Foundations of Artificial Intelligence

Alberto Maria Metelli

Exercise Session
10-15-2021

Registrations of the Exercise Sessions on WeBeep

Registration Links also available at

https://albertometelli.github.io/teaching/2021-teaching-fai

Exercise 2.2

Suppose you are trying to solve the following puzzle. The puzzle involves **numbers from 100 to 999**. You are given two numbers called **S** and **G**. You are also given a set of numbers called **bad**. **A move consists of transforming one number into another by adding 1 to one of its digits or subtracting 1 from one of its digits**; for instance, a move can take you from 678 to 679; or from 234 to 134. Moves are subject to the following constraints:

- You **cannot add to the digit 9 or subtract from the digit 0**. That is to say, no "carries" are allowed and the digits must remain in the range from 0 to 9.
- You c**annot make a move which transforms your current number into one of the numbers in the set bad**.
- You c**annot change the same digit twice** in two successive moves.

Exercise 2.2

Since the numbers have only 3 digits, there are at most 6 possible moves at the start. And since all moves except the first are preceded by another move which uses one of the digits, after the start there are at most 4 possible moves per turn. You solve the puzzle by getting from S to G in the fewest possible moves. Your task is to use A* search to find a solution to the puzzle.

1. Briefly list the information needed in the state description in order to apply A* to this problem.
2. Find a heuristic for use with A* search in this problem which is admissible and which does not require extensive mathematical calculation. Explain clearly why your heuristic is admissible.
3. Use your heuristic to carry out an A* search to find a solution when S = 567, G = 777, and bad = [666; 667]. For nodes that tie for best-node-to-expand, choose the node with higher path cost.

Solution Proposal - Modelization

State: (xyz,l) – xyz are the three digits and l $\in$ {1,2,3,-} is the last modified digit

Actions: (d,o) - d $\in$ {1,2,3} is the digit to be modified, o $\in$ {+,-} is the performed operation

Initial state: (567, -)

Goal test: (777,l) with l $\in$ {1,2,3,-}

Step cost: 1

Solution Proposal - Heuristic

Idea: sum of absolute differences between the digits of the current state and those of the goal state

If $G=(x_g y_g z_g, l)$ is a goal state:

$$h(xyz,l) = |x - x_g| + |y - y_g| + |z - z_g|$$

Is h admissible?

Solution Proposal - Heuristic

Idea: sum of absolute differences between the digits of the current state and those of the goal state

If $G=(x_g y_g z_g, l)$ is a goal state:

$$h(xyz,l) = |x - x_g| + |y - y_g| + |z - z_g|$$

Is h admissible?

Yes! Because it underestimates the number of moves to reach a goal state.

Is h consistent?

Solution Proposal - Heuristic

Idea: sum of absolute differences between the digits of the current state and those of the goal state

If $G=(x_g y_g z_g, l)$ is a goal state:

$$h(xyz,l) = |x - x_g| + |y - y_g| + |z - z_g|$$

Is h admissible?

Yes! Because it underestimates the number of moves to reach a goal state.

Is h consistent? Try at home!

A*

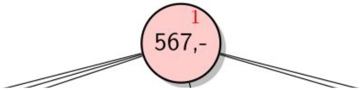
Elimination of Repeated States

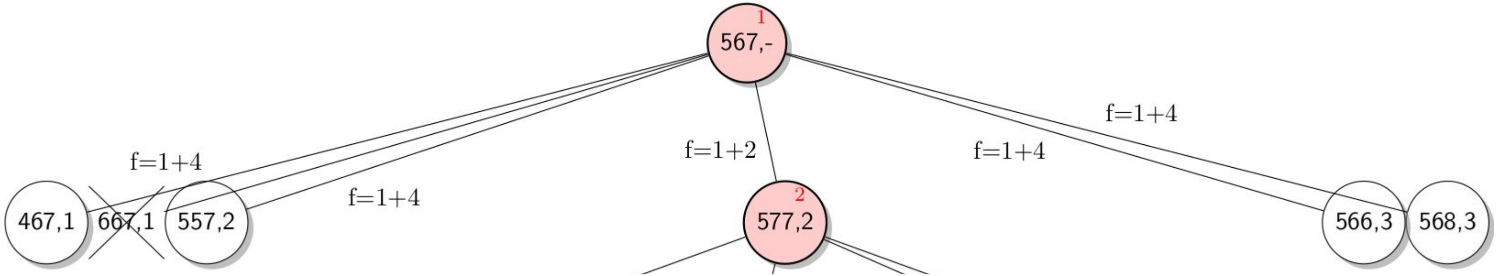
Tie breaking favoring the node with highest path cost

Initial state: (567,-)

Goal states: (777,l) with $l \in \{1,2,3,-\}$

Bad states: {666,667}

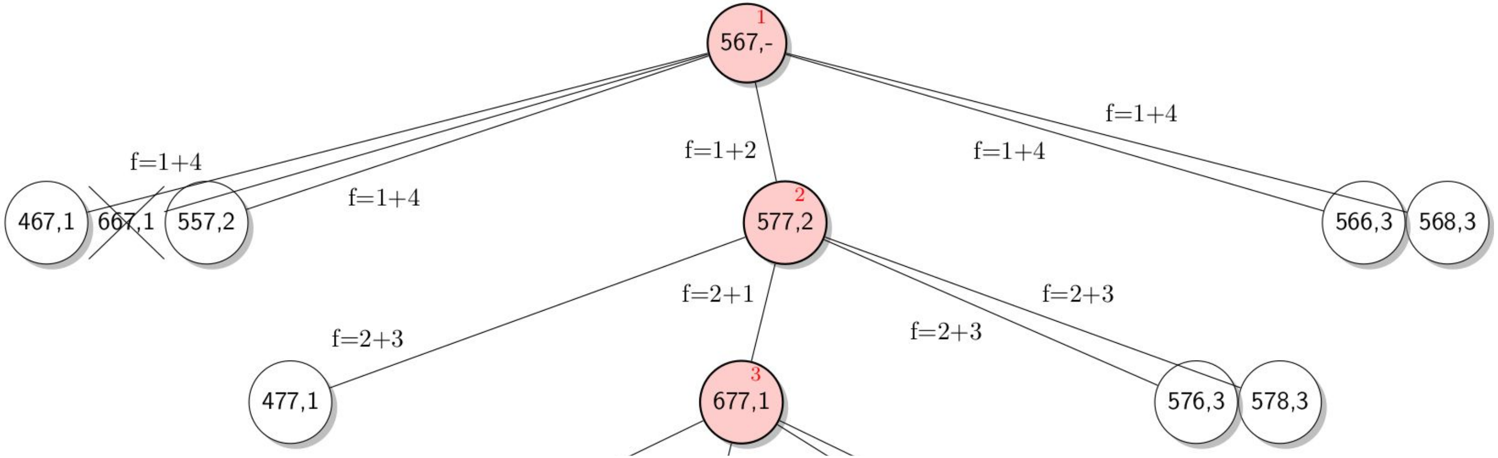

1
567,-

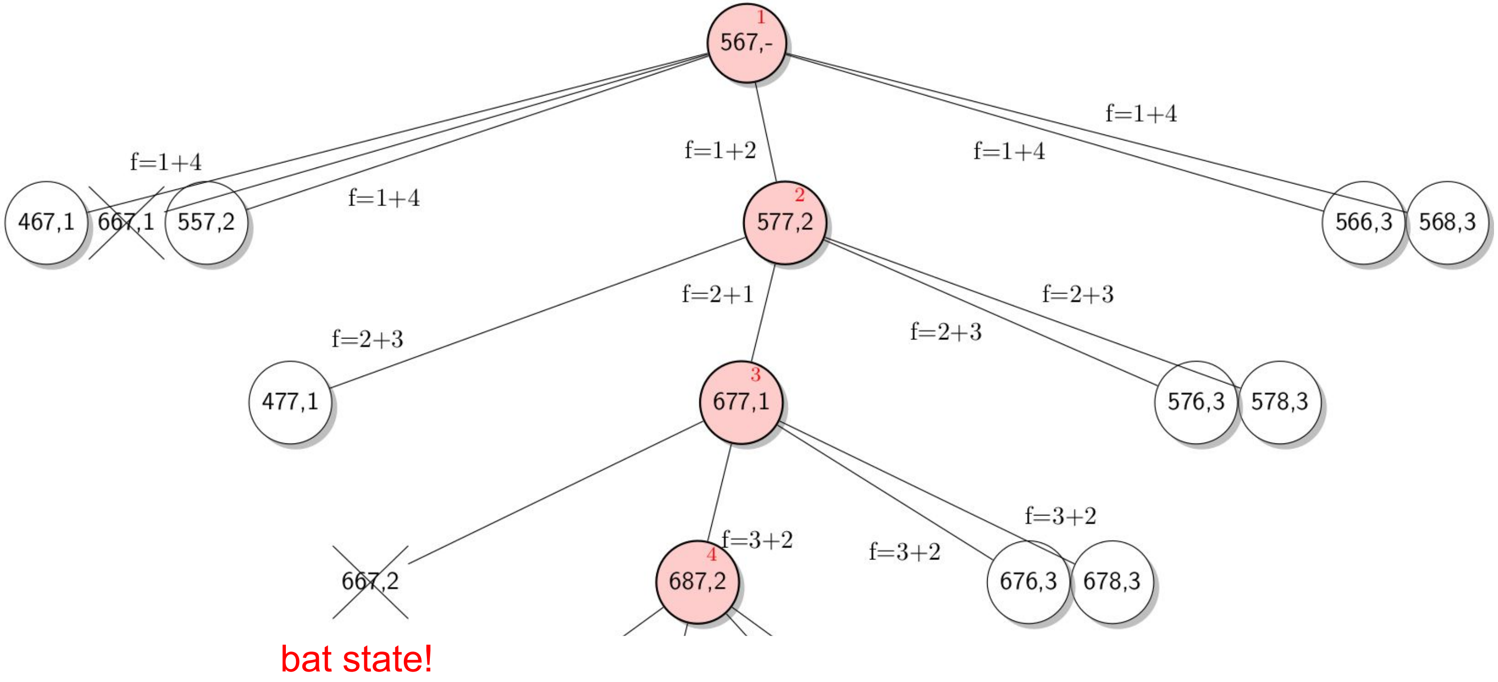


bat state!

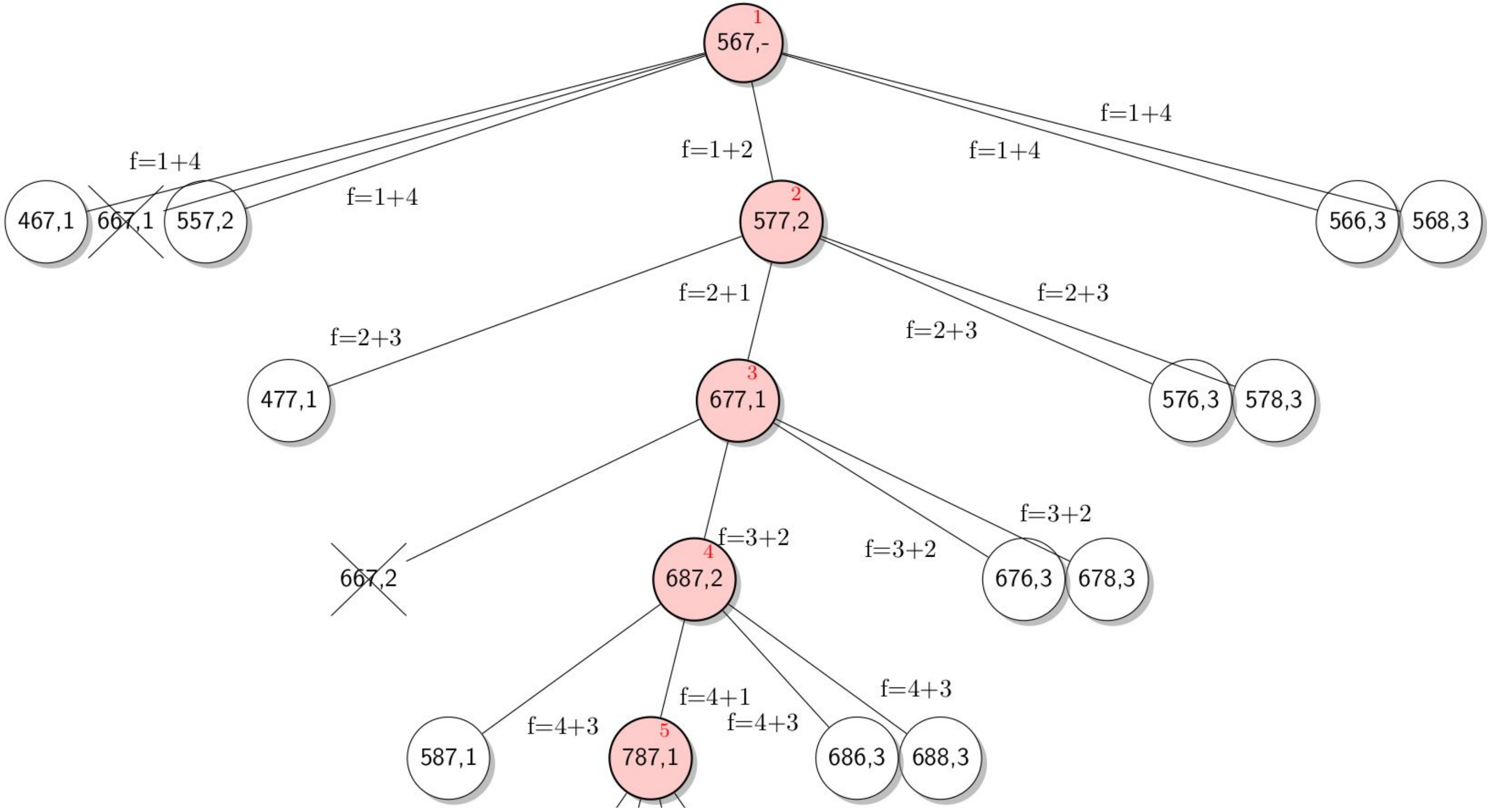
$$f = g + h$$

evaluation function   path cost   heuristic function

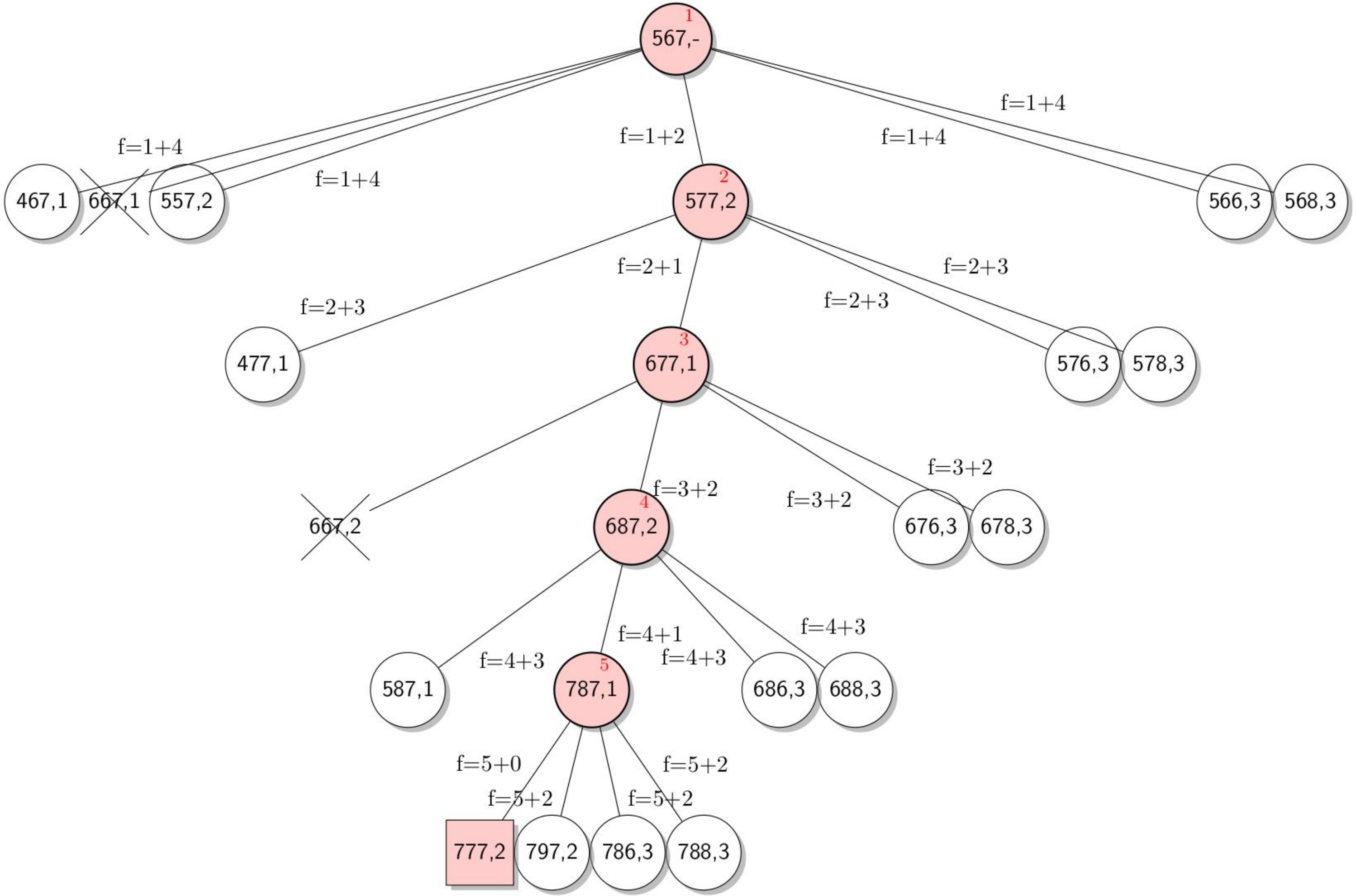

1
567,-
f=1+4
f=1+2
f=1+4
f=1+4
f=1+4
467,1
667,1
557,2
2
577,2
566,3
568,3
f=2+1
f=2+3
f=2+3
f=2+3
477,1
3
677,1
576,3
578,3



Tie breaking favoring the node with highest path cost



1
567,-
f=1+4
f=1+2
f=1+4
f=1+4
f=1+4
467,1
667,1
557,2
2
577,2
566,3
568,3
f=2+1
f=2+3
f=2+3
f=2+3
477,1
3
677,1
576,3
578,3
f=3+2
f=3+2
f=3+2
667,2
4
687,2
676,3
678,3
f=4+3
f=4+1
f=4+3
f=4+3
587,1
5
787,1
686,3
688,3



567,-
1
f=1+4
f=1+4
f=1+2
f=1+4
f=1+4
467,1
667,1
557,2
577,2
2
566,3
568,3
f=2+3
f=2+1
f=2+3
f=2+3
477,1
677,1
3
576,3
578,3
f=3+2
f=3+2
f=3+2
667,2
687,2
4
676,3
678,3
f=4+3
f=4+1
f=4+3
f=4+3
587,1
787,1
5
686,3
688,3
f=5+0
f=5+2
f=5+2
f=5+2
777,2
797,2
786,3
788,3

Exercise 3.3

Consider the following two-player zero-sum game. The game begins with a pile of seven bricks. On your move, you must split one pile of bricks into two piles. You may not split a pile of bricks into two equal piles. If it is your turn and all the piles of bricks have either one or two bricks, you have lost the game.

1. Formalize the problem
2. Apply the minimax algorithm for finding the best action for the max player at the root.
3. Apply the minimax algorithm with alpha-beta pruning for finding the best action for the max player at the root.