MODULE - I

Problem Characteristics
Production system Characteristics

20MCA188 ARTIFICIAL INTELLIGENCE (Elective-2)

Selma Joseph A.

Characteristics

The following are the important characteristics of a problem.

1. Problem is decomposable to smaller or easier problems.
2. Solution steps can be ignored or undone.
3. The problem universe is predictable.
4. There are obvious good solutions without comparisons to all other possible solutions.
5. Desired solution is a state of the universe or a path to a state.
6. Requires lots of knowledge; or, uses knowledge to constrain solutions.
7. Problem requires periodic interaction between humans and computer.

Examples illustrating problem characteristics

1. Problem is decomposable to smaller or easier problems

Example 1

Consider the problem: Evaluate the integral

$$\int (x^2 + \sin^2 x)\, dx.$$

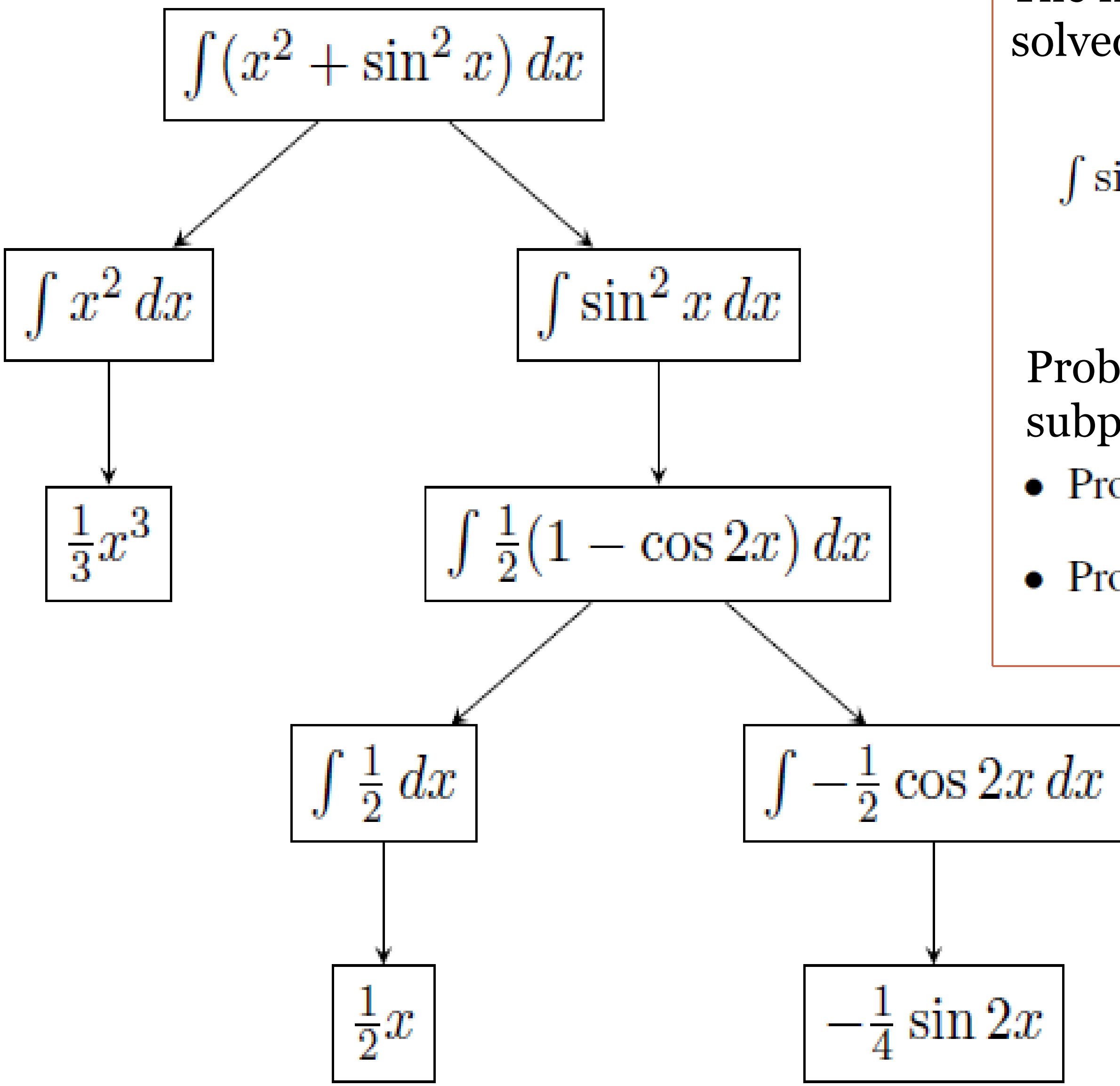
We show that this problem is decomposable to smaller subproblems Since

$$\int (x^2 + \sin^2 x)\, dx = \int x^2\, dx + \int \sin^2 x\, dx,$$

the problem can be decomposed into two simpler subproblems.

- Problem (i): Evaluate $\int x^2\, dx$.
- Problem (ii): Evaluate $\int \sin^2 x\, dx$.

Decomposing a problem into smaller subproblems



The first subproblem (i) can be easily solved to get $\int x^2\, dx = \frac{1}{3}x^3$.

$$\begin{aligned}\int \sin^2 x\, dx &= \int \frac{1}{2}(1 - \cos 2x)\, dx \\ &= \int \frac{1}{2}\, dx - \int \cos 2x\, dx,\end{aligned}$$

Problem (ii) can be divided into two subproblems:

- Problem (iii): Evaluate $\int \frac{1}{2}\, dx$.
- Problem (iv): Evaluate $\int -\frac{1}{2}\cos 2x\, dx$.

Can be easily solved as

$$\int \frac{1}{2}\, dx = \frac{1}{2}x$$

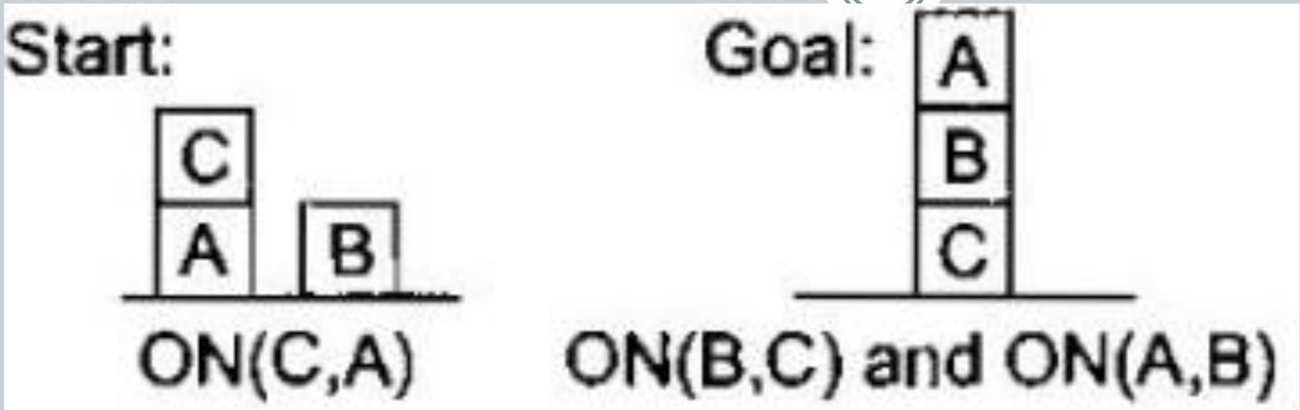
$$\int \frac{1}{2}\cos 2x\, dx = -\frac{1}{2}\sin 2x$$

Combining the solutions of the subproblems we get a solution of the given problem.

Example 2

Consider a simple blocks world problem

A simple blocks world problem



The following are the solution steps :

UNSTACK(C,A)
PUTDOWN(C)
PICKUP(B)
STACK(B,C)
PICKUP(A)
STACK(A,B)

Since, the steps are interdependent the problem cannot be decomposed into two independent problems.

2. Solution steps can be ignored or undone

Example 1

Suppose our goal is to prove a theorem in mathematics. On the way, we prove some preliminary
results, say Result 1, Result 2, Result 3, and then finally we prove the theorem.
The steps in the proof look like this:

Result 1
Result 2
Result 3
Theorem

Suppose, we later realize that Result 2 is not actually needed in proving the theorem.
Then, we may ignore Result 2 and present the steps of the proof as follows:

Result 1
Result 3
Theorem

In this example, the solution steps can be ignored.

Example 2

Consider the 8-puzzle. The solution involves a sequence of moves. In the process of finding a solution, after some moves, we realize that a certain previous move has be reversed. The previous move can be undone by backtracking the moves.
In this example, the solution steps can be ignored.

3. The problem universe is predictable

Example 1

In the 8-puzzle, every time we make a move we know exactly what will happen. This means that it is possible to plan an entire sequence of moves. Thus in this problem, the universe is predictable.

Example 2

In a game like bridge, this is not possible because a player does not know where the various cards are and what the other players will do on their turns. In this problem, the universe is unpredictable.

4. There are obvious good solutions without comparison to all other possible solutions

Example 1

Consider a mathematical problem. In general, there may be many methods for solving the problem. Any method is a good method without comparison to other methods provided it solves the problem. In general, any "any-path problem" is an example of a problem having obvious good solutions without comparison to other possible solutions.

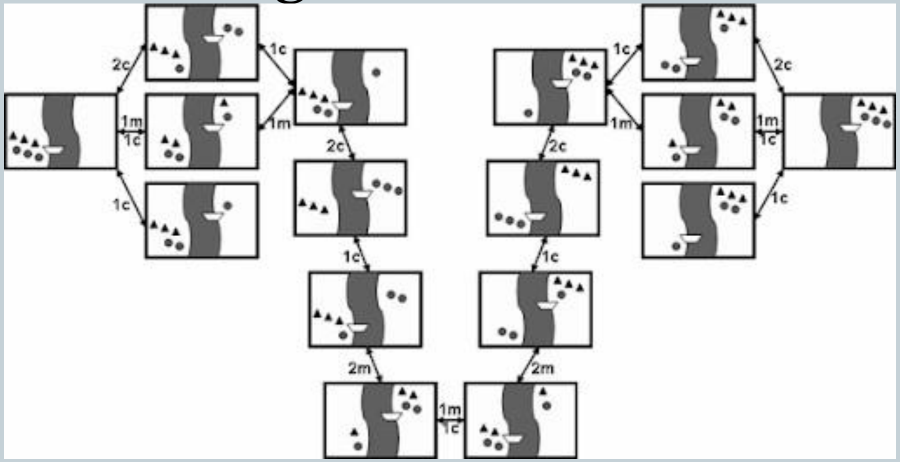
Example 2

A "best-path problem" is a problem having no obvious good solutions. The travelling salesman problem is an example for a best-path problem. The travelling salesman problem can be formulated as follows: "Given a list of cities and the distances between each pair of cities, what is the shortest possible route that visits each city exactly once and returns to the origin city?"

5. Desired solution is a state of the universe or a path to a state

Example 1

In the missionaries and cannibals problem, if we organise the various states in the form of a tree, it can be seen that the solution to the problem is a path connecting the various states



Example 2

In the cryptarithmetic puzzle, the solution to the problem is a state of the problem, namely, the state representing the assignment of digits to the letters in the puzzle.

6. Requires lots of knowledge; or, uses knowledge to constrain solutions

Example 1

Consider the problem of playing chess. The amount of knowledge required to play chess is very little: just the rules of the game! Additional knowledge about strategies may be used to make intelligent moves!!

Example 2

Consider the problem of scanning daily newspapers to decide which are supporting and which are opposing a political party in an upcoming election. It is obvious that a great deal of knowledge is required to solve this problem.

7. Problem requires periodic interaction between human and computer

Example

Even in a so called "fully automated system" situations constantly arise that call for human intervention. When the machines get thrown off track, or become faulty, experts have to be summoned to step in and troubleshoot the problems.

Production systems

- The term “production system” refers to many things. It may refer to a computer program which is used to provide a solution for a problem using a set of rules. It may also refer to a programming language for writing such programs. Further it can also be thought of as a model of computation that can be applied to implement search algorithms or as a model of human problem solving.
- Here, we define a production system as a structure having certain well defined components

Production systems…

Definition

A production system (also called production rule system) consists of the following:

1. Set of rules

This is the most important component of a production system. Briefly, a production rule is a rule of the form "C ! A" which may be interpreted as "given condition C, take action A". Thus, each rule in the set of rules consists of a left side (a pattern) that determines the applicability of the rule and a right side that describes the operation to be performed if the rule is applied.

2. Knowledge databases

The databases contain whatever information is appropriate for the particular task. Some parts of the database may be permanent.

The database includes a working memory whose elements may pertain only to the solution of the current problem. Working memory contains a description of the current state of the world in the problem-solving process. The description is matched against the conditions of the production rules. When a condition matches, its action is performed. Actions are designed to alter the contents of working memory.

Production systems…

3. Control strategy

The control strategy includes a specification of the order in which the rules are to be applied and a conflict resolution strategy.

(a) Order of application of rules

There are two methods of execution for a rule system:

-> Forward chaining

In forward chaining systems, the execution of rules starts from facts in databases and moves towards some goal or conclusion.

->Backward chaining

In backward chaining, the execution of rules starts from the goals and conclusions and works backwards to see if there is any data from which the goals and conclusions can be derived.

(b) Conflict resolution strategy

A system with a large number of rules and facts may result in many rules being true for the same fact; these rules are said to be in conflict. The conflict resolution strategy is a way of resolving the conflicts that manages the execution order of these conflicting rules. Some of the commonly used conflict resolution strategies are listed below:

- Choose arbitrarily.
- Choose the lowest numbered rule.
- Choose the rule containing the largest number of conditions.
- Choose the least recently used rule.
- Choose a rule where the condition is a new fact.
- Choose the rule with the highest priority (weight).

Production systems…

4. Interpreter

The interpreter repeats the following operations: all rules whose conditions are satisfied are found (rule matching), one of them is selected (conflict resolution), and its action is called (rule firing).
The interpreter (also referred to as rule applier) can be viewed as a “select-execute loop”, in which one rule applicable to the current state of the data base is chosen, and then executed. Its action results in a modified data base, and the select phase begins again. This cycle is also referred to as a “recognize-act cycle”.

Production system example: The water jug problem

Problem statement

We have two jugs of capacity 4 liters and 3 liters, and a tap with an endless supply of water. The objective is to obtain 2 liters of water exactly in the 4-liter jug with the minimum steps possible.

Solution

Step 1. Problem formulation as a production system

Represent a state of the problem by an ordered pair (x, y) of integers where x denotes the number of liters of water in the 4-liter jug and y the number of liters of water in the 3-liter jug.

1. Knowledge database

Since there is no information other than that contained in the statement of the problem, the database contains only the working memory. The working memory contains the following two facts:

{ (0,0) , (2, 0)} where (0 , 0) is the initial state and (2 , 0) is the goal state.

Production system example: The water jug problem..

2. Production rules

The set of production rules is given in Table :

Sl No.	State before action		State after action	Description of operation
1	(x, y) if $x < 4$	$\rightarrow$	$(4, y)$	Fill 4-liter jug
2	(x, y) if $y < 3$	$\rightarrow$	$(x, 3)$	Fill 3-liter jug
3	(x, y) if $x > 0$	$\rightarrow$	$(0, y)$	Empty 4-liter jug on the ground
4	(x, y) if $y > 0$	$\rightarrow$	$(x, 0)$	Empty 3-liter jug on the ground
5	(x, y) if $x + y \geq 4$ and $y > 0$	$\rightarrow$	$(4, y - (4 - x))$	Pour water from 3-liter jug into 4-liter jug until 4-liter jug is full
6	(x, y) if $x + y \geq 3$ and $x > 0$	$\rightarrow$	$(x - (3 - y), 3)$	Pour water from 4-liter jug into 3-liter jug until 3-liter jug is full
7	(x, y) if $x + y \leq 4$ and $y > 0$	$\rightarrow$	$(x + y, 0)$	Pour all water from 3-liter jug into 4-liter jug
8	(x, y) if $x + y \leq 3$ and $x > 0$	$\rightarrow$	$(0, x + y)$	Pour all water from 4-liter jug into 3-liter jug

3. Conflict resolution strategy

Select and fire the first rule in the conflict set that does not lead to a repeated state.

4. Interpreter

Interpreter is the select-execute loop.

Step 2. Application of the rules

The various steps in the solution procedure obtained by using the forward chaining method is shown in Table

Step No.	Working memory	Current state	Conflict set	Fire rule	Remarks
0	$\{(0,0),(2,0)\}$	$(0,0)$	–	–	Initial state
1	$\{(0,0),(2,0)\}$	$(0,0)$	1, 2	1	–
2	$\{(4,0),(2,0)\}$	$(4,0)$	2, 3, , 6	2	–
3	$\{(4,3),(2,0)\}$	$(4,3)$	3, 4, 5, 6	3	–
4	$\{(0,3),(2,0)\}$	$(0,3)$	1, 4, 7	7	–
5	$\{(3,0),(2,0)\}$	$(3,0)$	1, 2, 3, , 6, 8	2	–
6	$\{(3,3),(2,0)\}$	$(3,3)$	1, 3, 4, 5, 6	5	–
7	$\{(4,2),(2,0)\}$	$(4,2)$	2, 3, 4, 5, 6	3	–
8	$\{(0,2),(2,0)\}$	$(0,2)$	1, 2, 4, 7	7	–
9	$\{(2,0),(2,0)\}$	$(2,0)$	–	–	Goal state

Sequence of the applications of the production rules to reach the goal state in the water jug problem

Let us examine the working of the conflict resolution strategy. Consider Step No. 6 in Table. The current state is (3 , 3). If we examine the rules, it can be seen that this state satisfies the conditions of the Rules 1, 3, 4, 5, and 6 and hence the conflict set is {Rule 1, Rule 3, Rule 4, Rule 5, Rule 6}. The conflict resolution strategy we have adopted is "select and fire the first rule that does not lead to a repeated state".

If we select Rule 1 and fire it, the resulting state is (4 , 3) and we have obtained this state in Step No. 3.

Similarly if we select Rule 3 and fire it the resulting state is (0,3) which we have already obtained in Step No. 4.

Proceeding like this, we see that that the first rule in the conflict set that does not lead to a repeated state is Rule 5 and we select it and fire it to obtain the non-repeated state (4, 2).

Step No.	Working memory	Current state	Conflict set	Fire rule	Remarks
0	{(0, 0), (2, 0)}	(0, 0)	–	–	Initial state
1	{(0, 0), (2, 0)}	(0, 0)	1, 2	1	–
2	{(4, 0), (2, 0)}	(4, 0)	2, 3, 6	2	–
3	{(4, 3), (2, 0)}	(4, 3)	3, 4, 5, 6	3	–
4	{(0, 3), (2, 0)}	(0, 3)	1, 4, 7	7	–
5	{(3, 0), (2, 0)}	(3, 0)	1, 2, 3, 6, 8	2	–
6	{(3, 3), (2, 0)}	(3, 3)	1, 3, 4, 5, 6	5	–
7	{(4, 2), (2, 0)}	(4, 2)	2, 3, 4, 5, 6	3	–
8	{(0, 2), (2, 0)}	(0, 2)	1, 2, 4, 7	7	–
9	{(2, 0), (2, 0)}	(2, 0)	–	–	Goal state

Production system characteristics
(Classes of production systems)

Definitions

1. Monotonic and nonmonotonic production systems

A monotonic production system is a production system in which the application of a rule never prevents the later application of another rule that could also have been applied at the time the first rule was selected. A production system which is not monotonic is called a nonmonotonic production system.

2. Partially commutative production systems

A partially commutative production systems is a production system with the property that if the application of a particular sequence of rules transforms state x to state y then any allowable permutation of those rules also transforms state x into state y.

3. Commutative production systems

A production system that is both monotonic and partially commutative is called a commutative production system.

Production system characteristics
(Classes of production systems)

Remarks

1. It can be shown that, depending on how the operators are chosen, the 8-puzzle and the blocks world problem are partially commutative systems.
2. Production systems that are not partially commutative are useful in which irreversible changes occur. For example, the process to produce a desired chemical compound may involve may irreversible steps.
3. Commutative production systems are useful for solving ignorable problems like the problem of proving a theorem in mathematics.
4. Nonmonotonic partially commutative production systems are useful for problems in which the order of operations is not important. For example, robot navigation is such a problem.

summarises the characteristics of some of the well known production systems.

	Monotonic	Nonmonotonic
Partially commutative	Theorem proving	Robot navigation
Not partially commutative	Chemical synthesis	Bridge

The four categories of production systems

Advantages and disadvantages of production systems

Advantages

1. Provides excellent tools for structuring AI programs.
2. The system is highly modular because individual rules can be added, removed or modified independently.
3. There is separation of knowledge and control.
4. The system uses pattern directed control which is more flexible than algorithmic control.
5. Provides opportunities for heuristic control of the search.
6. Quite helpful in a real-time environment and applications.

Disadvantages

1. There is a lot of inefficiency in production systems. For example, there may be situations where multiple rules get activated during execution and each of these rules may trigger exhaustive searches.
2. It is very difficult to analyze the flow of control within a production system.
3. There is an absence of learning due to a rule-based production system that does not store the result of the problem for future use.
4. The rules in the production system should not have any type of conflict operations. When a new rule is added to the database it should ensure that it does not have any conflict with any existing rule.