Artificial Intelligence

Lecture 3, Chapter 3
Solving Problems by Searching

Supta Richard Philip

Department of CS
AIUB

AIUB, January 2025

Table of Contents

Table of Contents

Informed (Heuristic) Search Strategies

- An informed search strategy uses problem-specific knowledge as well as the definition of the problem.
- An informed search can find solutions more efficiently than an uninformed strategy. A general approach is called **best-first search**.
- Best-first search is an instance of the general TREE-SEARCH or GRAPH-SEARCH algorithm in which a node is selected for expansion based on an **evaluation function, f(n)**, so the node with the lowest evaluation is expanded first.
- Best-first graph search is identical to uniform-cost search, except for the use of f instead of g to order the priority queue.

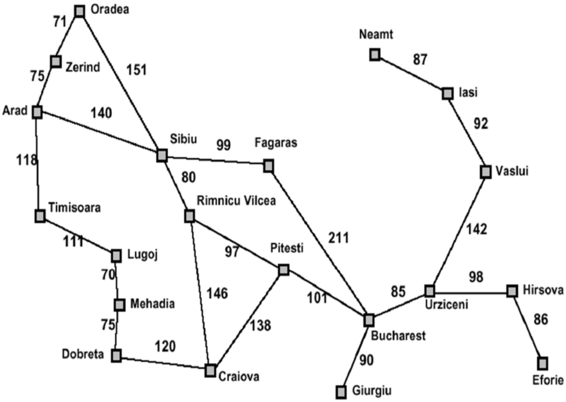
Informed (Heuristic) Search Strategies

- Most best-first algorithms include as a component of f a heuristic function, denoted h(n).
  h(n) = estimated cost of the cheapest path from the state at node n to a goal state.
- estimate the cost of the cheapest path from node n to goal node via the straight-line distance.
- if n is a goal node, then h(n) = 0

Greedy best-first search

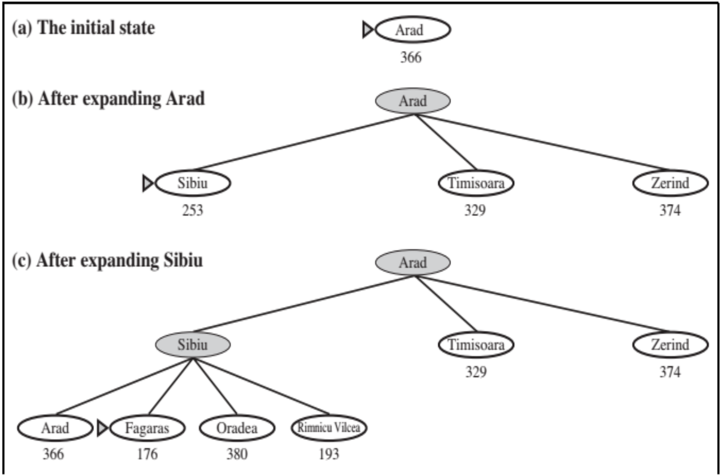
- Greedy best-first search tries to expand the node that is closest to the goal, on the grounds that this is likely to lead to a solution quickly. Thus, it evaluates nodes by using just the heuristic function; that is, $f(n) = h(n)$.
- Let us see how this works for route-finding problems in Romania; we use the straight line distance heuristic, which we will call h_{SLD}.
- Greedy best-first tree search is also incomplete even in a finite state space, much like depth-first search.

Greedy best-first search Cont.

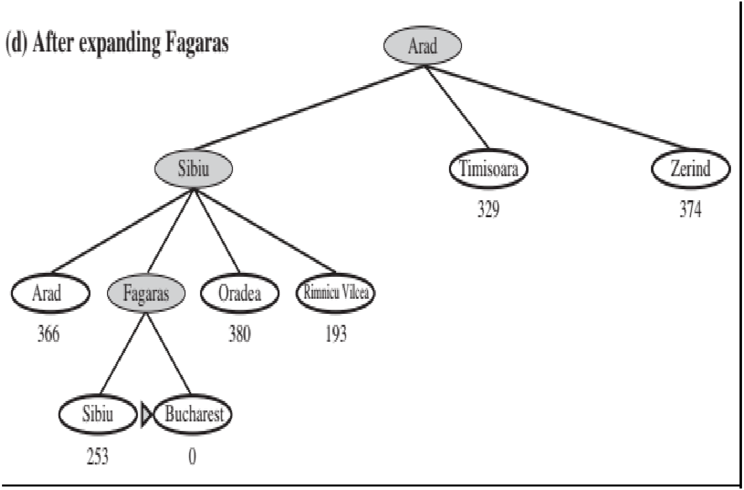


Straight−line distance to Bucharest	
Arad	366
Bucharest	0
Craiova	160
Dobreta	242
Eforie	161
Fagaras	178
Giurgiu	77
Hirsova	151
Iasi	226
Lugoj	244
Mehadia	241
Neamt	234
Oradea	380
Pitesti	98
Rimnicu Vilcea	193
Sibiu	253
Timisoara	329
Urziceni	80
Vaslui	199
Zerind	374

Greedy best-first search Cont.



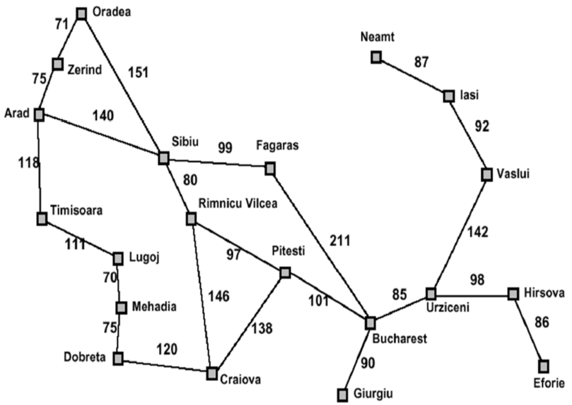
Greedy best-first search Cont.



A* search

- The most widely known form of best-first search is called A* search (pronounced "A-star Search").
- It evaluates nodes by combining g(n), the cost to reach the node, and h(n), the cost to get from the node to the goal $f(n) = g(n) + h(n)$.
- Since g(n) gives the path cost from the start node to node n, and h(n) is the estimated cost of the cheapest path from n to the goal,
  we have f(n) = estimated cost of the cheapest solution through n.

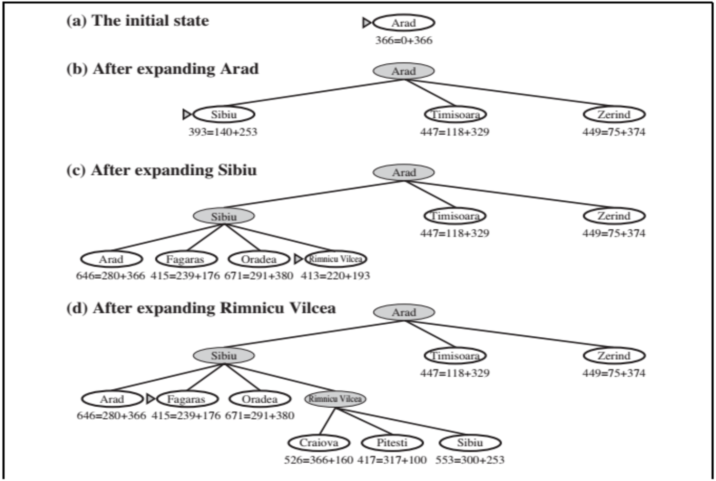
A* search cont.



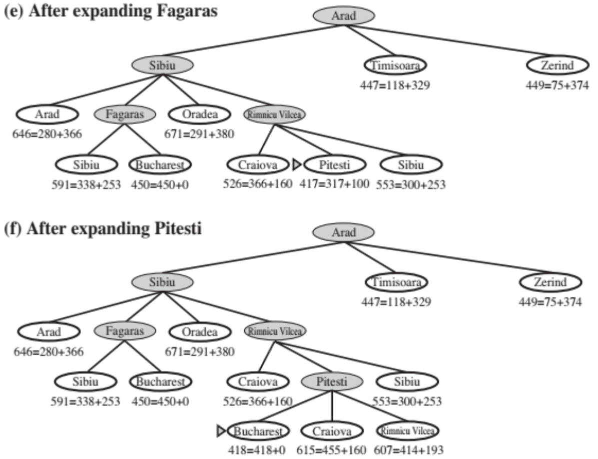
Straight−line distance to Bucharest

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

A* search cont.



A* search cont.



Conditions for optimality: Admissibility and consistency

- The first condition for **optimality** is that h(n) be an admissible heuristic. An admissible heuristic is one that never overestimates the cost to reach the goal.
- A second, slightly stronger condition called **consistency (or sometimes monotonicity)** is required only for applications of A* to graph search.
- A heuristic h(n) is consistent if, for every node n and every successor n‘ of n generated by any action a, the estimated cost of reaching the goal from n is no greater than the step cost of getting to n‘ plus the estimated cost of reaching the goal from n‘: $h(n) \leq c(n, a, n‘) + h(n‘)$.

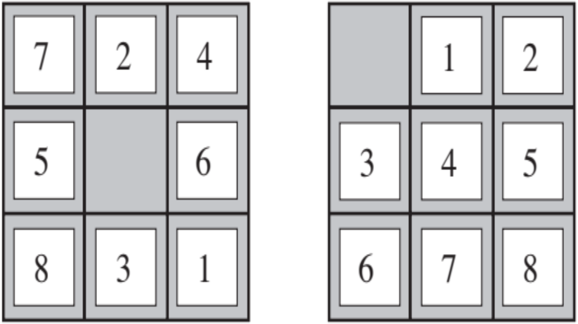
Conditions for optimality: Admissibility and consistency

- This is a form of the general **triangle inequality**, which stipulates that each side of a triangle cannot be longer than the sum of the other two sides.
- A* has the following properties: the tree-search version of A* is optimal if h(n) is admissible, while the graph-search version is optimal if h(n) is consistent.
- if h(n) is consistent, then the values of f(n) along any path are nondecreasing.
- The proof follows directly from the definition of consistency. Suppose n' is a successor of n; then $g(n') = g(n) + c(n, a, n')$ for some action a,
  $f(n') = g(n') + h(n') = g(n) + c(n, a, n') + h(n') \geq g(n) + h(n) = f(n)$.

Table of Contents

Heuristic Functions

- The 8-puzzle was one of the earliest heuristic search problems.



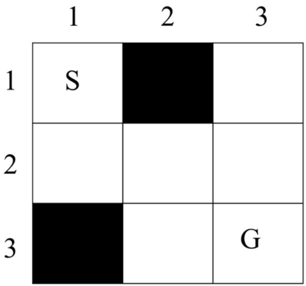
Start State   Goal State

Heuristic Functions 8-puzzle

- $h_1 =$ the number of misplaced tiles.
  All of the eight tiles are out of position, so the start state would have h1 = 8. h1 is an admissible heuristic because it is clear that any tile that is out of place must be moved at least once.
- $h_2 =$ the sum of the distances of the tiles from their goal positions.
  tiles cannot move along diagonals, the distance we will count is the sum of the horizontal and vertical distances. This is sometimes called the city block distance or Manhattan distance. h2 is also admissible because all any move can do is move one tile one step closer to the goal. Tiles 1 to 8 in the start state give a Manhattan distance of
  $h_2 = 3(1) + 1(2) + 2(3) + 2(4) + 2(5) + 3(6) + 3(7) + 2(8) = 18$.

Table of Contents

Exercise 1



- In the 3x3 grid, the robot can take action : left, right, up, and down and will start in location S(1,1) and the goal is location G(3,3). The heuristic of the problem is Manhattan distance $h = |x_2 - x_1| + |y_2 - y_1|$. Using the A* algorithm, determine the solution path and cost to the goal. Show expanded tree.

Exercise 1

	1	2	3
1	S		
2			
3			G

- In the 3x3 grid, the robot can take action : left, right, up, and down and will start in location S(1,1) and the goal is location G(3,3). The heuristic of the problem is Euclidean Distance $h = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$. Using the A* algorithm, determine the solution path and cost to the goal.

Table of Contents

References

- Stuart Russell and Peter Norvig. 2021. Artificial Intelligence: A Modern Approach (4th ed.). Pearson Education Inc., Boston, MA.
- Stuart Russell and Peter Norvig. 2009. Artificial Intelligence: A Modern Approach (3rd ed.). Prentice Hall.
- https://www.cs.cmu.edu/ 15281-s23/coursenotes/search/index.html