Agent:** Software agent can have keystrokes, file contents as sensory input and act on those inputs and display output on the screen.

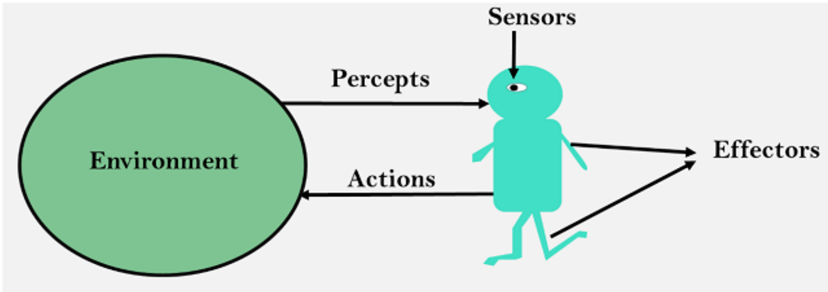
Hence the world around us is full of agents such as thermostat, cellphone, camera, and even we are also agents.

Before moving forward, we should first know about sensors, effectors, and actuator

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.



Intelligent Agents:

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.

Following are the main 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.

Note: Rational agents in AI are very similar to intelligent agents.

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.

Structure of an AI Agent

The task of AI is to design an agent program which implements the agent function. The structure of an intelligent agent is a combination of architecture and agent program. It can be viewed as:

Agent = Architecture + Agent program

Following are the main three terms involved in the structure of an AI agent:

Architecture: Architecture is machinery that an AI agent executes on.

Agent Function: Agent function is used to map a percept to an action.

$f:P^* \rightarrow A$

Agent program: Agent program is an implementation of agent function. An agent program executes on the physical architecture to produce function f.

PEAS Representation

PEAS is a type of model on which an AI agent works upon. When we define an AI agent or rational agent, then we can group its properties under PEAS representation model. It is made up of four words:

- **P:** Performance measure
- **E:** Environment
- **A:** Actuators
- **S:** Sensors

Here performance measure is the objective for the success of an agent's behavior.

PEAS for self-driving cars:

Let's suppose a self-driving car then PEAS representation will be:

Performance: Safety, time, legal drive, comfort

Environment: Roads, other vehicles, road signs, pedestrian

Actuators: Steering, accelerator, brake, signal, horn

Sensors: Camera, GPS, speedometer, odometer, accelerometer, sonar.

Agent Type	Performance Measure	Environment	Actuators	Sensors
Medical diagnosis system	Healthy patient, minimize costs, lawsuits	Patient, hospital, staff	Display questions, tests, diagnoses, treatments, referrals	Keyboard entry of symptoms, findings, patient's answers
Satellite image analysis system	Correct image categorization	Downlink from orbiting satellite	Display categorization of scene	Color pixel arrays
Part-picking robot	Percentage of parts in correct bins	Conveyor belt with parts; bins	Jointed arm and hand	Camera, joint angle sensors
Refinery controller	Maximize purity, yield, safety	Refinery, operators	Valves, pumps, heaters, displays	Temperature, pressure, chemical sensors
Interactive English tutor	Maximize student's score on test	Set of students, testing agency	Display exercises, suggestions, corrections	Keyboard entry

Figure 2.5 Examples of agent types and their PEAS descriptions.

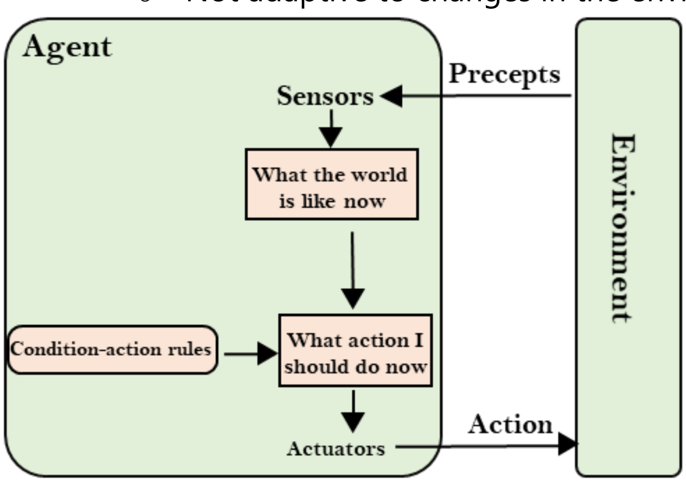
Types of AI Agents

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

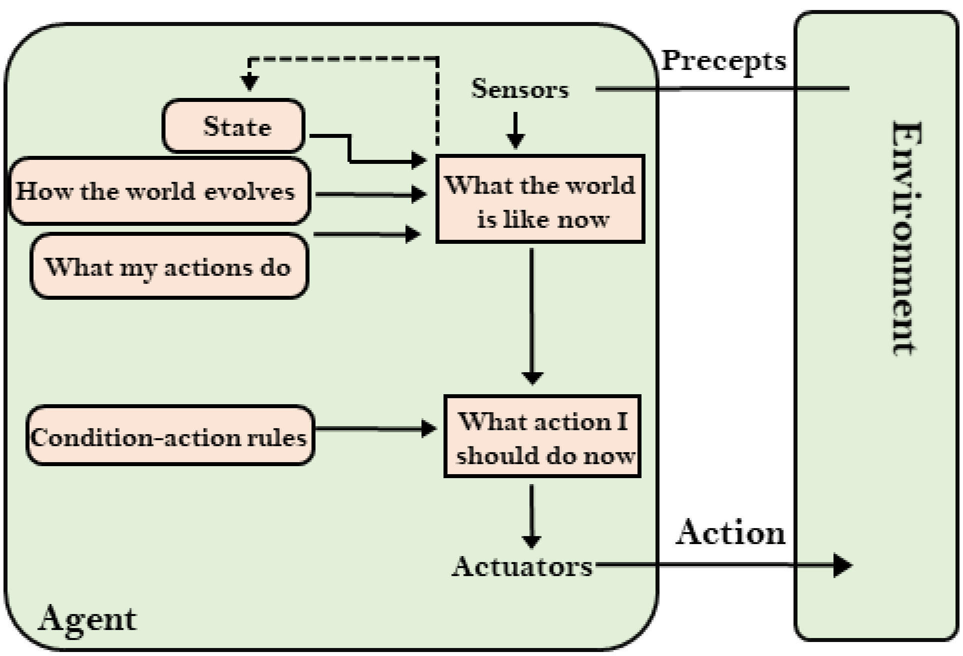
1. Simple Reflex agent:

- The Simple reflex agents are the simplest agents. 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.
- The Simple reflex agent does not consider any part of percepts history during their decision and action process.
- The Simple reflex agent 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.



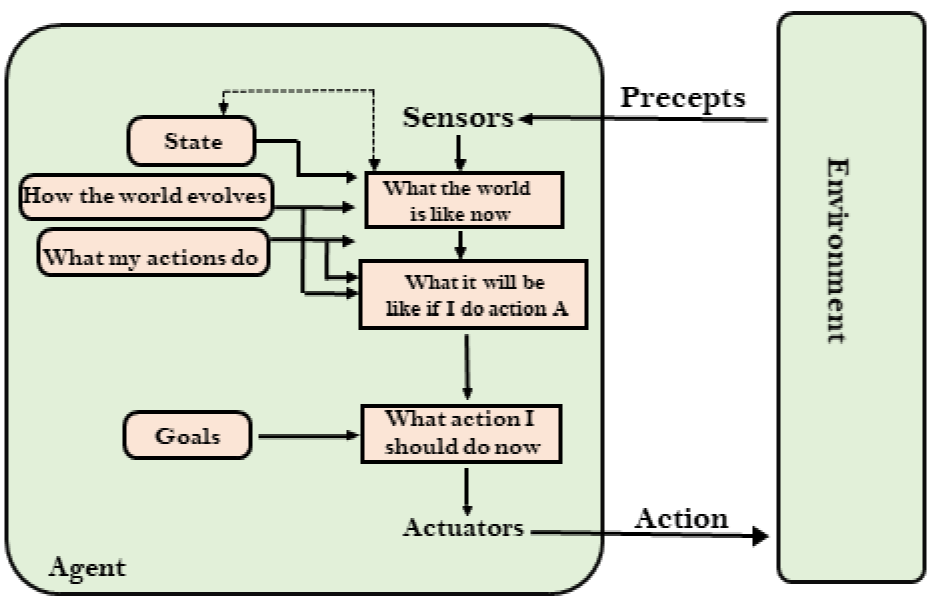
2. Model-based reflex agent

- The Model-based agent can work in a partially observable environment, and track the situation.
- A model-based agent has two important 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:
    a. How the world evolves
    b. How the agent's action affects the world.



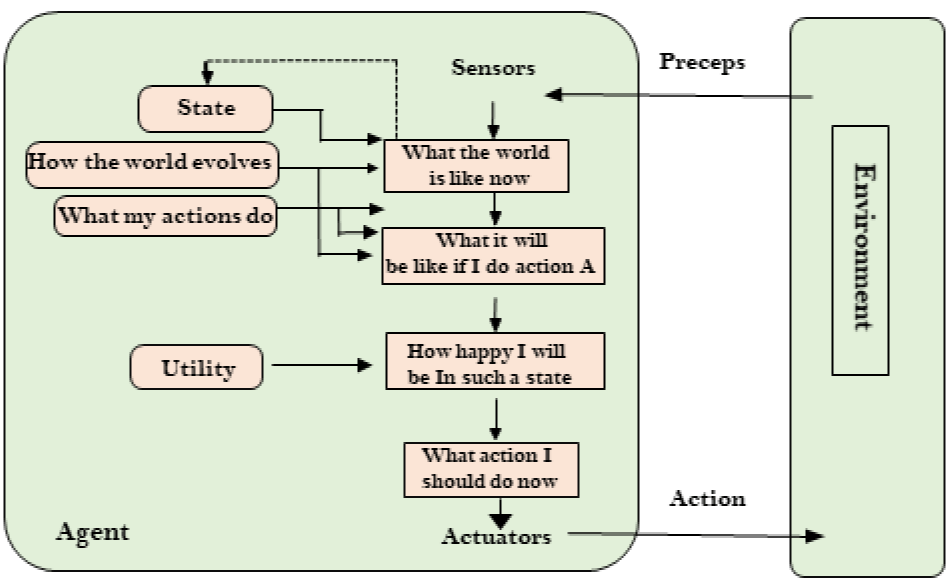
3. Goal-based agents

- 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.
- Goal-based agents 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.



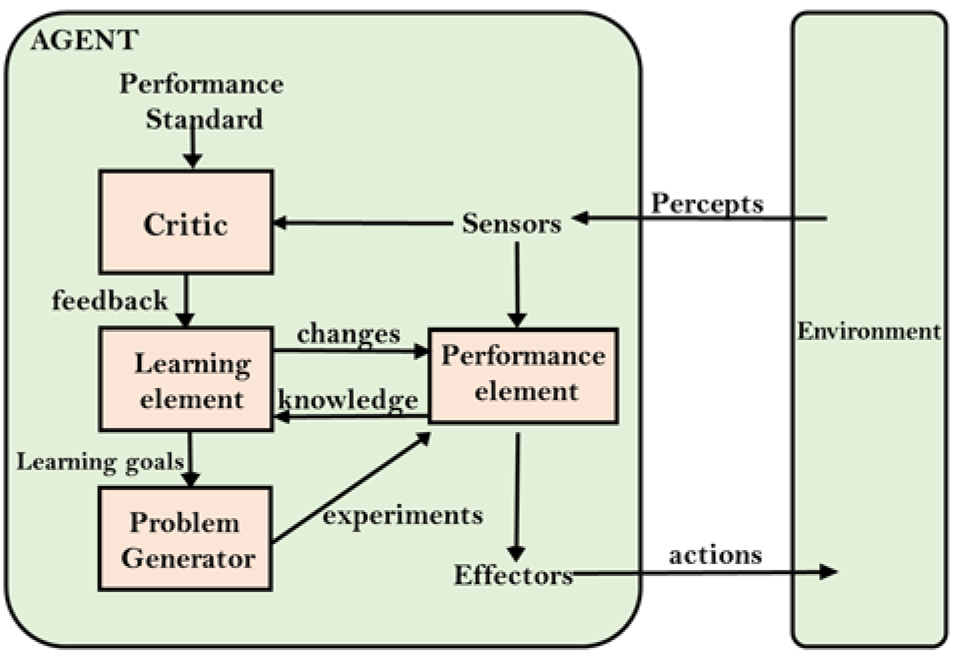
4. Utility-based agents

- These 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.
- Utility-based agent act based not only goals but also the best way to achieve the goal.
- The Utility-based agent is 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.



5. Learning Agents

- A learning agent in AI is the type of agent which 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.
- A learning agent has mainly four conceptual components, which are:
    a. **Learning element:** It is responsible for making improvements by learning from environment
    b. **Critic:** Learning element takes feedback from critic which describes that how well the agent is doing with respect to a fixed performance standard.
    c. **Performance element:** It is responsible for selecting external action
    d. **Problem generator:** This component is responsible for suggesting actions that will lead to new and informative experiences.
- Hence, learning agents are able to learn, analyze performance, and look for new ways to improve the performance.



AI environment

An environment in artificial intelligence is the surrounding of the agent. The agent takes input from the environment through sensors and delivers the output to the environment through actuators. There are several types of environments:

- 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

1. 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.

2. Deterministic vs Stochastic

- When a uniqueness in the agent's current state completely determines the next state of the agent, the environment is said to be deterministic.
- The stochastic environment is random in nature which is not unique and cannot be completely determined by the 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.

3. 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 output desired.

4. 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.

5. Dynamic vs Static

- An environment that keeps constantly changing itself when the agent is up with some action is said to be dynamic.
- 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.

6. 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.

7.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-** Where the previous move can affect all the following moves.

8. Known vs Unknown

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.

- **Problem Solving Agents**

Problem-solving in Artificial Intelligence

- ***a problem-solving refers to a state where we wish to reach (definite goal) from a present state or condition.***
- A **problem-solving agent** is a **goal-driven agent** and focuses on achieving the goal.
- A *Problem solving agent* / ***goal-based*** *agent* decides what to do ***by finding sequence*** *of* ***actions that lead to desirable states***.

Problem solving agents used to find sequence of actions to achieve goals.

Example:

Traveling from one city-1 to city-2

- **In order for an agent to solve a problem it should pass by 2 phases of formation:**
  - **Goal Formation**
  - **Problem formation**

- **Goals** helps to organize agents behavior by limiting the ***objectives***.
  - A goal is an achievable outcome (that is generally broad and longer term .)
  - **objective** (is shorter term) **defines measurable actions** to achieve an overall goal.
- **Goals** decides the actions it needs to achieve it.

GOAL FORMAULATION

It is the first and simplest step in problem-solving.

- It organizes the steps/sequence required to formulate *one goal out of multiple goals* as well as *actions to achieve that goal*.
- Goal formulation is based on the current situation and the agent's performance measure

Ex:-**Problem solving is about having a goal to achieve/Reach. From A to E .**

(from initial state to desirable state).

Problem Formulation

It is <u>the most important step of problem-solving</u> which decides *<u>what actions and states to consider</u>* for a given goal.

- The process of looking for a sequence of actions that reaches a goal is called **search**
- **A search** algorithm **takes a problem as input** and **returns a solution in the form of an action sequence**.
- Once solution is found, the actions it recommends can be carried out this is called **execution phase**.
- **"formulate, search, execute**" design for the agent,

- There are following five components involved in problem formulation:
  - **Initial State:** It is the starting state or initial step of the agent towards its goal.
  - **Actions:** It is the description of the possible actions available to the agent.
  - **Transition Model:** It describes what each action does.
  - **Goal Test:** It determines if the given state is a goal state.
  - **Path cost:** It assigns a numeric cost to each path that follows the goal.

The problem-solving agent selects **a cost function, which reflects its performance measure.**

❖ **an optimal solution has the lowest path cost among all the solutions.**

Simple “formulate, search, execute” design for the agent

```
function SIMPLE-PROBLEM-SOLVING-AGENT(percept) returns an action
  persistent: seq, an action sequence, initially empty
              state, some description of the current world state
              goal, a goal, initially null
              problem, a problem formulation

  state ← UPDATE-STATE(state, percept)
  if seq is empty then
      goal ← FORMULATE-GOAL(state)
      problem ← FORMULATE-PROBLEM(state, goal)
      seq ← SEARCH(problem)
      if seq = failure then return a null action
  action ← FIRST(seq)
  seq ← REST(seq)
  return action
```

Figure 3.1 A simple problem-solving agent. It first formulates a goal and a problem, searches for a sequence of actions that would solve the problem, and then executes the actions one at a time. When this is complete, it formulates another goal and starts over.

Cont…

➢ The agent design assumes the Environment is:

- **Static** : The entire process carried out without paying attention to changes that might be occurring in the environment.

- **Observable :** The initial state is known and the agent's sensor detects all aspects that are relevant to the choice of action.

- **Discrete:** countable actions
- **Deterministic:** The next state of the environment is completely determined by the current state and the actions executed by the agent.

Solution to the problem are single/ sequence of actions

Well-defined problems and solutions…

A well-defined problem can be described by:

i. **Initial state**

ii. **Operator or successor function** - the set of states reachable from x with one action

iii. **State space** - all states reachable from initial by any sequence of actions

iv. **Path** - sequence through state space

v. **Path cost** - function that assigns a cost to a path. Cost of a path is the sum of costs of individual actions along the path

vi. **Goal test** - test to determine if goal state is reached or not.

Cont…

vii. **The step cost** of taking action ‘a’ to go from state **x** to state **y** is denoted by c(x,a,y).

The step cost for are shown in next figure in **route distances**.

It is assumed that the step costs are non negative.

viii. **A Goal/Solution** to the problem is a path from the initial state to a goal state.

ix. **An optimal solution** has the lowest path cost among all solutions.

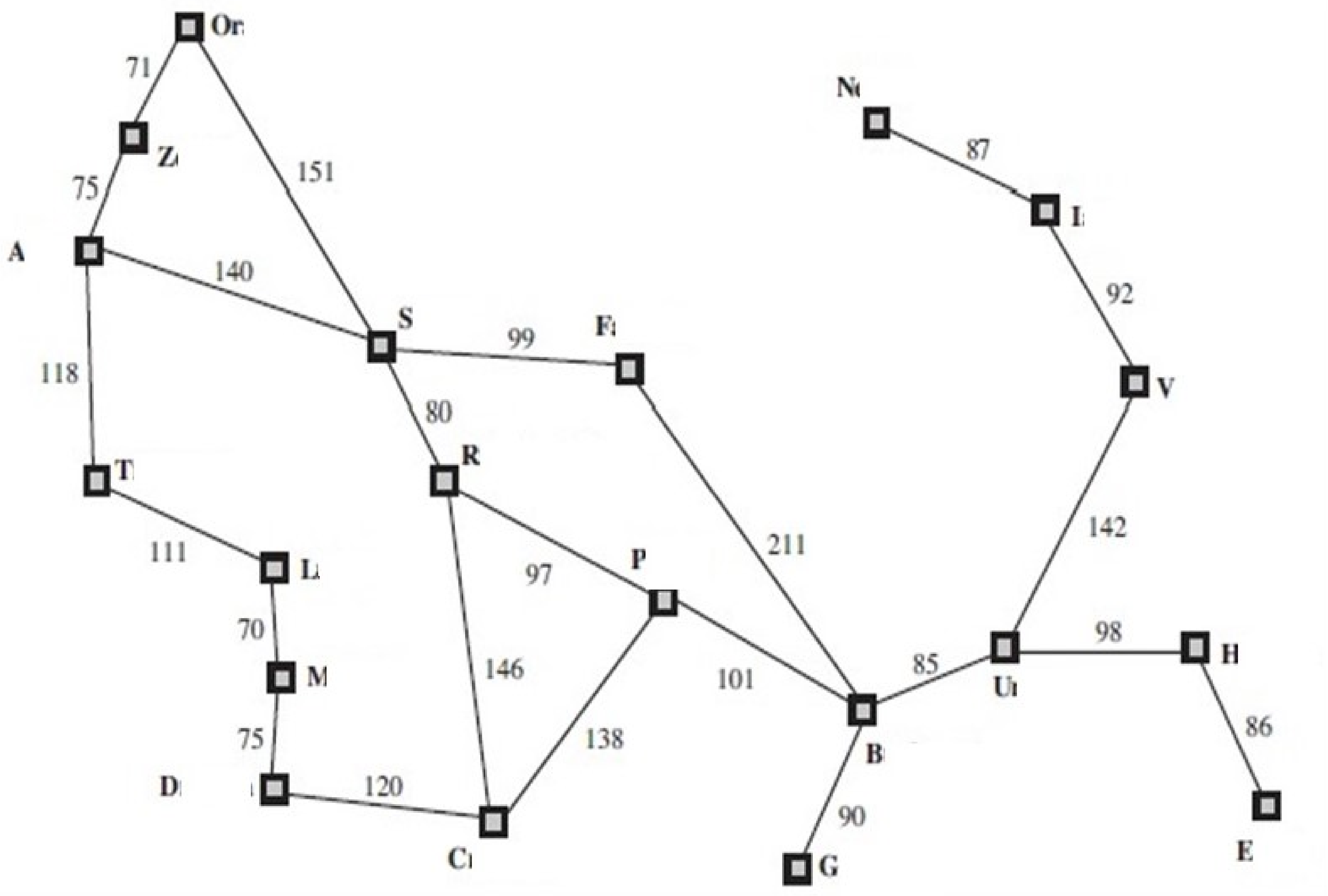


Figure 3.2 A simplified road map

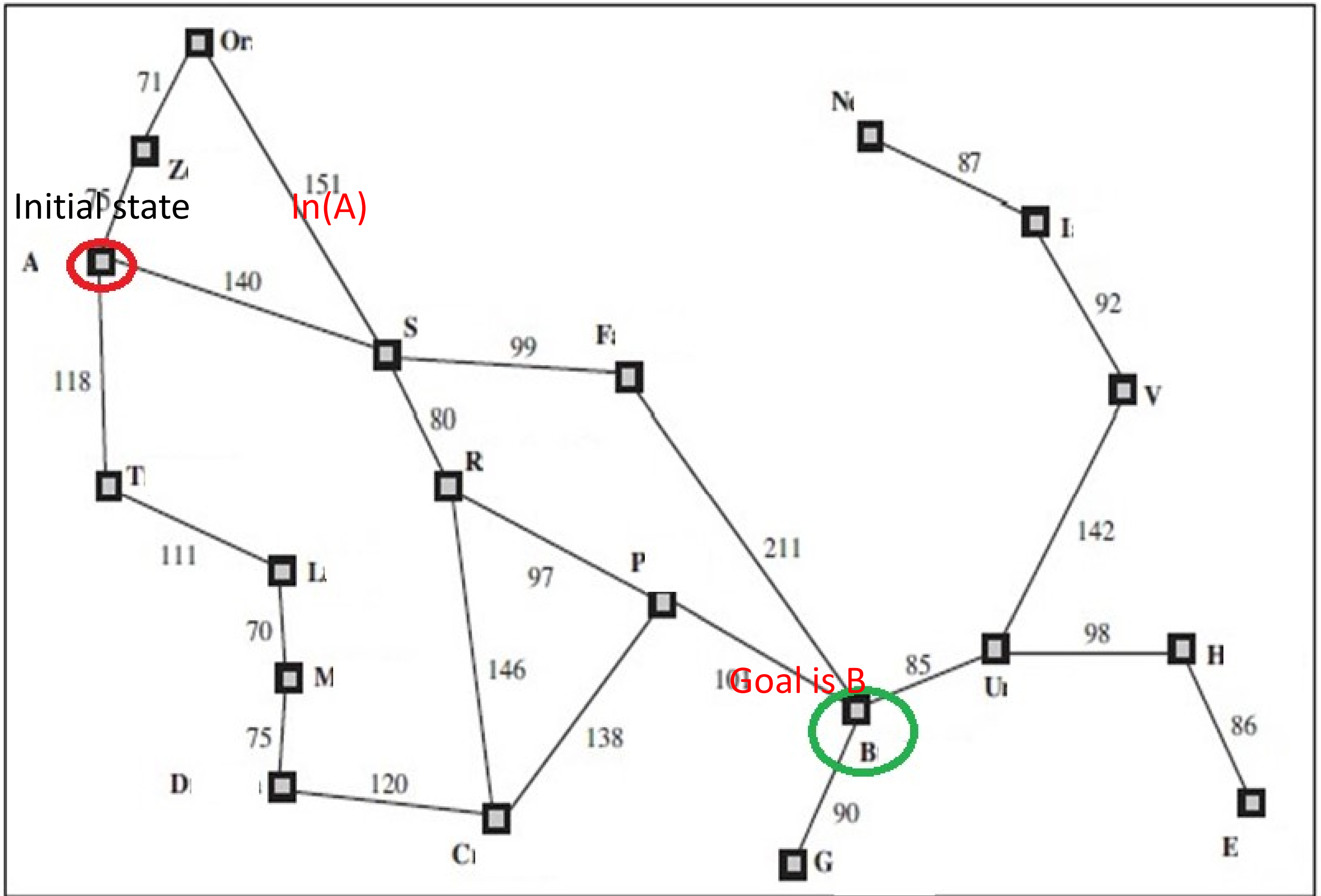
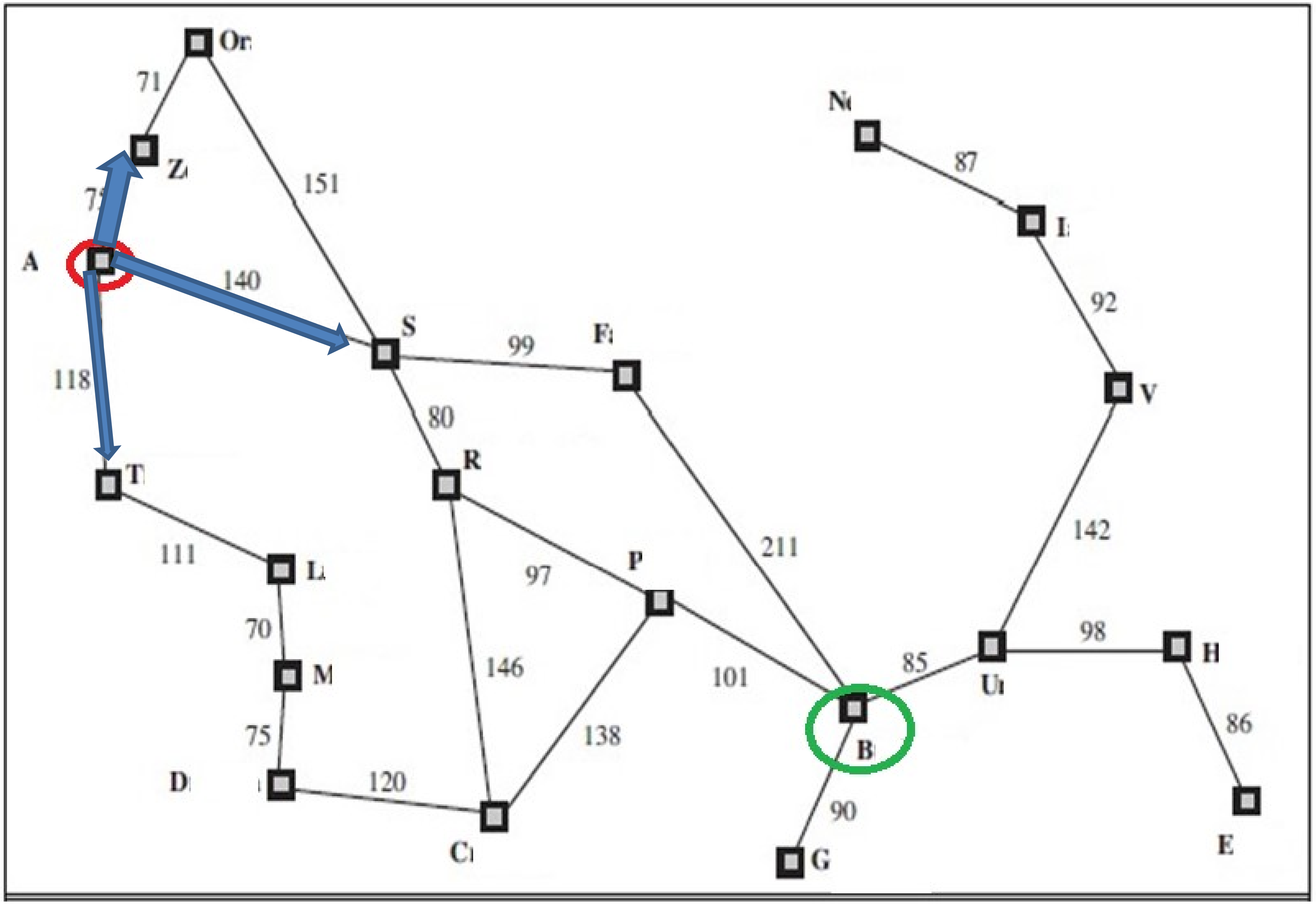
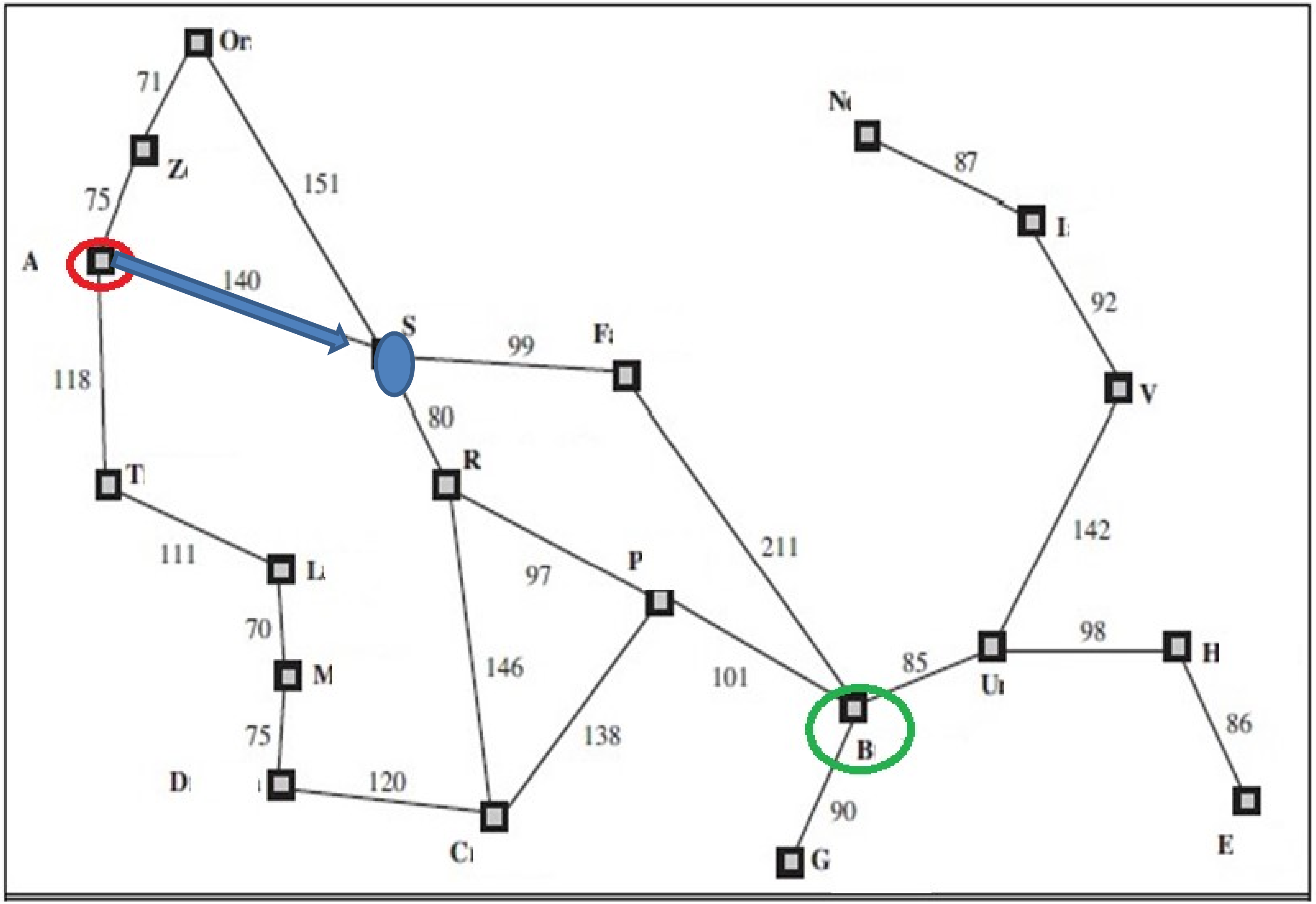


Figure 3.2 A simplified road map



The description of Possible Actions available to the agent

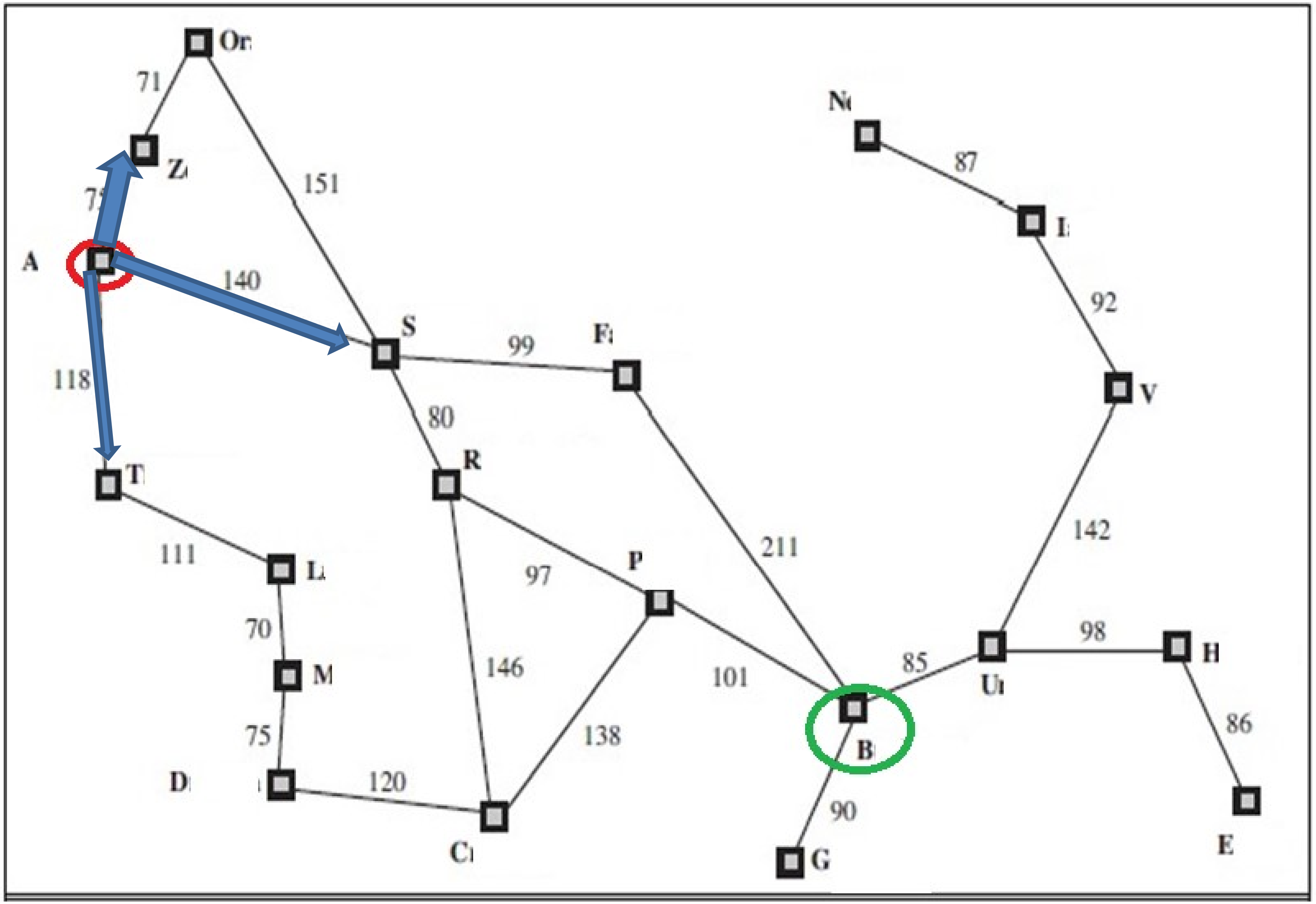
-From state In(A) the applicable actions are { Go (s), Go(Z), Go(T) }



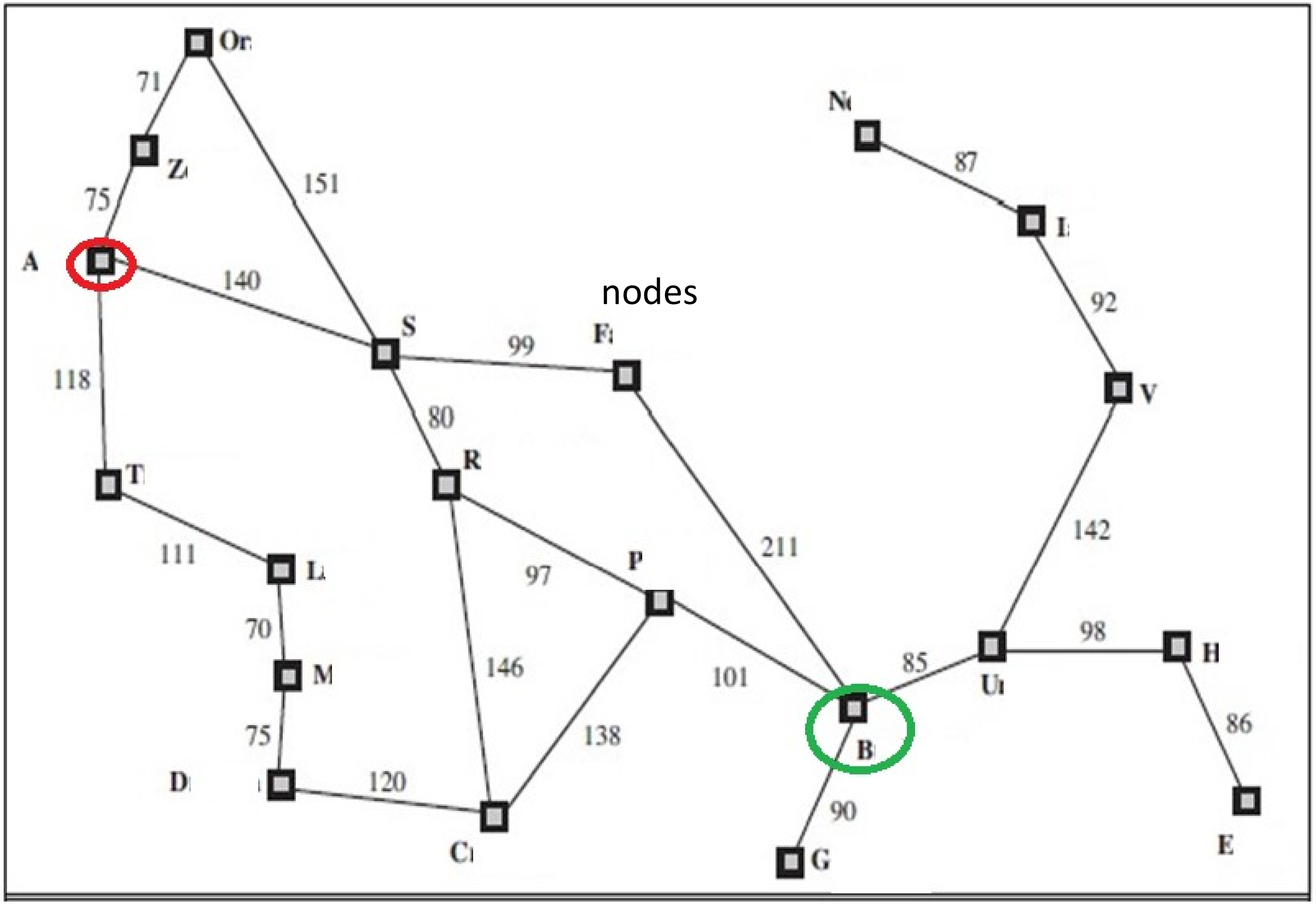
The description of what each action does (Transition Model)

successor =any state reachable from a state by single action

Result (In(A), Go(S))= In(S)



The **initial state**, **actions**, and **transition model** define **the state space of the problem** -—the set of all states reachable from the initial state by any sequence of actions.

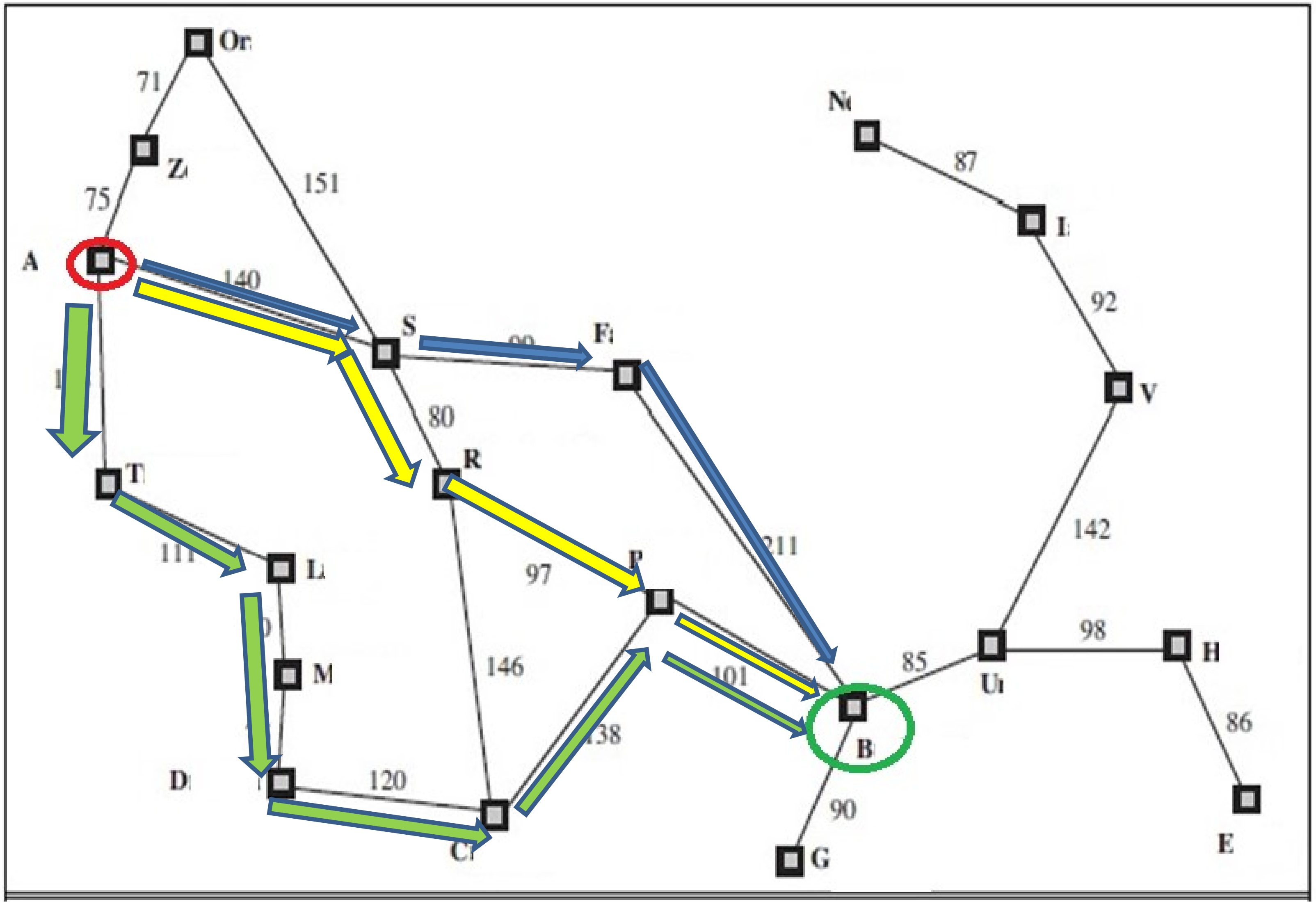


The state space forms a directed network or **graph** in which the **nodes** are states and the **links between nodes are actions.**

Optimal solution

We have defined the problem(including Goal)

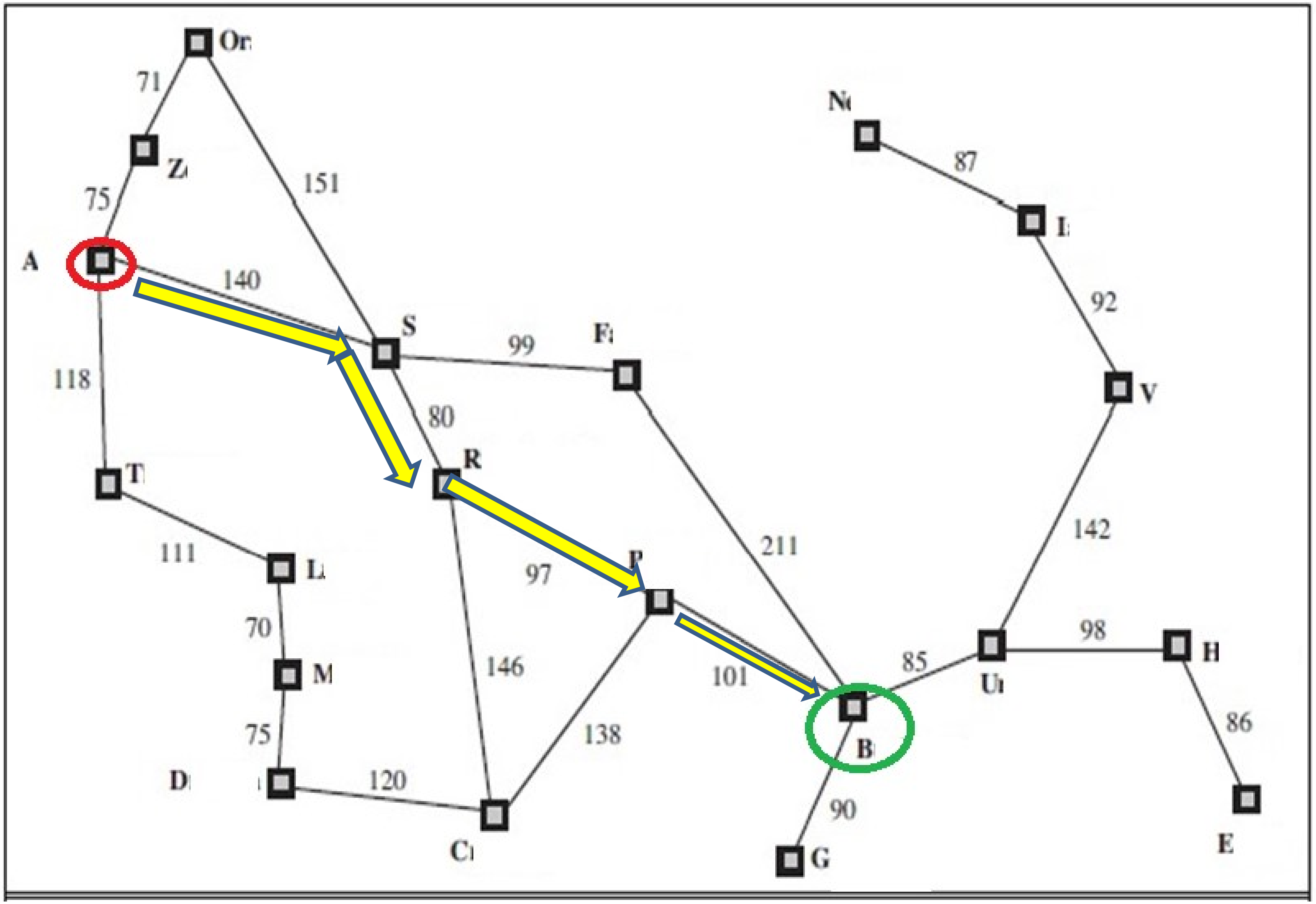
- A **solution** to a problem is:
  - an action sequence that leads from the initial state to goal state.
- Solution quality is measured by the **path cost** function.
- An **optimal solution** has the lowest path cost among all solutions.



A **path in the state space is a sequence** of states connected by a sequence of actions.

Abstraction when formulating problems

- There are so many details in the actual world!
- Actual World State = the travelling companions, the scenery out of the window, the condition of the road, the weather, etc.
- Abstract Mathematical State = In(A)
- We left out all other considerations in the state descriptionbecause they are irrelevant to the problem of finding a route to $\boxed{B}$



Or
71
Z
151
75
A
140
S
99
F
118
80
R
T
97
111
L
P
211
142
70
M
146
101
85
U
98
H
75
138
B
86
D
120
90
C
G
E
N
87
I
92
V

Example Problems…

➢ Toy vs Real World problems…

- ✓ A **toy problem is intended to illustrate or exercise various problem-solving methods.**

- ✓ It can be given a concise, exact description and hence is usable by different researchers to compare the **REAL-WORLD** performance of algorithms.

- ✓ A **real-world problem is one whose solutions people actually** care about.

Toy Problems…

Example 1: Vaccum world:

i. **States**: The agent is in one of two locations, each of which might or might not contain dirt. Thus there are $2 \times 2^2 = 8$ possible world states.

A larger environment with n locations has $nX2^n$ states.

i. **Initial state**: Any state can be designated as initial state.

ii. **Successor function :** This generates the legal states that results from trying the three actions (left, right, suck). The complete state space is shown in figure.

iv. **Goal Test:** This tests whether all the squares are clean.

v. **Path test:** Each step costs one, so that the path cost is the number of steps in the path.

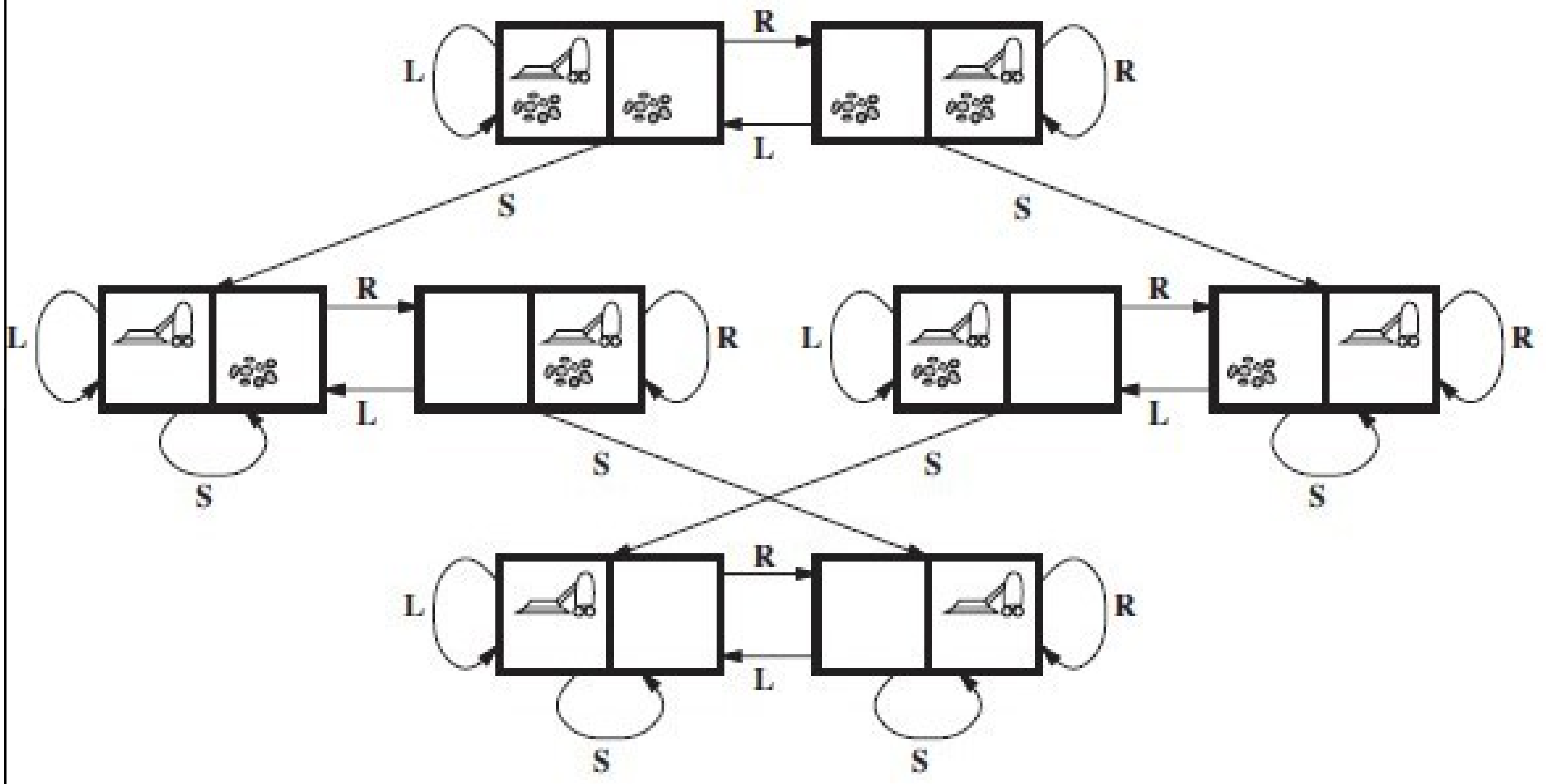


Figure 3.3 The state space for the vacuum world. Links denote actions: L = *Left*, R = *Right*, S = *Suck*.

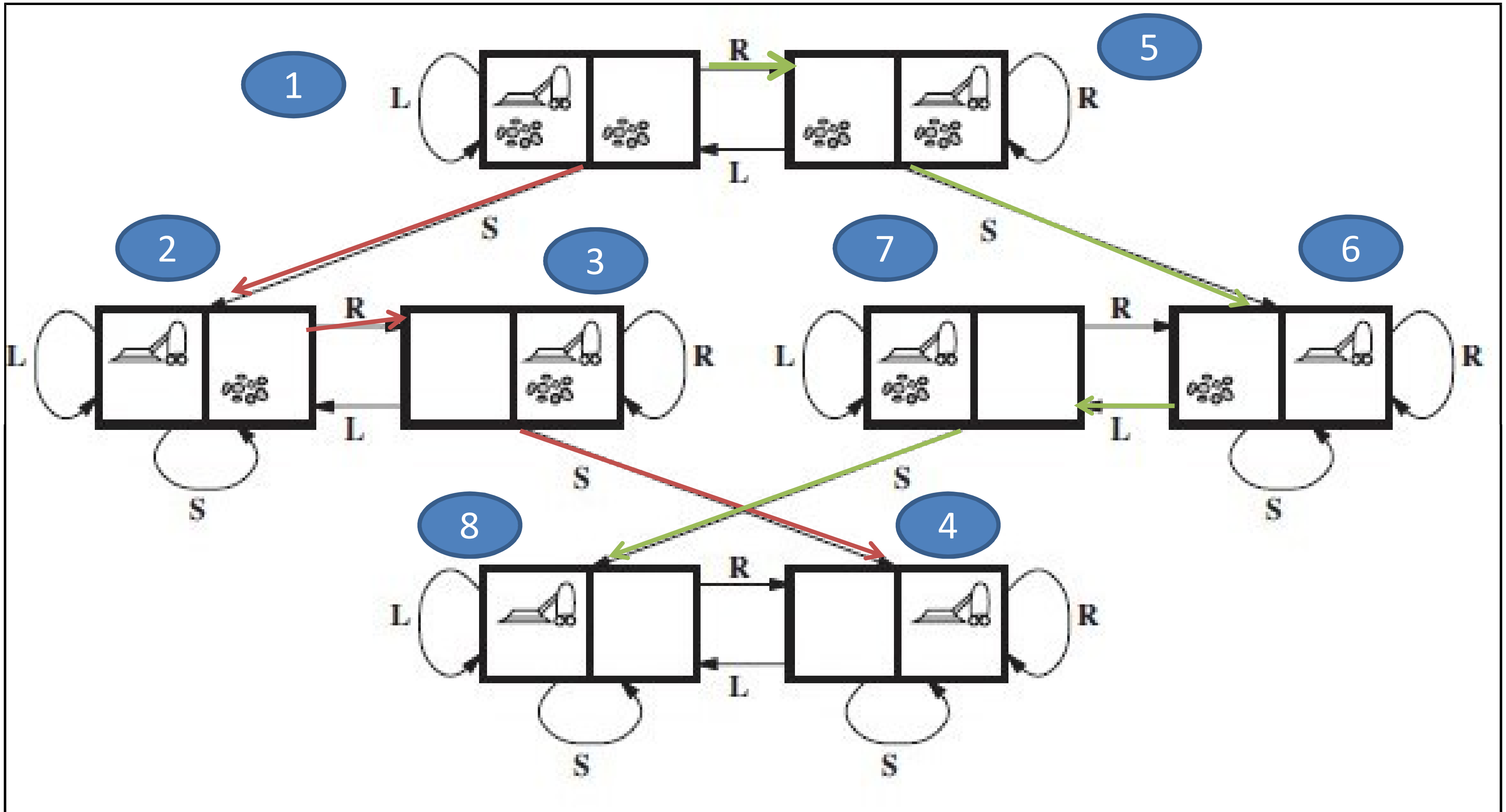


Figure 3.3 The state space for the vacuum world. Links denote actions: L = *Left*, R = *Right*, S = *Suck*.

Cont…

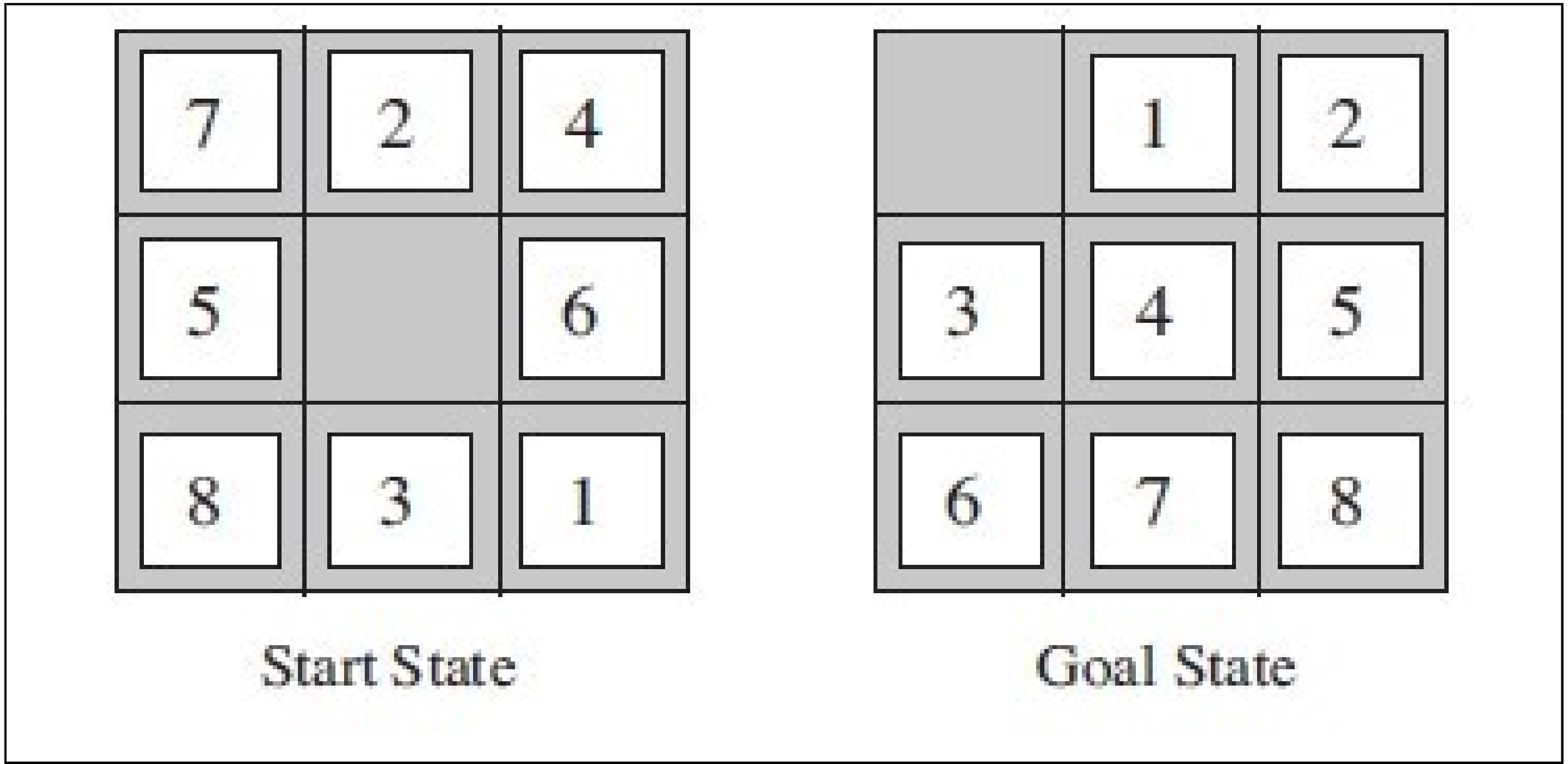
1. **States:** dirt and robot locations (ignore dirt amounts etc.)
2. **Actions:** Left, Right, Suck, NoOp
3. **Goal test:** No dirt
4. **Path cost:** 1 per action (0 for NoOp)

Example 2:The 8-puzzle…

- ✓An 8-puzzle consists of a 3x3 board with eight numbered tiles and a blank space.

- ✓A tile adjacent to the blank space can slide into the space. The object is to reach the goal state ,as shown in figure.

Cont…



A typical instance of the 8-puzzle.

Cont…

- **The problem formulation is as follows :**

i.States : A state description specifies the location of each of the eight tiles and the blank in one of the nine squares.

ii Initial state : Any state can be designated as the initial state. It can be noted that any given goal can be reached from exactly half of the possible initial states.

iii.Successor function : This generates the legal states that result from trying the four actions(blank moves Left, Right, Up or down).

Cont…

iv. **Goal Test:** This checks whether the state matches the goal configuration shown in figure(Other goal configurations are possible)

v. **Path cost :** Each step costs 1,so the path cost is the number of steps in the path.

Cont…

i. **States:** Integer locations of tiles (ignore intermediate positions)

ii. **Actions:** Move blank left, right, up, down (ignore unjamming etc.)

iii. **Goal test:** Goal state (given)

iv. **Path cost:** 1 per move

Example 3: 8 Queen Problem…

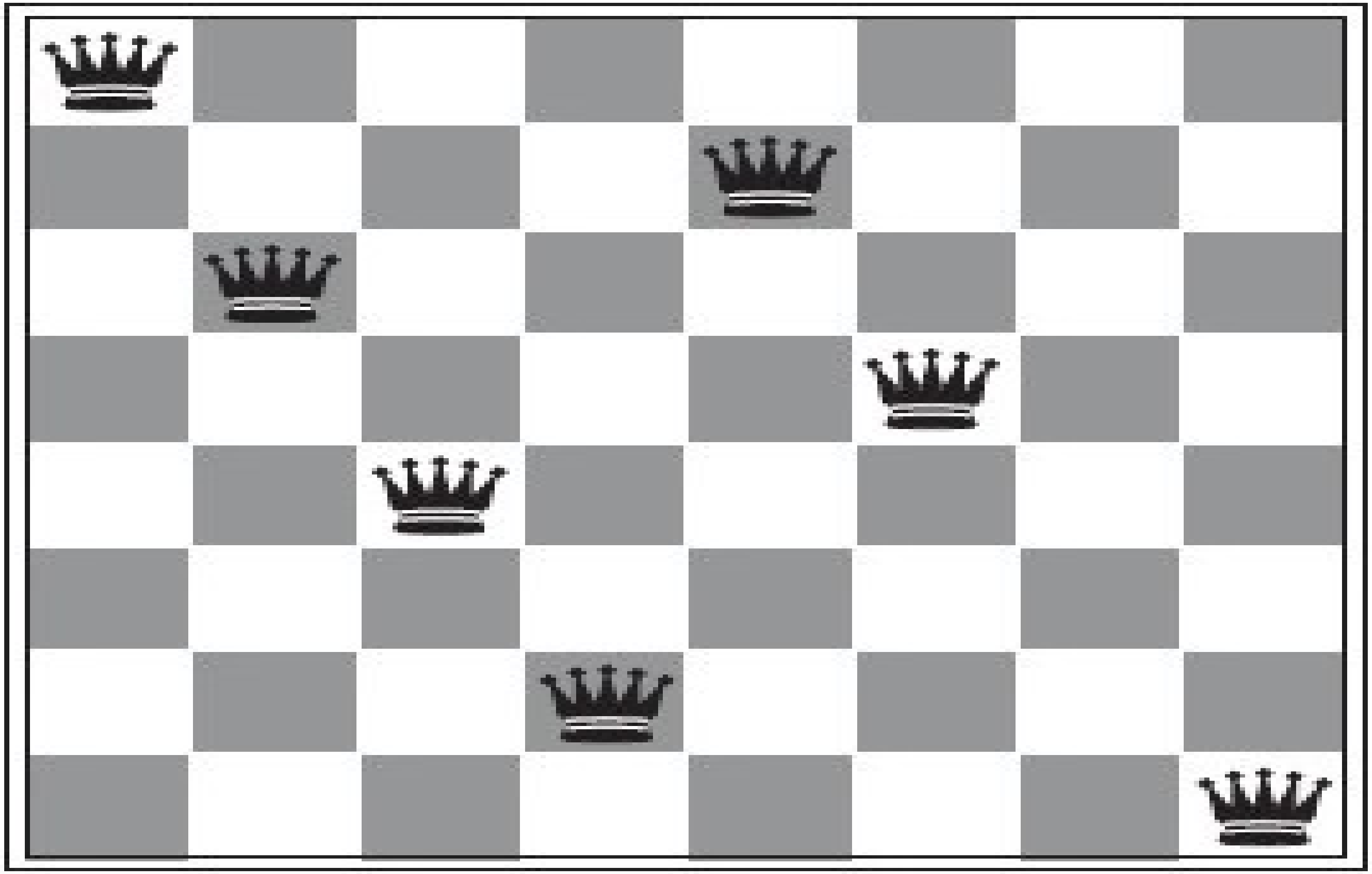
- ✓ The goal of 8-queens problem is to place 8 queens on the chessboard such that no queen attacks any other.(A queen attacks any piece in the same row, column or diagonal).

- ✓ An **Incremental formulation** involves operators that augments the state description, starting with an empty state for 8-queens problem, this means each action adds a queen to the state.

- ✓ A **complete-state formulation** starts with all 8 queens on the board and move them around.

- ✓ In either case the path cost is of no interest because only the final state counts.

Cont…



Cont…

- ✓ The first incremental formulation one might try is the following :

i. **States :** Any arrangement of 0 to 8 queens on board is a state.

ii. **Initial state :** No queen on the board.

iii. **Successor function :** Add a queen to any empty square.

iv. **Goal Test :** 8 queens are on the board, none attacked.

Cont…

- ✓ A better formulation would prohibit placing a queen in any square that is already attacked.

- ➢ **States** : Arrangements of n queens ($0 <= n < = 8$), one per column in the left most columns ,with no queen attacking another are states.

- ➢ **Successor function** : Add a queen to any square in the left most empty column such that it is not attacked by any other queen.

- ➢ This formulation reduces the 8-queen state space from 3×10^{14} to just 2057,and solutions are easy to find.