

## Problem 1

(a).

- **State:** Let the 4 gallon jug be  $A$ , the 7 gallon jug be  $B$ . The state can be represented as the amount of water in each jug, namely  $(W(A), W(B))$ .
- **Initial State:**  $(W(A) = 0, W(B) = 0)$
- **Goal Test:** The goal is reached when at least one of the following statement is true:  
 $W(A) = 2, W(B) = 2$
- **Actions:**
  1. Empty jug A:  $W(A) \leftarrow 0$ .
  2. Empty jug B:  $W(B) \leftarrow 0$ .
  3. Fill jug A:  $W(A) \leftarrow 4$ .
  4. Fill jug B:  $W(B) \leftarrow 7$ .
  5. Pour water from A to B.
  6. Pour water from B to A.
- **Cost:** The cost is accumulative along the path. Action 1., 2., 5. and 6. has no cost. The cost of action 3. or 4. is the amount of water that is actually filled.

(b). Figure 1 shows the state diagram of breath first search.  $ea$  and  $eb$  represents emptying jug  $A$  and  $B$ , respectively.  $fa$  and  $fb$  represents filling jug  $A$  and  $B$ , respectively.  $pa$  and  $pb$  represents pouring water from jug  $A$  and jug  $B$  to the other jug, respectively. The all empty state and the all full state are ignored.

(c). In this particular problem, it is optimal. We can verify this by checking the leaf nodes of tree. There are 4 nodes whose cost is smaller than the end node. However, they cannot reach the goal state with lower cost. For example, if we consider the leaf node  $(3, 0)$ ,  $cost : 8$ , we will find state  $(3, 0)$  has appeared before on other branch of the search tree, it will need a  $fb$  which add 7 cost to reach the goal, thus it cannot be better than the result found.

In general, BFS may not be optimal unless the cost of each step is the same.

(d). The naive  $A^*$  search will expand more nodes.

Since the heuristic is always giving 0 estimation, the  $A^*$  will expand all the node on the order of current cost. One observation from Figure 1 is that the end nodes has the larges cost among them. This means that if we use  $A^*$ , more nodes with cost smaller than the end node will be expanded before reaching the end node.

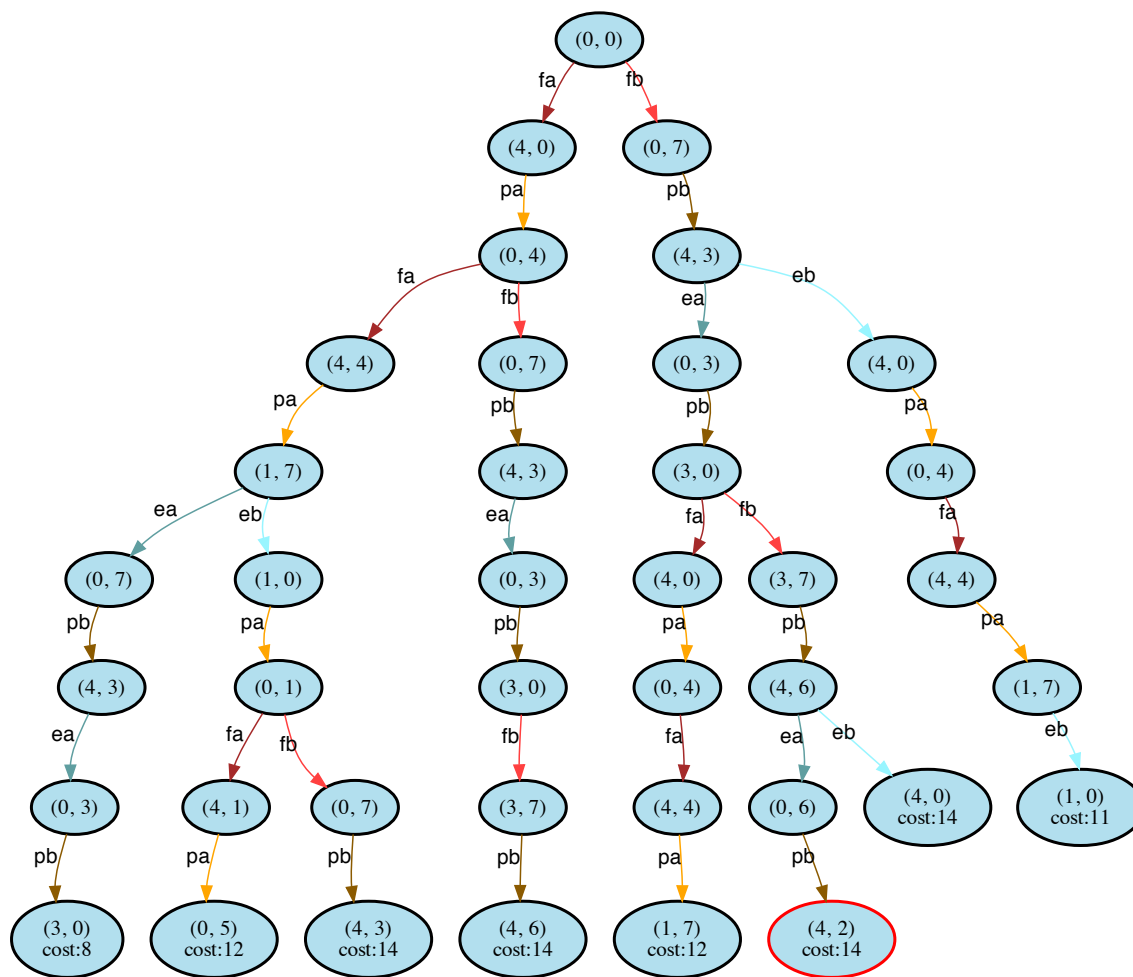


Figure 1: BFS state diagram of problem 1.

## Problem 2

(a). The state can be represented by the locations of the  $n$  vehicles. Since they are non-overlapping, the size of the state space is the number of unique arrangement of the  $n$  vehicles on the grid, which is  $\prod_{i=1}^n (n^2 + 1 - i)$ .

(b). Each vehicle can move 1 in each direction, move 2 in each direction (hop over) or stay. So the branching factor is at most  $9^n$ .

(c). The following is an admissible heuristic.

$$h_i = \frac{|x_i - (n - i - 1)| + |y_i - n|}{2} \quad (1)$$

Firstly, suppose the vehicle moves 1 on each step, then the manhattan distance from its current position to the target is an admissible heuristic. Now the vehicle can at most moves 2 on each step. This at most halves the number of steps.

(d). Suppose  $h_i$  is any admissible heuristic for any  $i$ , then:

1.  $\sum_{i=1}^n h_i$  may not be admissible. Here is a counter example. Let  $n$  be 3, and we use the heuristic in (c). It is easy to calculate that  $h_1 = 2$ ,  $h_2 = 1$  and  $h_3 = 2$ . Thus,  $\sum_{i=1}^n h_i = 5$ . This estimate is not admissible since you can use 3 steps to reach the goal:

- (a) move 1, 2 and 3 up;
- (b) move 1, 2 and 3 up;
- (c) switch 1 and 3.

2.  $\max(h_1, \dots, h_n)$  is admissible. This can be proved by contradiction. Assume  $\max(h_1, \dots, h_n)$  is not admissible, then there must be a vehicle, without the losing generality,  $j$ , which takes more step than  $h_j$  reach its destination. This contradicts with the assumption that  $h_j$  is admissible.
3.  $\min(h_1, \dots, h_n)$  is admissible. We have proved that  $\max(h_1, \dots, h_n)$  is admissible. Since it is always true that  $\min(h_1, \dots, h_n) < \max(h_1, \dots, h_n)$ ,  $\min(h_1, \dots, h_n)$  is admissible.

## Problem 3

## Problem 4