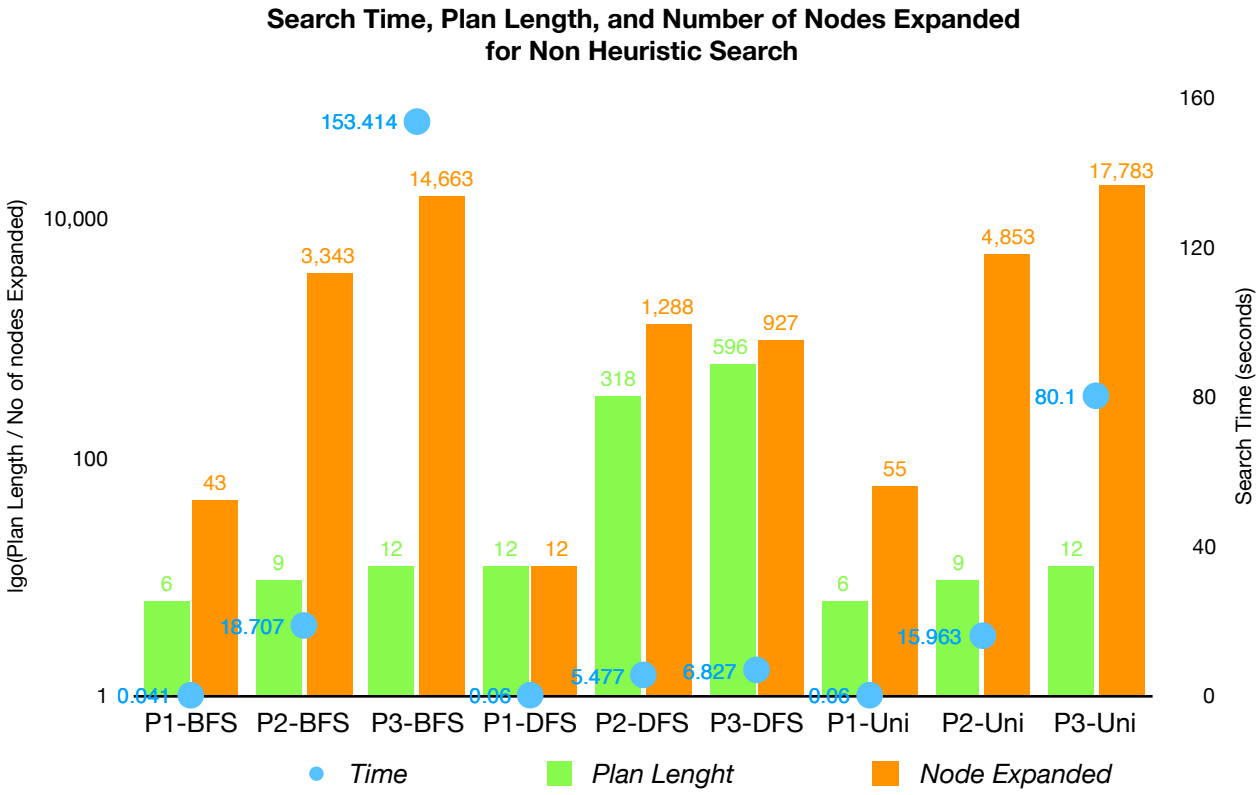
Heuristic Analysis

1. Optimal Plans for Air Cargo Problems

The optimal plans for each air cargo problem are presented in the table below.

Problems	Plan Length	Steps
Problem 1	6	Load(C1, P1, SFO) Fly(P1, SFO, JFK) Load(C2, P2, JFK) Fly(P2, JFK, SFO) Unload(C1, P1, JFK) Unload(C2, P2, SFO)
Problem 2	9	Load(C1, P1, SFO) Load(C2, P2, JFK) Load(C3, P3, ATL) Fly(P1, SFO, JFK) Fly(P2, JFK, SFO) Fly(P3, ATL, SFO) Unload(C3, P3, SFO) Unload(C1, P1, JFK) Unload(C2, P2, SFO)
Problem 3	12	Load(C1, P1, SFO) Fly(P1, SFO, ATL) Load(C3, P1, ATL) Fly(P1, ATL, JFK) Load(C2, P2, JFK) Fly(P2, JFK, ORD) Load(C4, P2, ORD) Fly(P2, ORD, SFO) Unload(C4, P2, SFO) Unload(C3, P1, JFK) Unload(C2, P2, SFO) Unload(C1, P1, JFK)

2. Non-Heuristic Searches Analysis



The Results of problem 2 and 3 with *Breath-First Tree Search*, *Depth Limit Search*, and *Recursive Best Search* had not been completed because they took longer than 10 minutes to complete.

Non-Heuristic Breath First Search (BFS) and Non-Heuristic Uniform Cost Search (Uni) are able to find optimal plan for each problem when the other search, a *Non-Heuristic Depth First Search (DFS),* is not. Note that DFS gave the longest plan length for each problem.

In term of time taken to complete all problems, DFS is the fastest search and BFS was the slowest to complete.

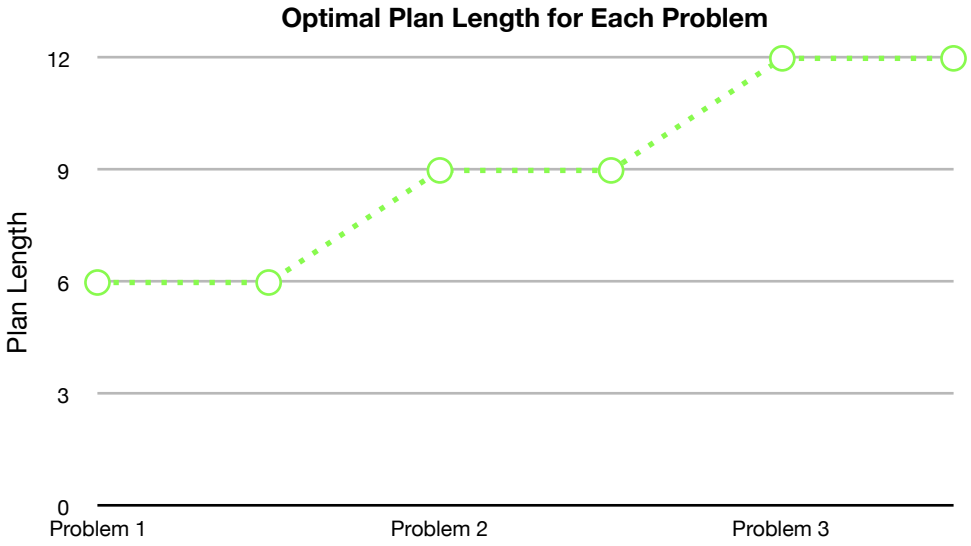
The results above showed the nature of each search. Firstly, BFS, which expands all the nodes at the same depth first, expanded each action to see whether goal can be completed. All possible actions were expanded and this reflected in large number of expanded node in problem 3.

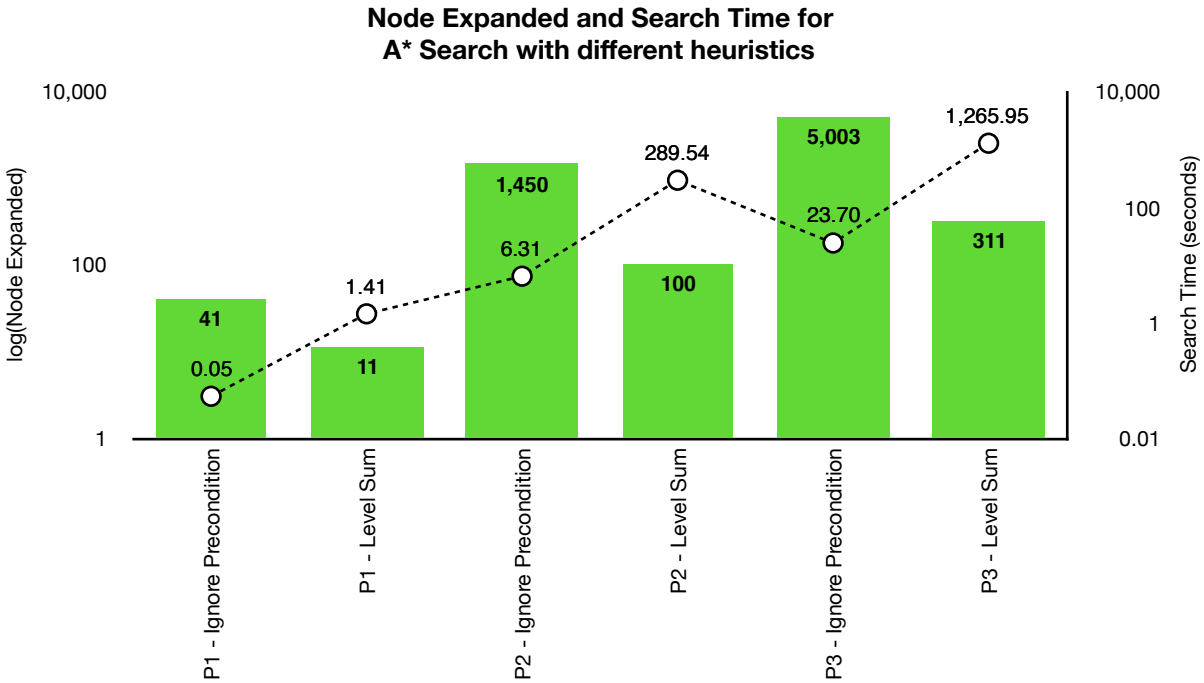
Secondly, DFS expands to the deepest need in the current frontier of search tree. However, the result also how DFS normally fails. It returned plan with the longest length for each problem because it gave the same action repeatedly while was expanding deeper nodes in the same frontier.

Lastly, Uniform Cost Search expands the node which cost is lowest. In the result, Uniform Cost Search expanded more node than BFS but took less time to completed all goals foe all problems.

3. A* Search with Different Heuristic Analysis

A* search with two different heuristics, which are *Ignore Precondition* and *Level Sum* were implemented. The results are presented below:





A* search delivered same optimality for all problems (Plan Length of 6, 9, and 12 for problem 1, 2 and 3 respectively).

However, a search time of A* search with *ignore precondition* heuristic was 0.05 seconds for problem 1, 6.31 seconds for problem 2, and 23.70 seconds for problem 3. In contrast, A* search took dramatically longer to finished its search with *level sum* heuristic, which was 1.14 seconds, 289.54 seconds, and 1,265.95 seconds.

On the other hand, number of nodes expanded were largely reduced when level sum heuristic was applied to A* search. For example in problem 3, A* search with *ignore precondition* heuristics expanded 5,003 nodes to completed search goals but it expanded only 311 nodes when *level sum* heuristic was applied.

Both heuristics search problems to be relaxed problem which helps A* search find optimal solution. The *ignore precondition* heuristic drops all preconditions for current action in each state. Thus A* search looks only for number of satisfied goals and this results in more nodes are expanded to complete search goal.

In contrast, *level sum* heuristics counts all goals in current state and find all actions need to construct plan to achieve goals. Therefore, this heuristic explores less node but takes longer time to complete.