Techniques of Artificial Intelligence

Exercises – Search Space

Andrei Covaci & Arthur Moraux

andrei.covaci@vub.be

March 21th, 2022

1. **Search strategies with example from lecture**

   An agent is in Arad, Romania, at the end of a touring holiday. It has a ticket to fly out of Bucharest the following day. The ticket is non-refundable, the agent's visa is about to expire and there are no seats available on flights for six weeks. Our holiday agent has adopted the goal of driving to Bucharest using the following map:

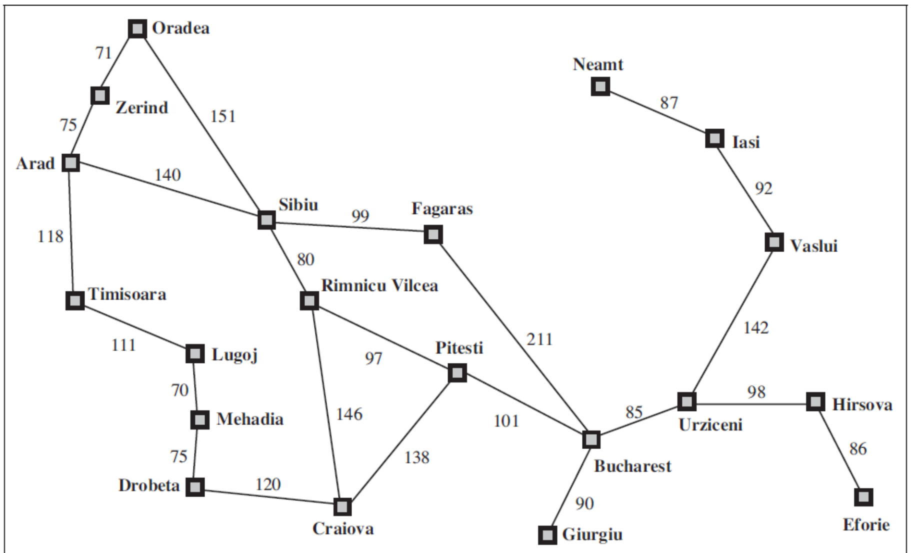


Figure 3.2 A simplified road map of part of Romania.

   Draw the search tree for the following strategies:

   (a) depth first search

   (b) breadth first search

   (c) uniform cost search

   (d) A*

Arad	366	**Mehadia**	241
Bucharest	0	**Neamt**	234
Craiova	160	**Oradea**	380
Drobeta	242	**Pitesti**	100
Eforie	161	**Rimnicu Vilcea**	193
Fagaras	176	**Sibiu**	253
Giurgiu	77	**Timisoara**	329
Hirsova	151	**Urziceni**	80
Iasi	226	**Vaslui**	199
Lugoj	244	**Zerind**	374

Figure 3.22 Values of h_{SLD}—straight-line distances to Bucharest.

2. **General search algorithms**

   add initial state to agenda;
   while *agenda not empty and solution not found* **do**
   | remove the first node N from the agenda;
   | **if** *N is the goal* **then**
   | | stop searching and return the solution;
   | **else**
   | | find all successors of N and add them to the agenda;
   | **end**
   end

   The outline of a general tree search algorithm is sketched above. What data structure must be used for the agenda to turn this outline into:

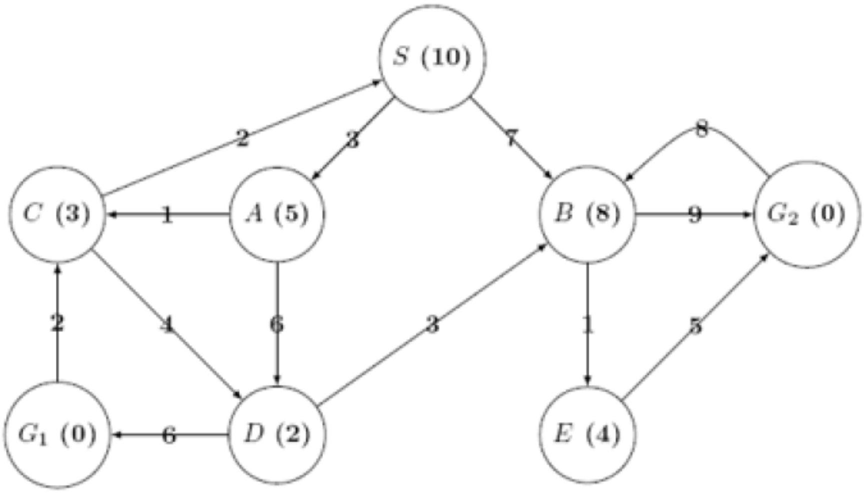
   (a) depth first search
   (b) breadth first search
   (c) Greedy first search
   (d) A* search

   For each of those search algorithms, start from the general algorithm above and add more details to it, such that it becomes the desired search algorithm.

3. **Search spaces**
   Consider the state space illustrated in the figure below.

   (a) S is the initial state
   (b) G1 and G2 are goal states
   (c) Arcs show actions between states (e.g., the successor function for state S returns {A, B}).
   (d) Arcs are labelled with actual cost of the action (e.g., the action from S to A has a cost of 3).
   (e) The numeric value in parentheses in each state is the state's h-value (e.g.,h(A) = 5).

   Give the sequence of states that will be visited, together with the total cost of reaching the goal state, for (1) depth first search, (2) breadth first search and (3) A*. You should assume the following on the operational details of the algorithms:

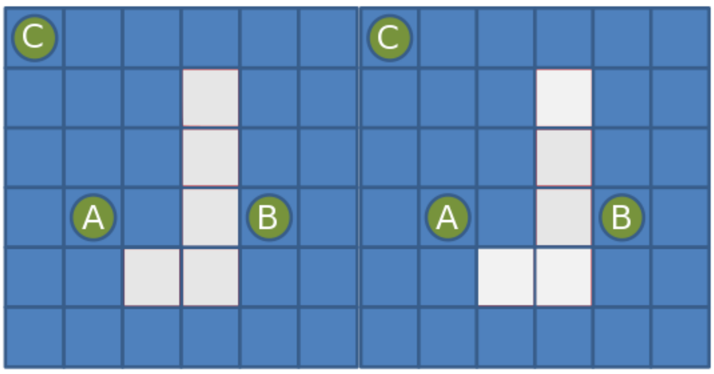


(a) The algorithm does not check if a state is revisited, so there may be several nodes with the same state in the search tree.

(b) The algorithm terminates only when it selects a goal node for expansion, not when it is generated by the successor function.

(c) The successor function always orders states alphabetically.

4. **Path planning**
Imagine a simple block's world as shown below. A block can be moved a single cell at a time in each of the four directions. We have to find the shortest path from configuration A to configuration B.



(a) Discuss whether depth first search and breadth first search will always find a path from A to B. Will they find the shortest path?

(b) Suppose that we want to use informed search in order to guide our search. We will use the Manhattan distance as a search heuristic. Discuss whether best first search will find the shortest path, use the left drawing to add f, g and h costs. Will A* do so? Use the right drawing to add f, g and h costs.

(c) Now suppose that there is a direct underground connection from C to B. Hence, the shortest path from A to B now is over C (it is only five steps). Argue whether A* will find this shortest path.

5. **Farmer, fox, chicken and grain**

   A farmer is on the left bank of a river with a fox, a chicken and a bag of grain, all of which he wishes to get to the right bank. There is a boat that can carry the farmer and at most ONE of those items at a time across the river. The farmer must be careful about what is left behind when he gets into the boat; if the fox is left alone with the chicken, the fox will eat the chicken; similarly, if the chicken is left with the grain, the chicken will eat the grain. Note that the fox will never eat the grain and that neither the fox nor the chicken will eat as long as the farmer is on the same bank as them.

   (a) Formulate the problem as a search problem by specifying an initial state, a general state, the operators, the goal test and the path cost.

   (b) Using your problem formulation, describe and illustrate the operation of the breadth-first and depth-first search algorithms. (Note that you need not conduct the search to completion, only so far as to demonstrate the operation of the algorithms.)

   (c) Which of these algorithms better solves the problem? Justify your answer with reference to space and time complexity, completeness and optimality.

   (d) Would you recommend an alternative search algorithm for either problem? Again, justify your answer.

6. **Sudoku**

   In the sudoku, you need to fill a 9×9 grid with numbers from 1 to 9 such that each number is only present once on each line, column and 3×3 smaller grid. For example starting from the following grid:

   (a) Formulate the problem as a search problem by specifying an initial state, a general state, the operators, the goal test and the path cost.

   (b) How many steps does it take to reach a goal? Is iterative-deepening search a good algorithm for this problem? Why?

   (c) How many moves are possible from the initial state (you can stop counting when you feel it is too much)? Compare to the length of the solutions found in previous question. Would you rather use breadth-first or depth-first search?

   (d) Describe informally how you would solve this problem by hand. Why do you think it is better than depth-first search?

   (e) Formulate your approach as a best-first search. It might require you to choose another path cost function.

7. **Labyrinth**

   This labyrinth is a 10×10 grid with walls placed randomly between adjacent cases. An agent is trying to find the shortest path from the entrance in the upper left to the exit in the lower right.

7			5		8			
		9	7		1	3		
	8					4		1
8				6			1	4
	3	7		1			9	
	9				2	8		
	2		9	5			4	
9		1			7			8
		5	1					2

(a) Formulate the problem as a search problem by specifying an initial state, a general state, the operators, the goal test and the path cost.

(b) Give an estimate of the branching factor of the problem, the number of states and the length of a solution. Would you rather use breadth-first of depth-first search for such an algorithm? Justify your answer. What if the labyrinth gets larger? How about bestfirst search?

(c) Propose an informal heuristic to find the shortest path faster. Formulate it as an admissible heuristic.

(d) Demonstrate the execution of the A* algorithm on this labyrinth. If your heuristic does not work as well as you expected, try to find a better one.

(e) Now consider that the agent is a robot that moves physically inside the labyrinth. It does not know the map of the labyrinth but it can sense its own orientation, which ways are open out of its current case and it knows that the exit is in the lower right corner. The robot wants to reach the exit as fast as possible. What does it change to the problem? Which algorithm would you use?

8. **Local Search**

Give the name of the algorithm which results from:

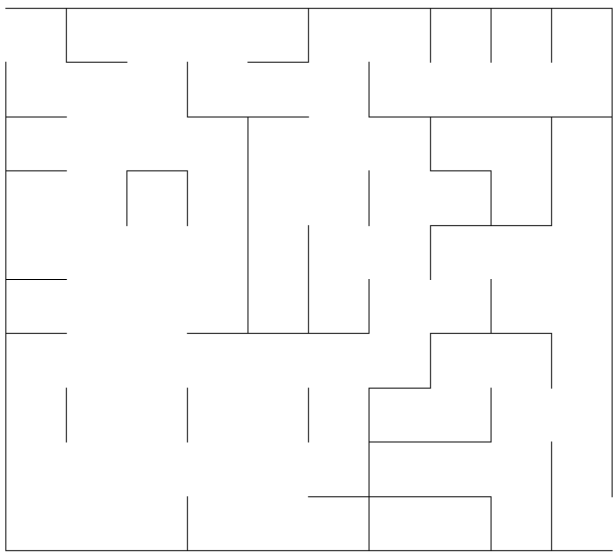
(a) Local beam search with $k = 1$

(b) Local beam search with one initial state and no limit on the number of states retained

(c) Simulated annealing with temperature $Temp = \infty$ at all times

(d) Simulated annealing with $T = 0$ at all times (and omitting the termination test).

(e) Genetic Algorithm with population size $N = 1$



9. **4-queens**

   Consider the initial state of the 4-queens problem shown below.

4	♛	♛	4
♛	5	5	♛
3	3	3	3
2	3	3	2

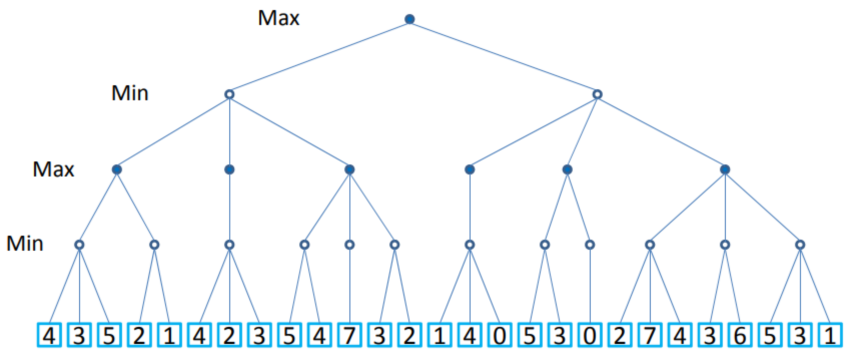
   Consider the hill-climbing local search with number of conflicts heuristic to find a valid state of the n-queens problem. The heuristic value of a state is the number of distinct queen pairs that can attack each other. The successors of a state are all possible states generated by moving a single queen to another square in the same column. Hill-climbing would at each step take the successor with the smallest number of conflicts.

   (a) Draw the tree of states that hill-climbing using the minimum conflict heuristic explores from the initial state. The successors of each state are those states that have least conflicts. Draw the tree corresponding to all possible exploration paths of hill-climbing.

   (b) Are all the leaves in the tree constructed in 1) solutions to the 4-queens problem? If not, how to achieve that all the leaves in the tree represent the solution?

10. **MiniMax**

    Perform the minimax algorithm on the figure below. First without, later with $\alpha\beta$-pruning.


Max
Min
Max
Min
4 3 5 2 1 4 2 3 5 4 7 3 2 1 4 0 5 3 0 2 7 4 3 6 5 3 1