

Above are the results from tall the tests being run except for A\* search with the max level and set level heuristic in problem 4.

1. Which algorithm or algorithms would be most appropriate for planning in a very restricted domain (i.e., one that has only a few actions) and needs to operate in real time?

A: If we want to use an algorithm that has a few actions and needs to operate in real time, the best option is a greedy best first search that uses the unmet goals heuristic as it consistently has the lowest ratio or time/actions for each problem. See below.



1. Which algorithm or algorithms would be most appropriate for planning in very large domains (e.g., planning delivery routes for all UPS drivers in the U.S. on a given day)

A: If we look at the ratio of nodes expanded per a given action, a greedy best search first approach with a level sum heuristic seems to be the best option. As can been below for a problem with a low number of actions, a greedy best first search with a level sum heuristic has similar efficiency when compared to a greedy best first search with a set level heuristic, but for more actions, we get better efficiency using a level sum heuristic.



1. Which algorithm or algorithms would be most appropriate for planning problems where it is important to find only optimal plans?

A: If we are only looking for optimal plans, A\* searches are better than greedy best first searches given how it will try to lower the overall cost. Based on the results form the different heuristics with A\* search, it seems like A\* search with the level sum heuristic seems to find the best balance between ratio of nodes expanded given an action and the time it takes per action.

