CHAPTER 1

UNIT I: Introduction	
1.	Introduction to Artificial Intelligence-History of AI- AI Techniques
2.	Problem Solving with AI- AI models-Data Acquisition and Learning Aspects in AI
3.	Problem-Solving Process – Formulating Problems-Problem Types and Characteristics- Problem Analysis and Representation
4.	Performance Measuring-Problem Space and Search-Toy Problems- Real-world problems-Problem Reduction Methods

Introduction to Artificial Intelligence

INTRODUCTION

Suppose you are on the way to some trip and travelling by road. With a large number of toll collection booths and endless multiple queues in front of you, you possibly ask your friend—"Hey which lane do we line up?" and judgmentally a friend suggests one of them. (But possibly other queue would have made your car pass by faster. Is it not dependent on the people working at the booths?) Can we have artificial intelligence (AI) to resolve this?

Imagine you pass through the booth and the toll is auto-deducted by means of in-vehicle unit specially designed for your car. Sensors capture the vehicle number and deduct the pre-deposited amount from the in-built chip. Such electronic toll collection system is found at Ahmedabad–Mumbai highway and a couple of other places in India. This system needs to be made more intelligent to make it work in Indian context across the various roads and highways in India. Certain changes can make it highly effective in terms of time and manpower utilisation in Indian context. The improved system can help the government by means of analysis of the records of the amount collection on week days and other statistics. At present, there are semi-automatic toll collection systems at Bangkok. Just imagine a toll system that is intelligent and vehicle-based. Such a system can direct the vehicle to proper lane and will be able to keep a track record of regularity of payments. In case, some very regular payee misses, the detection on one of the occasion can be possible. This can be possible with a prepaid card-based toll collection, where a prepaid card is used or even there can be a vehicle number-based toll collection, where it is connected to the bank account of vehicle owner. It can be different for interior roads and expressways. Hence, intelligent system can revolutionise the simple application of toll collection and different intelligent system applications right from planning to decision-making, and thus, can help in building a better system.

A simple sub-application is discussed here. At present, it is possible to suggest you to line up at a particular lane. The moment you approach a toll booth, the AI system would analyse the vehicles at the different lanes (including analysis of density and the size or even detection of some failure at one of the lane and so on) and intimate you about the lane number you should go in and the time that you would take.

Let us take one more example. It is often seen that the loading of material in trucks for transport is done manually. What would happen when a robot is employed? A simple robot can assist in saving the labour cost and help in improving the efficiency of the work. Such a robot is tuned to lift an object and place it in the truck. But is there any intelligence involved in it? Does this robot consider space requirements? No, but an intelligent one will definitely do it. An intelligent robot would take up the object and place it efficiently such that the space of the truck is utilised to the fullest. Thus, an intelligent robot would overcome space and time complexities. To add further, a robot without intelligence would fail to operate if the road/path is full of obstacles, but an intelligent one would perceive this and take a different route and carry out the task. Moreover, such an intelligent system can consider weight, size, material of which the articles in the box are composed (whether articles in box are brittle and to be kept in a particular way). Also, it can scan size of box, determine other properties of the box and load different boxes optimally.

And the last but not the least, all of us have heard the story of 'Snow White'. Her stepmother asked to the magical mirror "Who is the most beautiful lady among all?" and she got a reply. We know that it is a fairy tale. But what would happen if we have something of this sort in real? An intelligent app that would suggest some outfit that we can wear! Adding to this, a talking mirror-cum-dresser that would show how an outfit would look on us! Is this all feasible?—Definitely. With AI, we can have all these options and it is not just about dressing, but it can involve suggesting a suitable costume based on your mood, occasion, profession and so on. There are a few garment and fashion designing companies that give some sort of these facilities to recommend the best dress for you. These systems are at very rudimentary stage at present. They analyse the person's parameters, as said earlier and suggest the best garments, colours and combinations for you. Thus, intelligent systems have potential to change the way we have been using our resources resulting in more efficient, smart and friendly systems. These systems can adapt to the requirement and needs of daily operations as well as some of the very specific tasks.

So, artificial intelligence can help in making our life simple not only for taking decisions but also for the other aspects. Thus, artificial intelligence is slowly becoming a part of our day-to-day life and an essential part of all modern equipment as well. It is required for automated climate control in a car or an automated manufacturing unit in industry—Whether it is a washing machine or microwave oven—artificial intelligence is everywhere. These AI-based tools and techniques have not only improved accuracy but also made it possible to perform tasks, which were otherwise impossible without AI.

If we think about AI in holistic way, then it includes learning, searching and problem solving. The purpose of AI is to empower machine to solve the problems by making the machine intelligent. Whenever the term *intelligence* is referred, it is always referred in relevance with the human intelligence. In some cases, it is referred with reference to the reasonability in decision-making or the action taken. *AI* is defined by Rich and Knight as follows:

"AI is the study of how to make computers do things which, at the moment, people do better."

This definition catches the crux of AI and defines its purpose. This definition has its own limitations and covers one of the basic aspects of AI. It is not an easy task to define AI and all the books have done decent efforts to define AI based on some of its important facets. AI can be defined as follows:

It deals with the science that is about the efforts of making a machine behave intelligently and respond in a way as human would have responded and in due process, deliver reasonable answers. In other words, a branch of science and engineering that focuses on making machine intelligent is widely known as *AI*.

The scientists such as George Boole have led the foundation of AI logic years ago. Then, for last five decades many scientists were part of the movement of making machine intelligent. A broad categorisation of AI functions and objectives includes reasonable decision-making, demonstration of intelligence like humans and computational modelling to solve decision problems. In context with the above broad categorisation, AI can be defined in different ways as follows:

1. Machines which can think and have a capability to react like human beings
2. Systems that respond intelligently in the same way as the humans do
3. Computational models to solve various complex decision-making problems
4. Study of intelligent agents

The approaches followed while studying and representing AI include various empirical approaches—one coming from the philosophy of comparison with human and human reactions in case of different situations, while another is about use of mathematics and models for optimisation with reference to the desired outcome.

The horizon of AI includes techniques for knowledge transmission, knowledge representation, automated reasoning, and this is used to empower machines to behave intelligently. The purpose of overall data analysis and knowledge augmentation is to make machines learn and solve complex real-life, problems. AI is actually an ensemble of technologies, interactions and allied platforms which takes part in helping machine to demonstrate intelligence and reasonability. The human way of thinking, responding and decision-making is the only expectations and reference in this process.

Cognitive models deal with the computer knowledge-based models for AI. Various experimental techniques and theories about working and representation of human mind are part of the cognitive models. The study of human psychology and psychological analysis with reference to decision process is studied in these cognitive models. The fields of cognitive science and AI go hand in hand and have their application in natural language processing.

Further, for any action, there is an expected response from the aspect of rationality. Intelligent machines need to demonstrate the rational behaviour. We expect a human to react in a particular way to a given situation or respond to a particular decision scenario. Similarly, there is also an expectation from an intelligent machine to react rationally.

1.1 ARTIFICIAL INTELLIGENCE—HISTORY AND FOUNDATION

AI has been a part of mainstream research since last 60 years. But AI philosophy is as old as one thousand years. Statistics, analysis of patterns, use of formal systems have been parts of research for many years. Many philosophers including the great Aristotle tried to describe and represent human process of thinking and decision-making using symbols. In 20th century, the ideas in fiction started to realise in the form of computer. Indian and Greek philosophers developed various methods for formal reasoning. This was a structured approach towards problem solving. In the 1940s, Zuse devised high-level programming language and wrote the first chess program that could demonstrate chess playing. Before that chess playing was considered an intelligent activity that was not possible for machine. Later on, Leibniz envisioned and formulated a language of reasoning, where he mapped symbols for reasoning. When mathematical logic came to help in twentieth century, it assured that AI is very much possible.

Allan Turing devised a simple test of intelligence in 1950, where the response of machine is expected to be intelligent enough so that it is difficult to find out whether it is machine or human sitting on the other side. Since then, there is a quest for intelligent algorithm to build AI-based system to meet the expectations.

In 1956, John McCarthy insisted and made AI as a topic for conference at Dartmouth. In 1958, he (MIT) invented the Lisp language, which later became popular for AI-related programming.

Initially, AI was focussed on common sense reasoning and obvious reaction. In this common sense reasoning, AI was expected to perform some sort of general problem solving. The problem solving and decision-making was based on set of simple hypothesis. The problems were simple and did not require large knowledge base.

Slowly research began to handle large amount of knowledge and more complex relationships. It included domains like speech recognition and analysis, image processing and medical diagnosis. The complexity of the tasks and inferring mechanism kept on increasing. In case of noisy data and large information base for extraction of information, there were more challenges for the researchers. They slowly started developing intelligent systems to handle these complex research problems.

1.2 BIRTH OF ARTIFICIAL INTELLIGENCE

John McCarthy at Dartmouth College worked on research in the areas of automata theory, neural nets and study of intelligence. McCarthy gave a new term for AI, i.e., 'computational rationality'. Dartmouth is a place where formal workshop and conference on AI took place. This helped in exploring the field of AI, and the research in this field started to gain momentum. This further helped in realising the AI-related thoughts and ideas. The various concepts like intelligence, knowledge, reasoning, thoughts, cognition and learning started to formulate a platform for AI research. These concepts were further explored from the perspective of applying knowledge to perform a desired activity. These activities range from the normal activities to complex activities. This started a movement of building intelligent systems.

AI is one of the major components of intelligent systems. Let us take an example of simple mechanical intelligent system.

Figure 1.1 depicts a small example of a traditional mechanical intelligent system. Here, once the water reaches to a certain level, the water tap is closed to avoid overflow. In washing machines, these systems were replaced with the sensor-based level detector. Later, fuzzy logic came into the picture that allowed deciding the level of water dynamically based on the quantity of clothes. So, we are now having machines, with fuzzy logic included.

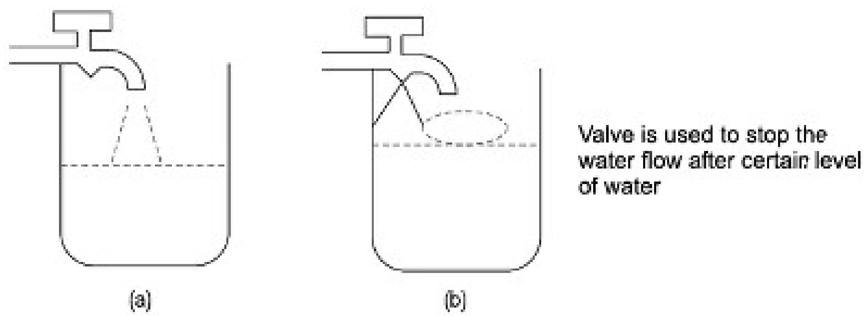


Figure 1.1 Mechanical intelligent system.

Traditional intelligent systems are the examples, wherein the knowledge acquired in past is programmed in the system. These systems were later replaced by electronic systems, which used some static historical knowledge to take decisions in future.

Let us consider an example related to traffic management to understand how the growth and evolution of AI can help in making things simple and manageable. We are aware of the growing traffic scenario and how much difficult it is to commute.

1. First, we begin with the history of the signals to control the traffic traditionally. They were based on some historic information. The switching time was static and independent of the road conditions, which did not account for the peak hours. This would have been sufficient if there had not been any change in the traffic.
2. The increase in the traffic resulted in total chaos. Gradually, there was a need for a tuner system that could change the signal timings with respect to the peak hours of the traffic. Was AI involved here? It could be just a simple decision-making system with a little intelligence to have the control of the signal or even manual setting could be done with no intelligence at all.
3. Gradually, the need for an intelligent system arised. A traffic jam at one road crossing could impact many roads. So, an observer with sensors like a camera to detect the current traffic conditions evolved an intelligent traffic monitor and controller that would tune the signal timing with this jam. This could be set up at every signal in operation. AI techniques, thus, can help us in solving problems in a better way.
4. At present, there is a need for an intelligent signal controller, wherein all the signals are synchronised and they can be controlled with a centralised controller. An AI-based system (that can consider events such as election rallies, major accidents, construction of road) can help in capturing the current scene at a particular route, and accordingly, dynamically change the signals timings. This information can even include the analysis and statistics of the traffic density to help in deciding the timings. The same should be reflected and transferred to the signals in proximity. It would, in turn, suggest the route too intelligently that would turn off some signals owing to some environment conditions. Such a system would need complex and advanced techniques of AI and is indeed a challenge to have one in country like India.

Thus, there is a need for AI-based systems, which are expected to react in dynamic situation and even when some information is absent.

To take up decisions, one needs to learn. The learning systems are data as well as experiment-oriented. The experience is encoded in the system in the form of data or rules. In these systems, pre-defined mapping exists that leads to decision rules. So, in order to build an intelligent system, good representation of knowledge and information is essential. AI struggled to meet the high expectation of the society and researchers because of the limitations in capturing the hidden knowledge and representation of that knowledge in decision-friendly format. But AI has so far been successful in offering many practical solutions in spite of its limitations. The search techniques, various learning methodologies and problem-solving methods embedded in AI help in resolving important practical problems. In this book, we consider all aspects and methodologies which come under AI. The purpose of this book is to realise and understand the practical aspects of AI. This book also covers the advanced and allied aspects of AI like pattern analytics, intelligent system modelling, concurrency aspects and applications of intelligent systems with reference to big data.

1.3 AI TECHNIQUES

AI deals with a large spectrum of problems. The spectrum of AI applications is spread across the domains, and even across the complexities of problems. This includes the following:

1. Various day-to-day practical problems
2. Different identification and authentication problems with their applications in security
3. Various classification problems resulting in decision-making
4. Interdependent and cross-domain problems

The generalisation may become very difficult in case of these problems, as there is a very little commonality among these different problems. But most of these problems are complex and hard to resolve. The very reason of the complexity is the dynamic nature of these problems unlike some routine mathematical problems. AI techniques need to look at these problems from analysis perspective and from the perspective of research initiatives to resolve them.

AI techniques need to be built from the problem-solving perspectives. The points raising need for AI technique are discussed below:

1. Need for analysis of voluminous and large amount of data. This data may not be confined to a single domain but may spread across the domains.
2. The analysis should be followed by the characterisation of miscellaneous data, then mapping of this data with reference to built-in knowledge, and then, building the knowledge further in this process.
3. Dealing with the constantly changing scenarios and situations, and the dynamic nature of data, the system and technique should react to the new scenario and situation. The situations are dynamic in nature, and static handling may not be useful.

4. The way in which data appears, the way it is used, the way it is organised and the way it should be used are different. Blindly using the data as it comes may result in wrong decisions.
5. Though in some cases, the huge data is available, but the relevant data is limited. Identification of relevant data, irrelevant data and outliers, and further, effective knowledge building based on limited relevant data are the challenges in front of AI techniques.

The main objective of AI techniques is to capture knowledge based on the data and information. There are different scenarios and the relevant data is captured. The AI techniques need to handle different problems. The broad categorisation of these problems can be as follows:

1. Structured problems
2. Unstructured problems
3. Linear problems
4. Non-linear problems

1.4 PROBLEM SOLVING WITH AI

AI has been very well used to solve structured problems. The *well-structured problems* are some of the very commonly faced problems during day-to-day life. These problems yield a right answer or right inference when an appropriate algorithm is applied. While *ill-structured problems* are the problems which do not yield a particular answer. In this case, there is possibility of more than one answer, and even a particular situation decides the correctness of the answer. Interestingly, ill-structured problems represent many of the real-world problems.

Some of the well-structured problems are given below:

1. Solving a quadratic equation to find out the value of X
2. Calculating path of the trajectory when a missile is fired
3. Calculating speed of ball when it reaches to batsman
4. Network flow analysis problems

Some examples of the ill-structured problems are given below:

1. Predicting how to dispose wet waste safely
2. Analysis of theoretical prepositions and adequacy of the same in a particular scenario
3. Identifying the security threats in big social gatherings

Solving ill-structured problems is challenging, since no list of specific and ordered operations or steps exists for them. Further, there is no well-defined criterion to evaluate the correctness of the outcome.

The behaviour of a typical well-structured problem is depicted in Figure 1.2, while the same for a ill-structured problem is depicted in Figure 1.3.

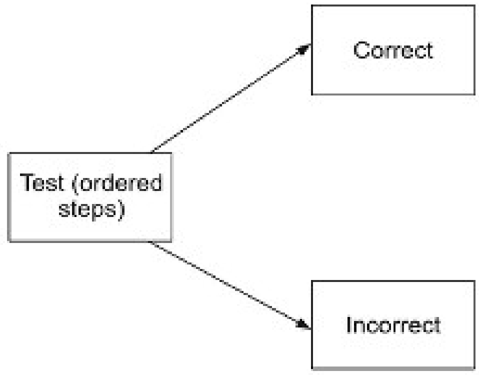


Figure 1.2 A typical well-structured problem analysis.

Figure 1.3 A typical ill-structured problem analysis.

Generally, abstracting the problem is possible in case of structured problems. The similarities and well-defined steps even allow some sort of generalisation in case of well-structured problems. The well-defined steps and well-defined way to measure accuracy allow to head systematically towards the goal state. In case of ill-structured problems, the uniqueness of problems and solution demands high level of problem-specific intelligence and makes it difficult to generalise.

EXAMPLES: A typical well-structured problem is the tic-tac-toe, shown in Figure 1.4.

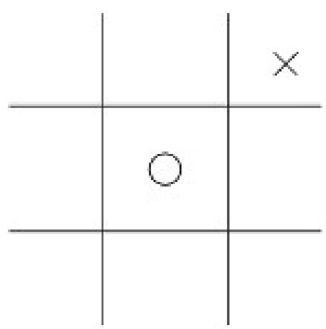


Figure 1.4 Tic-tac-toe.

Here, the final decision will depend on the value associated with all nine positions of tic-tac-toe. The legal values will be player 1 marked, player 2 marked and not marked. In this well-structured problem, solution will try to optimise chances of player 1 while minimising player 2 chances.

On the other hand unstructured problems are difficult to represent and model. There are possibilities of more than one goal states in case of unstructured problems. In most of the cases, exact goal state is not known. For example, systems to improve life expectancy of human being, expanding the business.

Linear problems are the ones which definitely have a solution or there will not be any solution. Speaking with respect to AI problem solving, these problems are the ones that typically fall under the classification category. Whereas, the problems that are not linear have to undergo some transformation for getting solution.

In case of non-linear problems, the relationship between input and output is not linear. Further decisions cannot be separated by simple linear classification function.

1.5 AI MODELS

One important aspect of building AI solutions is modelling the problem. Dunker introduced 'maze hypothesis' as a part of the psychological theory. In this particular hypothesis, the creative and intelligent tasks handled by human beings are modelled like a set of maze of paths from an initial node to a certain or resultant node. Human at any point of time analyses maze; for choices, he could find those which can lead to goal. These choices and maze-based approach can help in solving many multialternative solution problems.

Slowly, it became evident that all problems cannot be solved using maze models or the approach described above. This brought more focus on logic theory machines. Effective application of logic theory machines is found very useful in general problem solving, even this is found very useful for a wide spectrum of problems like chess problem. Chess can be viewed as a controlled environment in which computer is given a situation and a goal.

Figure 1.5 depicts the complexity of model building with reference to data and knowledge mapping.

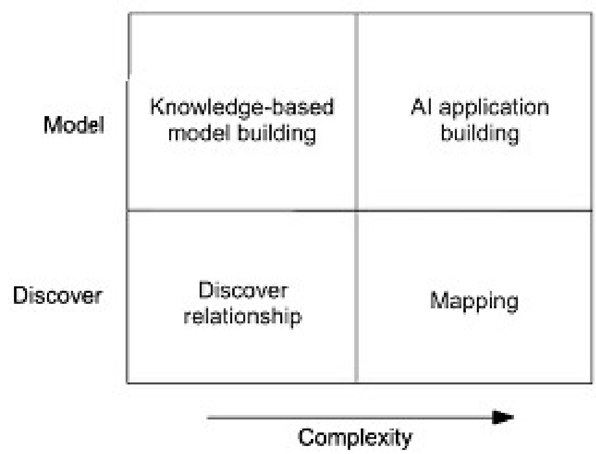


Figure 1.5 Model building and complexity.

A typical chess scenario is given in Figure 1.6. This is a much complex scenario than tic-tac-toe, but is still constrained. The chess program provided a sort of background for AI research. Two aspects that could be viewed from chess program were knowledge-based search and knowledge acquisition and representation. Models used for applications like chess programs were not effective for the other applications.

The advent of natural language processing and the need for man-machine dialogue made it more evident that the models used so far had their own limitations. Then, the formal models were proposed to solve AI problems. The requirement of complex problem solving gave birth to dynamic inductive models. Human behaviour and psychological study-based inductive dynamic models for creative problem solving slowly became popular.

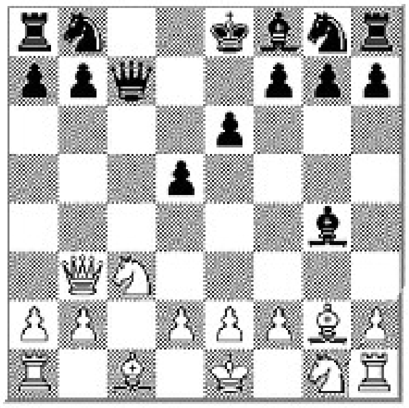


Figure 1.6 Chess: A complex scenario.

Let us have a look at the models.

Semiotic Models

These models are based on sign processes or signification and communication. The process of carrying meaning depends on codes. Semioticians classify signs or sign systems in relation to the problem. This meaning assignment and mapping process depends on the use of codes based on individual sounds or letters that humans use to form words or movements. In computers, these signs are determined for a logical sequence.

Statistical Models

Statistical models refer to representation and formalisation of relationships through statistical techniques. Most of the AI problems can be represented as statistical or pattern matching problems. Various learning models from AI perspective are based on statistics. The historical data is used here in decision-making. Statistical model employs probabilistic approaches and is typically a collection of probability density functions and distribution functions.

1.6 DATA ACQUISITION AND LEARNING ASPECTS IN AI

This section will introduce various AI-related topics on data acquisition and machine learning.

1. ***Knowledge discovery—Data mining and machine learning***: We start with some simple terms, i.e., information and data. Information can be referred to as pattern underlying the data, whereas the data refers to recorded facts. So, we define *data mining* or *knowledge discovery*

as the extraction of meaningful information that is previously unknown and can be useful potentially ahead. It is more concerned with data analysis and use of some techniques to identify and recognise the patterns that would yield good predictions. The mining process includes data cleaning, preprocessing, identifying and interpreting the patterns, understanding the application and generating the target data with the consolidated patterns.

Machine learning, as described by Tom Mitchell, is a field concerned with the study of algorithms that will improve its performance with experience. It is all about making machine behave intelligently based on the past experience.

Let us have a look at the relation between knowledge discovery and machine learning. *Knowledge discovery* is about finding understandable knowledge, while *machine learning* is more focussed on improving performance of an agent. Machine learning can be thought of as a broader concept, which has mining playing an implicit part. There is actually a very fuzzy distinction between them.

That is all about the concepts' part, but what about the applications? Data mining is a tool and holds core part in business intelligence (BI). Data mining plays a critical role in case of accurate and complex decision-making. Consider a simple example of a bank which wants to access credit risk of customers. Let us say Ram applies for loan. Should the loan be approved? Data about past credit history, timely payments, security, age, salary are some of the factors that are looked upon. Bank generally develops models using machine learning methods, with the parameters mentioned. The results predict whether Ram would default on loan or not.

2. *Computational learning theory* (*COLT*): Currently, a lot of research is done to study and analyse algorithms. In COLT, formal mathematical models are defined. These models help in analyzing the efficiency and complexity in terms of computation, prediction and feasibility of the algorithms. The analysis done provides a framework to take appropriate decisions for building better algorithms that would be effective in terms of data and time.

The computational learning theory finds its importance in the field of machine learning, pattern recognition, statistics and many more. With regard to machine learning, the goal of COLT is to inductively learn the target function. Learning theories help in understanding the explicit relevant aspects of the learner and the environment to classify easy and hard learning problems and in turn guiding the design learning systems. There are two frameworks for analysing the patterns—one is Probably Approximately Correct (PAC) and the other is mistake bound. The former identifies the classes of hypothesis that possibly can/cannot be learnt, whereas the latter tries to learn target function to series of trials.

3. *Neural and evolutionary computation*: A new technique in computation, i.e., neural and evolutionary computation is enabled to speed up the mining of data. Computation techniques that are based on biological properties fall under the category of evolutionary computing. Evolutionary computing is related to the study and use of these properties, consisting of evolutionary algorithms (of which genetic algorithm has been the most popular) that are basically used to solve multidimensional problem. The evolutionary computing finds its applications from the telecom domain to the financial decision-making, with optimisation as the base criterion.

In case of neural computing, the neural behaviour of human beings is stimulated to enable machine to learn. An artificial neural network is formed or configured for some specific application like pattern recognition or classification.

4. *Intelligent agents and multi-agent systems:* Intelligent agents and multi-agent systems (MAS) is a core part of intelligent systems, which allows timely decision-making in complex scenarios. An *agent* in simple terms, is a software program that assists user. An *intelligent agent* is the one which is flexible in terms of its action to get the desired outcome. It is goal-directed, reacts with the environment and acts accordingly. Consider an example of a student, who is pursuing a course in web designing. He uses search engine to get some notes for the subject. An intelligent agent will observe that he accesses the sites, which give him the detailed examples of the topics. So, each time he fires some query, the agent will give up sites that he is likely to refer based on the past experience. After some days, when he refers to the sites with illustrations, then the agent would need to change its behaviour pattern and act accordingly.

The capacity of an intelligent agent is restricted, and is dependent on the knowledge it has, the available resources and the different perspectives. The percept of individual agent is always limited. Complex tasks and decision-making demand combination of more than one percept of different intelligent agents. Hence, in many cases group of intelligent agents are required to solve the problems. This is a scenario of multi-agent system. So, in MAS, every agent's capability and its computation efficiency is exploited so that the overall performance is improved.

5. *Multi-perspective integrated intelligence:* For any problem to solve, each and every individual can have his own perspective. Some information might be present in some perspective, while it could be missing in other perspective, which could be effective in terms of decision-making. Utilising and exploiting this knowledge from different perspectives to build up an intelligent system giving accurate results, builds the Multi-perspective Intelligence (MPI) framework. Consider a scenario, where you want to apply for a job in a renowned company. You tend to seek feedback from some employees. Each will have his own perspective in relation to management, working environment, appraisals and so on. Some friend of yours might not be working, but is acquainted with the company. He would also have a different perspective. Based on this knowledge, possibly you could land upon a decision whether to take up the job or not. Information collected from different perspectives is used for final decision-making. This information collection can be continuous or discrete.

These learning approaches work in association with respect to the application they would be suited for. As said earlier, there is a very fuzzy line of distinction between them. And a good understanding of requirements and domain will result into accurate predictions and decision-making for solving a problem. The topics introduced here are just to make you aware of the type of work done with AI, though we will be discussing most them in detail in further chapters.

SUMMARY

Artificial intelligence has been a part of active scientific and engineering research for the last six decades. AI has gone through various stages of research during this period. Slowly, AI has become a part of the mainstream research, and many existing products and technologies are based on AI. The machines and equipment we use in our day-to-day life in some way or other use AI. Selection of model, analysis of data and building knowledge are some of the important aspects of building AI system. A large number of applications of AI need to be studied along with the practical aspects of AI to build a real-life AI solution. We want every appliance and every activity performed by machine to be intelligent. This intelligence is about understanding the problem at hand as well as the scenario and acting reasonably. (Intelligence is multi-faceted entity and this book tries to look at these facets more pragmatically).

KEYWORDS

1. **Semiotic models:** The models which are based on sign processes or signification and communication.
2. **Formal system:** It is a system based on assignment of meanings to the symbols.
3. **Natural language processing:** A computer science and linguistic branch dealing with the interactions between human being and machines/computers is known as natural language processing.
4. **Maze models:** Creative and intelligent tasks handled by human being are modelled like a set of maze of paths from an initial node to a certain or resultant node.
5. **Well-structured problems:** These problems yield a right answer or right inference when an appropriate algorithm is applied.
6. **Ill-structured problems:** These are the problems which do not yield a particular answer. In this case, there is possibility of more than one answer, and even a particular situation decides the correctness of the answer.
7. **Linear problem:** It is the problem which can be solved or where decision can be obtained by linear solution.
8. **Non-linear problem:** It is the problem which cannot be solved or separated by linear equations.

CONCEPT REVIEW QUESTIONS

1. Give example of one ill-structured problem with description and elaborate the method for solving that problem.
2. Describe various AI models.
3. Explain the model building concepts in AI.
4. What are the statistical models?
5. List various equipment in day-to-day life, where AI is used.
6. List milestones in AI evolution.

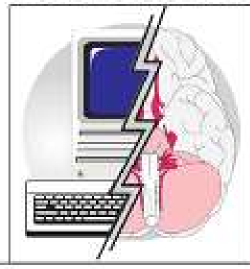
CRITICAL THINKING EXERCISE

1. Analyse knowledge-based complexity for the AI application for chess and tic-tac-toe.
2. Try to map different AI models with AI products.
3. List applications where formal systems are used.
4. Associate evolutionary systems with traditional one from technology aspects.

PROJECT WORK

1. Program a simple intelligent system for tic-tac-toe.
2. Draw a flow diagram where you use evolutionary system for washing machine.

CHAPTER 2



Problem Solving

Learning Objectives

- To identify a problem and design automated solution for any formalised task.
- To represent the problem in generic way so that the solutions do not remain instance-specific
- To understand the concept of state space and the role of search in the process
- To understand the complexity in solving the problem and decomposing into sub-parts to reduce it further with optimal solution
- To understand the ways to solve problems and search-based problem solving

INTRODUCTION

The field of AI comprises many fascinating areas but problem solving is fundamental to many of the AI-based applications. *Problem solving*, as the name suggests, is an area to deal with finding answer for some unknown situations. It involves understanding, representation, formulation and solving. This simple definition encapsulates two types of problems—simple and complex. The simple type of problems are the ones that are or can be solved by a deterministic procedure. There is a guarantee of a solution. For example solving a quadratic equation. So, when we say that the example falls under simple category, we are aware that there is a way to have this problem mapped into an algorithm that can be executed by the computer.

In the real world, the problems are complex. Most often the data that we have is partial. In other words, we are not fully aware of the scenario. There is lack of full information. So, the problems with such a data are complex problems. Let us take a simple example of a robot. For the robot, it is complex to carry out some specific tasks like searching some object or loading boxes in trucks. It is because of the movements (that it needs to carry out), which are dependent on the information at that particular instance. Since the information available at that instance is based on partially observable environment, it is incomplete.

Solving complex problems is indeed a complex and tricky task. For us, solving a problem at hand is not so difficult since we can reason out, perceive, learn, but for a machine, it is actually very difficult. While drawing conclusions, we can use the statistical methods, the mathematical modelling processes and so on to get the best solutions. AI focuses on mapping of these intellectual abilities into the machine to get the best solutions. In this chapter, we discuss about the real-world problems and their solutions.

So, what is the main job or the objective of problem solving? The objective is to find out set of actions so as to reach the set goal. Do we mean to say that we are having some sort of search? Search is, of course, a method for problem solving, but its details are given in the subsequent chapters. This chapter is focussed on the details of problem solving, the processes and the representations, with an overview of the search method that is applied in the process of solving.

Coming back to the point of discussion, where the important approach of problem solving is search in an action space. Action space comprises a set of actions that leads to some specific goal. The methods of problem solving can be categorised as follows:

1. General purpose
2. Special purpose

The general purpose method is applicable to a wide variety of problems and it is means-ends analysis. In means-ends analysis, the present situation is compared with the goal to detect the difference. It searches in action space to select an action that will reduce the difference. Say for example, if a person wants to cover 50 km distance, the problem solving process will search from memory for auto-rickshaws, buses, bicycles and so on for the means of transport. It would then discard walking and flying so as to reduce the search space for the next step. In case of special purpose, a particular problem is modelled with various assumptions, which are specific to that problem. Specific features of the situation are used in this method. For example: classifying legal documents with reference to a particular criminal case.

Sometimes, problem solving system behaves like an expert, who has gathered knowledge on a specific area over a period of time. Then, the expert uses his knowledge to relate to the current situation and derives analogy to find out the best possible solution. In AI approach, algorithm using historical knowledge can be treated as expert. Say for example: during diagnosis of a disease the doctors feed the diagnosis process and the disease-related symptoms. This helps in building the knowledge base of diagnosis and symptoms. The knowledge base of diagnosis and symptoms helps in diagnosing in future for similar scenarios. In case of new symptom, the doctor can update knowledge base based on new observations. This can help in diagnosis process, as it can provide pointers to arrive at conclusion.

Problem solving can also assist in planning and decision-making. Planning and decisions are the key parts of an intelligent system. Managers need efficient problem-solving methods for assisting and helping them with optimised decisions. An effective step-by-step sequence of actions that has low cost and reasonably less number of steps is important in case of planning. Planning is optimal ordering of actions to reach the solution in a given scenario, environment and constraints. Even it may be effective in manufacturing units, where assembling of goods needs to be done in a specific time and specific order so as to reduce the other overheads.

In many situations, the pattern of information is used to retrieve the right information from the search space. A chess player can make use of the pattern of moves to decide the next move. Problem solving, thus, tries to make optimal use of information and knowledge at hand to select a set of actions to reach the goal state.

2.1 PROBLEM-SOLVING PROCESS

The term *problem* is used in a situation, when the desired objective is not obvious. Arriving to an objective from initial situation is unknown initially and it consists of sequence of intermediate objectives. This process of solving a problem may vary from individual to individual. An individual's acumen, knowledge and skills—all put together to generate a solution.

Problem-solving is a process of generating solutions for a given situation. Figure 2.1 shows problem-solving process applied to achieve goal state. This process consists of sequence of well-defined methods that can handle doubts or inconsistency issues, uncertainty, ambiguity and help in achieving the desired goal.

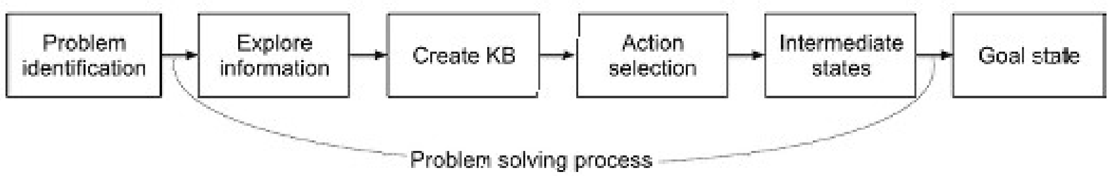


Figure 2.1 Problem-solving process.

The word *problem* encompasses computational tasks also. When we talk about problems like matrix inverse or solving quadratic or simultaneous equations, then the process involves computation to arrive at a solution. But in case of AI, the problem is formulated and solved by searching for a solution in space of possible solutions. Gradually, the problem-solving horizon is widened to include the other techniques of natural language processing, machine learning, game playing and so on. The problems may also include sophisticated information storage, retrieval, information extraction, decision-making and so on.

The term *problem* can be defined with following conditions:

1. Every problem is defined in a context. In this context, it has certain assumptions under initial conditions.
2. Every problem has a well-defined objective.
3. Solution to every problem consists of a set of activities. Each activity changes the state of problem, i.e., from the present state to the new state. This new state is closer to the solution state. Finally, initial state approaches the goal situation.
4. Previous knowledge and domain knowledge both are used as the resources during different states in the solution process.

So, the primary objective in solution process is the problem identification. Problem identification marks the problem space boundary. This precisely tells us what the achievable goal is and what information is to be used during the solution process. In case of an operational research, the objective is to optimise the operation cost of an item under boundary conditions. If they are not defined properly, number of operations and resources required may increase substantially. Hence, the solution may become complex, and sometimes, it may not be achievable. There are many real-world problems that cannot be solved if they are open for generic solution. But if it is limited to certain environment, it is solvable. Every problem needs different treatment depending on the goal, initial information and assumptions. General problem-solving techniques involve the following:

1. Problem definition
2. Problem analysis and representation
3. Planning
4. Execution
5. Evaluating solution
6. Consolidating gains

Let us take our discussion ahead to the formulation of the problems.

2.2 FORMULATING PROBLEMS

Identification of problem is the first step in problem-solving process. Many questions are to be answered while defining a problem. Once we get the answers to these questions, we can say that the problem definition is complete. To mention a few—What characterises the class of problem to be solved? What method (or technique) is used in the current stage? What information is necessary to specify a particular instance? During the course of getting the answers, specification of the achievable objective is very important. Every problem has certain initial conditions from which different actions are initiated. Finally, a set of actions or methods to be defined that cause the transitions are also important. A problem statement can have description of the data, method, procedures and algorithms that are used to solve it.

Once the problem is identified, we need to be very precise and specific with respect to the problem space along with the target that we need to achieve. At this stage, the assumptions or limitations of solution (if any) are also specified so that the solution quality is measured against the defined problem objectives and requirements.

The next important step here is the analysis and representation of the task knowledge. Usually, we understand the problem in terms of diagram, description and so on. But in AI, the target is to use machine to solve a problem, once the solution is planned and fed to it. This is done using state-space diagram. So, the problem is defined in terms of state. Each state is the abstraction of all available information. *Solution* to any problem is the collection of such different states and set of operations. This collection of states is termed as *state space*. Each of these states is achieved using the application of actions/operations to the previous state. During problem-solving process, an operator is applied to a state to move it to the next state. Then, another operator is applied and so on till the final state is achieved. This approach of generating a solution is called *state-space method*.

Figure 2.2 represents the steps involved in formulating problems.

Let us begin with the state-space approach for problem solving using a very well-known puzzle.

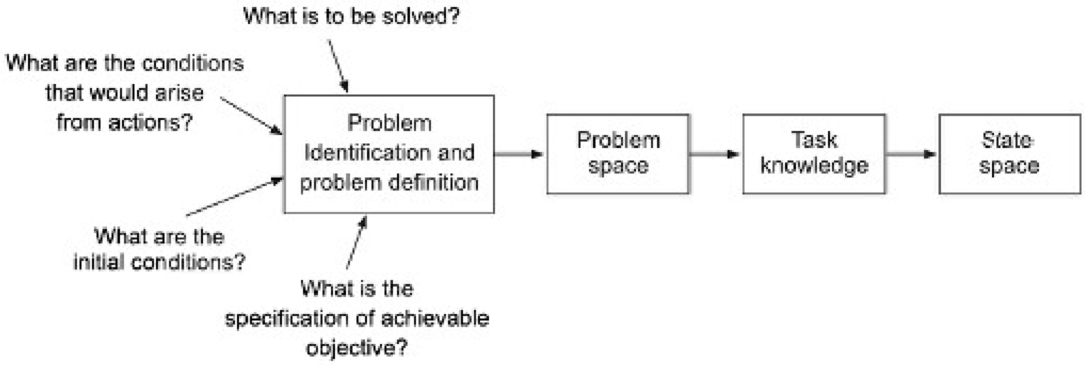


Figure 2.2 Problem formulating steps.

EXAMPLE: Consider a problem where three cells in the four-cell board are filled with single digits and one cell is left blank. The game is to change positions of the digit and blank cell of the board to arrive at new board positions. The rule of the game is blank cell can change the position with a digit by horizontal or vertical movement. Figure 2.3 represents the initial and the final states of this game.

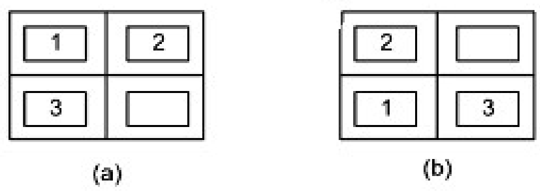


Figure 2.3 Initial and final state of the puzzle: (a) Initial state and (b) Final state.

PROBLEM: To reach from the initial state to the final state, with the minimum number of moves.

Solution: We now define the state space, and the operation and action space. Action space has the operations on the blank cell. Blank cell can move up (U), down (D), left (L) and right (R). Different board positions constitute the elements of state space. We designate every state by a symbol S_i. i represents the ith state. S_0 is the initial state and S_n is the final one. Figure 2.4 illustrates the states along with the operations. Here, in Figure 2.4, $S_n = S_3$.

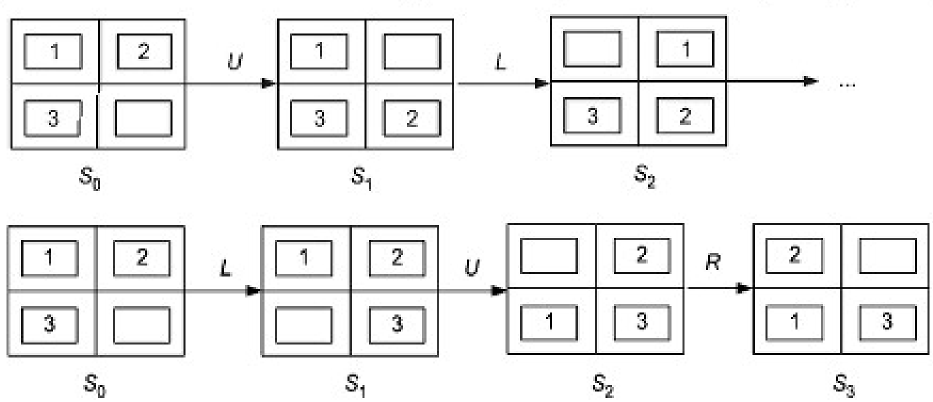


Figure 2.4 State transitions in the puzzle game.

From Figure 2.4, it is clear that there are actually two ways to solve the problem. The first solution starts with up, whereas the second starts with left. Clearly, the left move leads us to the goal in lesser number of steps than that of the up move (as the initial move). With each state change, the state is compared with the goal state. The puzzle discussed is simple, but it gets complicated when the digits are increased. What should be done under this changed scene? Let us put forth an algorithm for it.

1. State = initial state; existing state = state
2. While state ≠ final state
   (i) Existing state = state
   (ii) Apply operations from set $\{U, D, L, R\}$ to each state so as to generate new states.
   (iii) If new states ∩ existing states ≠ φ
      * Existing state = existing state ∪ new states
      * State = new state

   End while

A well defined problem, hence, is described in terms of

1. Initial state
2. Goal state
3. List of states
4. Operators or functions that change state or transition of state
5. Path (sequence of states leading to goal state)
6. Path cost (functions that assign a cost to the path)

2.3 PROBLEM TYPES AND CHARACTERISTICS

Problems can be categorised into different types. Let us look at these types.

1. *Deterministic or observable:* This type of problem is also termed as *single-state problem*. Each state is fully observable and it goes to one definite state after any action. Here, the goal state is reachable in one single action or sequence of actions. For example, vacuum cleaner with sensor. Here next state can be found using current state and action.

2. *Non-observable:* This type of problem comes under multiple-state problems. So, the problem-solving agent does not have any information about the state. Application of operator can lead problem to multiple states in this case. Hence, each state goes to a number of states after the application of an operator. Solution is a set of states leading to a set of goal states. In this type of problem, solution may or may not be reached. Let us take an example of vacuum cleaner. The goal state is to clean the floor rather clean floor. Action is to suck if there is dirt. So, in non-observable condition, as there is no sensor, it will have to suck the dirt, irrespective of whether it is towards right or left. Here, the solution space is the states specifying its movement across the floor.

3. *Non-deterministic or partially observable:* In this type of problem, the effect of action is not clear. In every new state, some new information is added and then operator acts on the state. Solution space is a tree structure of states. Solution is based on searching the tree and finding out the path for solution. If we take the same example, and now assume that the sensor is attached to it, then it will suck if there is dirt. Movement of the cleaner will be a tree that would be based on its current percept.

4. *Unknown state space:* Unknown state space problems are typically exploration problems. States and impact of actions are not known. There is a need to discover to understand the outcomes of actions. For example, online search that involves acting without complete knowledge of the next state or searching address without map.

2.4 PROBLEM ANALYSIS AND REPRESENTATION

Performance of a solution depends on the problem representation, as discussed earlier. This task can be simple or complex depending on the way it is described. A problem representation is a complete view of the problem and approach to solve it. Determining computational complexity for the solution is not possible until a formal representation of the problem is generated.

The performance of any intelligent system depends on the problem representation and formulation. A representation is a complete view of a problem or set of problems and an approach to solve these problems.

The problem representation is an important step to understand before we start solving the problem. The quality of solution and the appropriateness of solution mechanism rely on completeness of the problem definition.

Problem definition should satisfy the following criteria:

1. Compactness (must be able to restrict and define boundaries clearly)
2. Utility (must be compatible with good solution algorithms)
3. Soundness (should not report false positive or false negative)
4. Completeness (should not loose any information)
5. Generality (should be able to capture all or maximum instance of the problem)
6. Transparency (reasoning with the representation efficiently)

Let us take an example of 'Smart home' to understand these properties. The objective of the smart home can be:

To provide a secured home, with 24 hours surveillance, together with facilities of intruder detection system, detection of anomalies like fire, earthquake, etc. along with efficient communication and alarming system. Let us elaborate the criteria for this example.

- **Compactness:** It needs to define clearly the solution space.
- **Utility:** It needs to be compatible with the existing systems, say for example, social security systems available in the city and systems at government organisations like police stations.
- **Soundness:** It should not raise false alarm for intrusion or fire.
- **Completeness:** It should not loose information about the visitors or the historical information of previous instances and should use these instances to learn and handle similar future events.

- **Generality:** It should be able to handle all similar events irrespective of changing environment. For example, the alarming system should be able to work properly irrespective of the season and day of the time.
- **Transparency:** The reasoning behind the action taken should be visible to the user.

Problem representation with different aspects

Problem representation is very critical and important in problem solving. Many different opinions and views should be investigated and explored in this representation. First and foremost, it needs machine language so as to describe the logic and specific encoding rules for the problems. Further, the problem needs to be mapped into a solution space that is actually expanded during the solution process or it can be in the form of a state space that has all the feasible states and transitions. A representation comprising data structures and programs executing on them to derive inference can also be a different view in the representation.

Researchers in 1950's worked extensively on human problem-solving process, where they encoded different tasks in the problem space through state diagrams. In these diagrams, initial situation, desired goal and various intermediate states or concepts are used.

So, problem space consists of partial solutions, which finally determine the complete solution. This approach is further extended for AI, where problems are different in many ways. The moves in case of AI can be abstracted from the real-world states. The steps of problem solving can be modelled into a system such that a machine can execute these well-defined steps and find out the solution. This automatic problem-solving system is termed as *problem solver*.

So, a *problem solver* is an algorithm or a methodology that accepts problem description, domain description and boundary conditions as input, and then, searches for a solution. The solver terminates when it finds a solution or when its solution space is exhausted or it meets the boundary condition that is defined in input. The boundary condition may include time boundary or solution boundary. When the solution space does not converge, then termination takes place either due to time or tolerance as the acceptable output variation is reached.

Coming towards the point of problem description, a problem description contains domain description and problem instances. Domain description includes properties of simulated world. For example, in case of tower of Hanoi in which possible states and moves are the description of the problem domain. Problem instances are the initial states and the goal states of the problem. So, for tower of Hanoi, problem instance may be the initial position of the disk and final goal is the desired position.

Figure 2.5 shows the problem instances for initial and goal states and possible states in the process of solving.

Problem representation consists of domain description and solver algorithm that operates on this description. The starting point for a complete problem representation is the determination of typical knowledge structure and interdependency among knowledge at different layers. A typical knowledge structure is depicted in Figure 2.6. Design knowledge plays an important role in the problem-solving process and it is at the root level in the diagram. This design knowledge actually controls and establishes the flow of information

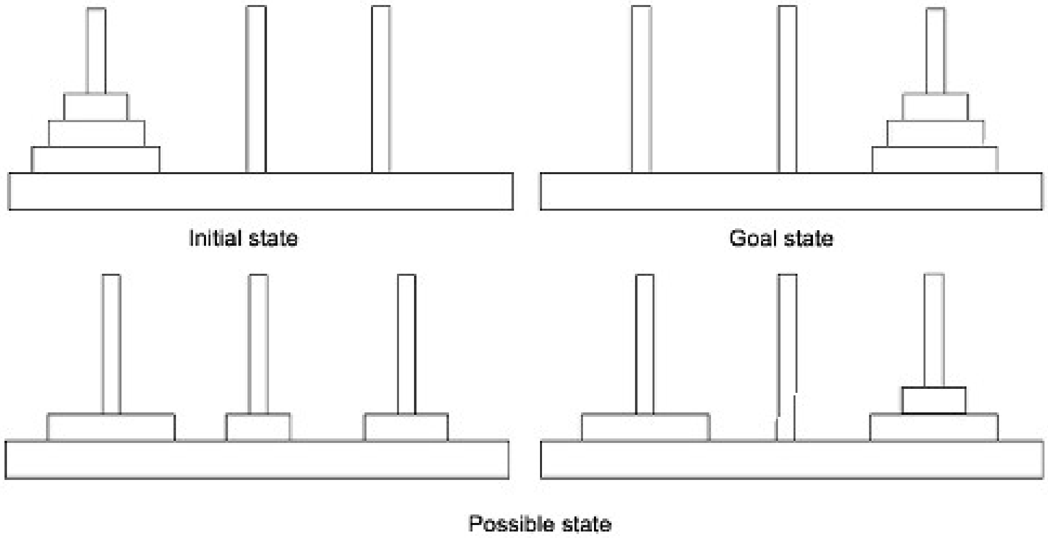


Figure 2.5 Tower of Hanoi: States.

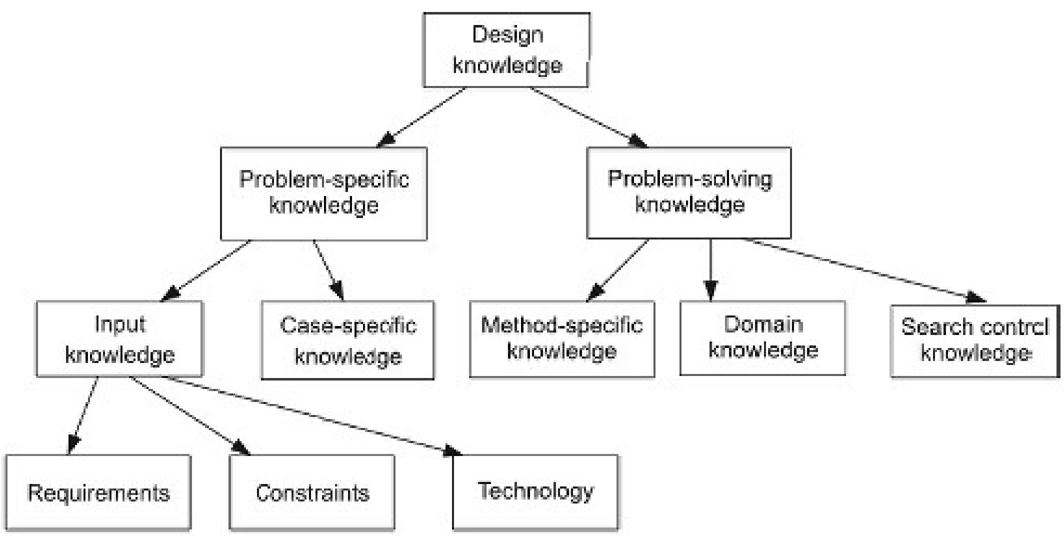


Figure 2.6 Knowledge involved in problem-solving process.

from different knowledge layers to execute the operations in order to reach the desired goal. Basically, design knowledge is used to encapsulate solution strategy by formulating the problem. So, it directly makes use of the problem-solving knowledge and problem-specific knowledge. Problem-solving knowledge is a generative way of problem-solving. In AI paradigm, problem-solving deals with the searching of solutions under a pre-defined boundary. So, search techniques come under the pre-requisite knowledge for devising a solution. Domain knowledge is a definitional knowledge, which is related to the problem domain. Whereas, problem-specific knowledge includes explicit representations of all objects, classes, their relations, constraints, etc. This can include case-specific knowledge also. This specific knowledge is required to solve a particular problem. It may comprise functions, structural goals or typical requirements of a problem. It also includes input knowledge. Problem requirements are also one type of input. At the same time, constraints are used to express explicit input knowledge of a problem and verify inconsistency of a solution. There may be different forms of constraints like logical, arithmetic or structural.

We now discuss about this problem formulation and problem representation with an example of water jug problem.

The problem is defined as follows:

There are two jugs without scale of measurement. One can make a jug empty or fill it by pouring water in it. One of the two jugs has 5 gal capacity. The other is smaller and has 2 gal capacity. There is 3 gal water in the big jug and 1 gal in the smaller one.

PROBLEM: To find out a way to empty 2 gal jug and fill 5 gal jug with 1 gal water.

Let us first convert this problem into a state-space (problem problem formulation step). So, we need to define the states, actions and goals.

States: Amount of water in the jugs

Actions:
1. Empty the big jug.
2. Empty the small jug.
3. Pour water from small jug to big jug.
4. Pour water from big jug to small jug.

Goal: To get the specified amount of water (1 gal) in big jug and empty the smaller jug.

Path cost: The number of actions applied. (Minimum the number of actions, better is the solution).

Once we have the problem formulated, the next step is to represent. The states and the operators are to be precisely defined so that they can be programmed and executed. Problem representation includes initial condition, initial values, dependencies (if any) among the states and the variables. When these parameters are defined, we say that the problem is well-defined. The quality of solution is also dependent on the precise presentation of the problem that considers all the essentialities.

Representation: States can be defined as the amount of water in both the jugs (b, s), where b is the amount of water in the bigger jug and s is the amount of water in the smaller jug.

Initial state: (5, 2)

Goal state: (1, 0)

Operators: Let us discuss about the selection of operators for this problem. At this stage, operators are to be defined carefully with all possible constraints. Applicable operators are to (i) empty the jug or (ii) fill the jug. We need to apply constraints or pre-conditions for these operators. So, the pre-condition to empty jug is the jug was not emptied earlier.

One situation is when one of the jugs is not empty, then there are two alternatives—(i) Empty big (initially, big jug is not empty and it is emptied to fill in the small jug) and (ii) Empty small (initially, small jug is not empty and then it is emptied to fill in the big jug).

Other situation is when one of the jugs is empty. Again, there are two options—big jug is empty and small jug is not empty. For small to be emptied, the states will be $(s, 0)$. That means the big jug has the contents of small and small jug has nil. Similarly, it can be (*b*, 0). The next state after this can have possibilities that (*b* – 2, 2) since small is empty so, it will have 2 gal and 2 are reduced from big (this is if *b* >= 2). But if *b* < 2, then the next state will be (0, *b*), where big will be empty and *s* will have *b* contents.

Not making things complicated, we will solve it and have a path cost calculation. The sequence of operations to be performed on the two jugs is shown in Figure 2.7. The solution is found by searching through the action space defined as operators 1 to 4.

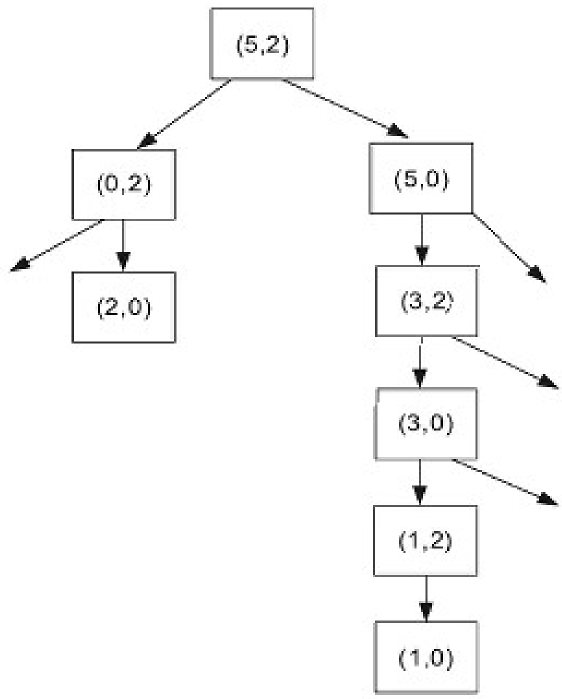


Figure 2.7 Water jug problem.

In Figure 2.7, the initial state, the intermediate state and the final states have been presented in the form of tree structure. The initial and final states are marked bold, while the intermediate states are marked by the application of the operators. So, searching for solution is same as searching for the tree. Let us look at the details of it.

Initial state: (5, 2)
Goal state: (1, 0)
Operators: These are as follows:

1. Empty big (remove the water from big jug).
2. Empty small (remove the water from small jug).
3. Big is empty (water in small jug to be poured in big jug).
4. Small is empty (water in big jug is to be poured in small jug).

Figure 2.7 shows that the goal state is achieved with the action of sequences 2, 4, 2, 4, and 2. Thus, the precise operator definitions along with the goal and the initial states representations assist in solving the problem. So, we have discussed about the basic representations to solve the problem, but on what basis is the performance accounted? Let us consider the same in the next section.

2.5 PERFORMANCE MEASURING

There are various factors that are to be considered when it comes to problem solving. The performance is governed on different things. Let us begin with the factors.

There are three outcomes of a problem solver—finding a solution, terminating with failure after search space is exhausted and hitting a time bound like terminating after certain number of iterations or trials. We get rewards when problem is solved, but pay for time required for solving. So, the reward is dependent on the specific problem and its solution. Investment of longer time and resource will make the result negative if the problem is quite simple like arranging the numbers.

When it comes to performance gain, it can be measured by four parameters—problem, time, resource and result. Result is, of course, success or failure. Success means the problem has been solved successfully. Even though success is achieved, quality of the solution is also important. For example, let us consider an intelligent system providing driving assistance that suggests optimal route in terms of the shortest distance. The suggested shortest route may not be the preferred one in terms of quality owing to the road conditions on that route. Thus, the quality impacts the preferred solution. How close the solution is to the real-world value and the possibility to have the final state to be achieved in every course of execution also need to be looked at. We are well-versed with the time and space complexities. The cost of the solution is measured in terms of the time and the resource cost required to find the solution. So, it may be the sum of the storage (space) or the hardware configuration and the execution time. One thing to mention is that gain reduces with the time and storage.

2.6 PROBLEM SPACE AND SEARCH

As we have discussed earlier, the problem is represented as the state space. Let us understand the role of search in detail here.

Search is a general algorithm that helps in finding the path in the state space. Every problem (as said before) can be solved with the help of search. It considers one or more path. The path may lead to the solution or might be dead end. In case of dead end, backtrack should occur. The search algorithm, makes use of control strategy like that of forward or backward search. Further, it needs to identify and adopt a strategy to explore the states. In broader sense, there are two types of strategies—informed search and uninformed search.

Informed search does not guarantee a solution, but there is high probability of getting a solution. In this approach, a heuristic that is specific to the problem is used to control the flow of solution path. Heuristic is a skill-based technique used to solve the problem. It is based on common sense, rule of thumb, educated guesses or intuitive judgment. It is high order cognitive process. A heuristic function is used in search algorithm to rank the alternatives at each multiple selection path in order to take decision about which path should be followed. When speedy process is required, informed search is preferred over the uninformed.

In case of uninformed strategy, it does not consider specific nature of the problem. This strategy is very simple and can be generalised to any problem. Uninformed strategy generates all possible states in the state space and checks for the goal state. Hence, it always finds a solution if it exists. But the approach is time consuming, as the search space is large. The strategy is preferred in applications where any error in the algorithm has serious consequences. This method is also used to benchmark results of the other algorithms.

The search methods are evaluated on the basis of completeness (if there exists a solution, it needs to find it), optimality (the obtained solution is the best one), time complexity and space complexity.

This was an overview of the problem solving and the search relationship. We will have a detailed study of these methods and strategies in next two chapters dealing with search in detail.

2.6.1 Defining the Problem as a State-Space Search

The problem at hand can be solved by searching the state space to find out the path that leads to a final state from the initial state. The basic notion behind the state-space search is that we can solve the given problem by checking the steps considering the fact that they might lead us towards the solution. So, each action takes a step ahead to a different state.

While defining the problem as a state-space search, we mean to say that the goal should be properly formulated. As said earlier, the states and the operators should be clearly indicated. There should be transparency in describing the rules and they should be as generalised as possible.

2.6.2 Issues in Design of Search Programs

In problems solving, we apply search. The search leads us to goal state. So, can we say the result or the solution we have obtained is dependent on the search process? Yes, indeed it is. Search techniques guide to reach a solution in the space. The way we have obtained the solution for the jug problem by applying the rules, the entire path is said to be the solution path, starting from the state (5, 2) to the goal.

We, as human beings, can analyse the state space and identify the solution. This happens because of the cognitive skills that we possess. But while searching, the state space is traversed. If an error occurs, backtrack should occur so as to evaluate the alternative path. So, there is a need of a systemic procedure for forward and backward movement at the same time systematic method for traversing states in the state space.

When we are having a search program implemented to solve the problems along with the direction of search, the other factor that needs to be highlighted is the selection of the rules. Which rules should be applied so as to reach the goal state is equally important. Further, the representation, as discussed earlier, is also a significant issue. This representation is about the knowledge that maps the node in the search process.

To summarise, the following issues are observed while designing search problems:

1. State representation and identifying relationships among states
2. Proper selection of forward and backward movement to determine optimal path to goal state
3. Rule selection

2.7 TOY PROBLEMS

Toy problems are not real-world problems. They are generated for fun. These problems are played on board or paper. Hence, the problem environment is controlled. The problem formulation is, thus, possible. All boundary conditions and the constraints are known. Hence, it is very much possible to convert them to computer programs. Many games like 8-puzzle, tic-tac-toe, backgammon are toy problems. Let us understand these problems as a problem-solving venture.

2.7.1 Tic-tac-toe Problem

It is a very well known pencil and paper game for two players 'x' and 'o'. (In fact, we have grown up playing this). Each player marks a 3*3 grid by 'x' and 'o' in turn. The game starts with 'x' and ends with win or loss or draw. The player who puts respective mark in a horizontal, vertical or diagonal line wins the game. If played without a fault, it leads to draw.

We begin now with the formulation of the game as a problem that can be solved by search strategy. By this, it is possible to convert it to a computer program. Let us consider the rules, say the rule of the game is to put 'x' and 'o' in the cell. Game gets over when either 'x' or 'o' are at the diagonal positions, as indicated in Figure 2.8.

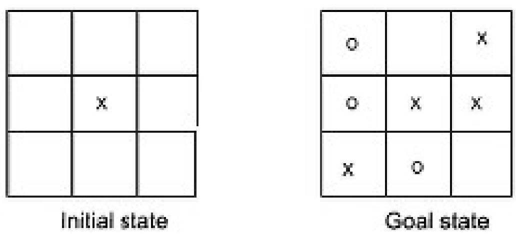


Figure 2.8 Tic-tac-toe game.

Formulating the problem in state-space search, we get

Initial state: State in Figure 2.8.

States: Figure 2.9 with different, 'x' and 'o' positions constitutes the states in the space. The generation of states in the state space is shown in Figure 2.9.

Operators: Adding 'x' or 'o' in cells one by one

Goal: To reach the final/winning position

Path cost: Each step costs 1 so that the path cost is the length of the path.

2.7.2 Missionaries and Cannibals

The missionaries and cannibals problem is the one, where three missionaries and three cannibals are on one side of a river. There is a boat that can hold one or two people. We need to find a way to get everyone to the other side of the river without ever leaving a group of missionaries in one place outnumbered by cannibals in that place. A maximum of two objects (missionaries/cannibals) can travel to other side and minimum of one has to be there in the boat for the boat to come back.

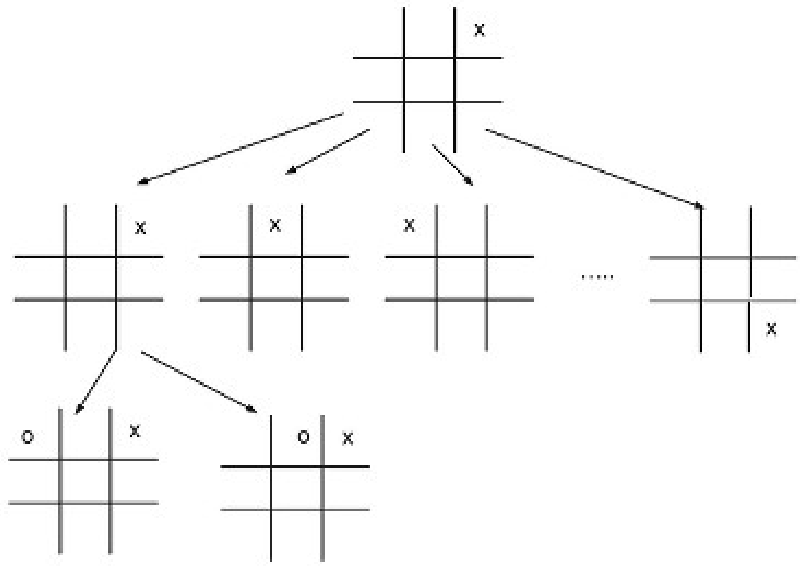


Figure 2.9 Tic-tac-toe state space.

The first step towards solving the problem is to find out the operators that involve taking one or two people from one side to another in the boat. A state represents the time when missionaries and cannibals are in the boat or on the other side. Because boat can hold only two people, there is no question of outnumbering. But the crossings are important. So, there are constraints while applying the operators.

States: Let the state consists of sequence of three numbers representing the number of missionaries, cannibals and boat. So, the initial state is (3, 3, 1).

Operators: They carry one missionary, one cannibal, two missionaries, two cannibals or one of each across in the boat.

Let us formulate the problem.

Goal state: It is the state, where missionaries and cannibals have reached the other side of the river. So, the final state is (0,0,0).

Initial state: (3, 3, 1)

Operator: Putting missionary and cannibal in boat such that the cannibal does not outnumber missionary and there may be one/two people in the boat at a time.

Path cost: Number of crossings

This is a simple problem for a computer to solve if it is represented rightly as state-space problem.

Figure 2.10 represents the initial and goal states and the snapshot of slate space diagram. The diagram begins with initial state where the boat, missionaries and cannibals are on one side of the river. It can be represented in other ways as well, where the final state can also be added in the representation with values as 0, 0, 0 for the missionaries, cannibals and the boat; and changing them with every transition from one end of river to other.

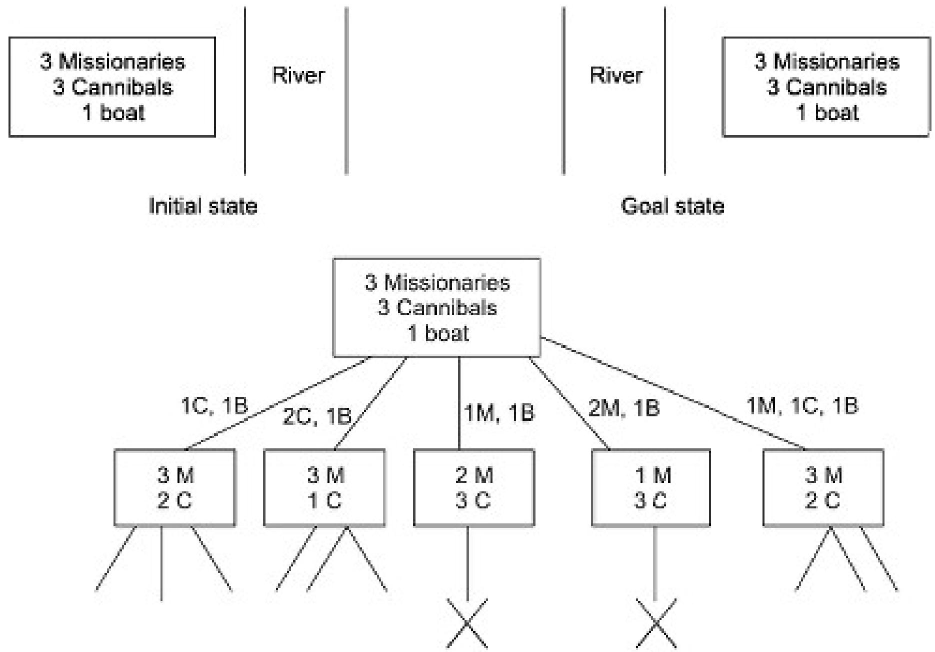


Figure 2.10 Missionaries and cannibals problem: Initial and goal states and state space.

2.8 REAL-WORLD PROBLEMS

Toy problems are used to illustrate different problem-solving methods and compare the performances of the algorithms. These problems are given concise and exact description. But the real-world problems, tend to have a single agreed upon description. The research on problem solving actually focuses on capturing properties of real-world problems. The cognitive processes of researchers attempt to solve these problems. Couple of day-to-day real-life problems are discussed below.

Route Finding

Route finding is defined in terms of specified locations and transition along links between them. Route finding algorithms are used widely in a variety of applications like automated travel advisory system, airline travel planning, car systems that provide directions and so on. The other applications can be video stream in computer network.

Let us consider airline travel planning problems that is solved by an automated travel advisory system. The objective is to arrive to a destination with the minimum cost. Each state comprises airport location and the current time. Let us formulate the problem.

Initial state: Starting point of the journey

Goal state: Final destination

Operators/Actions: Flight from current location to reach the next location in such a way so that there is enough time difference in the next flight and in transfer within the airport if there is a connecting flight.

Path cost: Total of travel fare, travel time, waiting time, time of the day, type of airline and can include special package, mileage awards, etc.

Most often, the commercial travel systems use a problem formulation of this kind, with many additional complications such as backup reservations on alternative flights. This is done to the extent that these are justified by the cost and likelihood of the original plan.

Travelling Salesman Problem

Travelling salesman is the most famous touring problem, where the objective is to find a tour—the shortest one. So, given number of cities and a starting city, the problem solving needs to cover all the cities, where each city is traversed just once minimising the tour cost and returning back to the starting city. There are many variants to this problem. Let us understand the representation of the standard one from Figure 2.11 below, where C_1 is the starting city.

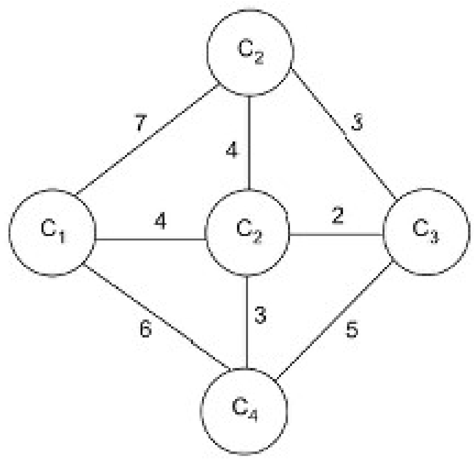


Figure 2.11 Travelling salesman: Cities and distances.

The problem can be formulated as

Initial state: The starting city for the tour is C_1.

Goal state: The complete trip ends at C_1.

Operator: Travel from one city to another with the constraint that the city is not visited before.

Path cost: Travelling time

2.9 PROBLEM REDUCTION METHODS

Problem reduction is a strategic approach to reduce complexity of a problem. A general approach for solving a large and complex problem is to decompose it into some smaller problems. Further apply this decomposition process to each of these problems iteratively till they get solved easily or they cannot be split. This paradigm of problem solving is called *problem reduction*. This approach of problem reduction can be applied to the applications of any domain in which top-down decision-making strategy is applicable.

Many a times, a problem seems to be hard to solve. These types of problem are reduced to a number of small sub-problems. Once the sub-problems are solved, the solution to the hard problems can be achieved. This is the core concept of the reduction methods.

We are well-versed with the representation in state-space search and sequences of actions that are to be followed with the search so as to reach the goal. Other approach is to decompose the goal. In the sense, more simple sub-goals are formed. As an example, when a trip is to be planned from city A to city B, which involves a sequence of actions like travel by air, travel by train and so on then, the trip can be split into many parts. The parts may be reach the station, get a train and reach city B. There can be many alternatives parts for this trip too, which could be go to airport, fly to city B. So, each of these alternatives has some travel cost and time associated with it. The best plan needs to be taken into account.

When the problem is divided into, a set of sub-problems, each of these sub-problems can be solved separately, and then, the combinations of the solutions generate the final solution. When the problem is divided (say decomposed), an AND arc is generated in the graph. This graph essentially indicates several sub-problems and solutions.

So, what the graph is all about and how to represent it? Here a simple state-space search is presented as a tree structure, where each successor node represents an alternative action to be taken. This type of graph is called OR graph. In case of problem reduction process, the problem space is represented as AND/OR graph of states. As discussed earlier, when we decompose the problem into sub-problems, each of the sub-problems is solved separately. Decomposing the problem generates an AND arc in the graph. Each of the AND arc can point to any number of successor nodes. These nodes should be solved so that the arcs may lead to more arcs. This indicates that there are more sub-problems and solutions.

In case of AND/OR graphs, the successors of AND nodes must be achieved and for OR nodes, one of the successors must be achieved. Figure 2.12 shows AND/OR graph for travel plan from city A to city B. Assume C to be intermediate city to reach B. The problem is divided into two sub-problems—travel by train and travel by air. The AND arc shows the states in each sub-problem. The child nodes or the leaf nodes can be further expanded. The travel cost at each node can be computed and shown in the graph. Thus, the best plan can be obtained with the graph traversal.

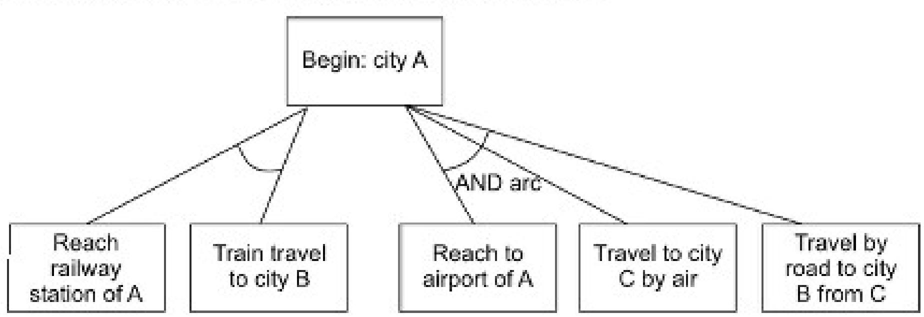


Figure 2.12 AND/OR graph.

Most often, the problems can be formulated as state space or problem reduction. State-space search is preferred when the solution can be expressed as the final state or as a path from the initial state to the final one. The set of actions can be defined so as to transform the states, whereas problem reduction will be useful when a problem can be decomposed into independent sub-problems. So, in short, the selection of approach is dependent on the type of the problem.

SUMMARY

Problem solving is an important topic in AI. We face problems at every point of time and look upon solutions for them. But the thing which is important here is how to solve a problem and what actually the problem is. Problem solving deals with the formulation of problem in terms of state and operators so that the goal is achieved. There are various ways to solve the problem, but a solution is acceptable only when it matches the required cost and time constraints or rather when it is cost-effective along with the quality. Problem solving is not just reaching the goal state, but it is rather reaching the solution optimally and following the constraints and rules defined in the problem.

While solving problems, heuristic and other advanced search techniques can be used to reach the goal. The use of AND/OR graphs helps in the evaluation of options to solve the complex problems by reduction method. Appropriate mapping and state representations help in finding out the solutions, with understanding of the problems and consideration of different paths.

KEYWORDS

1. **Problem solving:** It is the process of generating solutions for given complex and difficult situations or problems.
2. **Simple problems:** These are the problems that can be solved by deterministic procedure, where the solution is guaranteed.
3. **Complex problems:** The problems that lack information and have partial data are complex problems. The scenario is also not fully available here.
4. **General-purpose method of problem solving:** It is a means-ends analysis method, where a comparison of the current state with the goal state takes place to find out the difference and search is carried out to decide the states to reduce the difference.
5. **Special purpose method of problem solving:** In this, a particular problem is modelled with various assumptions, which are specific to that problem.
6. **Deterministic or observable problems:** Single-state problems where each state is fully observable and it goes to one definite state under operator. The goal state is reachable in one single action or sequence of actions.
7. **Non-observable problems:** These are multiple state problems. The problem solving agent does not have any information about the state. So, each state can be transformed to a number of states after the application of operator.

8. **Non-deterministic or partially observable problems:** In every new state, some new information is added and then operator acts on the state. Some information is available here.
9. **Unknown state-space problems:** These are most often known as exploration problem. States and actions are not known here.
10. **State space:** A problem is defined in terms of state. A state represents the status of solution at a given step of solution procedure. A solution to any problem is the collection of such different states and set of operations. This collection of states is termed as state space.
11. **Informed search strategy:** It is a search that does not guarantee a solution, but there is high probability of getting a solution. It makes use of heuristic.
12. **Uninformed search strategy:** It is a strategy that generates all possible states in the state space and checks for the goal state. It always finds a solution if it exists.
13. **Problem reduction:** It is a strategic approach to solve a problem so as to reduce the complexity, where the problem is decomposed into sub-parts.

MULTIPLE CHOICE QUESTIONS

1. There are four letters a, b, c, d available for registration number to be placed on number plate of any car. Number plate can contain only 3 letters and the letters can be used in any order. How many different numbers can be generated?
(a) 10 (b) 12 (c) 20 (d) 24
2. Ten teams are playing in a cricket tournament. The winning team of each match progresses to the next round and the loosing team is out of the game. In the 2nd round, each team plays with the other teams, after which a final round is played. How many matches will be played?
(a) 12 (b) 15 (c) 11 (d) 16
3. In problem reduction, state space is given by
(a) AND graph (b) OR graph (c) AND/OR graph (d) Tree diagram
4. A problem is reduced to 5 sub-problems. How many AND arc will be there in the graph?
(a) 5 (b) 10 (c) 2 (d) Cannot say
5. A problem is always represented as
(a) State search (b) State space (c) Search space (d) Problem space
6. A fully observable problem belongs to the category of
(a) Multi-state problem (b) Two-state problem
(c) Single-state problem (d) Cannot say
7. Which of the following can be used in problem solving as the structure of state space?
(i) Graphs (ii) Trees (iii) Queues
(a) (i) only (b) (i) and (ii) (c) (ii) only (d) (ii) and (iii)

8. The solution quality in problem solving is measured on the basis of (most appropriate)
   (a) Number of states (b) Time factor
   (c) Optimality (d) Space factor
9. Consider the word *Dealing*. In how many ways can the letters be arranged such that the vowels always remain together?
   (a) 120 (b) 720 (c) 420 (d) 620
10. For a traffic alerting problem-solving agent, the type of problem category it is likely to fall is
    (a) Fully observable (b) Partially observable
    (c) Non-observable (d) Unknown state space

CONCEPT REVIEW QUESTIONS

1. Describe the problem formulation steps with example.
2. How many problem types exist? Explain each type with an example.
3. Describe how search techniques are useful for finding solution to a problem.

CRITICAL THINKING EXERCISE

1. Consider a TV channel that is to telecast six shows, viz., 'Friends', 'KBC', 'Indian Idol', 'Dancing Star', 'Big Boss', 'Sailaab'. The following constraints are to be satisfied:
   (a) 'Friends' must be shown before 'KBC' and 'Indian Idol'.
   (b) 'KBC' should be telecasted before 'Dancing Star'.
   (c) 'Big Boss' must be telecasted after 'Indian Idol' and 'Sailaab'.

   Solve the problem to telecast the shows. Discuss and argue on the state representations.
2. Suppose there are four candidates contesting for party ticket for assembly elections. The party wants to select the best candidate. Formulate a problem for this situation with possible approach to find out the solution. (Assume data as required.)

PROJECT WORK

1. Write a program to implement the first assignment of critical thinking exercise.
2. Write a program to represent the same assignment using AND/OR graph.