Heuristic Analysis

In this report, different logistics planning problems for an Air Cargo transport system will be solved by using a planning search agent, such as depth first graph search or a star search.

Local machine information:

Notebook Model	Processor	Memory	Operating System
Lenovo T440s	Intel i7-4600U @ 2.10GHz x 4	8Gb	Ubuntu 16.1

Problem 1:

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))

Search Agent	Expansions	Goal Tests	New Nodes	Plan length	Time elapsed(Second)
breadth_first_search	43	56	180	6	0.02992305399993711
breadth_first_tree_search	1458	1459	5960	6	0.7373642840002503
depth_first_graph_search	21	22	84	20	0.011693734000346012
depth_limited_search	101	271	414	50	0.0887045230001604
uniform_cost_search	55	57	224	6	0.032715945999825635
recursive_best_first_search h_1	4229	4230	17023	6	2.2608719750001
greedy_best_first_graph_search h_1	7	9	28	6	0.004016191000118852
astar_search h_1	55	57	224	6	0.034398275000057765
astar_search h_ignore_preconditions	41	43	170	6	0.02896680699996068
astar_search h_pg_levelsum	11	13	50	6	3.857598983999651

Based on the report shown above, greedy_best_first_graph_search h_1 obtain the result fastest and recursive_best_first_search h_1 is the slowest searching agent. In this problem, If uninformed planning algorithms is going to be used, depth_first_graph_search is the best in terms of the speed. Then, if A* search is going to be used, astar_search h_ignore_preconditions will be the best one.

Problem 2:

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))

Search Agent	Expansions	Goal Tests	New Nodes	Plan length	Time elapsed(Second)
breadth_first_search	3343	4609	30509	9	13.948505666000528
breadth_first_tree_search	n/a	n/a	n/a	n/a	n/a
depth_first_graph_search	624	625	5602	619	4.676244258999759
depth_limited_search	n/a	n/a	n/a	n/a	n/a
uniform_cost_search	4853	4855	44041	9	99.96627871800047
recursive_best_first_search h_1	n/a	n/a	n/a	n/a	n/a
greedy_best_first_graph_search h_1	998	1000	8982	15	18.14441489299952
astar_search h_1	4853	4855	44041	9	105.94995907199973
astar_search h_ignore_preconditions	1506	1508	13820	9	39.937825807999616
astar_search h_pg_levelsum	86	88	84	9	2677.278856242

n/a: The search agent cannot obtain the result within 10 minutes.

Based on the report above, depth_first_graph_search obtain the result fastest Then, breadth_first_tree_search, depth_limited_search and recursive_best_first_search h_1 are failed to obtain the result within 10 minutes. If uninformed planning algorithms is going to be used, depth_first_graph_search is the best in terms of the speed. Then, if A* search is going to be used, astar_search h_ignore_preconditions will be the best one.

Problem 3:

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) ∧ Airport(ORD))
Goal(At(C1, JFK) ∧ At(C3, JFK) ∧ At(C2, SFO) ∧ At(C4, SFO))

Search Agent	Expansions	Goal Tests	New Nodes	Plan length	Time elapsed(Second)
breadth_first_search	14663	18098	129631	12	591.1350850649997
breadth_first_tree_search	n/a	n/a	n/a	n/a	n/a
depth_first_graph_search	408	409	3364	392	8.107768466999914
depth_limited_search	n/a	n/a	n/a	n/a	n/a
uniform_cost_search	18223	18225	159618	12	616.3229179369998
recursive_best_first_search h_1	n/a	n/a	n/a	n/a	n/a
greedy_best_first_graph_search h_1	5578	5580	49150	22	209.35098051299974
astar_search h_1	18223	18225	159618	12	590.61647025
astar_search h_ignore_preconditions	5118	5120	45650	12	388.98022772900003
astar_search h_pg_levelsum	n/a	n/a	n/a	n/a	n/a

n/a: The search agent cannot obtain the result within 10 minutes.

Based on the report above, depth_first_graph_search obtain the result fastest and breadth_first_tree_search and recursive_best_first_search h_1 are failed to obtain the result within 10 minutes. If uninformed planning algorithms is going to be used, depth_first_graph_search is the best in terms of the speed. Then, if A* search is going to be used, astar_search h_ignore_preconditions will be the best one.

To conclude, depth_first_graph_search usually be the fastest searching agent. It is hard to say that non-heuristic search is always better in all these problem. In problem 2, we can see that lots of non-heuristic searching agent cannot obtain the result within 10 minutes. It seems to explain that although heuristic search may calculate the result slower than non-heuristic search, heuristic search can still obtain the result within a certain time consult. On the other hand, non-heuristic seems hard to guarantee to finish the search within a certain amount of time.