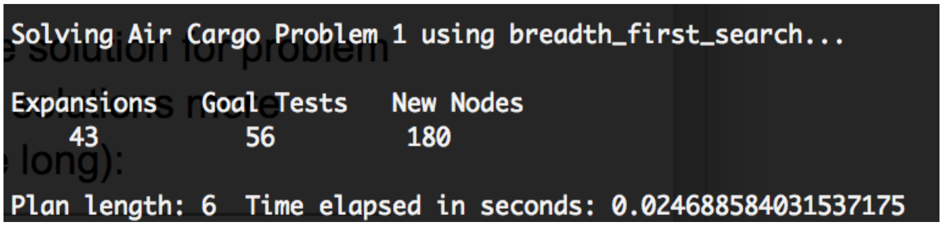
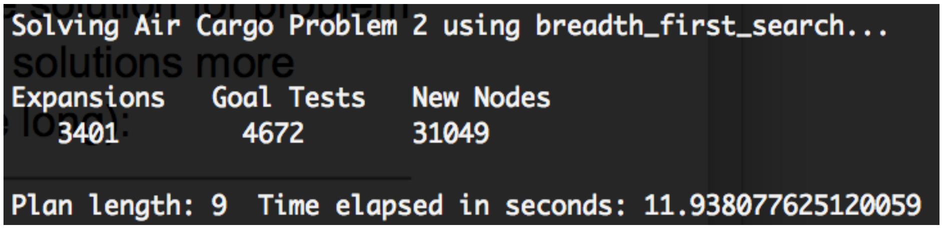
Heuristic Analysis

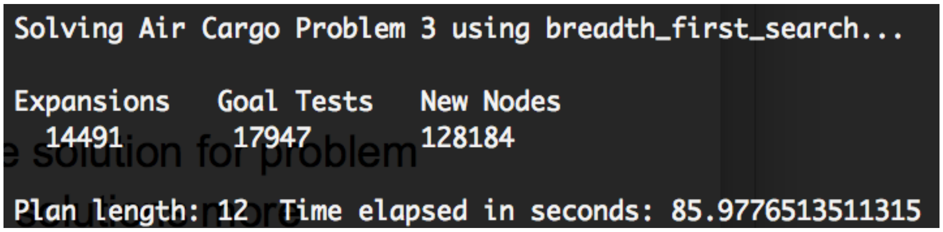
Since the breadth-first search will always provide a solution with minimal plan length. I began analysis by determining the shortest possible plan lengths per problem and ran the BFS first:



```
Solving Air Cargo Problem 1 using breadth_first_search...

Expansions   Goal Tests   New Nodes
    43          56          180

Plan length: 6  Time elapsed in seconds: 0.024688584031537175
```

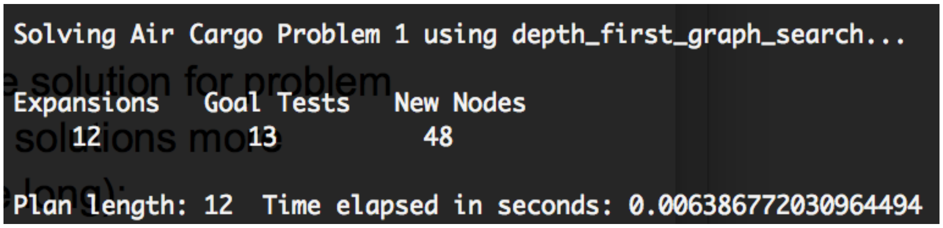


```
Solving Air Cargo Problem 2 using breadth_first_search...

Expansions    Goal Tests    New Nodes
   3401         4672        31049

Plan length: 9  Time elapsed in seconds: 11.938077625120059
```

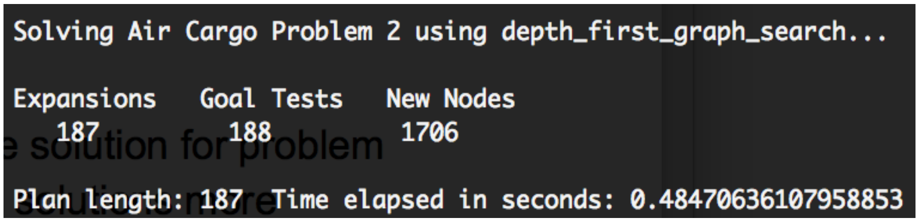


```
Solving Air Cargo Problem 3 using breadth_first_search...

Expansions    Goal Tests    New Nodes
  14491        17947        128184

Plan length: 12  Time elapsed in seconds: 85.9776513511315
```

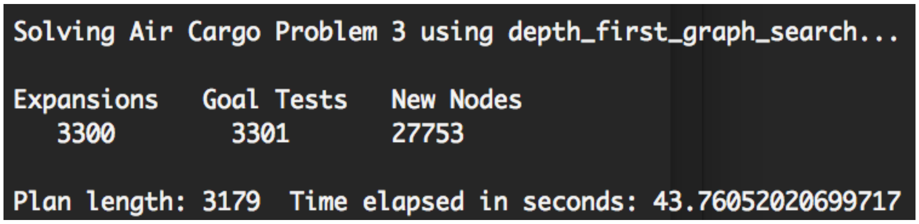
However, I found that it takes quite a long time to arrive at the solution for problem 3 with BFS. I next took a look at depth-first search thinking. While, it might provide solutions more quickly, but longer plans



```
Solving Air Cargo Problem 1 using depth_first_graph_search...

Expansions    Goal Tests    New Nodes
    12           13            48

Plan length: 12  Time elapsed in seconds: 0.006386772030964494
```



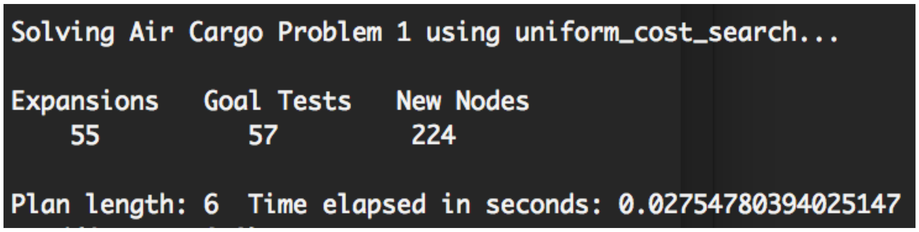
```
Solving Air Cargo Problem 2 using depth_first_graph_search...

Expansions   Goal Tests   New Nodes
   187          188         1706

Plan length: 187  Time elapsed in seconds: 0.48470636107958853
```



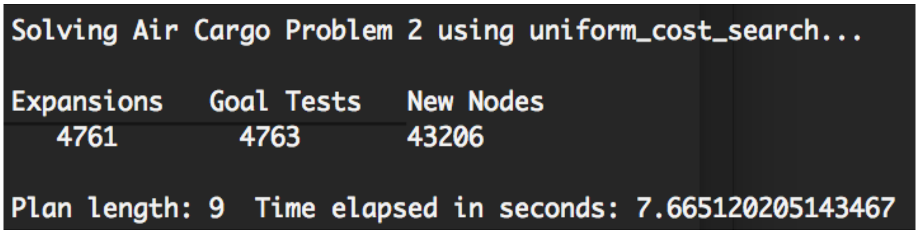
```
Solving Air Cargo Problem 3 using depth_first_graph_search...

Expansions   Goal Tests   New Nodes
   3300        3301       27753

Plan length: 3179  Time elapsed in seconds: 43.76052020699717
```

DFS did indeed provide answers more quickly, however the plan lengths are extremely hugme.

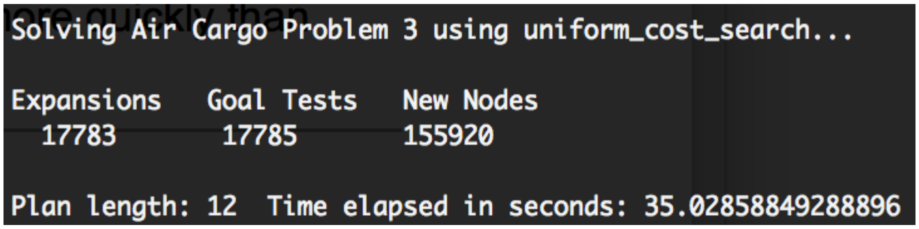
I next ran uniform cost search to see if it could provide an optimal solution more quickly than BFS:



```
Solving Air Cargo Problem 1 using uniform_cost_search...

Expansions   Goal Tests   New Nodes
    55           57          224

Plan length: 6  Time elapsed in seconds: 0.02754780394025147
```



```
Solving Air Cargo Problem 2 using uniform_cost_search...

Expansions   Goal Tests   New Nodes
   4761         4763       43206

Plan length: 9  Time elapsed in seconds: 7.665120205143467
```

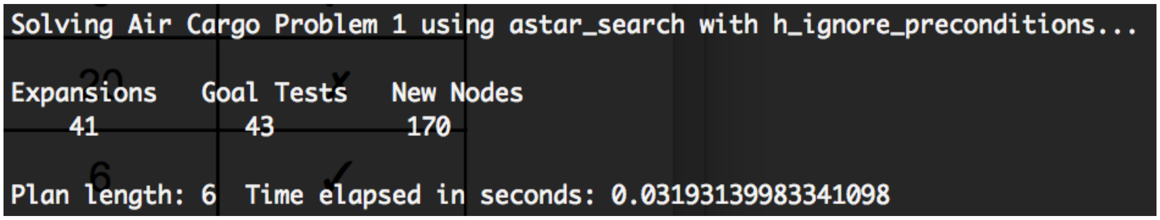


```
Solving Air Cargo Problem 3 using uniform_cost_search...

Expansions   Goal Tests   New Nodes
  17783        17785       155920

Plan length: 12  Time elapsed in seconds: 35.02858849288896
```

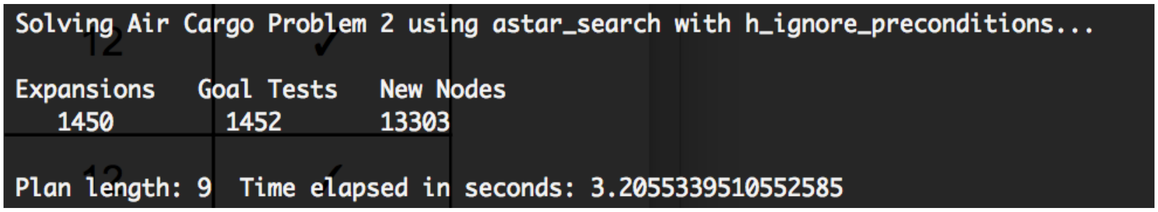
Overall, uniform cost search performed the best out of the 3 uninformed search algorithms I attempted- it found optimal solutions pretty quickly for all 3 problems:

	Number	Node	Goal	Time	Plan Length	Optimal Plan
BFS	1	43	56	0.02	6	Yes
DFS	1	12	12	0.006	12	No
Uniform	1	55	57	0.027	6	Yes
BFS	2	3401	4672	11.9	9	Yes
DFS	2	187	188	0.48	187	No
Uniform	2	4761	4763	7.67	9	Yes
BFS	3	14491	17947	86.00	12	Yes
DFS	3	3300	3301	43.76	3179	No
Uniform	3	17783	17785	35.03	12	Yes

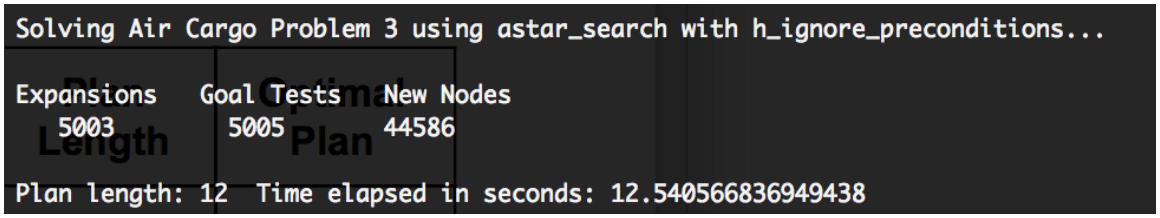
Next, I ran A* with the ignore ignore-preconditions heuristic:



```
Solving Air Cargo Problem 1 using astar_search with h_ignore_preconditions...

Expansions   Goal Tests   New Nodes
    41          43           170

Plan length: 6  Time elapsed in seconds: 0.03193139983341098
```

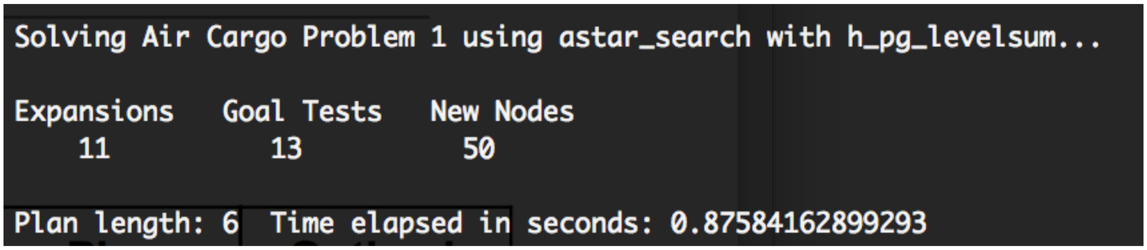


```
Solving Air Cargo Problem 2 using astar_search with h_ignore_preconditions...

Expansions   Goal Tests   New Nodes
   1450        1452       13303

Plan length: 9  Time elapsed in seconds: 3.2055339510552585
```

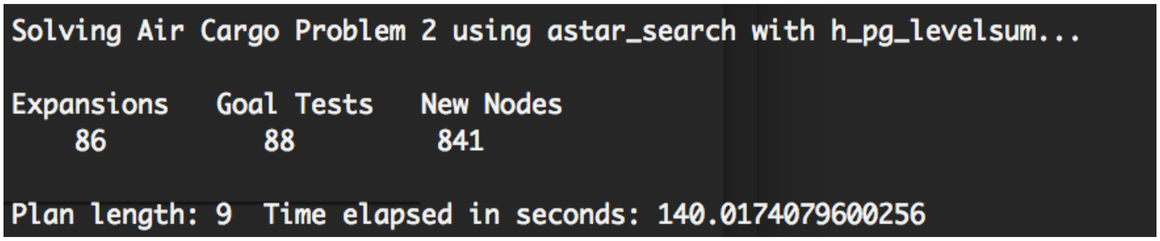


```
Solving Air Cargo Problem 3 using astar_search with h_ignore_preconditions...

Expansions   Goal Tests   New Nodes
   5003        5005       44586

Plan length: 12  Time elapsed in seconds: 12.540566836949438
```

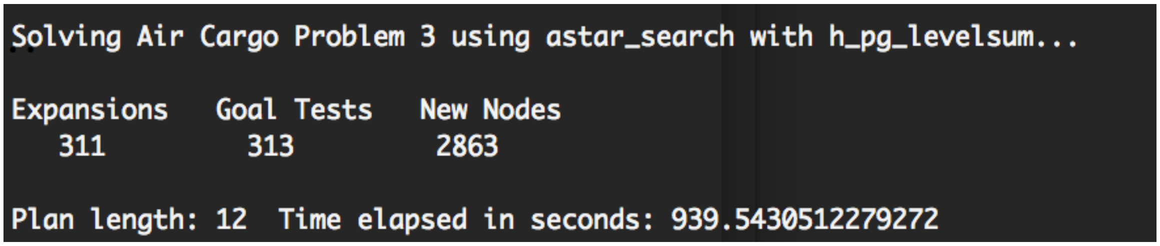
And then A* with the level-sum heuristic:



```
Solving Air Cargo Problem 1 using astar_search with h_pg_levelsum...

Expansions   Goal Tests   New Nodes
    11           13          50

Plan length: 6  Time elapsed in seconds: 0.87584162899293
```



```
Solving Air Cargo Problem 2 using astar_search with h_pg_levelsum...

Expansions   Goal Tests   New Nodes
    86           88         841

Plan length: 9  Time elapsed in seconds: 140.0174079600256
```



```
Solving Air Cargo Problem 3 using astar_search with h_pg_levelsum...

Expansions   Goal Tests   New Nodes
   311          313         2863

Plan length: 12  Time elapsed in seconds: 939.5430512279272
```

We can see the results for the different heuristics here:

	Number	Node	Goal	Time	Plan length	Optimal Plan
ignore-preconditions	1	41	43	0.03	6	Yes
level-sum	1	11	13	0.88	6	Yes
ignore-preconditions	2	1450	1452	3.21	9	Yes
level-sum	2	86	88	140.02	9	Yes
ignore-preconditions	3	5003	5005	12.54	12	Yes
level-sum	3	311	313	939.54	12	Yes

Overall, the levelsum was quite slow to perform due to the heavy cost of the calculation for the hueristic- but it was able to minimize the number of nodes that had to be examined suggesting that the heuristic was very good.

Preconditions was a great balance in that the cost of calculating the heuristic was quite low and it did manage to lead the search generally in a good direction, allowing it to perform faster than DFS while providing an optimal solution.

Ignoring preconditions is a frequently used heuristic in planning search (Artificial Intelligence A Modern Approach, 3rd ed., Russell and Norvig, p. 376) as is level sum (Artificial Intelligence A Modern Approach, 3rd ed., Russell and Norvig, p. 382), nevertheless it can be better to pick a cheaper to calculate heuristic over a smarter but harder to calculate one.

The following table describes an optimal sequence of actions to solve each of the air cargo problems provided using the highlighted approaches from the tables above:

Problem	Search Type	Optimal Sequence of Actions

Air Cargo Problem 1	BFS	Load(C1, P1, SFO)
		Load(C2, P2, JFK)
		Fly(P2, JFK, SFO)
		Unload(C2, P2, SFO)
		Fly(P1, SFO, JFK)
		Unload(C1, P1, JFK)

Air Cargo Problem 2	BFS	Load(C1, P1, SFO)
		Load(C2, P2, JFK)
		Load(C3, P3, ATL)
		Fly(P2, JFK, SFO)
		Unload(C2, P2, SFO)
		Fly(P1, SFO, JFK)
		Unload(C1, P1, JFK)
		Fly(P3, ATL, SFO)
		Unload(C3, P3, SFO)

Air Cargo Problem 3	A* Search h_ignore_preconditions	Load(C2, P2, JFK)
		Fly(P2, JFK, ORD)
		Load(C4, P2, ORD)
		Fly(P2, ORD, SFO)
		Unload(C4, P2, SFO)
		Load(C1, P1, SFO)
		Fly(P1, SFO, ATL)
		Load(C3, P1, ATL)
		Fly(P1, ATL, JFK)
		Unload(C3, P1, JFK)
		Unload(C2, P2, SFO)
		Unload(C1, P1, JFK)