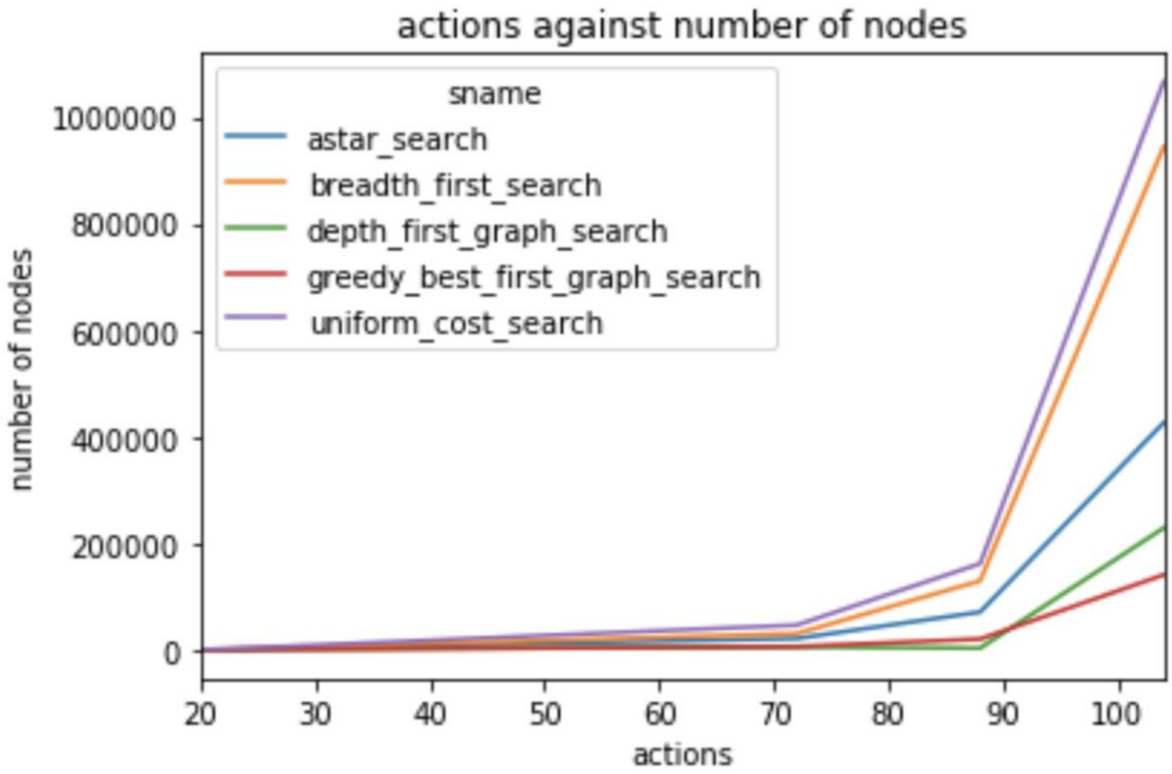
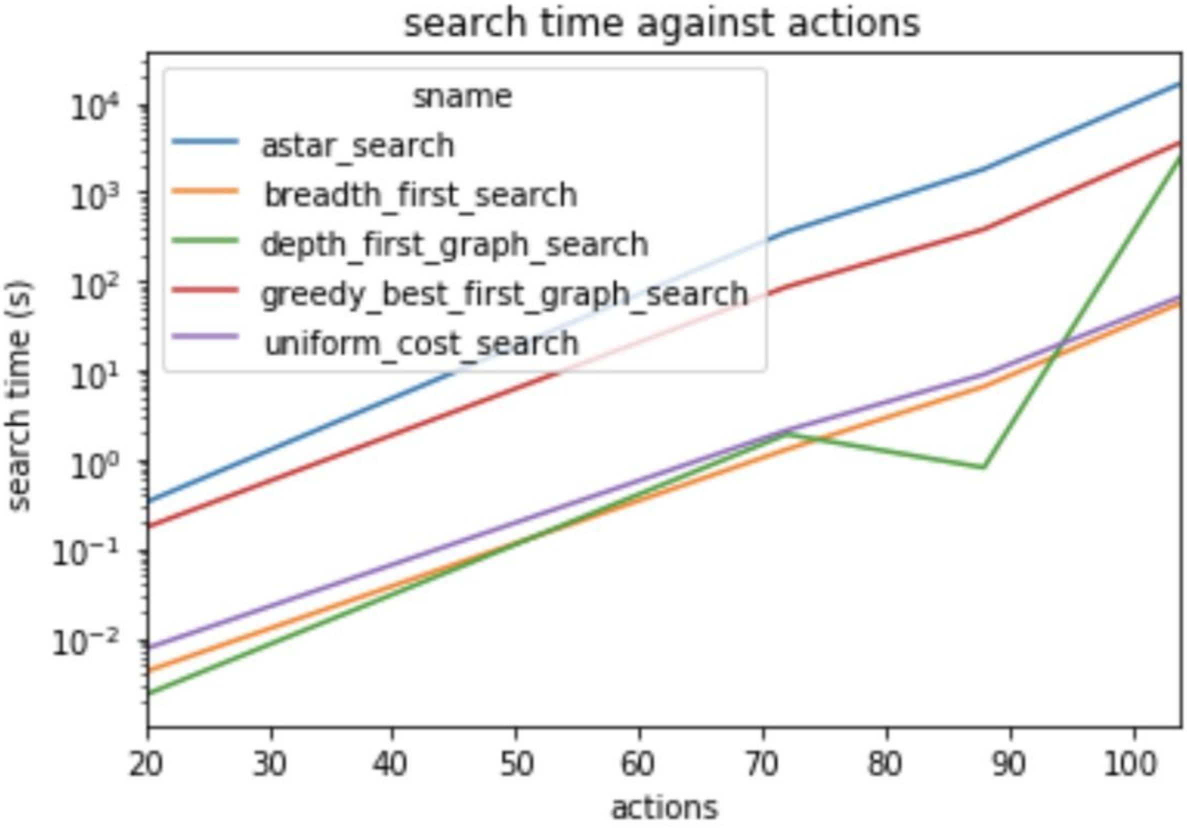
Udacity classical planning Report

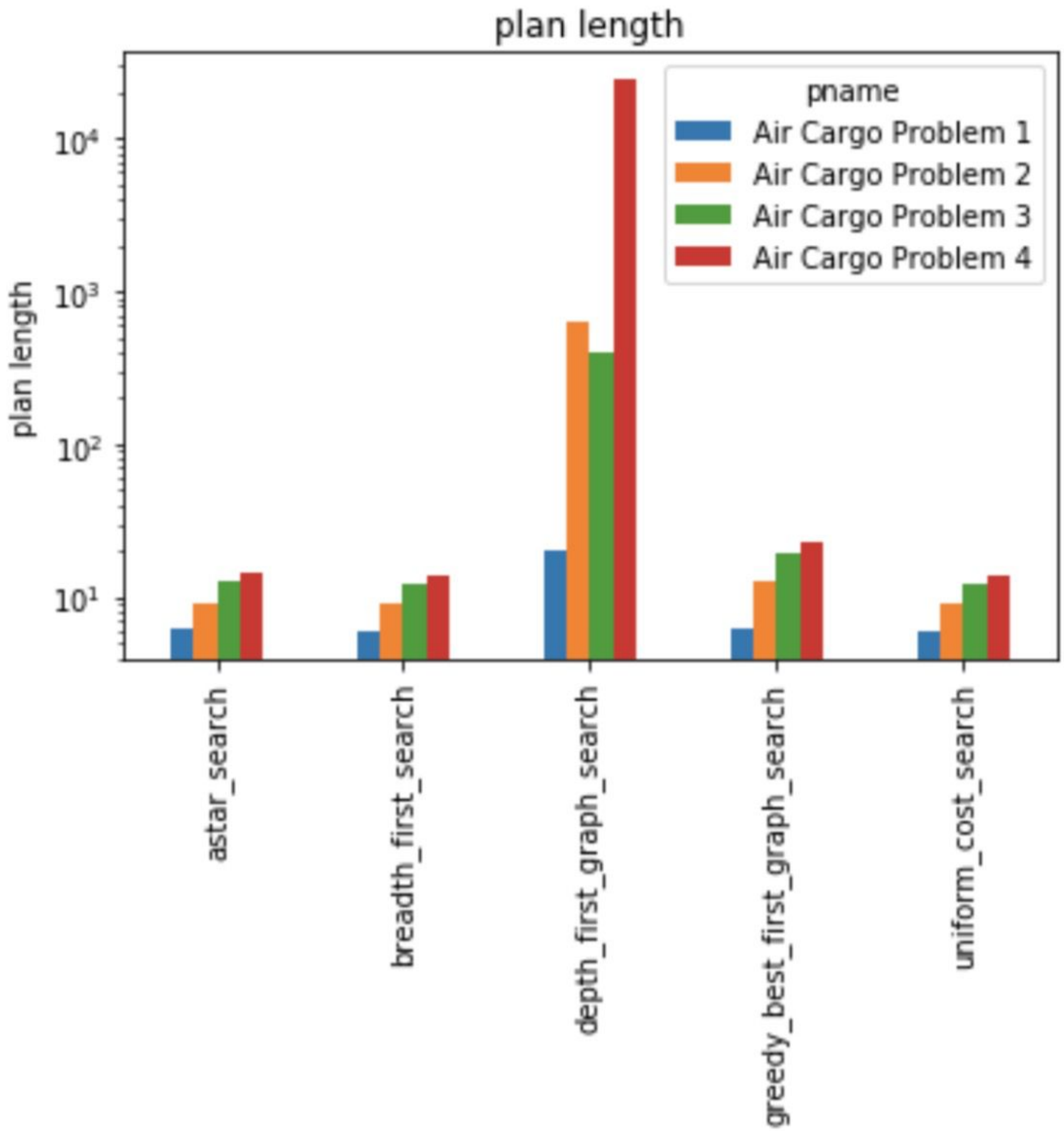
Number of nodes expanded against number of actions in the domain



Search time against the number of actions in the domain



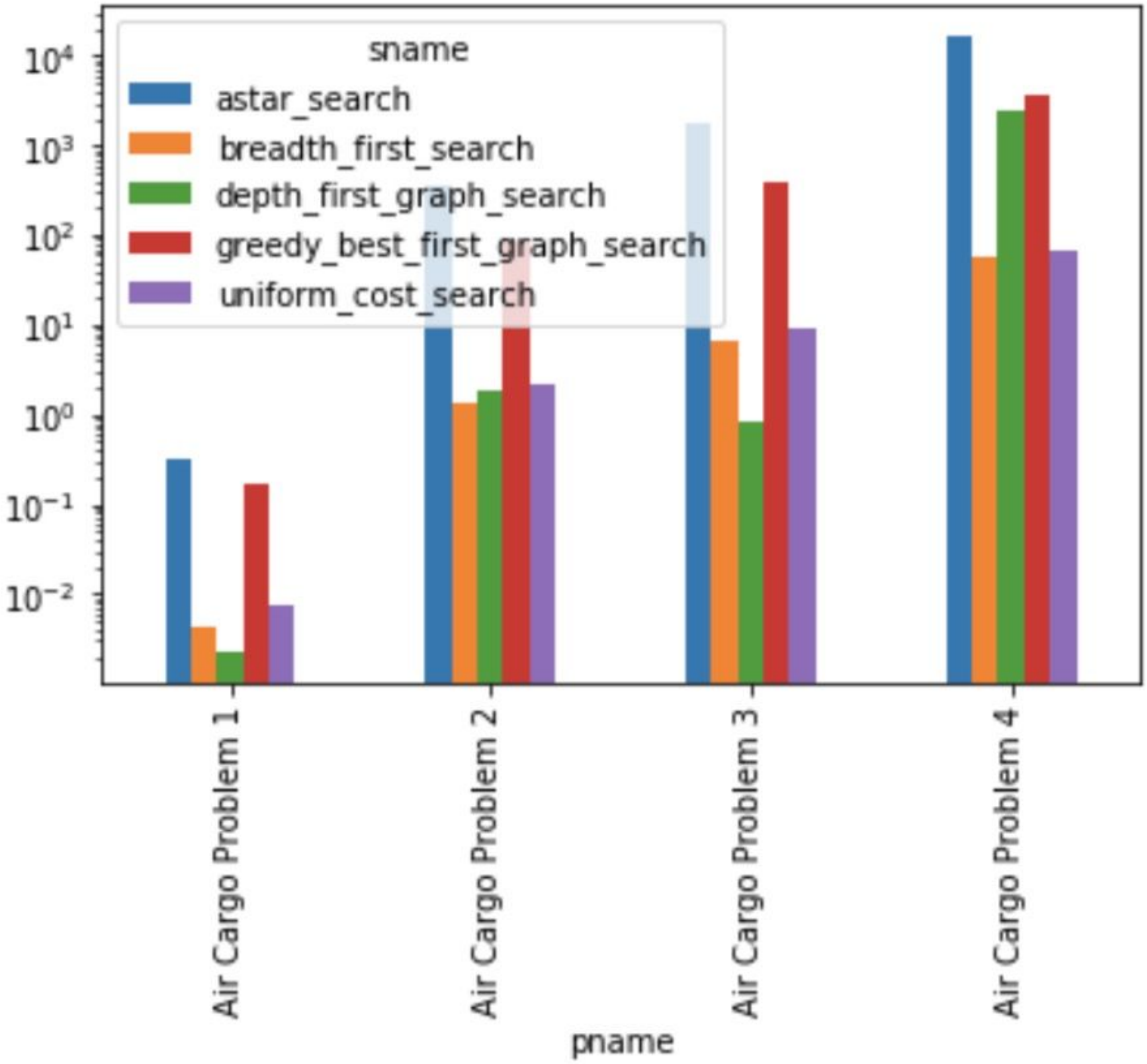
Length of the plans returned by each algorithm on all search problems



Questions:

- 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?
    - Breadth first search has the lowest search time and the shortest plan length while taking up a similar level of number of nodes to all the other algorithms
- 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)
    - In a very large domain, a uniform cost search has the fastest search time while giving the shortest plan length. However, it will require more memory as the number of nodes is significantly higher than most other algorithms.
    - If memory becomes a problem, greedy breadth first graph search may be a better choice although it requires a longer search time.
- Which algorithm or algorithms would be most appropriate for planning problems where it is important to find optimal plans?
    - A_star search, breath_first search, and uniform_cost_search have similar performances

Search time of each algorithm


sname
astar_search
breadth_first_search
depth_first_graph_search
greedy_best_first_graph_search
uniform_cost_search
10^4
10^3
10^2
10^1
10^0
10^{-1}
10^{-2}
Air Cargo Problem 1
Air Cargo Problem 2
Air Cargo Problem 3
Air Cargo Problem 4
pname