AIND - Planning Lab Analysis

This document intends to make an analysis in order to fully complete AIND Planning Lab. The problem described is an Air Cargo Transport System. Three planning problems are being analysed at this report.

Optimal plans for the problems

Air Cargo Problem 1

Problem	**Optimal plan (length=6)**
Init(At(C1, SFO) ∧ At(C2, JFK) ∧ At(P1, SFO) ∧ At(P2, JFK) ∧ Cargo(C1) ∧ Cargo(C2) ∧ Plane(P1) ∧ Plane(P2) ∧ Airport(JFK) ∧ Airport(SFO)) Goal(At(C1, JFK) ∧ At(C2, SFO))	1. Load(C1, P1, SFO) 2. Load(C2, P2, JFK) 3. Fly(P2, JFK, SFO) 4. Unload(C2, P2, SFO) 5. Fly(P1, SFO, JFK) 6. Unload(C1, P1, JFK)

Air Cargo Problem 2

Problem	**Optimal plan (length=9)**
Init(At(C1, SFO) ∧ At(C2, JFK) ∧ At(C3, ATL) ∧ At(P1, SFO) ∧ At(P2, JFK) ∧ At(P3, ATL) ∧ Cargo(C1) ∧ Cargo(C2) ∧ Cargo(C3) ∧ Plane(P1) ∧ Plane(P2) ∧ Plane(P3) ∧ Airport(JFK) ∧ Airport(SFO) ∧ Airport(ATL)) Goal(At(C1, JFK) ∧ At(C2, SFO) ∧ At(C3, SFO))	1. Load(C1, P1, SFO) 2. Load(C2, P2, JFK) 3. Load(C3, P3, ATL) 4. Fly(P2, JFK, SFO) 5. Unload(C2, P2, SFO) 6. Fly(P1, SFO, JFK) 7. Unload(C1, P1, JFK) 8. Fly(P3, ATL, SFO) 9. Unload(C3, P3, SFO)

Air Cargo Problem 3

Problem	**Optimal plan (length=12)**
Init(At(C1, SFO) ∧ At(C2, JFK) ∧ At(C3, ATL) ∧ At(C4, ORD) ∧ At(P1, SFO) ∧ At(P2, JFK) ∧ Cargo(C1) ∧ Cargo(C2) ∧ Cargo(C3) ∧ Cargo(C4) ∧ Plane(P1) ∧ Plane(P2) ∧ Airport(JFK) ∧ Airport(SFO) ∧ Airport(ATL) ∧	1. Load(C2, P2, JFK) 2. Fly(P2, JFK, ORD) 3. Load(C4, P2, ORD) 4. Fly(P2, ORD, SFO) 5. Load(C1, P1, SFO) 6. Fly(P1, SFO, ATL)

Airport(ORD)) Goal(At(C1, JFK) ∧ At(C3, JFK) ∧ At(C2, SFO) ∧ At(C4, SFO))	7. Load(C3, P1, ATL) 8. Fly(P1, ATL, JFK) 9. Unload(C1, P1, JFK) 10 Unload(C2, P2, SFO) 11. Unload(C3, P1, JFK) 12. Unload(C4, P2, SFO)

Search methods data

Air Cargo Problem 1

Method	Expansions	Goal Tests	New Nodes	Plan Length	Time elapsed in seconds
Non-heuristic search					
breadth_first_search	43	56	180	6	0.0583
breadth_first_tree_search	1458	1459	5960	6	1.520
depth_first_graph_search	21	22	84	20	0.022
depth_limited_search	101	271	414	50	0.154
uniform_cost_search	55	57	224	6	0.073
Heuristic search					
astar_search h_1	55	57	224	6	0.0789819739293307
astar_search h_ignore_preconditions	41	43	170	6	0.0849287880118936 3
astar_search h_pg_levelsum	11	13	50	6	3.0833986899815 5

Air Cargo Problem 2

Method	Expansions	Goal Tests	New Nodes	Plan Length	Time elapsed in seconds
Non-heuristic search					
breadth_first_search	3343	4609	30509	9	19.28120
breadth_first_tree_search	It tooks more than 3 minutes				
depth_first_graph_search	624	625	5602	619	4.861831
depth_limited_search	It tooks more than 3 minutes				
uniform_cost_search	4853	4855	44041	9	61.605972755000186
Heuristic search					
astar_search h_1	4853	4855	44041	9	61.375491166953
astar_search h_ignore_preconditions	1506	1508	13820	9	20.807023262954317
astar_search h_pg_levelsum	86	88	841	9	321.5026430550497

Air Cargo Problem 3

Method	Expansions	Goal Tests	New Nodes	Plan Length	Time elapsed in seconds
Non-heuristic search					
breadth_first_search	14663	18098	129631	12	132.755
breadth_first_tree_search	It took more than 5 minutes				
depth_first_graph_search	408	409	3364	392	2.39021
depth_limited_search	It took more than 3 minutes				
uniform_cost_search	18151	18153	159038	12	445.728
Heuristic search					
astar_search h_1	18151	18153	159038	12	504.1941834639292
astar_search h_ignore_preconditions	5118	5120	45650	12	130.6192178729687
astar_search h_pg_levelsum	404	406	3718	12	2100.5661585619673

Search methods Analysis

Air Cargo Problem 1

- Depth search methods performed worse than others because it generated the longest plan to reach the goal
- Breadth first search reached the goal in less time than all others
- But heuristic methods expanded fewer nodes but it took more time to run.

- Uniform_cost_search found almost was as good as breadth_first_search

Air Cargo Problem 2

- Again depth search methods had a bad performance. They reach the goal, but with a length of 619 and depth_limited_search didn't reach the end because it took more than 3 minutes
- A* search with ignore preconditions had an interesting performance. It took only 1.52 seconds more than breadth_first_search but it expanded only 1506 nodes (1837 less than breadth_first_search)
- A* search levelsum expanded 86 nodes but it took 321.502 seconds to run (301 seconds more than h_ignore_preconditions
- Uniform_cost_search found an optimal plan, but it took 41 seconds more than h_ignore_preconditions

Air Cargo Problem 3

- As expected, depth search methods had a bad performance.
- A* search with ignore preconditions was the best in here. It took 130 seconds (2 seconds less than breadth_first_search) and it only expanded 5118 nodes (9545 less than breadth_first_search)
- A* search with level sum expanded fewer nodes, but it's expensive computationally, taking more time to compute
- We can conclude that, as the problem gets more complex, A* search with ignore preconditions performs better than any other non-heuristic search method