Heuristic Analysis

Optimal Solutions

	air_cargo_p1	air_cargo_p2	air_cargo_p3
1	Load(C1, P1, SFO)	Load(C1, P1, SFO)	Load(C1, P1, SFO)
2	Fly(P1, SFO, JFK)	Fly(P1, SFO, JFK)	Fly(P1, SFO, ATL)
3	Unload(C1, P1, JFK)	Unload(C1, P1, JFK)	Load(C3, P1, ATL)
4	Load(C2, P2, JFK)	Load(C2, P2, JFK)	Fly(P1, ATL, JFK)
5	Fly(P2, JFK, SFO)	Fly(P2, JFK, SFO)	Unload(C1, P1, JFK)
6	Unload(C2, P2, SFO)	Unload(C2, P2, SFO)	Unload(C3, P1, JFK)
7		Load(C3, P3, ATL)	Load(C2, P2, JFK)
8		Fly(P3, ATL, SFO)	Fly(P2, JFK, ORD)
9		Unload(C3, P3, SFO)	Load(C4, P2, ORD)
10			Fly(P2, ORD, SFO)
11			Unload(C2, P2, SFO)
12			Unload(C4, P2, SFO)

Uninformed Search

breadth_first_search	air_cargo_p1	air_cargo_p2	air_cargo_p3
optimality	✅	✅	✅
time elapsed (s)	0.24	84.01	481.47
# expansions	43	3343	14663
# goal tests	56	4609	18098
plan length	6	9	12

depth_first_graph_search	air_cargo_p1	air_cargo_p2	air_cargo_p3
optimality	❌	❌	❌
time elapsed	0.12	16.32	10.86
# expansions	21	624	408
# goal tests	22	625	409
plan length	20	619	392

depth_limited_search	air_cargo_p1	air_cargo_p2	air_cargo_p3

optimality	❌	❌	━
time elapsed	0.60	5276.41	(stoped after 3h)
# expansions	101	222719	━
# goal tests	271	2053741	━
plan length	50	50	━

 `Breadth-first search` was the only uninformed algorithm that found optimal solutions for all problems, and ran in a reasonable amount of time. `Depth-first search` took less time and expansions to find a solution, but the plan length was considerably larger, due to the fact that `DFS` kept adding actions to the solution until the goal was reached (path length is almost as large as the number of expansions). The worst performing algorithm was `Depth limited search`, searching deep into the state space, but not deep enough as to brute-force the solution (like `DFS`). The solutions that `DLS` was able to find both have the length of 50 (the depth limit of the algorithm).

A* Search

h_ignore_preconditions	air_cargo_p1	air_cargo_p2	air_cargo_p3
optimality	✅	✅	✅
time elapsed	0.23	32.56	124.65
# expansions	41	1421	4589
# goal tests	43	1423	4591
plan length	6	9	12

h_pg_levelsum	air_cargo_p1	air_cargo_p2	air_cargo_p3
optimality	✅	✅	❌
time elapsed	2.10	186.18	744.12
# expansions	11	79	242
# goal tests	13	81	244
plan length	6	9	13

The `ignore preconditions` heuristic (being a relaxed version of the original problem, and thus - admissible) was guaranteed to find the optimal solution for all problems. It was able to find a solution faster than using a planning graph and with `levelsum` heuristic, which did not find an optimal solution for all problems (the heuristic is not admissible). However, using `levelsum` resulted in considerably fewer expansions, but the time spent computing the heuristic for each expansion cancels that.

Conclusions

The heuristic that yielded the best results was `ignore preconditions`. Compared to non-heuristic search, it expanded less nodes than `BFS`, but still found optimal solutions for each problem (compared to `DFS`). While planning graph and with `levelsum` heuristic solution for the third problem was not optimal, it was very close (1 extra action on solution) while expanding fewer nodes than any other search algorithm.