TYPES OF SEARCH, SEARCH METHODOLOGIES, CLASSICAL SEARCH METHODOLOGIES

Fundamentals of Artificial Intelligence

Session 07b

Pramod Sharma
pramod.sharma@prasami.com

2

Recap…

pra-sâmi

3

What is AI?

- Views of AI fall into four different perspectives --- two dimensions:
  - Thinking versus Acting
  - Human versus Rational

	Human-like Intelligence	"Ideal" Intelligent/ Pure Rationality
Thought/ Reasoning ("modeling thought" "brain")	**2. Thinking humanly**	**3. Thinking Rationally**
Behavior/Actions "behaviorism" "mimics behavior"	**1. Acting Humanly**	**4. Acting Rationally**

Seems to be a Good Idea

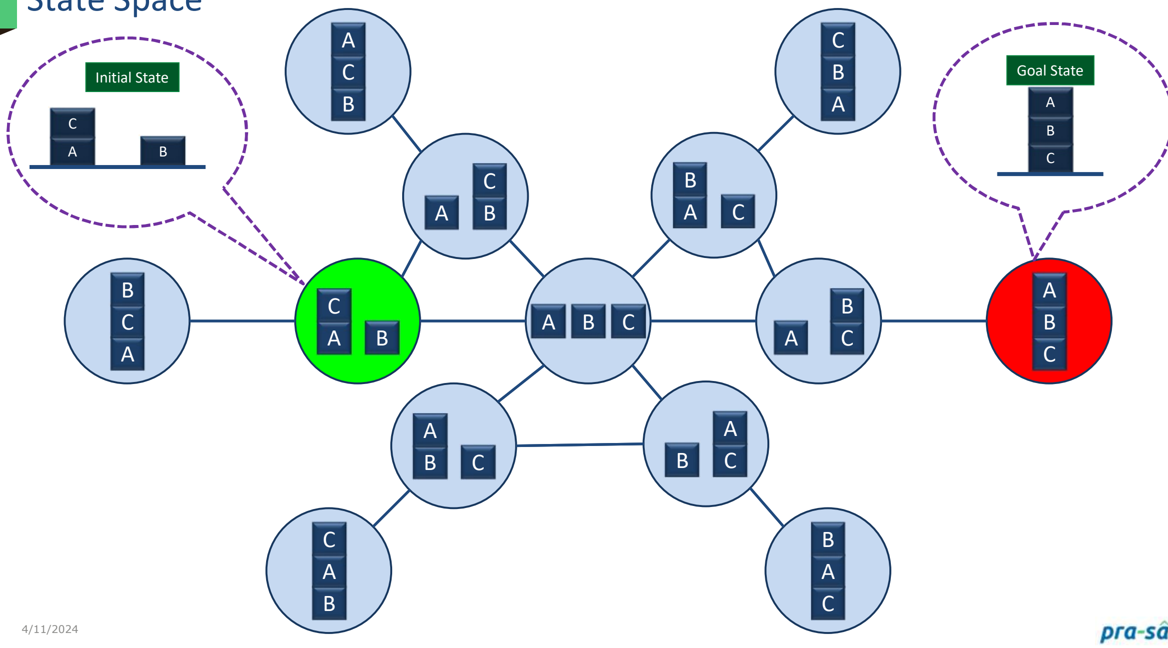
4

Agent's Knowledge Representation

Atomic	States are indivisible; No internal structure	Search on atomic states; Heuristic and Blind searches
Propositional (aka Factored)	States are made of state variables that take values (Propositional or multi-values or Continuous)	Search + Inference in (prop logic) and probabilistic (Bayes Nets) representations
Relational	States Describe the objects in the world and their inter-relations	Search + Inference in predicate logic (or relational prob. Models)
First-order	States Describe the objects in the world and their inter-relations along with Functions over the objects	Search + Inference in first order logic

5

State Space



4/11/2024

pra-sâmi

6

PA_FAI_01: Construct a Turing Machine

Course & Batch : ____________________________
Roll No: ______________________________________
Name: _____________________________________
Date of submission:_________________________

- Construct a Turing Machine for the language L = $\{0^n1^n2^n\}$ where $n \geq 1$
- Answer to include
  - Algorithm
  - Representative transition diagram
  - Initial Tape and Final Tape
  - Similar to what you saw in slides 29 and 30
- Submission:
  - PDF or Image to be uploaded in the shared drive
  - Upload file Name: PRN_<*last 4 digits of PRN*>_<*name*>.<*ext*>

4/11/2024

pra-sâmi

7

Agenda

- Goal Formulation by Example
- Search by example : Toy and real world Problems
- Type of Search
- Informed search/ Heuristic Search

4/11/2024

pra-sâmi

8

Goal Formulation

4/11/2024

pra-sâmi

9

Goal Formulation

Party… may be…

The decision problem is a complex one involving many tradeoffs and careful reading of guidebooks.

Overall a tough call…

4/11/2024

pra-sâmi

10

Goal Formulation

But…

Not reaching for meeting is out of question!

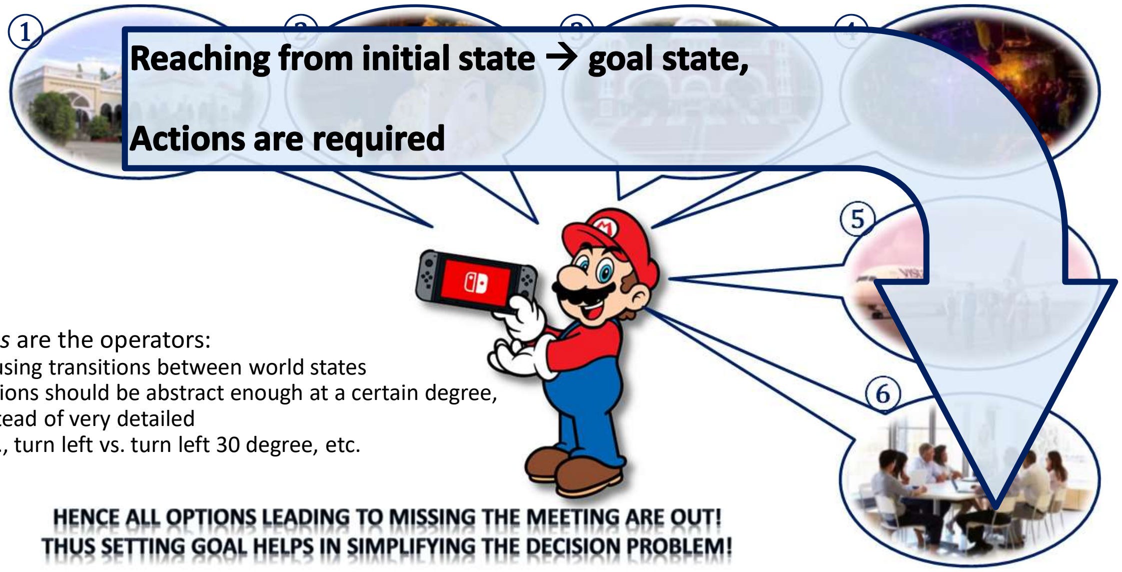
➔ The goal is formulated!

4/11/2024

pra-sâmi

11

Goal Formulation



Actions are the operators:

- ❖ Causing transitions between world states
- ❖ Actions should be abstract enough at a certain degree, instead of very detailed
- ❖ E.g., turn left vs. turn left 30 degree, etc.

HENCE ALL OPTIONS LEADING TO MISSING THE MEETING ARE OUT!
THUS SETTING GOAL HELPS IN SIMPLIFYING THE DECISION PROBLEM!

pra-sâmi

12

Recap - From our Example

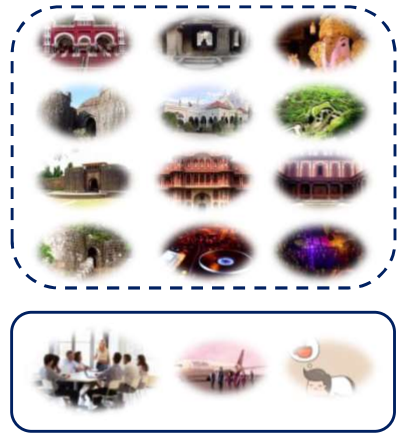
- ❑ Formulate Goal
  - ❖ Be In the Meeting
- ❑ Formulate Problem
  - ❖ States : Locations of interest
  - ❖ Actions : reach location, enjoy the time there
- ❑ Find Solution
  - ❖ Sequence of activities

pra-sâmi

13

Search

- Because there are many ways to achieve the same goal
  - Those ways are together expressed as a tree
  - Multiple options of unknown value at a point,
    - The agent can examine different possible sequences of actions, and choose the best
  - This process of looking for the best sequence is called search
  - The best sequence is then a list of actions, called solution

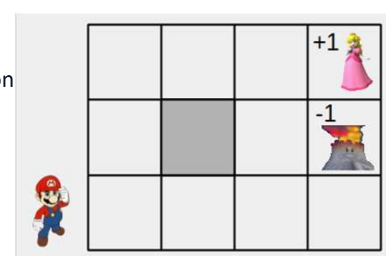


pra-sâmi

14

Search Algorithm

- Defined as
  - Taking a problem and Returns a solution
- Once a solution is found
  - The agent follows the solution and carries out the list of actions – execution
- Design of an agent
  - "Formulate, Search, Execute"



pra-sâmi

15

The Goal Test

- Applied to the current state to test
  - If the agent is in its goal state,
  - We don't want to continue to work if we have already reached the goal
- Sometimes there is an explicit set of possible goal states.
  - E.g. Reach the meeting in time
- Sometimes the goal is described by the properties
  - Instead of stating explicitly the set of states
- Example: Chess
  - The agent wins if it can capture the KING of the opponent on next move (checkmate) no matter what the opponent does

16

A Path Cost Function

- Assigns a numeric cost to each path
- Select a performance measure denoted by g
- To distinguish the best path from others
- Usually the path cost is
  - The sum of the step costs of the individual actions (in the action list)

17

Well-defined Problems and Solutions

- Together a problem is defined by
  - Initial state (that the agent starts in)
  - Actions (The set of possible actions)
  - Transition model (description of what each action does)
    or
    Successor functions (refer to any state reachable from given state by a single action)
  - Goal test
  - Path in the state space:
    - Any sequence of states connected by a sequence of actions.
  - Path cost function
- The solution of a problem is then
  - A path from the initial state to a state satisfying the **goal test**
- Optimal solution
  - The solution with **lowest path cost** among all solutions

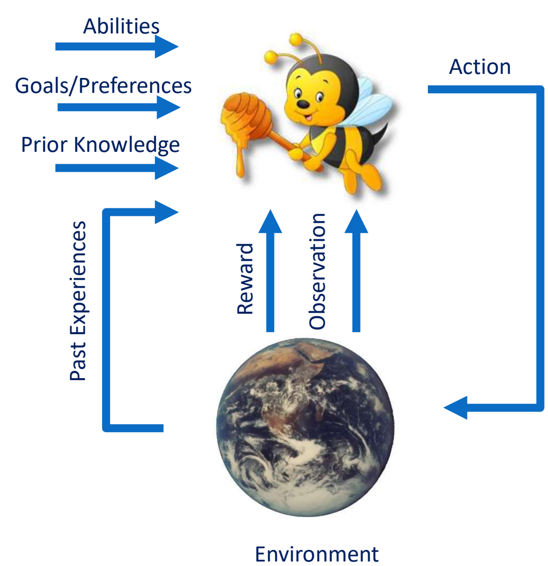
4/11/2024

pra-sâmi

18

Problem-Solving Agents

- Let's assume that agent is some machine which is interacting with environment to solve a problem (read perform a task)
- It solves problem by
  - Finding sequences of actions that lead to desirable states (goals)
  - Use more advanced factored or structured representations are usually called planning agents
- To solve a problem,
  - The first step is the goal formulation, based on the current situation



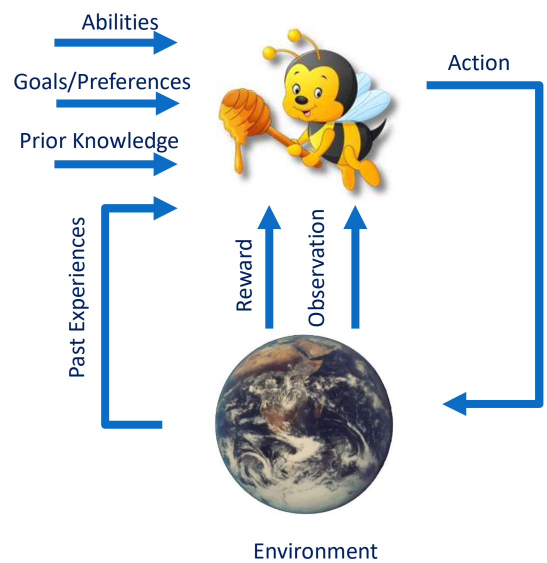
4/11/2024

pra-sâmi

19

Problem-Solving Agents

- Goal formulation
  - What is the goal state
  - What are important characteristics of the goal state
  - How does the agent know that it has reached the goal
  - Are there several possible goal states
    - Are they equal or are some more preferable
- Problem formulation
  - What are the possible states of the world relevant for solving the problem
  - What information is accessible to the agent
  - How can the agent progress from state to state

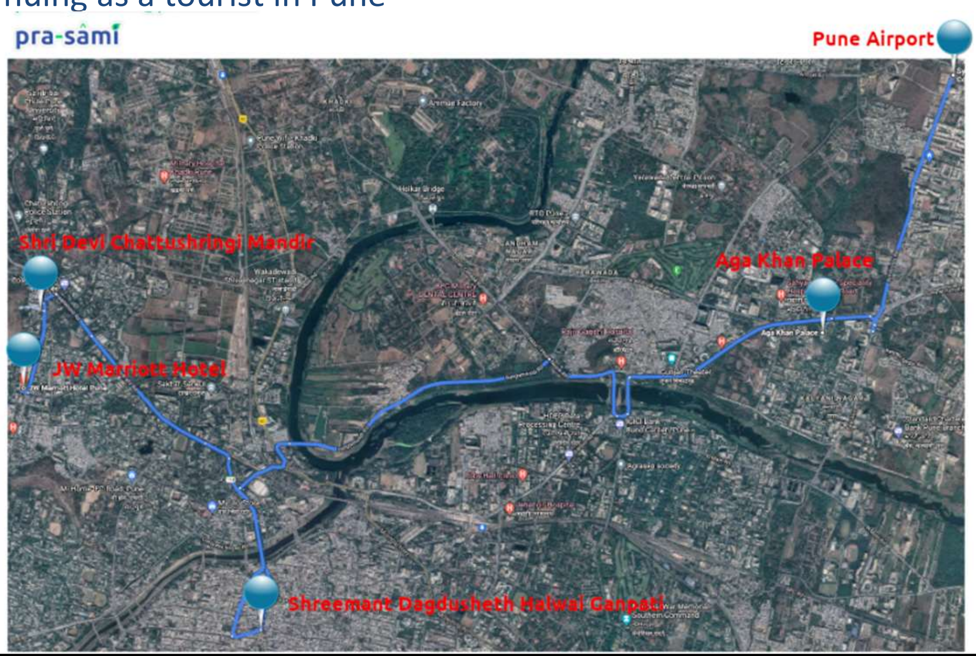


4/11/2024

pra-sâmi

20

Continuing as a tourist in Pune

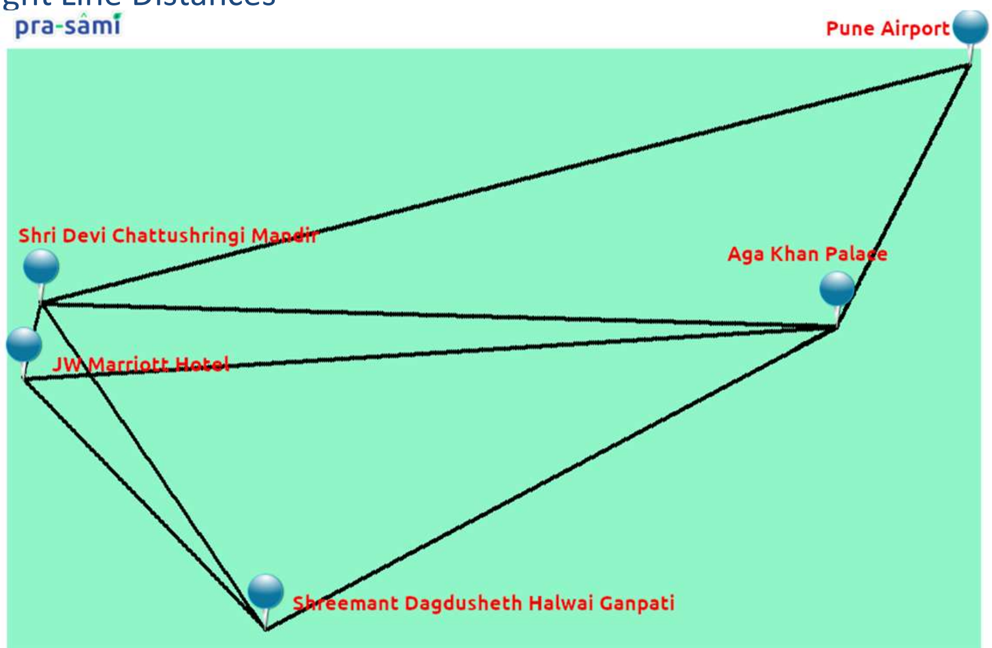


4/11/2024

pra-sâmi

21

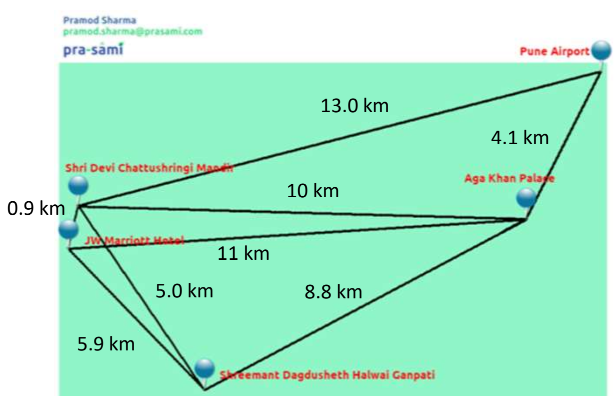
Straight Line Distances



4/11/2024

22

Straight Line Distances



Actual Distance by Road

Locations		SLD (km)
From	to	
Airport	Aga Khan Palace	2.87
Airport	Shreemant Dagdusheth Halwai Ganpati	8.80
Airport	Shri Devi Chattushringi Mandir	9.32
Airport	JW Marriott	9.73
Aga Khan Palace	Shreemant Dagdusheth Halwai Ganpati	6.31
Aga Khan Palace	Shri Devi Chattushringi Mandir	7.78
Aga Khan Palace	JW Marriott	7.95
Shreemant Dagdusheth Halwai Ganpati	Shri Devi Chattushringi Mandir	3.07
Shreemant Dagdusheth Halwai Ganpati	JW Marriott	3.35
Shri Devi Chattushringi Mandir	JW Marriott	0.82

Straight Line Distance

4/11/2024

23

Abstraction

- While planning a day in Pune, we focus on the destinations and time it will take:
  - En route
  - At the destination
- Besides the four components (Initial state, Actions, Successor function, Goal test) for problem formulation
  - Anything else?
- We are not worried about other aspect of the journey
  - Cab? Which one? How many red lights? Cops on the way? etc.
- Abstraction
  - The process to take out the irrelevant information
  - Leave the most essential parts to the description of the states (Remove detail from representation)
  - Conclusion: Use the most important parts that are contributing to search only

4/11/2024

pra-sâmi

24

Real-World Example - Route-Finding Problem

- Route-finding algorithms are used in a variety of applications.
  - Web sites
  - In-car systems that provide driving directions,
  - Routing video streams in computer networks,
  - Military operations planning,
  - Airline travel-planning systems

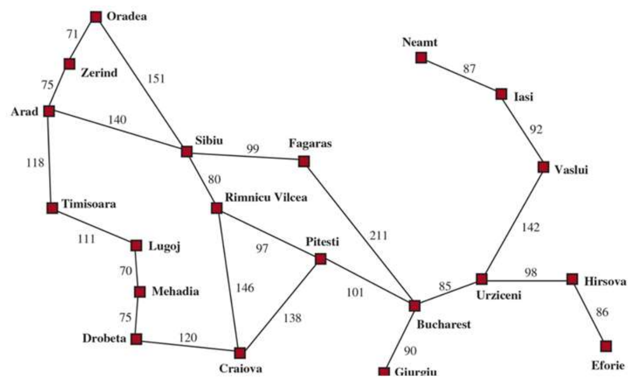
4/11/2024

pra-sâmi

25

Searching For Solutions – another example

- On holiday in Romania; currently in Arad
- Flight leaves tomorrow from Bucharest
- Formulate goal:
  - Be in Bucharest
- Formulate problem:
  - States: various cities
  - Actions: drive between cities
- Find solution:
  - Sequence of cities, e.g., Arad, Sibiu, Fagaras, Bucharest



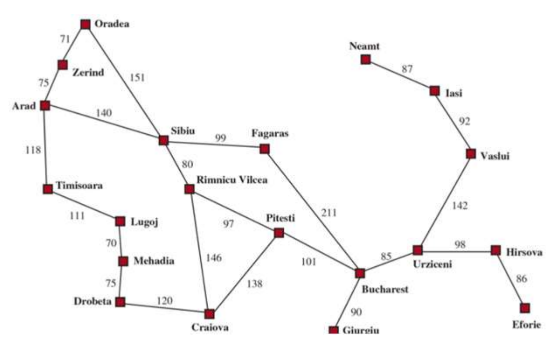
4/11/2024

pra-sâmi

26

Single-state problem formulation

- A problem is defined by four items:
  - **Initial State** e.g., "at Arad“
  - **Actions** or successor function S(x) = set of action–state pairs
    - e.g., S(Arad) = { < Arad → Zerind, Zerind >, … }
  - **Goal test**, can be explicit or implicit .
    - Explicit in this case, e.g., x = "at Bucharest“
  - **Path cost** (additive)
    - e.g., sum of distances
- A solution is a sequence of actions leading from the initial state to a goal state

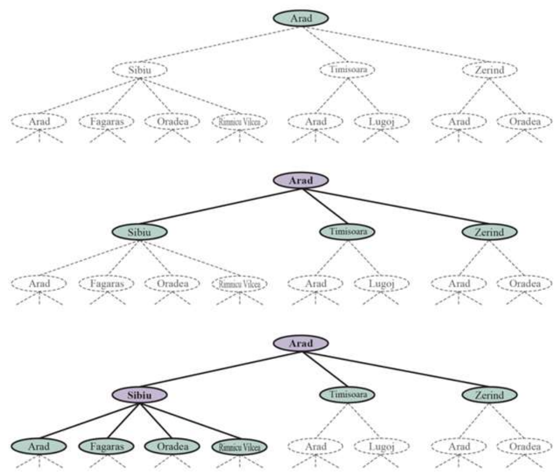


4/11/2024

pra-sâmi

Search Tree

- Initial state
  - The root of the search tree
- Expanding
  - Applying successor function to the current state
  - Thereby generating a new set of states
- Leaf nodes
  - The states having no successors
- Fringe
  - Nodes that have not been expanded yet.



pra-sâmi

Components of a Node

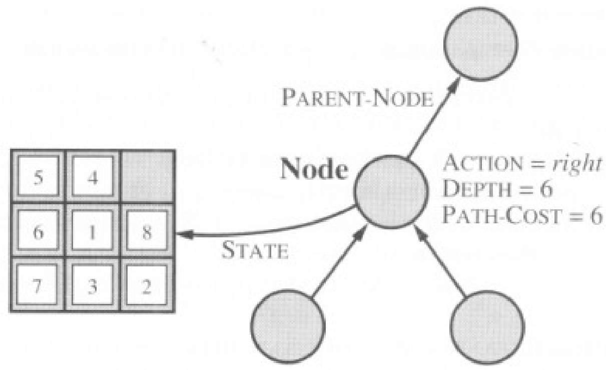
- State: which state it is in
- Parent Node: from which node it is generated
- Action: which action applied to its parent-node to generate it
- Path Cost: the cost, g(n), from initial state to the node n itself
- Depth: number of steps along the path from the initial state

pra-sâmi

29

Node Vs. State

- A node is a bookkeeping data structure used to represent the search tree
  - It includes state variables, a pointer to the parent node, its depth in the tree, and the path cost to reach it.
- A state corresponds to a configuration of the world
  - The states form a graph.
- Nodes are on particular paths, as defined by PARENT pointers , whereas states are not
- Two different nodes can contain the same world state if that state is generated via two different search paths



pra-sâmi

30

Search Strategies

Uninformed Search (Blind Search)	Informed Search (Heuristic Search)
Breadth First Search	Best First Search
Uniform Cost Search	Greedy Best First Search
Depth First Search	A* Search
Depth First Limited Search	Recursive Best First Search
Iterative deepening depth-first search	Beam Search
Bi-directional Search	
Constraint Satisfaction Search	

pra-sâmi

31

Informed Search / Heuristic Search

4/11/2024

pra-sâmi

32

Heuristics and Algorithms

- A correct algorithm will find you the best solution given good data and enough time
  - It is precisely specified
- A heuristic gives you a workable solution in a reasonable time
  - It gives a guided or directed solution

4/11/2024

pra-sâmi

33

Heuristic Search

- A heuristic is a rule or principle used to guide a search
  - It provides a way of giving additional knowledge of the problem to the search algorithm
  - Must provide a reasonably reliable estimate of how far a state is from a goal, or the cost of reaching the goal via that state
- Heuristic search is also known as informed search
- Important aspect: formation of heuristic function h(n)
  - A way of calculating or estimating such distances/cost
- Map Distance: heuristic function can be straight line distance (SLD)

Locations		SLD (km)
From	to	
Airport	Aga Khan Palace	2.87
Airport	Shreemant Dagdusheth Halwai Ganpati	8.80
Airport	Shri Devi Chattushringi Mandir	9.32
Airport	JW Marriott	9.73
Aga Khan Palace	Shreemant Dagdusheth Halwai Ganpati	6.31
Aga Khan Palace	Shri Devi Chattushringi Mandir	7.78
Aga Khan Palace	JW Marriott	7.95
Shreemant Dagdusheth Halwai Ganpati	Shri Devi Chattushringi Mandir	3.07
Shreemant Dagdusheth Halwai Ganpati	JW Marriott	3.35
Shri Devi Chattushringi Mandir	JW Marriott	0.82

34

Evaluation function

- There are an infinite number of possible heuristics
  - Criteria is that it returns an assessment of the point in the search
- If an evaluation function is accurate, it will lead directly to the goal
- More realistically, this usually ends up as “seemingly-best-search”
- Traditionally, the lowest value after evaluation is chosen as we usually want the lowest cost

35

Heuristic evaluation functions

- Estimate of expected utility value from a current position
  - E.g. value for pieces left in chess
  - Way of judging the value of a position
- Humans have to do this as we do not evaluate all possible alternatives
  - These heuristics usually come from years of human experience
- Performance of a game playing program is very dependent on the quality of the function

pra-sâmi

36

Heuristics for the 8-puzzle

- Number of tiles out of place (h1)
- Manhattan distance (h2)
  - Sum of the distance of each tile from its goal position
  - Tiles can only move up / down or left/right → city blocks

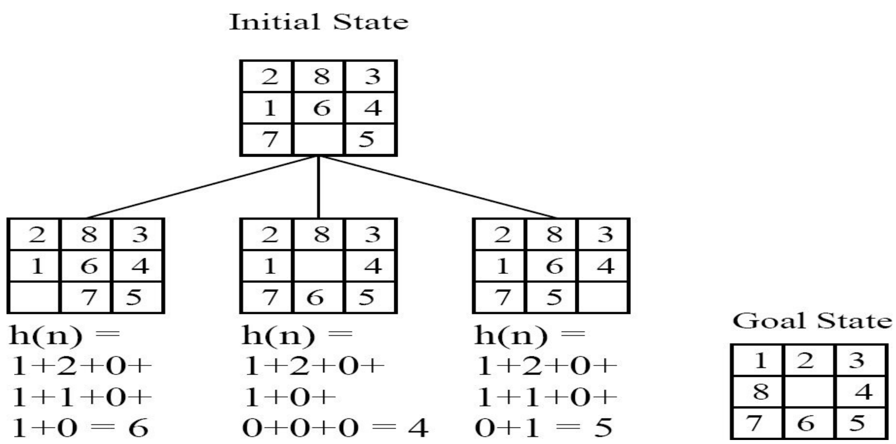
0	1	2
3	4	5
6	7	

pra-sâmi

37

The 8-puzzle

- Using a heuristic evaluation function:
  - $h_2(n)$ = sum of the distance each tile is from its goal position



pra-sâmi

38

H1 and H2 Functions

Goal state

0	1	2
3	4	5
6	7	

Current state

0	1	2
3	4	5
6		7

$h_1=1$
$h_2=1$

Current state

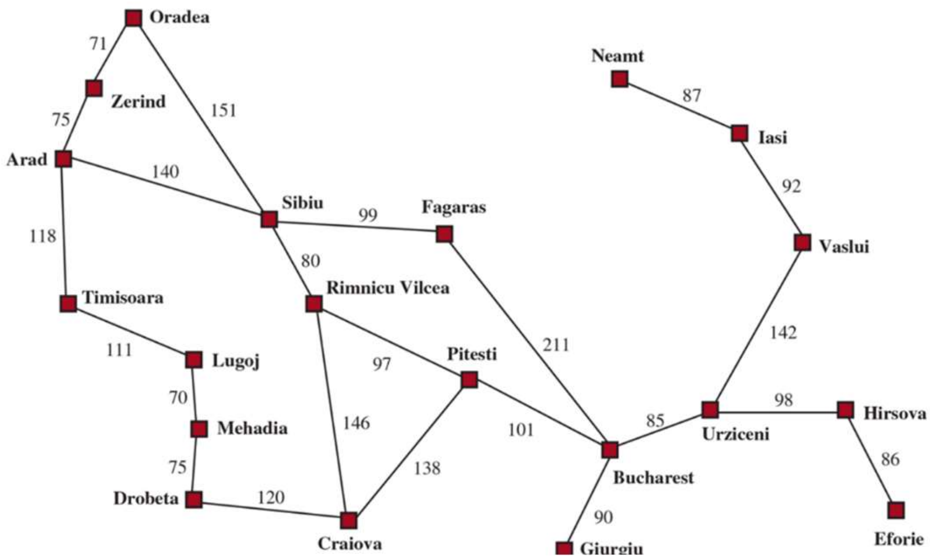
0	2	5
3	1	7
6		4

$h_1=5$
$h_2=0+1+1+0+2+1+0+2=7$

pra-sâmi

39

Heuristic Search : Heuristic Function



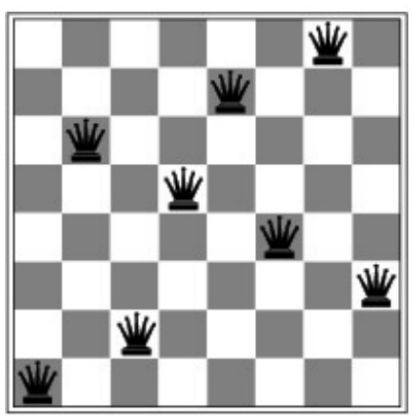
Straight–line distance to Bucharest	
Arad	366
Bucharest	0
Craiova	160
Dobreta	242
Eforie	161
Fagaras	178
Giurgiu	77
Hirsova	151
Iasi	226
Lugoj	244
Mehadia	241
Neamt	234
Oradea	380
Pitesti	98
Rimnicu Vilcea	193
Sibiu	253
Timisoara	329
Urziceni	80
Vaslui	199
Zerind	374

4/11/2024

pra-sâmi

40

Example: 8 - Queens



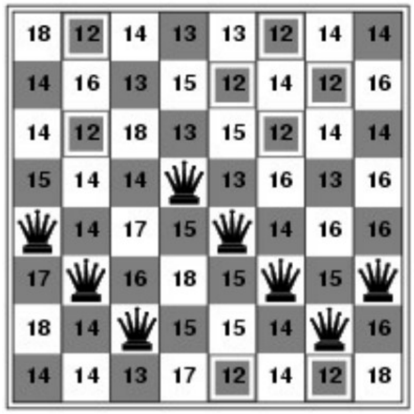
- A local minimum with h = 1

4/11/2024

pra-sâmi

41

Example: 8 - Queens



- h = number of pairs of queens that are attacking each other, either directly or indirectly
- h = 17 for the above state

4/11/2024

pra-sâmi

42

Heuristic Search : Best-First Search

4/11/2024

pra-sâmi

43

Heuristic Search: Best-First Search

How do we decide which node from the frontier to expand next?

- Idea: use an evaluation function f(n) for each node
  - f(n) provides an estimate for the total cost.
  - Expand the node n with smallest f(n).
- Implementation:
  - Order the nodes in the fringe (frontier) increasing order of cost.
- Special cases:
  - Greedy best-first search
  - A* search

44

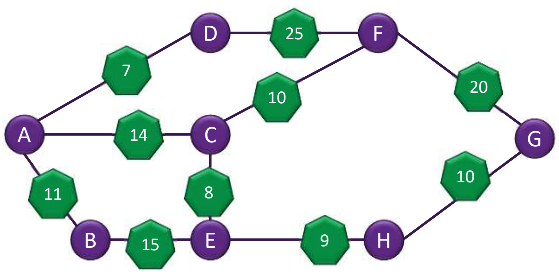
Heuristic Search: Greedy-Best Search

- Tries to expand the node that is closest to the goal
- Evaluates using only heuristic function :
  - f(n) = h(n)
- Possibly lead to the solution very fast
- Problem :
  - Can end up in sub-optimal solutions (doesn't take notice of the distance it travels).
- Complexity and time: $O(b^m)$
- Complete & optimal : No (stuck in infinite loop)

45

Heuristic Search : Greedy-Best Search

From → to nodes	SLD	From → to nodes	SLD
A → G	40	E → G	19
B → G	32	F → G	17
C → G	25	H → G	10
D → G	35	G → G	0



- Start with priority-queue = initial-state
- While priority-queue not empty do:
  - Remove best node from the priority-queue
  - If it is the goal node, return success. Otherwise find its successors
  - Apply evaluation function to successors and add to priority-queue
  - Evaluation function f(n) is estimate of total cost

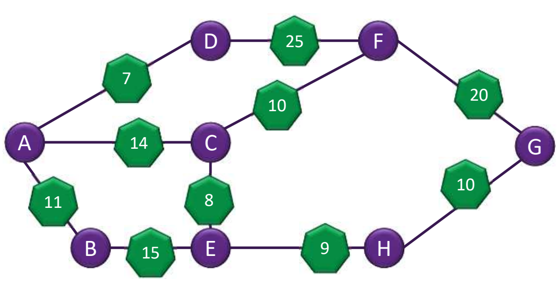
4/11/2024

pra-sâmi

46

Heuristic Search : Greedy-Best Search

From → to nodes	SLD	From → to nodes	SLD
A → G	40	E → G	19
B → G	32	F → G	17
C → G	25	H → G	10
D → G	35	G → G	0



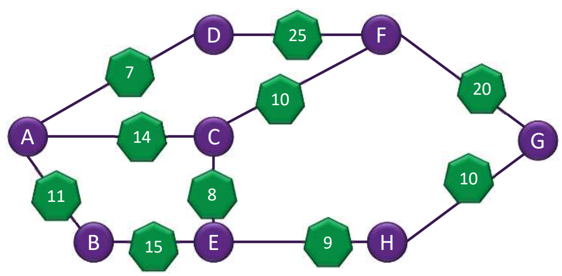
4/11/2024

pra-sâmi

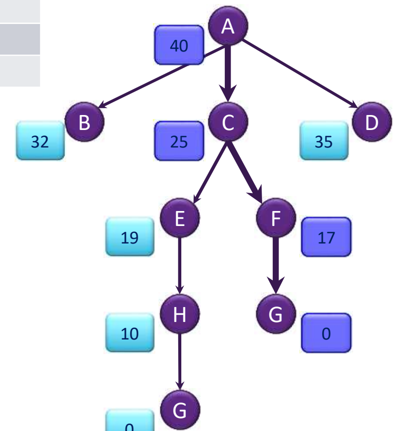
47

Heuristic Search : Greedy-Best Search

From → to nodes	SLD	From → to nodes	SLD
A → G	40	E → G	19
B → G	32	F → G	17
C → G	25	H → G	10
D → G	35	G → G	0



Open	Closed
A	
~~A~~, C, B, D,	A
~~C~~, F, E, B, D	A, C
~~F~~, G, E, B, D	A, C, F

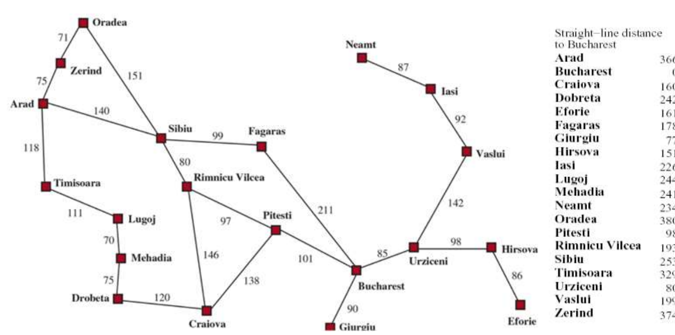


Note that A-C-E-H-G is shorter. A-C-F-G is fast but suboptimal solution!

pra-sâmi

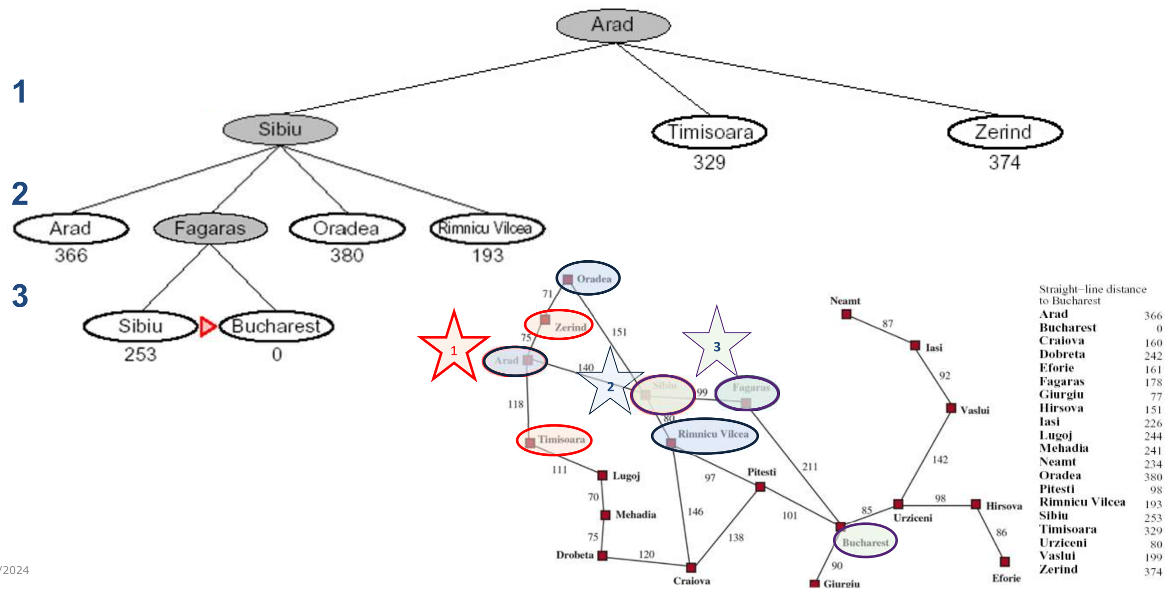
48

Heuristic Search : Greedy-Best Search



49

Heuristic Search : Greedy-Best Search



4/11/2024

50

Heuristic Search : A* Algorithm

4/11/2024

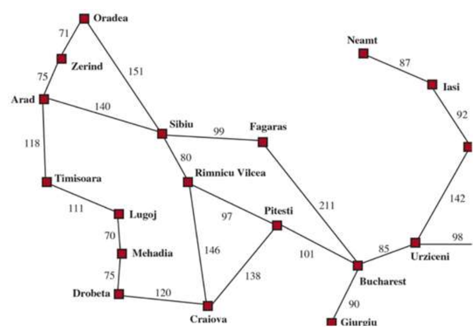
51

Heuristic Search : A* Algorithm

- Widely known algorithm – (pronounced as “A star” search)
- Evaluates nodes by combining g(n) “cost to reach the node” and h(n) “cost to get to the goal”
- f(n) = g(n) + h(n),   f(n) ➔ estimated cost of the cheapest solution
- Complete and optimal : since evaluates all paths
- Time ? : a bit time consuming
- Space ? : lot of it!

52

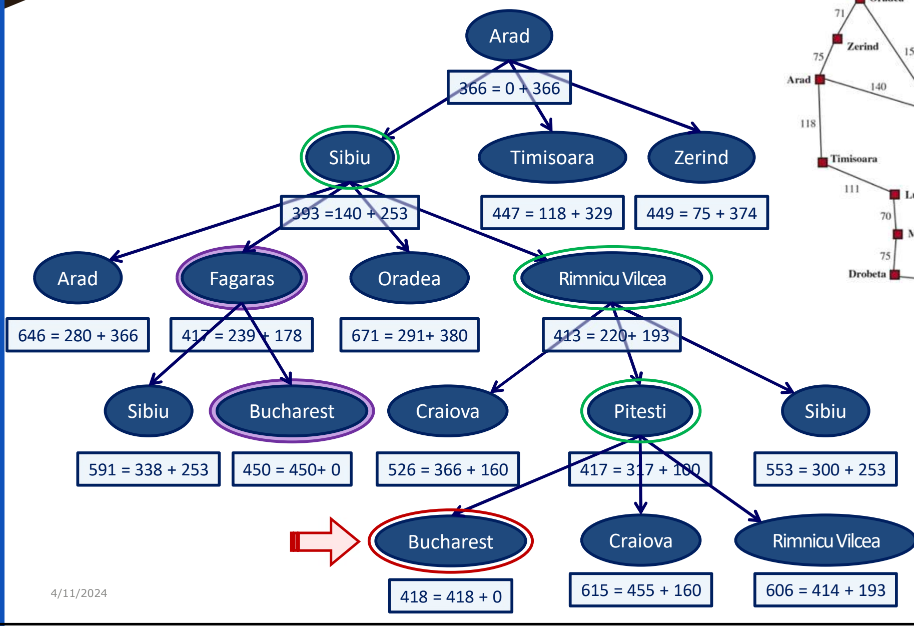
Heuristic Search : A* Algorithm

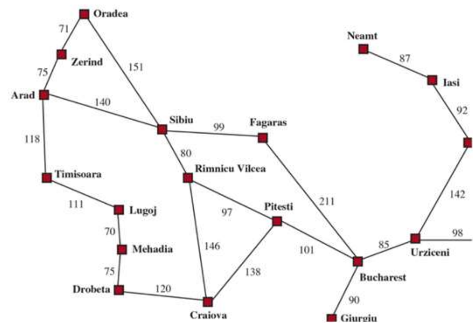


Straight−line distance to Bucharest	
Arad	366
Bucharest	0
Craiova	160
Dobreta	242
Eforie	161
Fagaras	178
Giurgiu	77
Hirsova	151
Iasi	226
Lugoj	244
Mehadia	241
Neamt	234
Oradea	380
Pitesti	98
Rimnicu Vilcea	193
Sibiu	253
Timisoara	329
Urziceni	80
Vaslui	199
Zerind	374

53

Heuristic Search : A* Algorithm





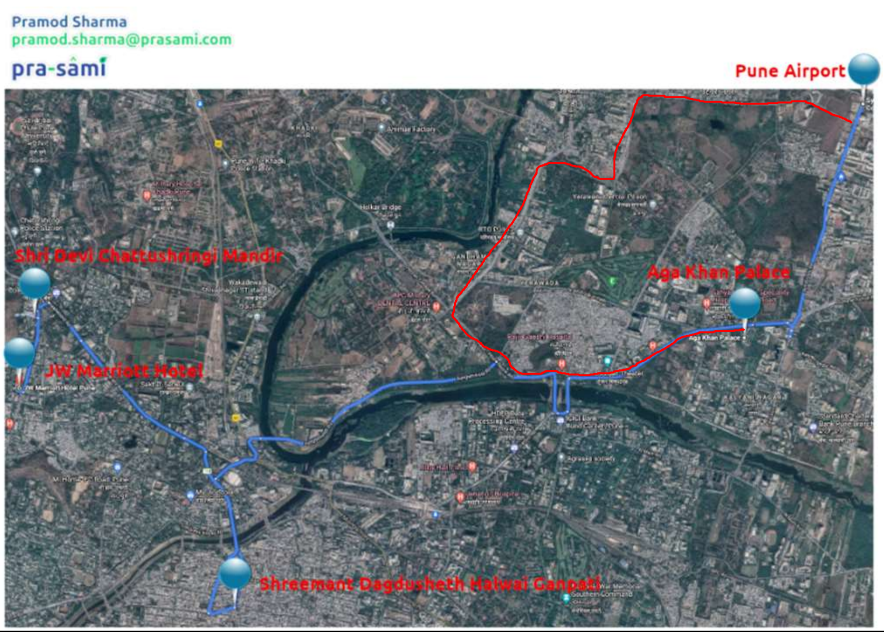
Straight–line distance to Bucharest

Arad	366
Bucharest	0
Craiova	160
Dobreta	242
Eforie	161
Fagaras	178
Giurgiu	77
Hirsova	151
Iasi	226
Lugoj	244
Mehadia	241
Neamt	234
Oradea	380
Pitesti	98
Rimnicu Vilcea	193
Sibiu	253
Timisoara	329
Urziceni	80
Vaslui	199
Zerind	374

4/11/2024

54

A* Algorithm : Memory Hungry



pra-sâmi

4/11/2024

55

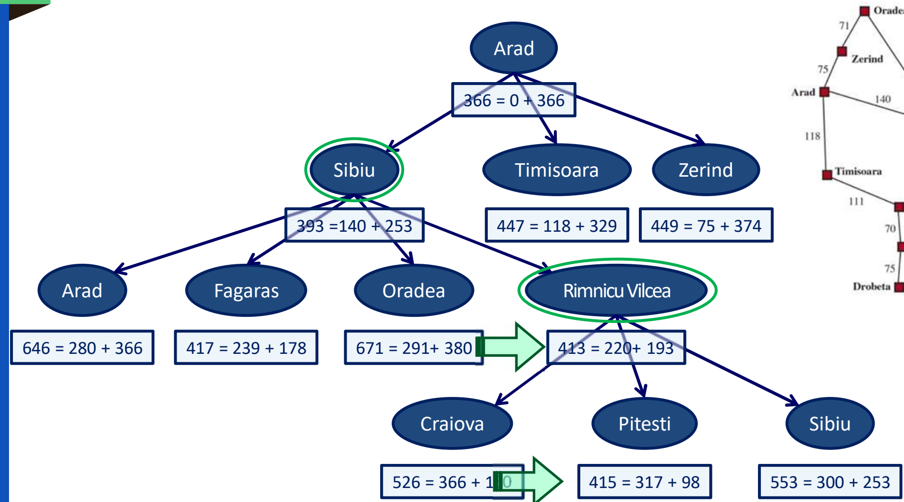
Memory Bound Search - RBFS

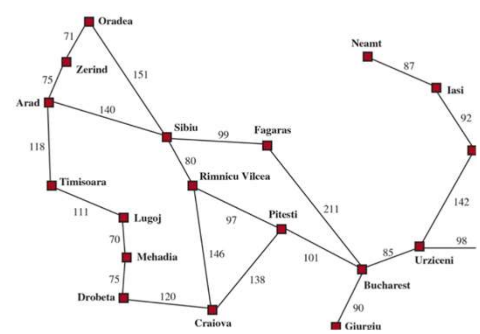
- Recursive Best First Search is similar to A* algorithm
  - Both are recursive
  - Difference
    - A* keeps all nodes in memory
    - RBFS keeps current path and sibling nodes only
  - When to stop search
    - It no longer looks good
  - Forget the sub tree to save space

pra-sâmi

56

Memory Bound : Recursive Best First Search



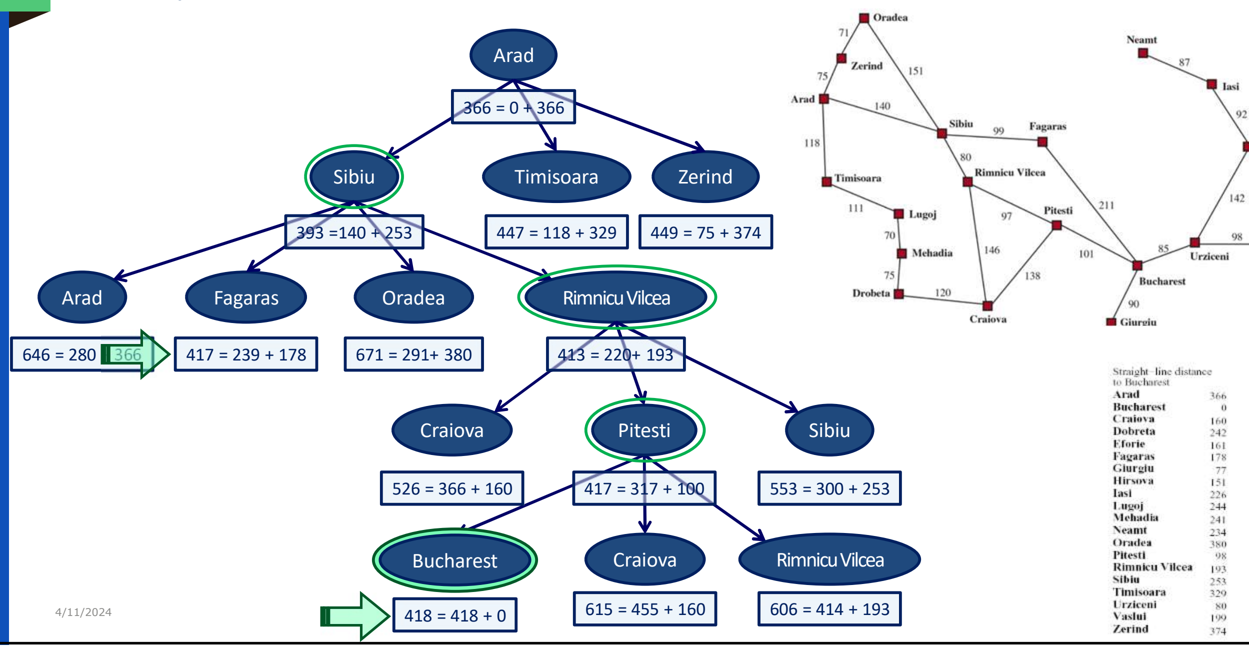


Straight-line distance to Bucharest

Arad	366
Bucharest	0
Craiova	160
Dobreta	242
Eforie	161
Fagaras	178
Giurgiu	77
Hirsova	151
Iasi	226
Lugoj	244
Mehadia	241
Neamt	234
Oradea	380
Pitesti	98
Rimnicu Vilcea	193
Sibiu	253
Timisoara	329
Urziceni	80
Vaslui	199
Zerind	374

57

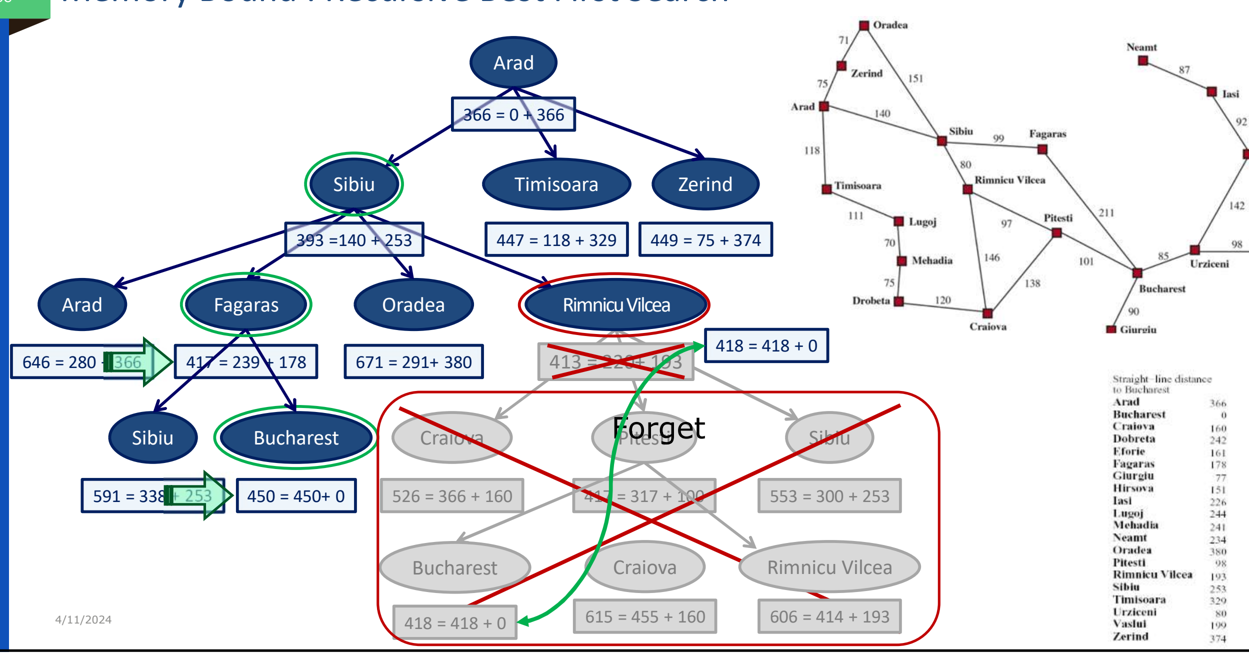
Memory Bound : Recursive Best First Search



4/11/2024

58

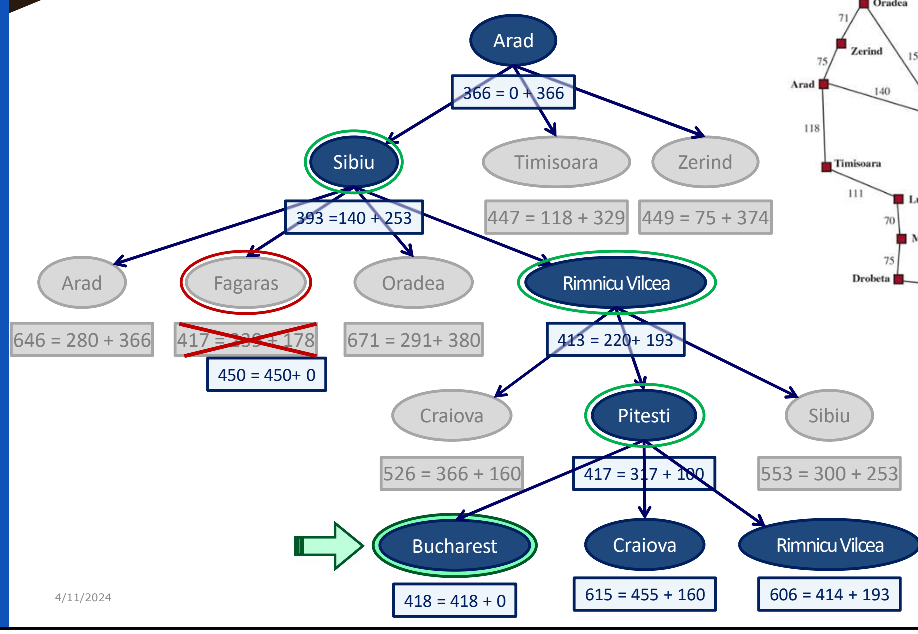
Memory Bound : Recursive Best First Search

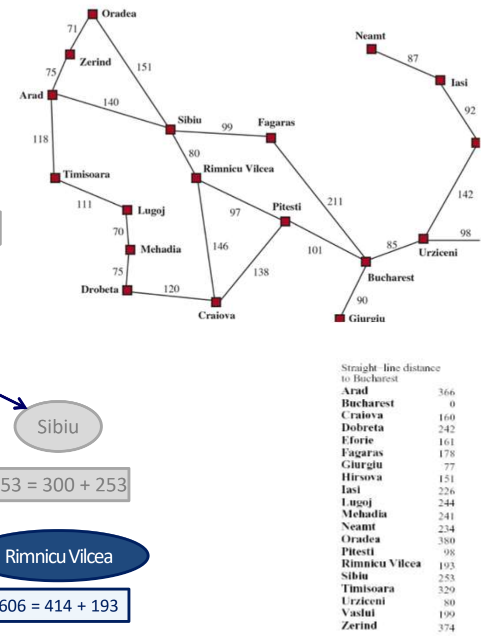


4/11/2024

59

Memory Bound : Recursive Best First Search





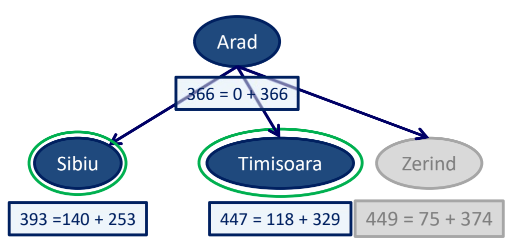
4/11/2024

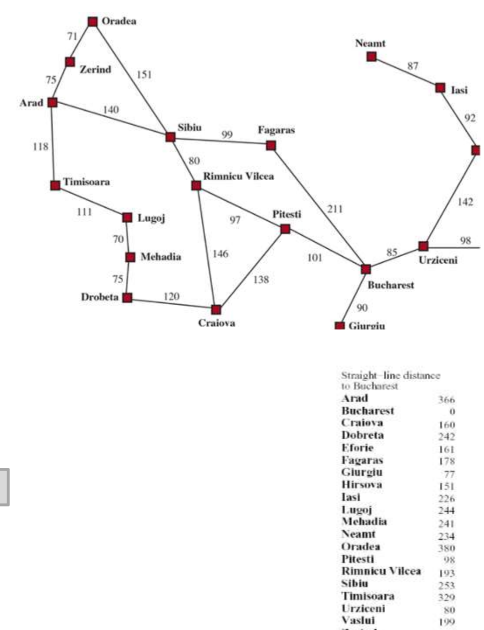
60

Beam Search

- Beam search: explore a graph by expanding the most optimistic node in a limited set.
  - Is an optimization of ***best-first search*** that reduces its memory requirements.

- Beam Search with β = 2

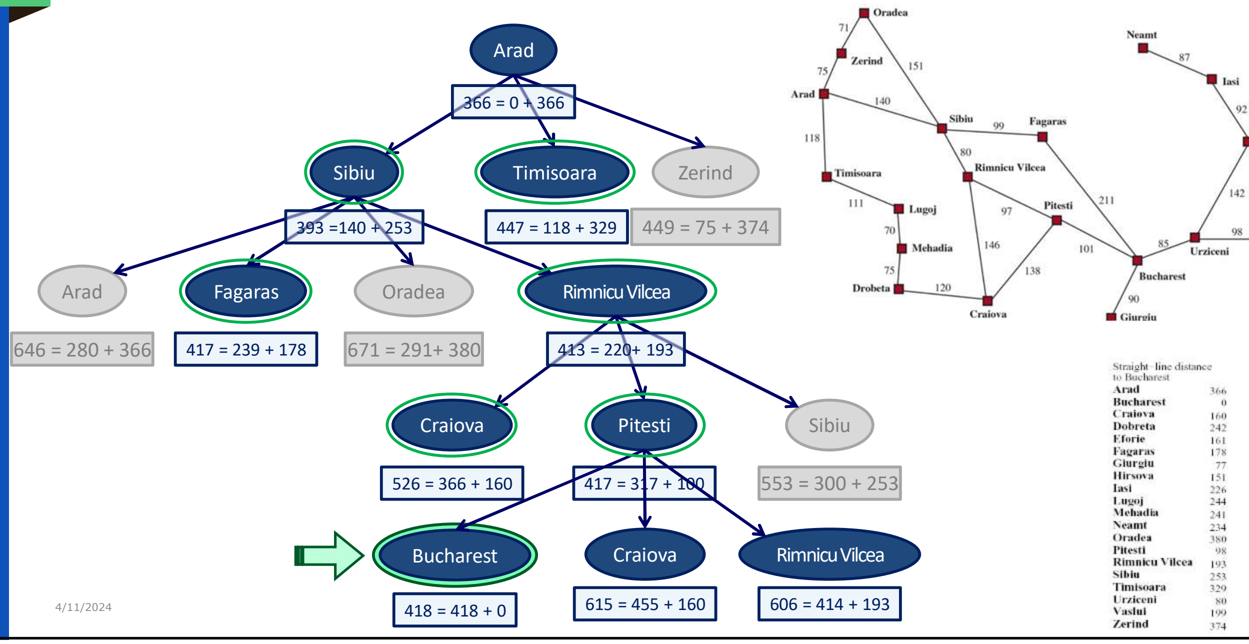




4/11/2024

61

Beam Search (β = 2)



62

Reflect…

- Before an agent can start searching for solutions:
  - A goal must be identified and
  - A well-defined problem must be formulated
- A problem consists of five parts: the initial state, a set of actions, a transition model describing the results of those actions, a goal test function, and a path cost function (optional)
- The environment of the problem is represented by a state space
- A path through the state space from the initial state to a goal state is a solution
- Search algorithms treat states and actions as atomic: they do not consider any internal structure they might possess
- Search algorithms are judged on the basis of completeness, optimality, time complexity, and space complexity.
- Complexity depends on b, the branching factor in the state space, and d, the depth of the shallowest solution

63

Reflect…

- Informed search methods may have access to a heuristic function h(n) that estimates the cost of a solution from n.
  - The generic best-first search algorithm selects a node for expansion according to an evaluation function
  - Greedy best-first search expands nodes with minimal h(n). It is not optimal but is often efficient.
  - A^* search expands nodes with minimal f(n) = g(n) + h(n).
  - A^* is complete and optimal, provided that h(n) is admissible (for TREE-SEARCH) or consistent (for GRAPH-SEARCH). The space complexity of A^* is still prohibitive.
  - RBFS (recursive best-first search) is robust, optimal search algorithms that use limited amounts of memory; given enough time, they can solve problems that A^* cannot solve because it runs out of memory.
- The performance of heuristic search algorithms depends on the quality of the heuristic function.
- One can sometimes construct good heuristics by relaxing the problem definition, by storing pre-computed solution costs for sub-problems in a pattern database, or by learning from experience with the problem class.

4/11/2024

pra-sâmi

64

Reflect

- A heuristic is a way of trying ___________
  - To discover something or an idea embedded in a program
  - To search and measure how far a node in a search tree seems to be from a goal
  - To compare two nodes in a search tree to see if one is better than the other is
  - All of the mentioned
- What is the term used for describing the judgmental or commonsense part of problem solving?
  - Heuristic
  - Critical
  - Value based
  - Analytical
- A search algorithm takes ____ as an input and returns ____ as an output.
  - Input, output
  - Problem, solution
  - Solution, problem
  - Parameters, sequence of actions
- A problem in a search space is defined by one of these state
  - Initial state
  - Last state
  - Intermediate state
  - All of the mentioned

4/11/2024

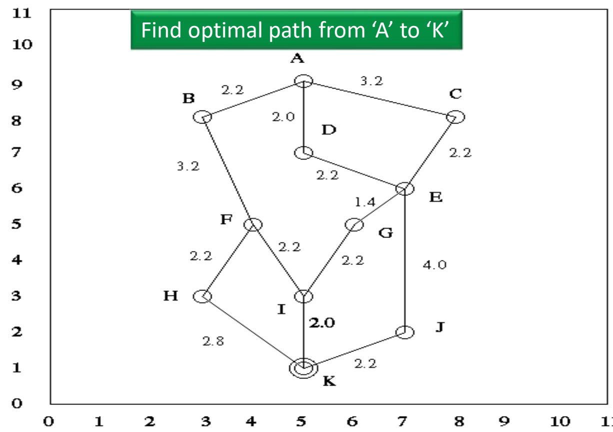
pra-sâmi

65

PA_FAI_02: A* Algorithm

Course & Batch : ____________________________
Roll No: ____
Name: ____________________________________
Date of submission:_____15-04-2024___________________

Node	Coordinates	SL Distance to K
A	(5,9)	8.0
B	(3,8)	7.3
C	(8,8)	7.6
D	(5,7)	6.0
E	(7,6)	5.4
F	(4,5)	4.1
G	(6,5)	4.1
H	(3,3)	2.8
I	(5,3)	2.0
J	(7,2)	2.2
K	(5,1)	0.0



Answer to include
- Optimal Path
- Open and close list
- g(n) : cost to reach the node
- h(n) : cost to get to the goal
- And f(n) = g(n) + h(n), : estimated cost of the cheapest solution

pra-sâmi

66

Next Session…

- Introduction
- Breadth-first search
- Depth-first search
- Bidirectional search

pra-sâmi

67

pra-sâmi