An analysis of three air cargo problems with various search functions and heuristics
By Rohit Patel

We ran the three air cargo problems using various search techniques and heuristics. A discussion of the results follows. A summary of the test results is provided in the appendix.The optimal plans for each of the problems is as follows:

air_cargo_p1
Load(C1, P1, SFO)
Load(C2, P2, JFK)
Fly(P1, SFO, JFK)
Fly(P2, JFK, SFO)
Unload(C1, P1, JFK)
Unload(C2, P2, SFO)

air_cargo_p2
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(C1, P1, JFK)
Unload(C2, P2, SFO)
Unload(C3, P3, SFO)

air_cargo_p2
Load(C1, P1, SFO)
Load(C2, P2, JFK)
Fly(P1, SFO, ATL)
Load(C3, P1, ATL)
Fly(P2, JFK, ORD)
Load(C4, P2, ORD)
Fly(P1, ATL, JFK)
Unload(C1, P1, JFK)
Fly(P2, ORD, SFO)
Unload(C2, P2, SFO)
Unload(C3, P1, JFK)
Unload(C4, P2, SFO)

The following algorithms were run on each of the problems, and the table below shows a summary of time elapsed in seconds:

Time Elapsed in Seconds			
Search Type	air_cargo_p1	air_cargo_p2	air_cargo_p3
astar_search with h_ignore_preconditions	0.1	43.7	172
astar_search with h_pg_levelsum	0	93.5	389.3
breadth_first_search	0.2	44.6	272.4
depth_first_graph_search	0.1	9.1	6.3
depth_limited_search	0.4	>10 mins	> 10 mins
recursive_best_first_search	14.3	> 10 mins	> 10 mins
uniform_cost_search	0.2	53.8	247.5

We note informed search algorithms perform better than uninformed, however, depth first search is quite fast in arriving at the solution for these particular problems. We will look at the results in light of the optimality of the answer to get more insights.

Plan Length			
Search Type	air_cargo_p1	air_cargo_p2	air_cargo_p3
astar_search with h_ignore_preconditions	6	9	12
astar_search with h_pg_levelsum	6	9	13
breadth_first_search	6	9	12
depth_first_graph_search	20	619	392
depth_limited_search	50	50	NA
recursive_best_first_search	6	NA	NA
uniform_cost_search	6	9	12

We note that depth first search, while quick, often produces very suboptimal plans, as is expected if the search goes deep in unwanted directions (AIND, Udacity Sec 14-15)[2]. As expected, the breadth first search always produces the optimal route (AIND, Udacity Sec 14-15)[2], and is highly efficient (from a time perspective) with regards to the problems we have studied. This is because breadth_first_search, while searches for more nodes than astar_search with h_ignore_preconditions, it is must faster exploring each node.

Note the performance of the a-star search with two different heuristic functions. With h_ignore_preconditions, the search always leads to an optimal solution, this is because it is an admissible heuristic function. However, with the h_pg_levelsum heuristic function, that is not always the case. This is because the function is inadmissible (AIMA, pg 392)[1]. We should note however that the plans produce by this heuristic functions are always close to optimal. We also note that informed searches typically are more computationally intensive, for each node explored, and thus can be outperformed by uninformed searches. This is especially visible when comparing breadth_first_search with astar_search with h_ignore_preconditions for our problems. Following tables highlight the phenomenon:

Expansions			
Search Type	**air_cargo_p1**	**air_cargo_p2**	**air_cargo_p3**
astar_search with h_ignore_preconditions	41	1310	4478
astar_search with h_pg_levelsum	11	74	229
breadth_first_search	43	3343	14663
depth_first_graph_search	21	624	408
depth_limited_search	101	222719	NA
recursive_best_first_search	4229	NA	NA
uniform_cost_search	55	4689	17665

Expansions/sec			
Search Type	**air_cargo_p1**	**air_cargo_p2**	**air_cargo_p3**
astar_search with h_ignore_preconditions	280	30	26
astar_search with h_pg_levelsum	223	1	1
breadth_first_search	286	75	54
depth_first_graph_search	309	69	65
depth_limited_search	276	73	NA
recursive_best_first_search	296	NA	NA
uniform_cost_search	298	87	71

References:

1) **Russell, Stuart, and Peter Norvig. "Artificial intelligence : a modern approach",** ***Pearson Education, Limited, 2016***
2) **Udacity, “Artificial Intelligence Nanodegree”, Course Videos, Sec 14-15**

Appendix: A summary of the test results

Problem	Search Type	Expansions	Goal Tests	New Nodes	Plan Length	Time Elapsed in Seconds	Expansions/sec	Goal Tests/sec	New Nodes/sec
air_cargo_p1	astar_search with h_ignore_preconditions	41	43	170	6	0.1	280	294	1161
air_cargo_p1	astar_search with h_pg_levelsum	11	13	50	6	0	223	263	1012
air_cargo_p1	breadth_first_search	43	56	180	6	0.2	286	372	1196
air_cargo_p1	depth_first_graph_search	21	22	84	20	0.1	309	323	1234
air_cargo_p1	depth_limited_search	101	271	414	50	0.4	276	742	1133
air_cargo_p1	recursive_best_first_search	4229	4230	17023	6	14.3	296	296	1193
air_cargo_p1	uniform_cost_search	55	57	224	6	0.2	298	309	1216
air_cargo_p2	astar_search with h_ignore_preconditions	1310	1312	11979	9	43.7	30	30	274
air_cargo_p2	astar_search with h_pg_levelsum	74	76	720	9	93.5	1	1	8
air_cargo_p2	breadth_first_search	3343	4609	30509	9	44.6	75	103	684
air_cargo_p2	depth_first_graph_search	624	625	5602	619	9.1	69	69	615
air_cargo_p2	depth_limited_search	222719	2053741	2054119	50	3051	73	673	673
air_cargo_p2	uniform_cost_search	4689	4691	42590	9	53.8	87	87	792
air_cargo_p3	astar_search with h_ignore_preconditions	4478	4480	39564	12	172	26	26	230
air_cargo_p3	astar_search with h_pg_levelsum	229	231	2081	13	389.3	1	1	5
air_cargo_p3	breadth_first_search	14663	18098	129631	12	272.4	54	66	476
air_cargo_p3	depth_first_graph_search	408	409	3364	392	6.3	65	65	532
air_cargo_p3	uniform_cost_search	17665	17667	154975	12	247.5	71	71	626