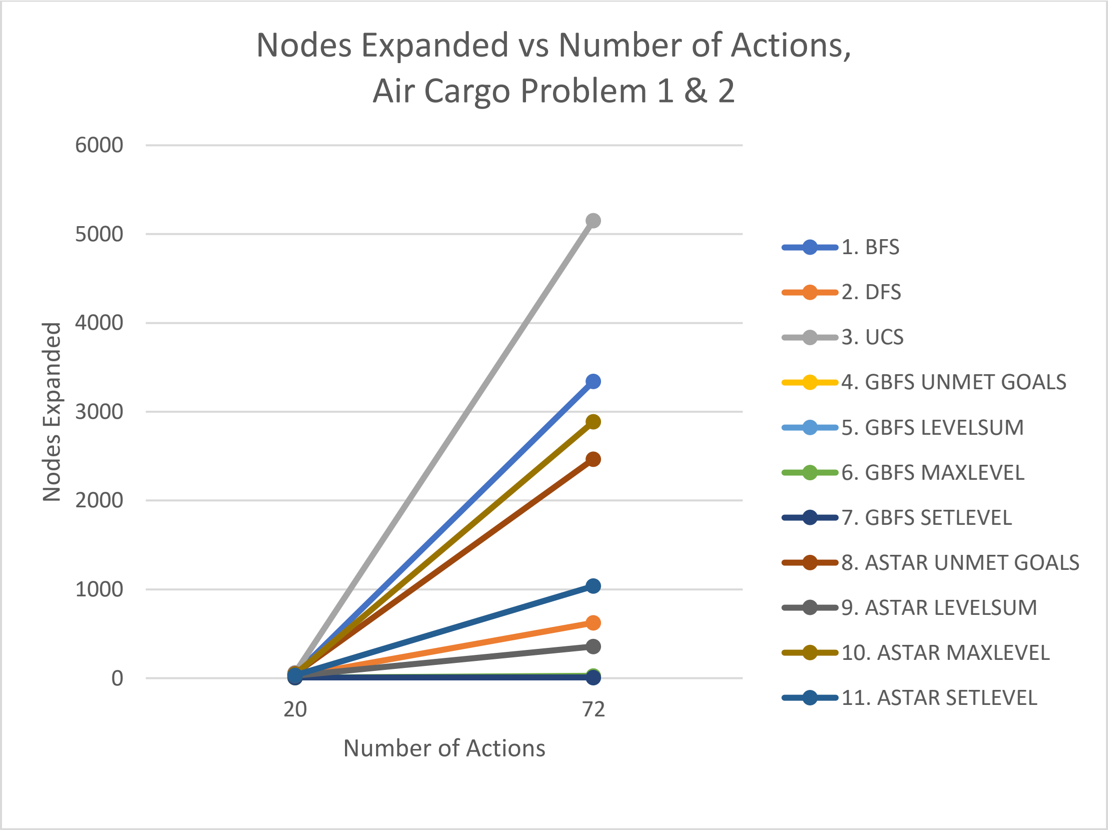
# Build a Forward Planning Agent

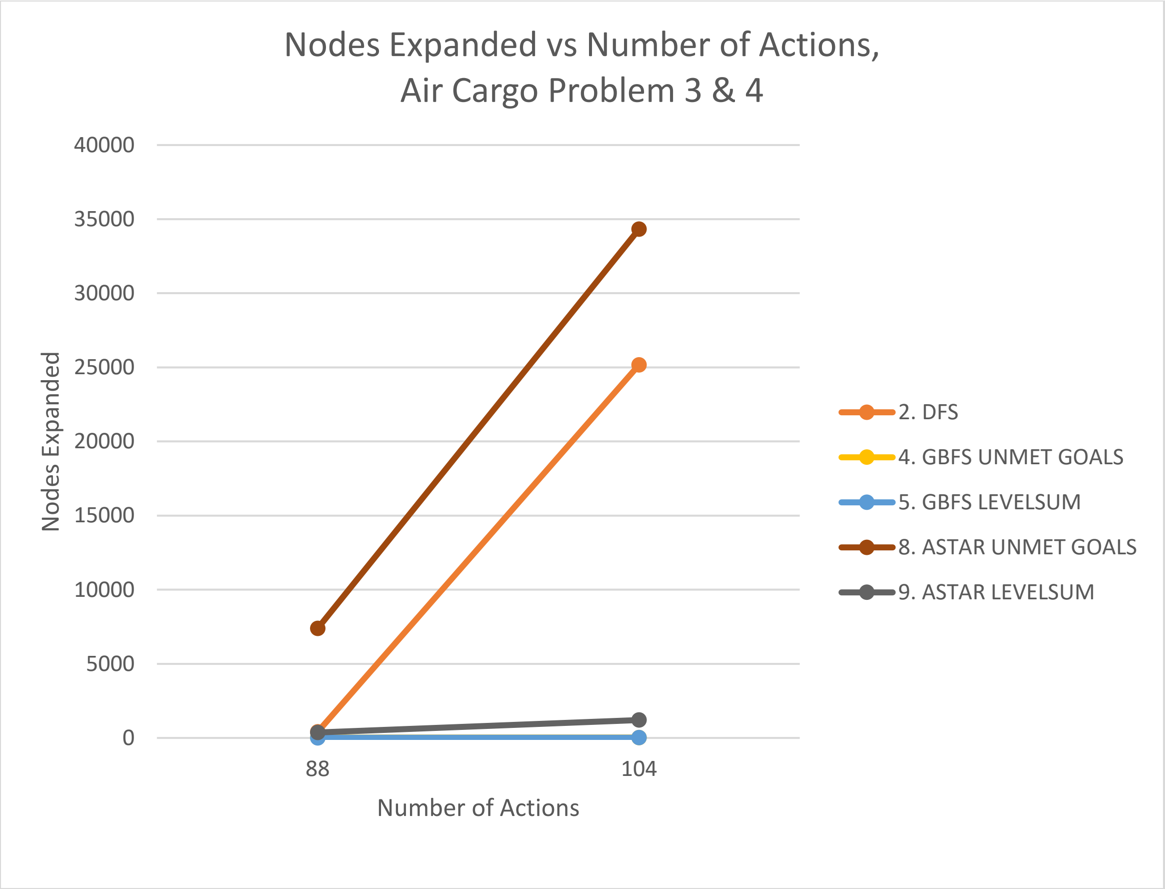
*By Jonfranco Shattuck*

Udacity – Artificial Intelligence Nanodegree

# Search Complexity

| | *AIR CARGO PROBLEM* | *1* | *2* | *3* | *4* |
|---|---|---|---|---|---|
| | **Number of Actions** | **20** | **72** | **88** | **104** |
| **Nodes Expanded** | *1. BFS* | 43 | 3343 | 14663 | 99736 |
| | *2. DFS* | 21 | 624 | 408 | 25174 |
| | *3. UCS* | 60 | 5154 | 18510 | 113339 |
| | *4. GBFS UNMET GOALS* | 7 | 17 | 25 | 29 |
| | *5. GBFS LEVELSUM* | 6 | 9 | 14 | 17 |
| | *6. GBFS MAXLEVEL* | 6 | 27 | 21 | 56 |
| | *7. GBFS SETLEVEL* | 6 | 9 | 35 | 107 |
| | *8. ASTAR UNMET GOALS* | 50 | 2467 | 7388 | 34330 |
| | *9. ASTAR LEVELSUM* | 28 | 357 | 369 | 1208 |
| | *10. ASTAR MAXLEVEL* | 43 | 2887 | 9580 | 62077 |
| | *11. ASTAR SETLEVEL* | 33 | 1037 | 3423 | 22606 |



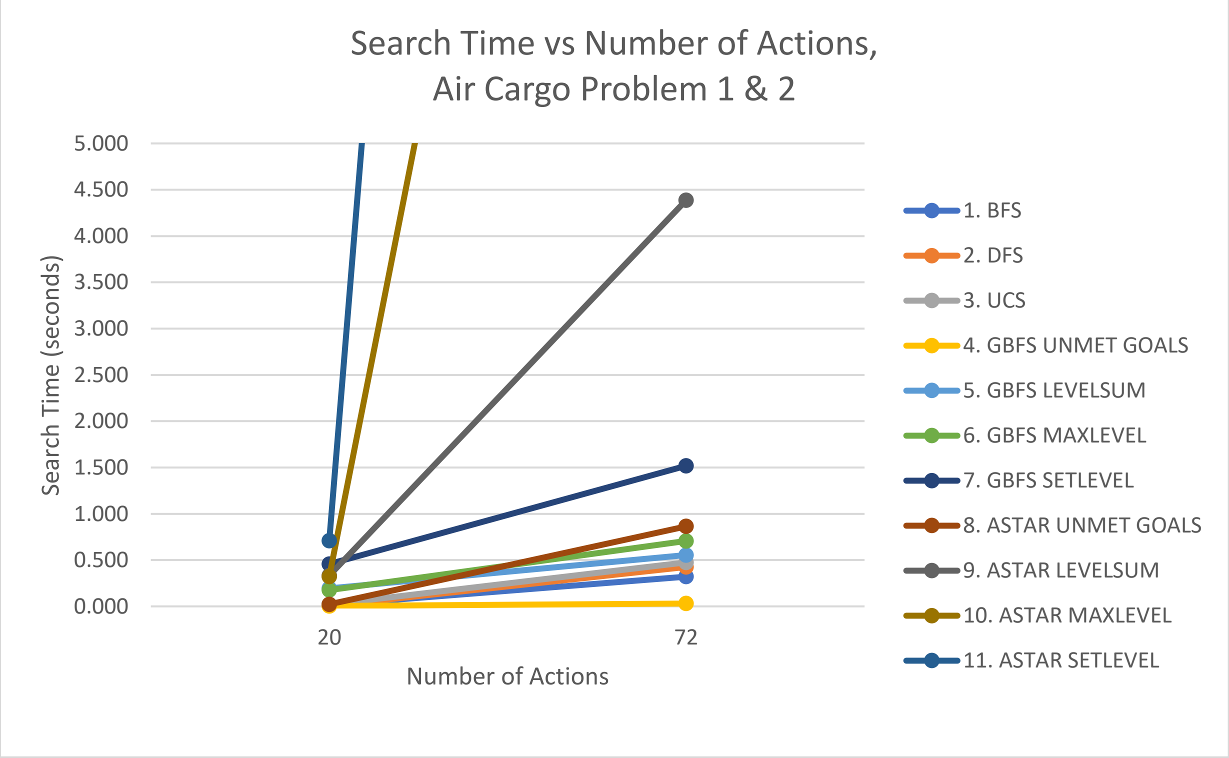


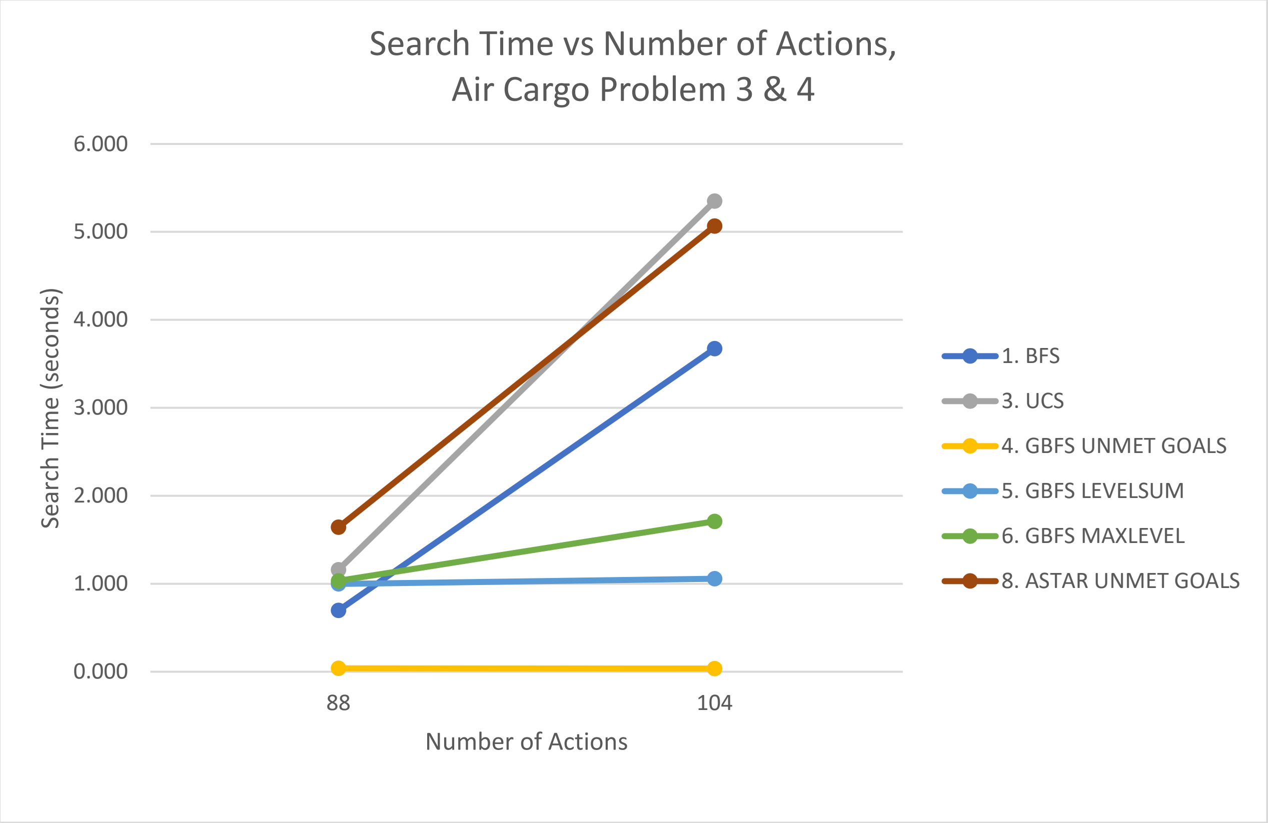
**Analysis:**

The search complexity of the uninformed searches increases drastically as the problem size increases, exponentially for BFS and UCS. The Greedy Best First Searches performed the best across the board in terms of search complexity, some scaling approximately linearly with the problem size. The A-Star searches had a search complexity that performed better than the uninformed searches (at least, BFS and UCS), but still significantly worse than the GBFS.

# Search Time

| | *AIR CARGO PROBLEM* | *1* | *2* | *3* | *4* |
|---|---|---|---|---|---|
| | **Number of Actions** | **20** | **72** | **88** | **104** |
| **Search Time (seconds)** | *1. BFS* | 0.016 | 0.320 | 0.697 | 3.674 |
| | *2. DFS* | 0.006 | 0.425 | 0.191 | 720.470 |
| | *3. UCS* | 0.020 | 0.475 | 1.160 | 5.349 |
| | *4. GBFS UNMET GOALS* | 0.004 | 0.030 | 0.039 | 0.037 |
| | *5. GBFS LEVELSUM* | 0.194 | 0.554 | 0.998 | 1.057 |
| | *6. GBFS MAXLEVEL* | 0.176 | 0.706 | 1.034 | 1.710 |
| | *7. GBFS SETLEVEL* | 0.457 | 1.518 | 4.662 | 14.290 |
| | *8. ASTAR UNMET GOALS* | 0.022 | 0.863 | 1.643 | 5.067 |
| | *9. ASTAR LEVELSUM* | 0.333 | 4.386 | 8.434 | 36.536 |
| | *10. ASTAR MAXLEVEL* | 0.321 | 19.840 | 127.082 | 1250.124 |
| | *11. ASTAR SETLEVEL* | 0.709 | 47.746 | 268.545 | 2621.855 |



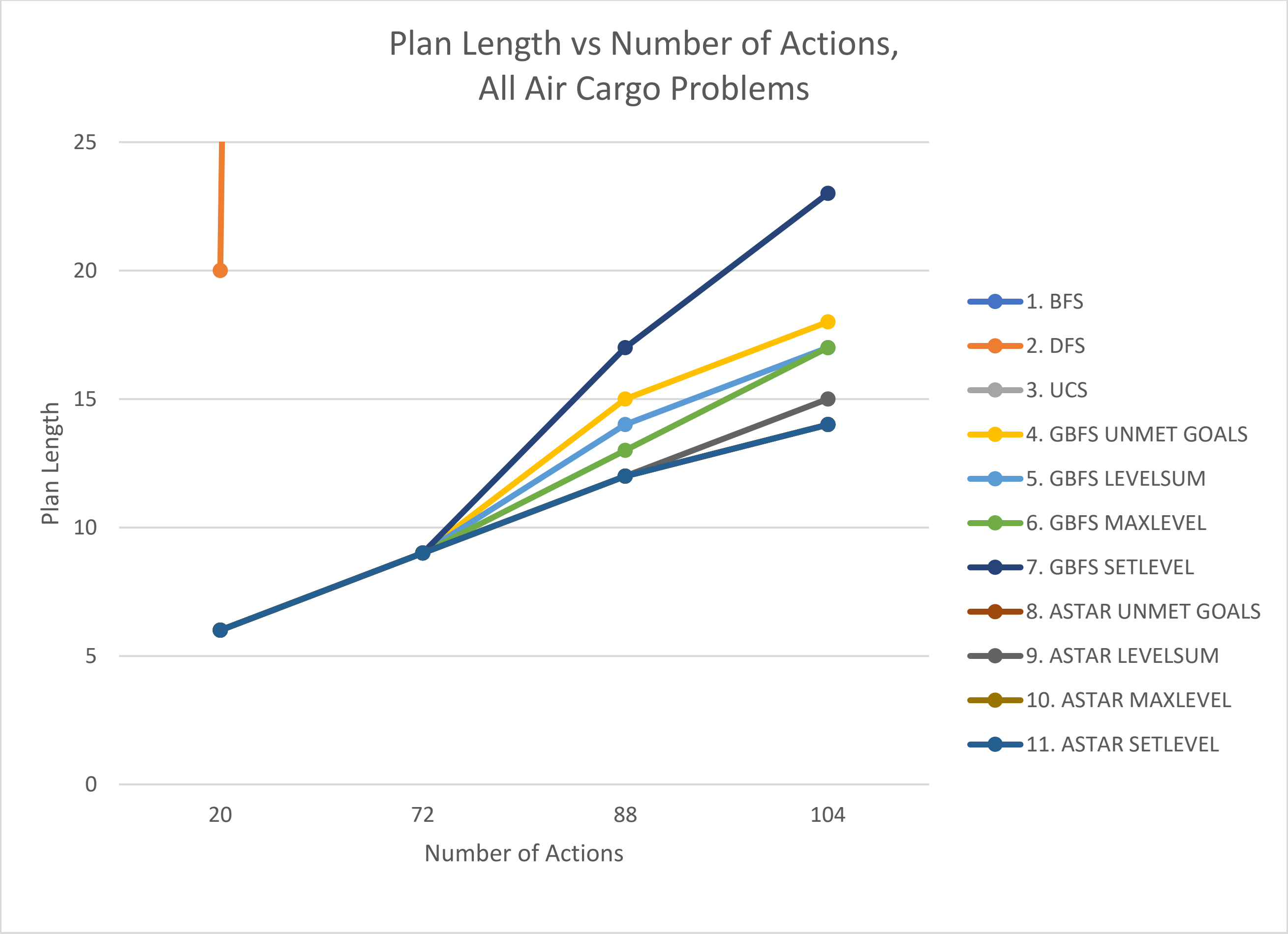


**Analysis:**

The uninformed searches performed similarly to their search complexity in terms of problem scaling, with DFS being the notable exception. DFS has the random chance of executing the search quickly or getting lost in loops of self-defeating decisions which can lead to very long run times. The GBFS had very quick runtimes, with the outlier being the set level heuristic. GBFS Unmet Goals had extremely good runtime performance, remaining almost unchanged as the problem size increased. The A-Star searches performed very differently depending on the heuristic given. The Unmet Goals heuristic performed very well, on par with BFS and UCS, but still worse than most GBFS searches. The Max Level and Set Level heuristics performed quite terribly as the problem size increased, taking some dozens of minutes to complete at worst!

# Optimality

| | *AIR CARGO PROBLEM* | *1* | *2* | *3* | *4* |
|---|---|---|---|---|---|
| | **Number of Actions** | **20** | **72** | **88** | **104** |
| **Optimality (Length of Plan)** | *1. BFS* | 6 | 9 | 12 | 14 |
| | *2. DFS* | 20 | 619 | 392 | 24132 |
| | *3. UCS* | 6 | 9 | 12 | 14 |
| | *4. GBFS UNMET GOALS* | 6 | 9 | 15 | 18 |
| | *5. GBFS LEVELSUM* | 6 | 9 | 14 | 17 |
| | *6. GBFS MAXLEVEL* | 6 | 9 | 13 | 17 |
| | *7. GBFS SETLEVEL* | 6 | 9 | 17 | 23 |
| | *8. ASTAR UNMET GOALS* | 6 | 9 | 12 | 14 |
| | *9. ASTAR LEVELSUM* | 6 | 9 | 12 | 15 |
| | *10. ASTAR MAXLEVEL* | 6 | 9 | 12 | 14 |
| | *11. ASTAR SETLEVEL* | 6 | 9 | 12 | 14 |



# Report 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?**

The Greedy Best First Searches (with heuristics Unmet Goals, Level Sum, and Max Level) perform very well in terms of both search complexity and search time. As long as optimality is not of high preference, these algorithms can quickly find solutions as the algorithm is run repeatedly, with GBFS Unmet Goals notably performing the best.

**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)?**

For very large domains and a scenario where optimality means cost savings, the pros and cons of the GBFS and A-Star searches would have to be compared against each other. If an enormous amount of computing power is available, it might be worth it to attempt a search with A-Star Unmet Goals. It may scale better than BFS would yet would provide close to an optimal solution, which would translate to cost savings. Otherwise, and more likely for a very large domain, GBFS searches would allow for realistic compute times that can be repeated multiple times a day. The exact heuristic chosen would again depend on a preference of optimality vs search time.

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

With only optimality as the requirement, BFS, UCS, and A-Star Unmet Goals would all work while having similar run times. BFS and UCS are guaranteed to find optimal solutions, while A-Star is optimal only if the heuristic meets certain requirements.