Problem Characteristics of AI

Lecture Four

Dr. Luma Alsaffar

Problem Characteristics

1. Is problem decomposable into set of(nearly) independent smaller or easier sub problems?
2. Can solution steps be ignored or at least undone if they prove unwise?
3. Is the problem's universe predictable?
4. Is a good solution to the problem obvious without comparison to all other possible solutions?
5. Is a desire solution a state of the world or a path to a state?
6. Is a large amount of knowledge absolute required to solve the problem, or is knowledge important only to certain the search?
7. Can a computer that is simply given the problem return the solution, or will the solution of problem require interaction between the computer and a person?

فيما يلي الخصائص المهمة للمشكلة

- يمكن تفكيك المشكلة إلى مشاكل أصغر أو أسهل
- يمكن تجاهل خطوات الحل أو التراجع عنها
- يمكن التنبؤ بعالم المشكلة
- توجد حلول جيدة واضحة دون مقارنات مع جميع الحلول الأخرى الممكنة.
- الحل المطلوب هو حالة من حالات الكون أو مسار إلى حالة.
- يتطلب الكثير من المعرفة؛ أو يستخدم المعرفة لتقييدالحلول.
- تتطلب المشكلة تفاعلًا دوريًا بين البشر والحاسوب.

Seven problem characteristics

1. Decomposable Problem
   - Block world problem
2. Can solution steps be ignored or undone?
   - Ignorable : theorem proving
     - solution steps can be ignored
   - Recoverable : 8 puzzle
     - solution steps can be undone (backtracking)
   - Irrecoverable : chess
     - solution steps can not be undone

- Can solution steps be ignored or at least undone if they prove to be unwise?
  - In real life, there are three types of problems:
    - Ignorable (Theorem proving)
    - Recoverable (Backtracking)
    - Irrecoverable. (Chess)
- Is the knowledge Base consistent?
  - **Target problem:** A man is standing 150 ft from a target. He plans to hit the target by shooting a gun that fires bullets with velocity of 1500 ft/sec. How high above the target should he aim?
  - **Note:** 9.81 meter/sec = g(gravity)

    9810 Millimeter(mm) = 9.81 * 1000

    1 feet = 300 mm

    9810/300 = 32 feet = 1 g

    Time=Distance / Velocity

Solution:

- Velocity of bullet is 1500 ft./sec and distance is 150 ft, so bullet takes 0.1 sec to reach (t=d/v) the target.

- Assume bullet travels in a straight line.

- Due to gravity, the bullet falls at a distance: $(1/2)\,gt^2 = (1/2)(32)(0.1)^2 = 0.16$ft.

- So if man aims up 0.16 feet high from the target, then bullet will hit the target.

- Now there is a contradiction to the fact that bullet travel in a straight line because the bullet in actual will travel in an arc. Therefore there is inconsistency in the knowledge used.

Is the problem Decomposable?

By this method we can solve large problem easily.

Ex: Decomposable problem

Symbolic Integration

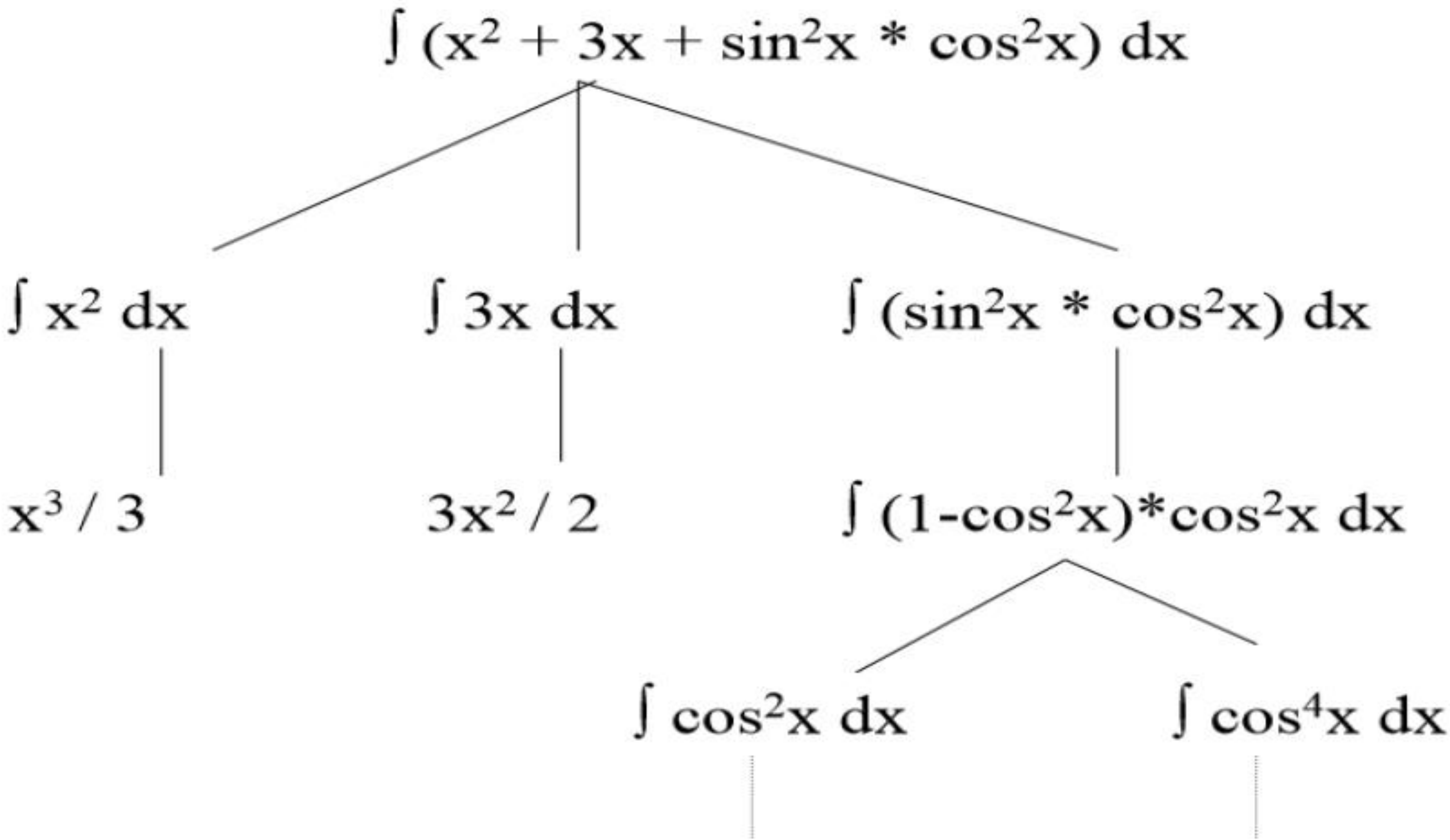
$$\int (x^2 + 3x + \text{Sin}^2x.\text{Cos}^2x)dx$$

Can be divided to

Integral of x^2

Integral of $3x$

Integral of $\text{Sin}^2x.\text{Cos}^2x$, which can be further divided to $(1 - \text{Cos}^2x).\ \text{Cos}^2x$ ....



∫ (x² + 3x + sin²x * cos²x) dx
∫ x² dx
∫ 3x dx
∫ (sin²x * cos²x) dx
x³ / 3
3x² / 2
∫ (1-cos²x)*cos²x dx
∫ cos²x dx
∫ cos⁴x dx

Examples illustrating problem characteristics

Problem is decomposable to smaller or easier problems

Example 1

Consider the problem: Evaluate the integral

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

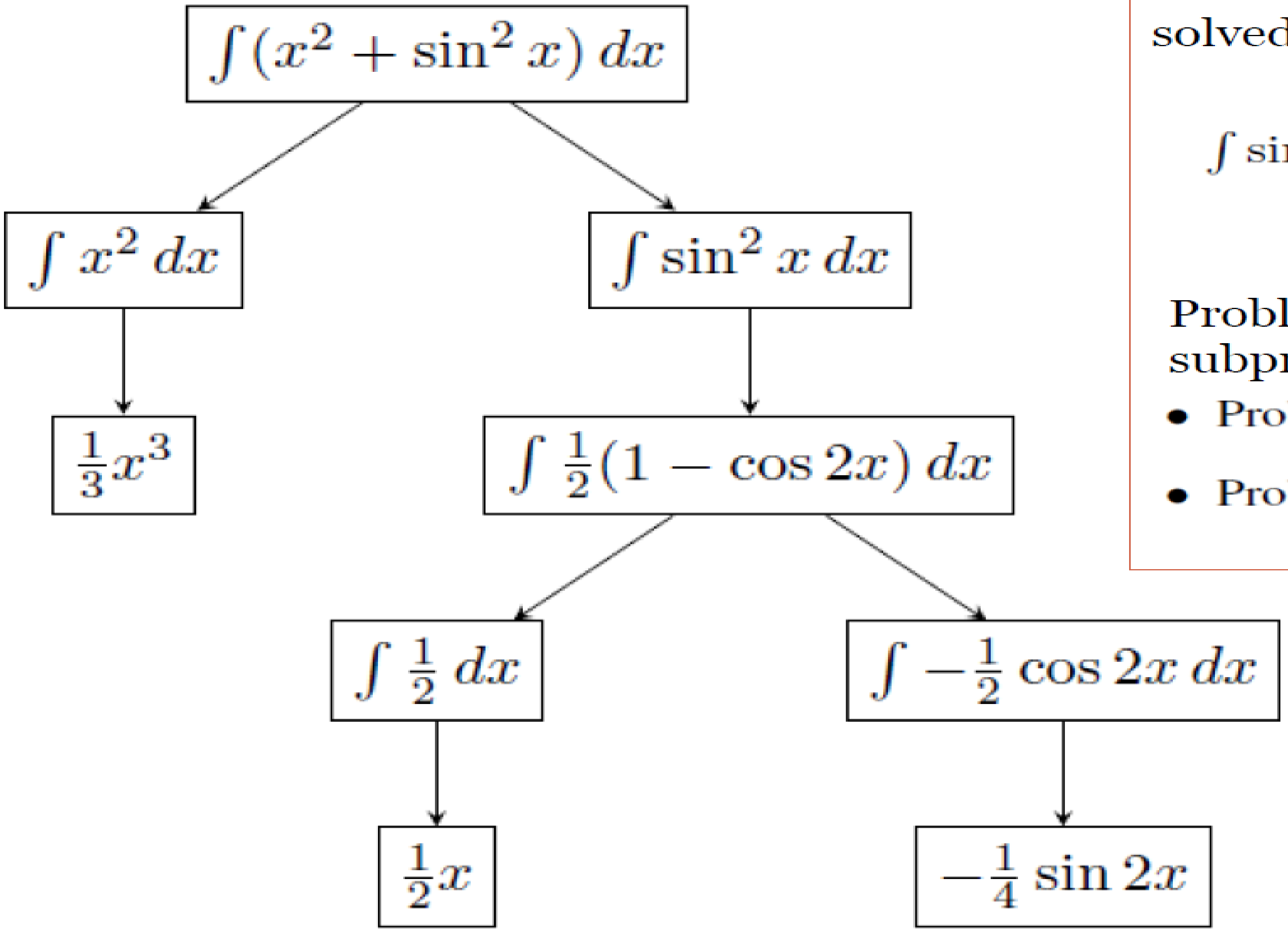
We show that this problem is decomposable to smaller subproblems Since

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

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 $\int x^2\, dx = \frac{1}{3}x^3$.

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

Problem (ii) can be divided into two subproblems:

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

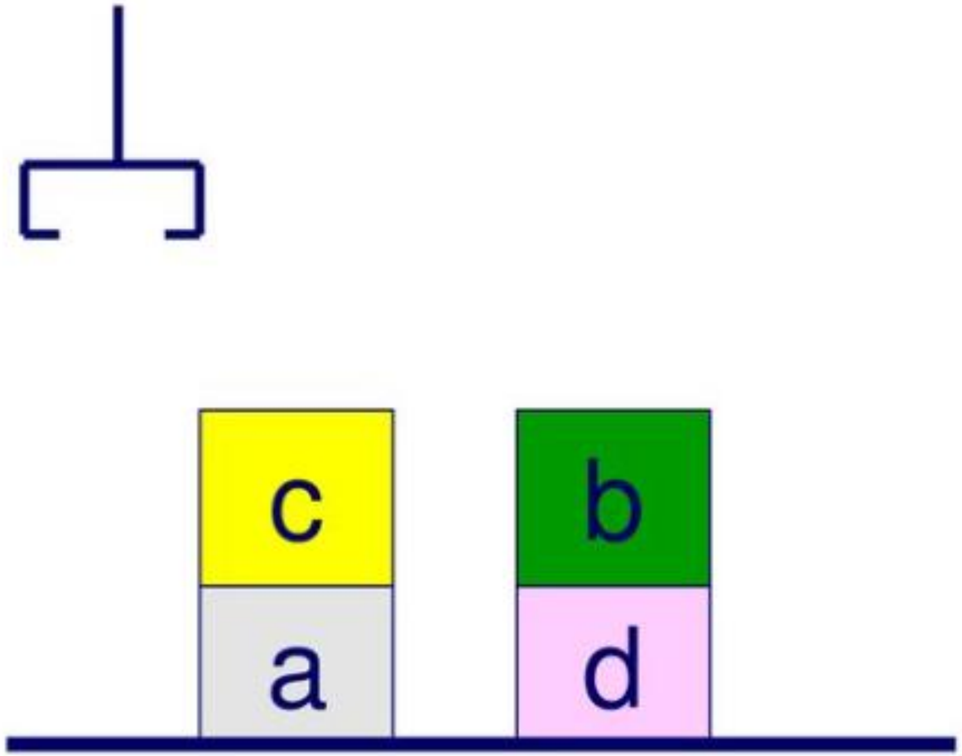
Can be easily solved as

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

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

A blocks world

- on(c,a).
- on(b,d).
- ontable(a).
- ontable(d).
- clear(b).
- clear(c).
- hand_empty.



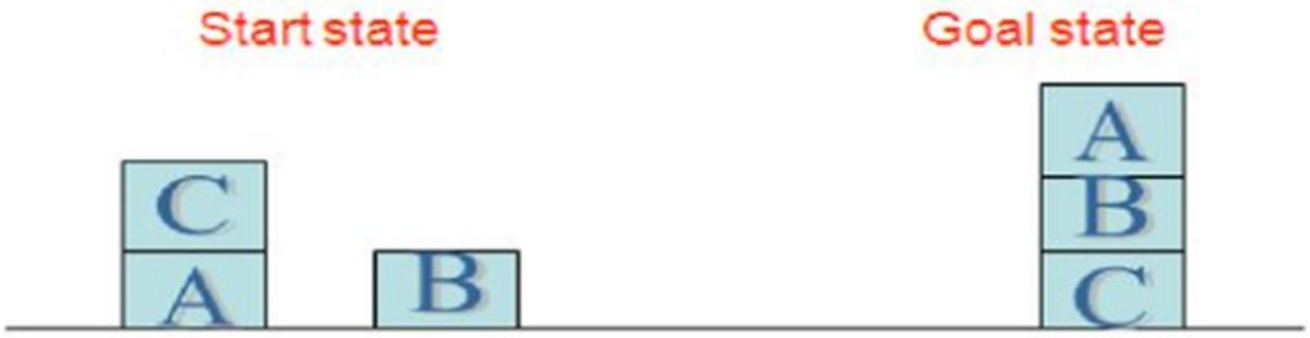
Is the problem Decomposable?

Ex: Non- decomposable problems

Block World Problem

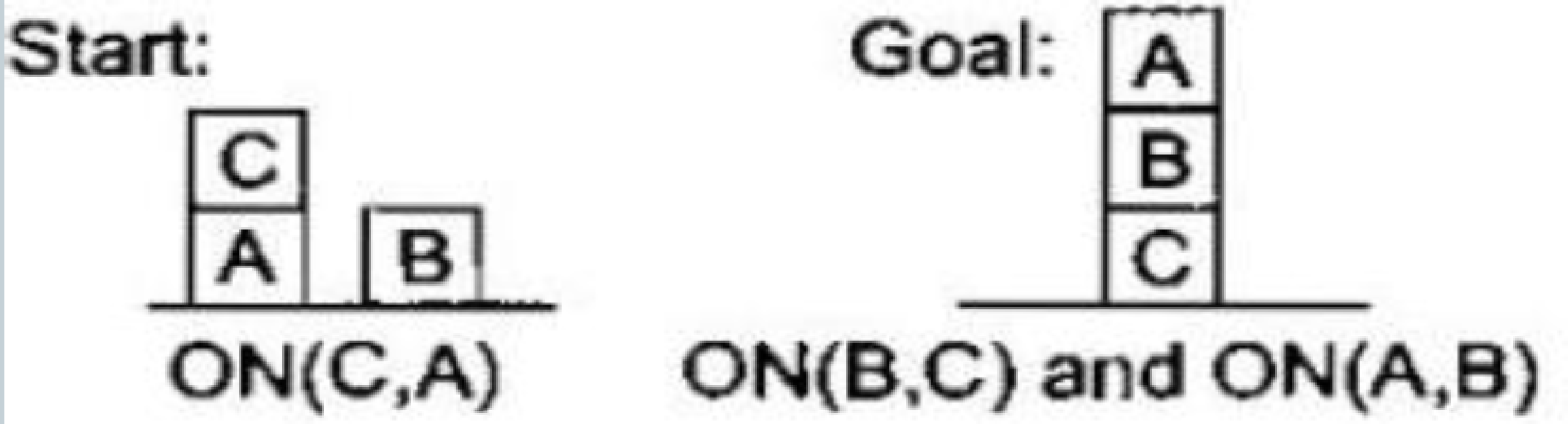
Assume that only two operations are available:

1. CLEAR(x)[Block x has nothing on it]->ON(x,Table)[Pick up x and put on the table]
2. Clear(x) and Clear(y)->ON(x,y)[Put x on y]



ON(B,C) and ON(A,B)

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.

Can Solution steps be ignored or undone?

- **Ignorable problem:** in which solution steps can be ignored.

Ex:- Theorem Proving

Suppose we are trying to prove a mathematical theorem. We proceed by first proving a lemma that we think will be useful. Eventually, we realize that the lemma is not help at all.

Every thing we need to know to prove theorem is still true and in memory, if it ever was. Any rule that could have been applied at the outset can still be applied. *All we have lost is the effort that was spent exploring the blind alley.*

Can Solution steps be ignored or undone?

- Recoverable problem: in which solution steps can be undone.

Ex:- The 8-Puzzle

The 8-puzzle is a square tray in which are placed, eight square tiles and remaining 9^{th} square is uncovered. Each tile has number on it. A tile that is adjacent to blank space can be slide in to that space. A game consist of a starting position and a specific goal position.

We might make stupid move.

We can backtrack and undo the first move. Mistakes can still be recovered from but not quite as easy as in theorem proving.

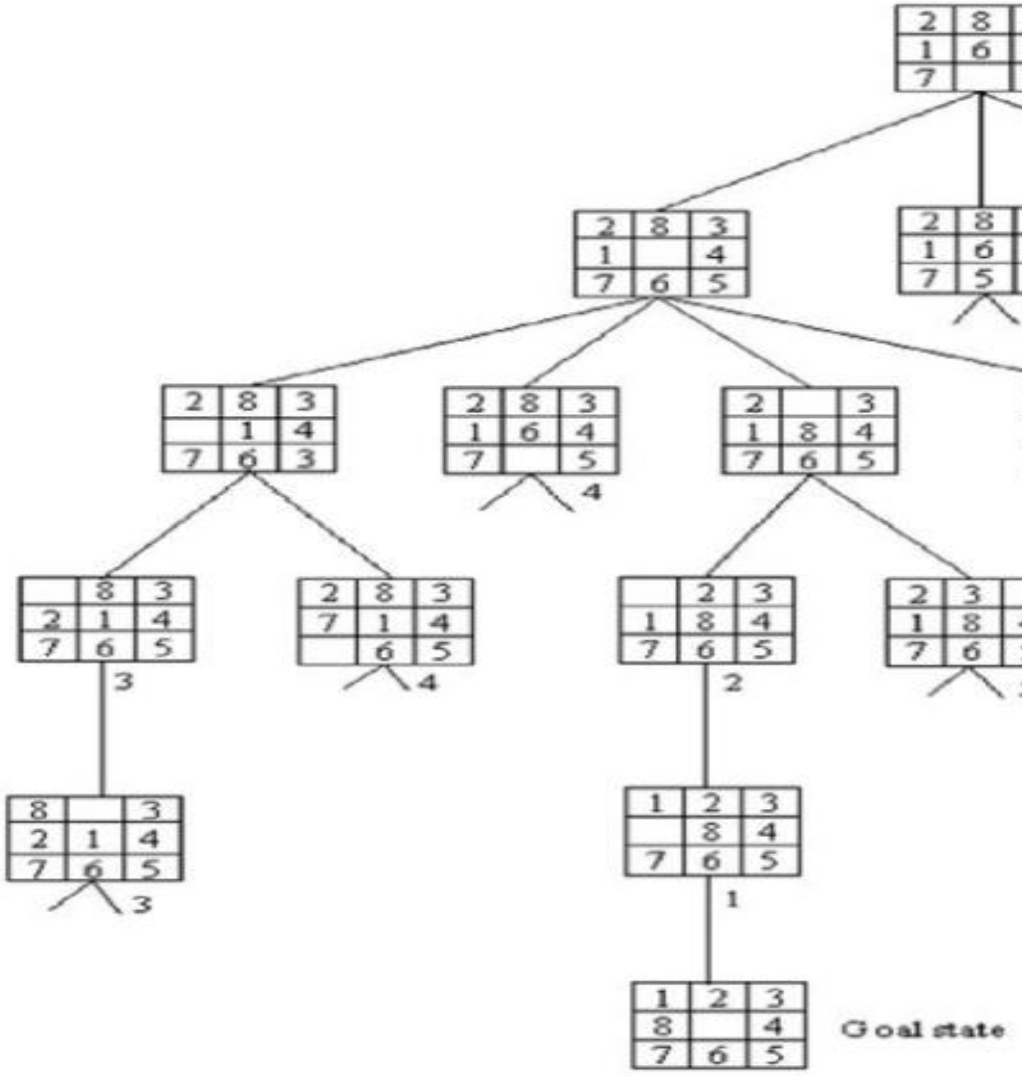
2	8	3
1	6	4
7		5

Initial State

1	2	3
8		4
7	6	5

Goal state

8-Puzzle



Can Solution steps be ignored or undone?

- Irrecoverable problem: in which solution steps cannot be undone.

Ex:- Chess

Suppose a chess playing program makes a stupid move and realize it a couple of move later. It cannot simply play as though it never made the stupid move. Nor can it simply backup and start the game over from that point. All it can do is to try to make best of the current situation and go on from there.

Can Solution steps be ignored or undone?

- **Ignorable problem** can be solved using a simple control structure that never backtracks. Such a control structure is easy to implement.
- **Recoverable problem** can be solved by slightly more complicated control strategy that does something mistakes and **backtracking** will be necessary to recover from such mistakes.
- **Irrecoverable** problems, solved by a system that expends a great deal of effort making each decision since each the decision must be final.
- **Some irrecoverable** problems can be solved by recoverable style methods used in a **planning process** , in which an entire sequence of steps is analyzed in advance to discover where it will lead before first step is actually taken.

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.

Seven problem characteristics

5. Is the solution a state or a path ?
   - consistent…… interpretation for the sentence
   - water jug problem ➔ path / plan
6. What is the role of knowledge?

   knowledge for perfect program of chess

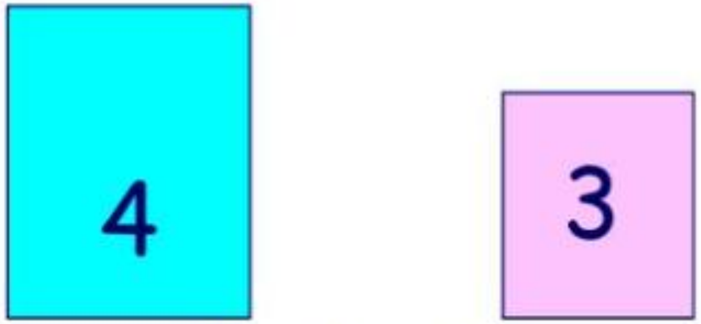
   (need knowledge to constrain the search)

   newspaper story understanding

   (need knowledge to recognize a solution)
7. Does the task require interaction with a person? solitary/ conversational

Defining the problem

- A water jug problem: 4-gallon and 3-gallon



  - no marker on the bottle
  - pump to fill the water into the jug
  - How can you get exactly 2 gallons of water into the 4-gallons jug?

A state space search

(x,y) : order pair

x : water in 4-gallons → x = 0,1,2,3,4

y : water in 3-gallons → y = 0,1,2,3

start state : (0,0)

goal state : (2,n) where n = any value

Rules :

1. Fill the 4 gallon-jug (4,-)
2. Fill the 3 gallon-jug (-,3)
3. Empty the 4 gallon-jug (0,-)
4. Empty the 3 gallon-jug (-,0)

A state space search

(x,y) : order pair

x : water in 4-gallons → x = 0,1,2,3,4

y : water in 3-gallons → y = 0,1,2,3

start state : (0,0)

goal state : (2,n) where n = any value

Rules :

1. Fill the 4 gallon-jug (4,-)
2. Fill the 3 gallon-jug (-,3)
3. Empty the 4 gallon-jug (0,-)
4. Empty the 3 gallon-jug (-,0)

Water jug rules

1	(x, y) if $x < 4$	$\rightarrow (4, y)$	Fill the 4-gallon jug
2	(x, y) if $y < 3$	$\rightarrow (x, 3)$	Fill the 3-gallon jug
3	(x, y) if $x > 0$	$\rightarrow (x - d, y)$	Pour some water out of the 4-gallon jug
4	(x, y) if $y > 0$	$\rightarrow (x, y - d)$	Pour some water out of the 3-gallon jug
5	(x, y) if $x > 0$	$\rightarrow (0, y)$	Empty the 4-gallon jug on the ground
6	(x, y) if $y > 0$	$\rightarrow (x, 0)$	Empty the 3-gallon jug on the ground
7	(x, y) if $x + y \geq 4$ and $y > 0$	$\rightarrow (4, y - (4 - x))$	Pour water from the 3-gallon jug into the 4-gallon jug until the 4-gallon jug is full

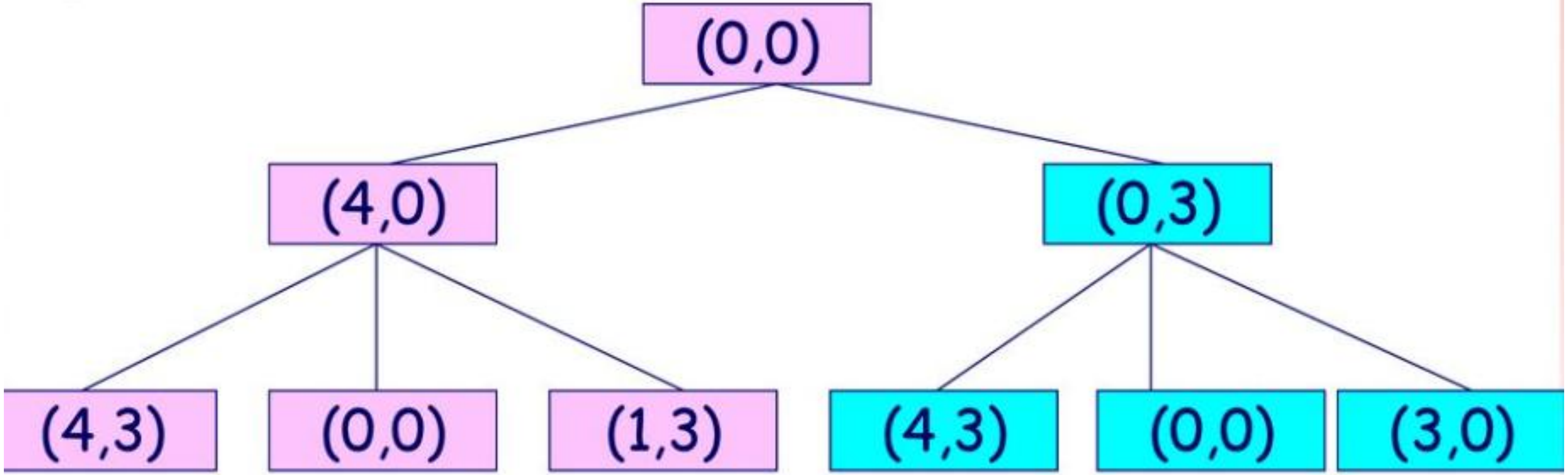
Water jug rules

8	(x, y) if $x + y \geq 3$ and $x > 0$	$\rightarrow (x - (3 - y), 3)$	Pour water from the 4-gallon jug into the 3-gallon jug until the 3-gallon jug is full
9	(x, y) if $x + y \leq 4$ and $y > 0$	$\rightarrow (x + y, 0)$	Pour all the water from the 3-gallon jug into the 4-gallon jug
10	(x, y) if $x + y \leq 3$ and $x > 0$	$\rightarrow (0, x + y)$	Pour all the water from the 4-gallon jug into the 3-gallon jug
11	$(0, 2)$	$\rightarrow (2, 0)$	Pour the 2 gallons from the 3-gallon jug into the 4-gallon jug
12	$(2, y)$	$\rightarrow (0, y)$	Empty the 2 gallons in the 4-gallon jug on the ground

Formal description of a problem

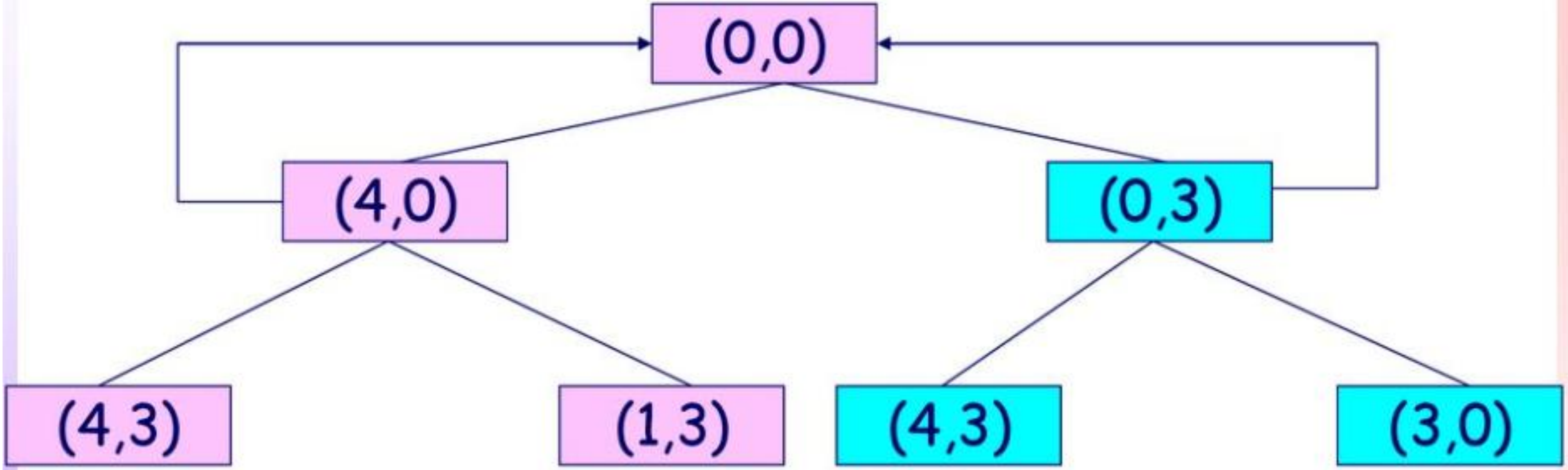
1. Define a state space that contains all the possible configurations of the relevant objects.
2. Specify state/states that describes the situation of start state.
3. Specify state/states that describes the situation of goal state.
4. Specify the set of rules.
    - assumption, generalization

Search Tree



- Water jug problem.

Search Graph



- Water jug problem.
  - Cycle
  - When will the search terminate?

Is universe predictable?

Certain-outcome problem

Ex: <u>8-Puzzle</u>

Every time we make a move, we know exactly what will happen. This is possible to **plan** entire sequence of moves and be confident that we know what the resulting state will be.

Uncertain-outcome problem

Ex: play Bridge

One of the decisions we will have to make is which card to play on the first trick. What we would like to do is to plan entire hand before making the 1st hand. But now it is not possible to do such planning with certainty since we cannot know exactly where all the cards are or what the other players will do on their turn.

Is a good solution Absolute or Relative ?

Any-path problem

Ex: Answer-question System

Consider the problem of answering the question based on following facts:

1. Marcus was a man.
2. Marcus was a Pompean.
3. Marcus was born in 40 AD.
4. All men are mortal.
5. All Pompeans died when volcano erupted in 79 AD.
6. No mortal lives longer than 150 years.
7. Now it is 1991 AD.

" Is Marcus alive?"

Question answering question

1. Marcus was a man.
2. Marcus was a Pompeian.
3. Marcus was born in 40 A.D.
4. All men are mortal.
5. All Pompeians died when the volcano erupted in 79 A.D.
6. No mortal lives longer than 150 years.
7. It is now 1991 A.D.

Is Marcus alive?

Solution 1

1. Marcus was man.	axiom 1
4. All men are mortal.	axiom 4
8. Marcus is mortal.	1,4
3. Marcus was born in 40 A.D.	axiom 3
7. It is now 1991 A.D.	axiom 7
9. Marcus' age is 1951 years.	3,7
6. No mortal lives longer than 150 years	axiom 6
10. Marcus is dead.	8,6,9

Solution 2

7. It is now 1991 A.D.	axiom 7
5. All Pompeians died in 79 A.D.	axiom 5
11. All Pompeians are dead now.	7,5
2. Marcus was a Pompeian.	axiom 2
12. Marcus is dead……	11,2

Is a good solution Absolute or Relative ?

1. Marcus was a man	- Axiom1
4. All men are mortal	-Axiom4
8. Marcus is Mortal	- 1&4
3. Marcus was born in 40 AD	-Axiom3
7. Now it is 1991 AD	-Axiom7
9. Marcus age is 1951 years	- 3&7
6. No mortal lives longer than 150 years	-Axiom6
10. **Marcus is dead**	-6,8,9

OR

7. It is now 1991AD	-axiom 7
5. All pompeians died in 79 AD	-axiom 5
11. All pompeians are died now	-7 & 5
2. Marcus was a pompeian	-axiom 2
12. **Marcus is dead**	-11,2

Since all we are interested in is the answer to question, it does not matter **which path we follow**.

If we do follow one path successfully to the answer, there is no reason to go back and see if some other path might also lead to a solution.

Understanding a sentence

- The bank president ate a dish of pasta salad with the fork.

- bank = financial institution / a side of a river
- dish = eat dish / eat pasta
- pasta salad : dog food → food with dog meat?
- with a fork : with ....her friend. / with vegetable.
- solution : state of the world

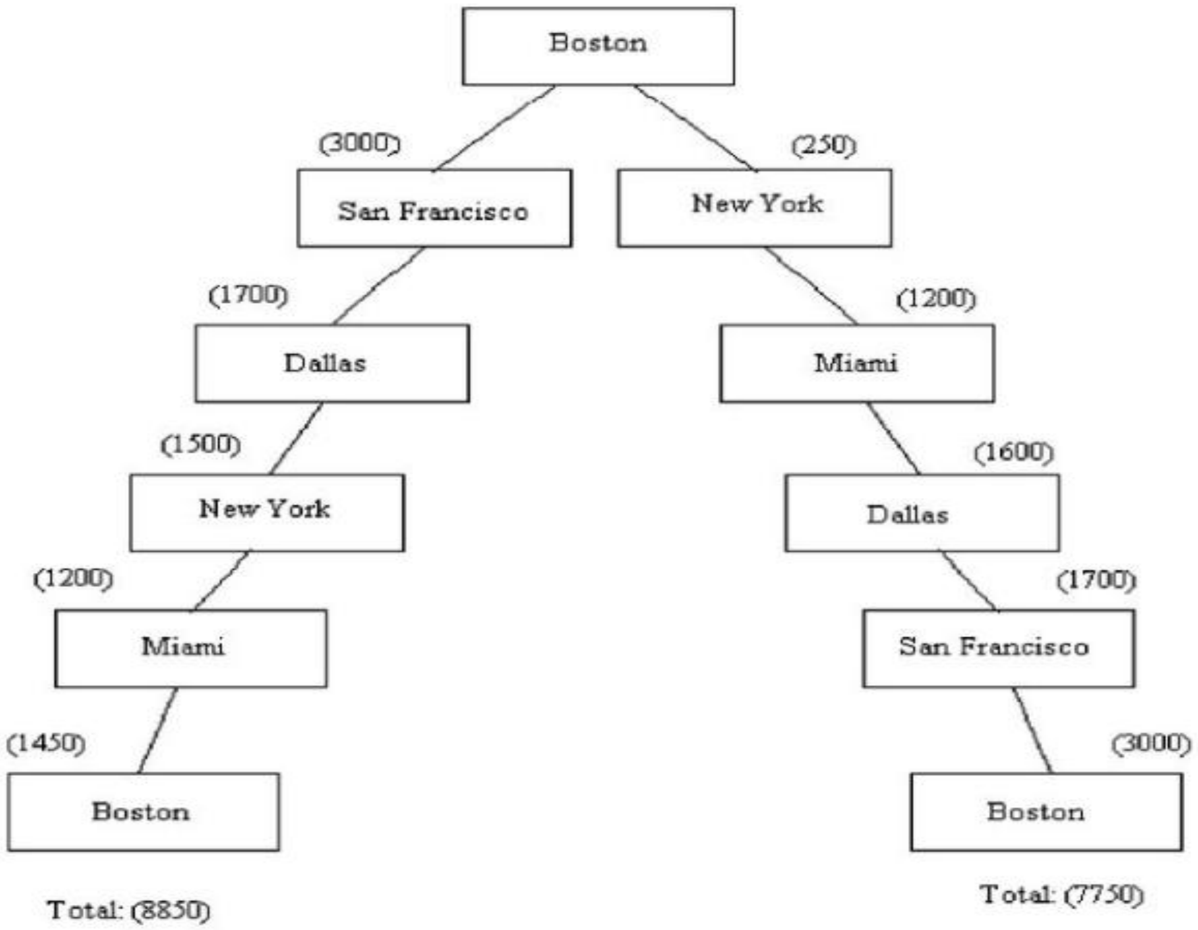
Is a good solution Absolute or Relative ?

Best-path problem

Ex: Traveling Salesman Problem

- Given a road map of n cities, find the **shortest** tour which visits every city on the map exactly once and then return to the original city (*Hamiltonian circuit*)

	Boston	New York	Miami	Dallas	S.F.
Boston		250	1450	1700	3000
New York	250		1200	1500	2900
Miami	1450	1200		1600	3300
Dallas	1700	1500	1600		1700
S.F.	3000	2900	3300	1700	


Boston
(3000)
San Francisco
(1700)
Dallas
(1500)
New York
(1200)
Miami
(1450)
Boston
Total: (8850)
(250)
New York
(1200)
Miami
(1600)
Dallas
(1700)
San Francisco
(3000)
Boston
Total: (7750)

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.

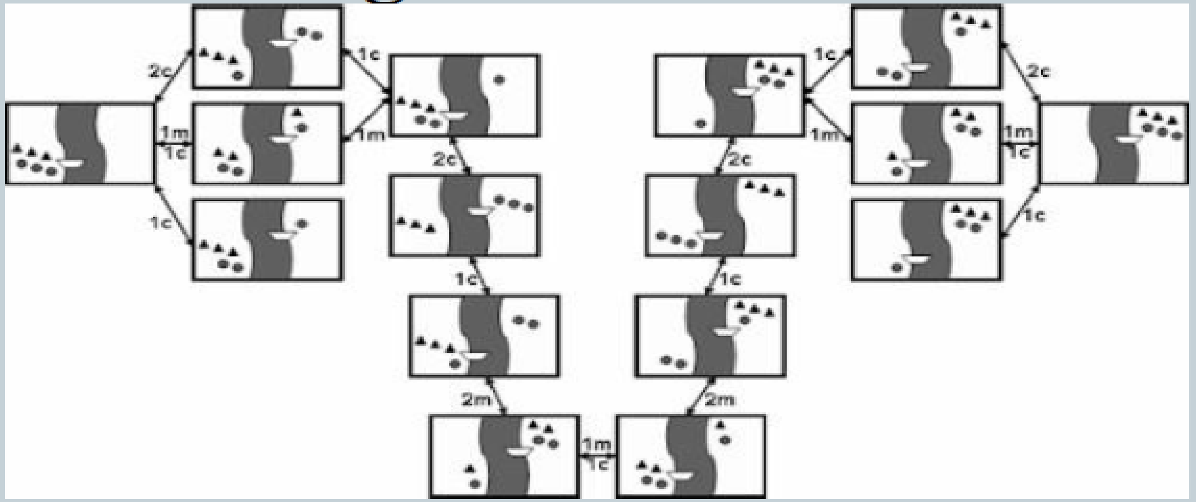
Example 2

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?"

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.

Is a good solution Absolute or Relative ?

- Best-path problems are, in general, computationally harder than any-path problems.

- Any-path problems can often be solved in a reasonable amount of time by using heuristics that suggest good paths to explore. If the heuristics are not perfect, the search for a solution may not be as direct as possible, but that does not matter.

- For true best-path problems, however, no heuristic that could possibly miss the best solution can be used. So a much more exhaustive search will be performed.

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.

Is the solution a State or Path ?

Solution is a path to state

<u>Ex: Water jug problem</u>

Here is not sufficient to report that we have solved the problem and the final state is (2,0).

Here we must report is <u>not the final state</u> but the <u>path that we found to that state.</u>

Thus a statement of solution to this problem must be a sequence of operations (some time called *apian*) that produce the final state.

Solution is a state of world

<u>Ex: Natural language understanding</u>

To solve the problem of finding the interpretation we need to produce interpretation itself. No record of processing by which the interpretation was found is necessary.

"The **<u>bank</u>** president ate a **<u>dish</u>** of pasta salad with the fork".

What is the role of knowledge?

Knowledge is important only to constrain the search for solution

Ex: playing chess

Suppose you have ultimate computing power available.

How much knowledge **would be required by a perfect program**?

just the rule for determining legal moves and some simple control mechanism that implement an appropriate search procedure.

Knowledge is required even to be able to recognize a solution

Ex: Scanning daily news paper to decide which are supporting the democrates and which are supporting the republicans in some upcoming elections.

you have ultimate computing power available.

How much knowledge **would be required by a perfect program?**

This time answer is great deal. It would have to know:

- The name of candidates in each party.
- For supporting republicans; you want to see done is have taxes lowered.
- For supporting democrats; you want to see done is improved education for minority students.
- And so on......

Does the task require interaction with person?

Solitary:

in which the computer is given a problem description and produces an answer with no intermediate communication and with no demand for an explanation of the reasoning process.

Level of interaction b/w computer and user is **problem-in solution-out.**

EX: Theorem Proving

Conversational:

in which there is intermediate communication between a person and the computer, either to prove additional assistance to computer or to prove additional information to user, or both.

Ex: Medical diagnosis

Relationship b/w problems and production systems

- For any solvable problem, there exist an infinite number of production systems that describe ways to find solution. Some will be more natural or efficient than other.
- Any problem that can be solved by any production system can be solved by a commutative one, but the commutative one may be so unwieldy as to be practically useless.
- So in formal sense, there is no relation ship b/w kind of problems and kind of production system since all problems can be solved by all kinds of system.
- But in practical sense, there definitely is such a relationships b/w kind of problems and kind of systems that lend themselves naturally to describing those problems.

Relationship b/w problems and production systems

Ignorable problems; where creating new things rather than changing old once

Change occur but can be reversed and in which order of operation is not critical

	Monotonic	Nonmonotonic
Partially Commutative	Theorem Proving	Robot Navigation, 8-puzzle
Not Partially Commutative	Chemical synthesis	Bridge, Chess

where creating new thins by changing old once

Reverse not possible and order matter.