

Techniques of Artificial Intelligence

Exercises – Search Space

Dipankar Sengupta
Dipankar.Sengupta@vub.ac.be

Roxana Rădulescu
Roxana.Radulescu@vub.ac.be

February 15, 2016

1. Fundamentals for search

Suppose you have a data structure D with the push and pop operation. Suppose you have the following sequence of push operations: push(1), push(3), push(2).

- (a) Which elements are returned by two consecutive pop operations if D is a stack, if D is a queue?
- (b) Now suppose that D is a priority queue and that the priority of each element is the value of the element itself. Which elements are returned by two consecutive pop operations now?

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. How can this outline be turned into:

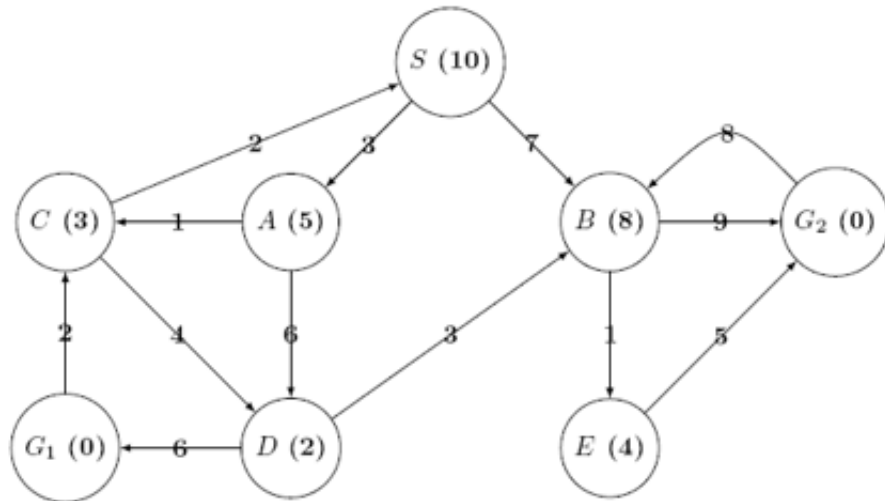
- (a) depth first search
- (b) breadth first search
- (c) best first search
- (d) A* search

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 states 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.