Planning Search Project

Optimal Problem plans

Problem 1	**Plan 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))	Load(C1, P1, SFO) Load(C2, P2, JFK) Fly(P1, SFO, JFK) Fly(P2, JFK, SFO) Unload(C1, P1, JFK) Unload(C2, P2, SFO)
Problem 2	**Plan 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))	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(C3, P3, SFO) Unload(C1, P1, JFK) Unload(C2, P2, SFO)
Problem 3	**Plan 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))	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) Fly(P2, ORD, SFO) Unload(C4, P2, SFO) Unload(C3, P1, JFK) Unload(C1, P1, JFK) Unload(C2, P2, SFO)

Non-heuristic search result metrics analysis

Compare and contrast non-heuristic search result metrics (optimality, time elapsed, number of node expansions) for Problems 1,2, and 3. Include breadth-first, depth-first, and at least one other uninformed non-heuristic search in your comparison; Your third choice of non-heuristic search may be skipped for Problem 3 if it takes longer than 10 minutes to run, but a note in this case should be included.

Having in mind that node expansions, node creations and goal test operations all are a part of total time needed for search algorithm to find a plan it is reasonable to compare algorithms just by the total time it took search algorithm to find the solution, given that variations of cpu time available for the algorithm did not differ substantially.

Problem 1

Amongst *Breadth first search*, *Depth first graph search* and *Uniform cost search* best in searching for optimal path were *Breadth first search* and Uniform cost search as they found optimal solution plan twice shorter compared to *Depth first graph search.*
In terms of time required to find a plan *Depth first graph search* outperformed other two requiring roughly a fifth of time.

Problem 2

Similarly as with Air Cargo Problem 1 *Breadth first search* and *Uniform cost search* outperformed *Depth first graph search*. But this time all three problems took similar amount of time find solution ~15s. Regardless of Uniform cost search algorithm performing somewhat more intermediate steps like node expansions, goal tests and node creations compared to Breadth first search the total amount of time needed to perform search for Uniform cost search was notably smaller. Cause for that maybe either implementation, algorithm inherent intricacies (one may require more RAM or perform more IO operations) or failure to ensure identical environment for running algorithms or all of the above.

Problem 3

Air Cargo Problem 3 was again similar to previous findings. Uniform cost search found solution ~2 times faster compared to Breadth first search. Both of them found optimal solutions. Interestingly enough depth first search found a solution ~35 times faster than breadth first search and ~20 times faster than uniform cost search on the other hand the solution path is far from optimal - requiring ~58 times more actions to reach a problem solution.

From the perspective of the three problems depth first search tends to find solution significantly faster or as fast as others but at the cost of solution optimality.

Heuristic search result metrics

Compare and contrast heuristic search result metrics using A with the "ignore preconditions" and "level-sum" heuristics for Problems 1, 2, and 3.*

For every problem both A* with the "ignore preconditions" and "level-sum" heuristics found optimal plans. The main difference as seen in *Heuristic search experiment analysis data table* is time required for a plan to be found. The difference is from one to two orders of magnitude in favour of ignoring preconditions. But looking at the number of expansions, goal tests and new nodes the difference is not that significant even opposite, number of such operations is ~2-3 times smaller for level-sum.

On the other hand it has to be noted that to find a problem solution one has to find a path within the state graph a path connecting the initial state to a goal state. By ignoring preconditions the problem is relaxed by adding more edges to the graph, making it strictly easier to find a path, and thus is easier to search. Ignore preconditions heuristic drops all preconditions from actions. Every action becomes applicable in every state, and any single goal fluent can be achieved much faster - in one step (if there is an applicable action—if not, the problem is impossible).

Best heuristic

What was the best heuristic used in these problems? Was it better than non-heuristic search planning methods for all problems? Why or why not?

Looking only at the performed experiments it is easy to name A* search with *ignore preconditions* heuristic as the winner. The path that is found is optimal in all experiments performed, speed at which solutions were found are superior to other heuristic and non-heuristic alternatives (if compared only with algorithms with optimal solutions).

Tables and figures

Provide tables or other visual aids as needed for clarity in your discussion.

Non-heuristic search experiment analysis data table

Air Cargo Problem	Search	Expansions	Goal Tests	New Nodes	Plan length	Time elapsed
1	breadth_first_search	43	56	180	6	0,055953412
2	breadth_first_search	3343	4609	30509	9	16,59158764
3	breadth_first_search	14663	18098	129631	12	128,7606812
1	depth_first_graph_search	12	13	48	12	0,010361294
2	depth_first_graph_search	1669	1670	14863	1444	15,39022454
3	depth_first_graph_search	592	593	4927	571	3,59602124
1	uniform_cost_search	55	57	224	6	0,04754372902
2	uniform_cost_search	4852	4854	44030	9	15,55863006
3	uniform_cost_search	18235	18237	159716	12	69,73866134

Heuristic search experiment analysis data table

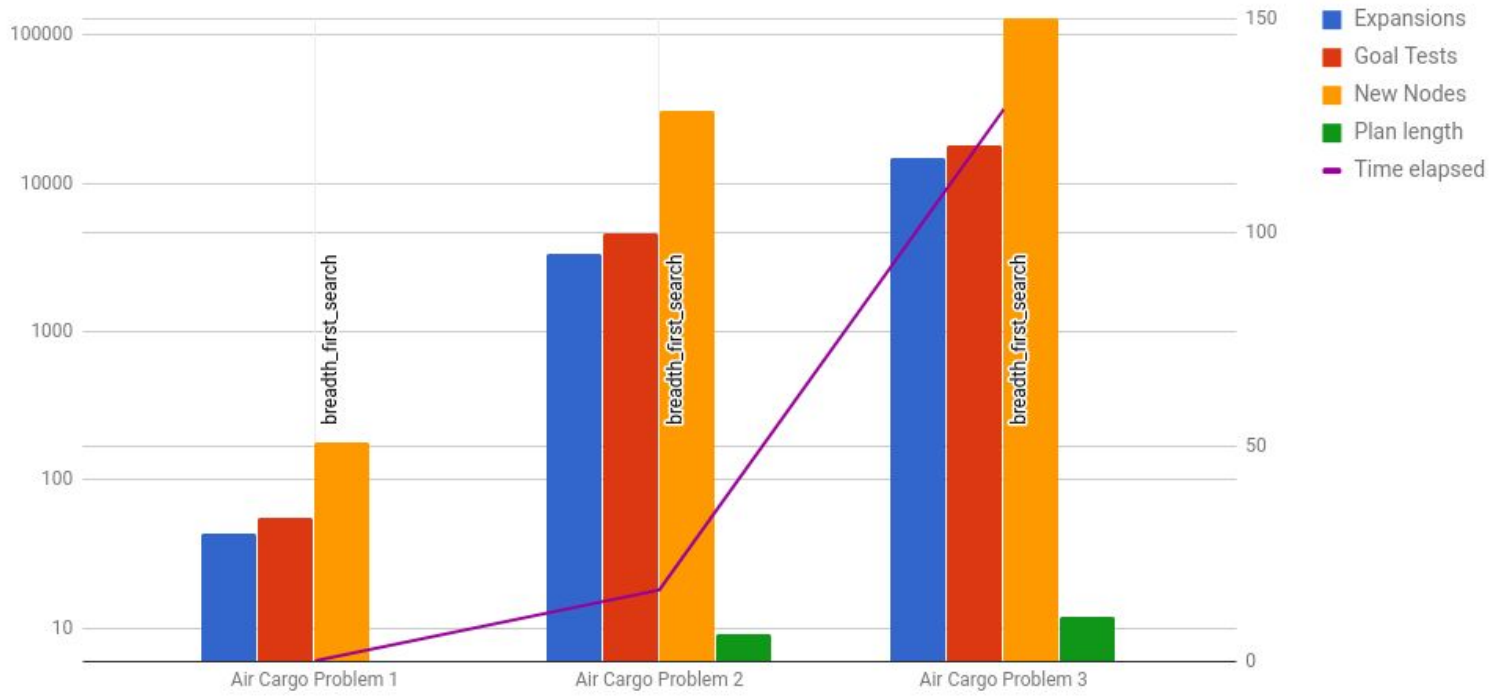
Air Cargo Problem	Search	Heuristic	Expansions	Goal Tests	New Nodes	Plan length	Time elapsed
1	astar_search	h_ignore_preconditions	41	43	170	6	0,054207379
2	astar_search	h_ignore_preconditions	1450	1452	13303	9	5,97876217
3	astar_search	h_ignore_preconditions	5040	5042	44944	12	24,75135223
1	astar_search	h_pg_levelsum	39	41	158	6	0,973297466
2	astar_search	h_pg_levelsum	1129	1131	10232	9	352,0276291
3	astar_search	h_pg_levelsum	2015	2017	17827	12	1368,924876

Full experiment data table

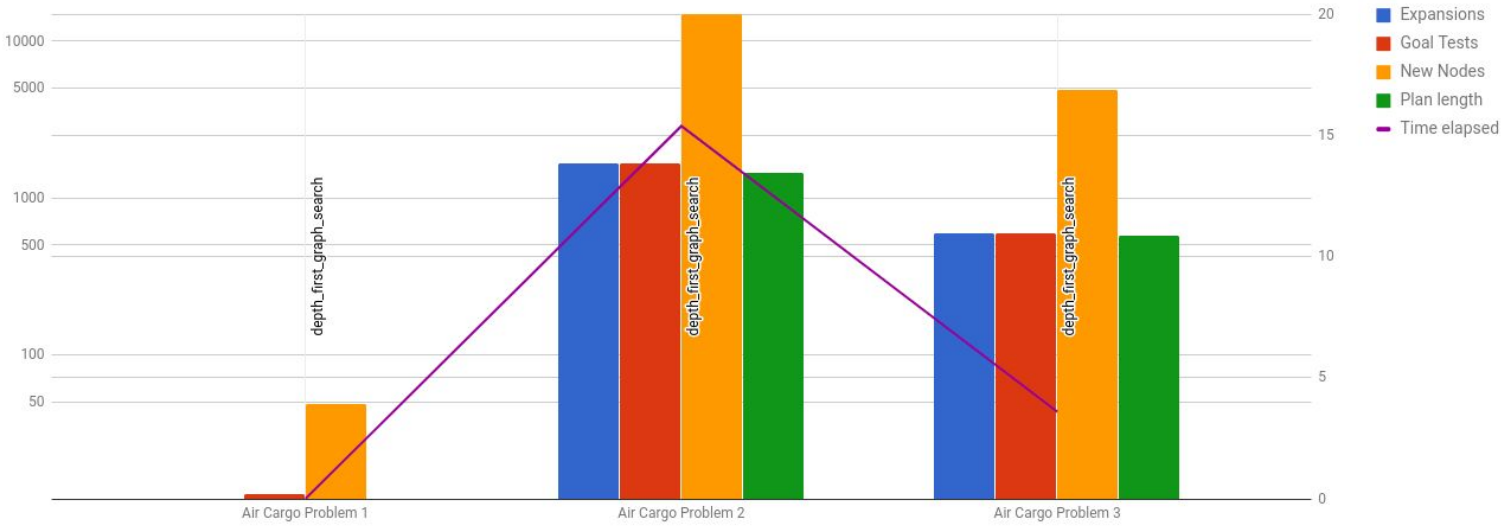
Air Cargo Problem	Search	Heuristic	Expansions	Goal Tests	New Nodes	Plan length	Time elapsed
1	astar_search	h_1	55	57	224	6	0,05033136701
2	astar_search	h_1	4852	4854	44030	9	14,8720157
3	astar_search	h_1	18235	18237	159716	12	69,36647613
1	astar_search	h_ignore_preconditions	41	43	170	6	0,054207379
2	astar_search	h_ignore_preconditions	1450	1452	13303	9	5,97876217
3	astar_search	h_ignore_preconditions	5040	5042	44944	12	24,75135223
1	astar_search	h_pg_levelsum	39	41	158	6	0,973297466

2	astar_search	h_pg_levelsum	1129	1131	10232	9	352,0276291
3	astar_search	h_pg_levelsum	2015	2017	17827	12	1368,924876
1	breadth_first_search		43	56	180	6	0,055953412
2	breadth_first_search		3343	4609	30509	9	16,59158764
3	breadth_first_search		14663	18098	129631	12	128,7606812
1	breadth_first_tree_search		1458	1459	5960	6	1,229366892
1	depth_first_graph_search		12	13	48	12	0,010361294
2	depth_first_graph_search		1669	1670	14863	1444	15,39022454
3	depth_first_graph_search		592	593	4927	571	3,59602124
1	depth_limited_search		101	271	414	50	0,119278093
2	depth_limited_search		222719	2053741	2054119	50	1388,242634
1	greedy_best_first_graph_search	h_1	7	9	28	6	0,007127704972
2	greedy_best_first_graph_search	h_1	990	992	8910	17	3,03281837
3	greedy_best_first_graph_search	h_1	5615	5617	49438	22	20,61434709
1	recursive_best_first_search	h_1	4229	4230	17029	6	3,668217345
1	uniform_cost_search		55	57	224	6	0,04754372902
2	uniform_cost_search		4852	4854	44030	9	15,55863006
3	uniform_cost_search		18235	18237	159716	12	69,73866134

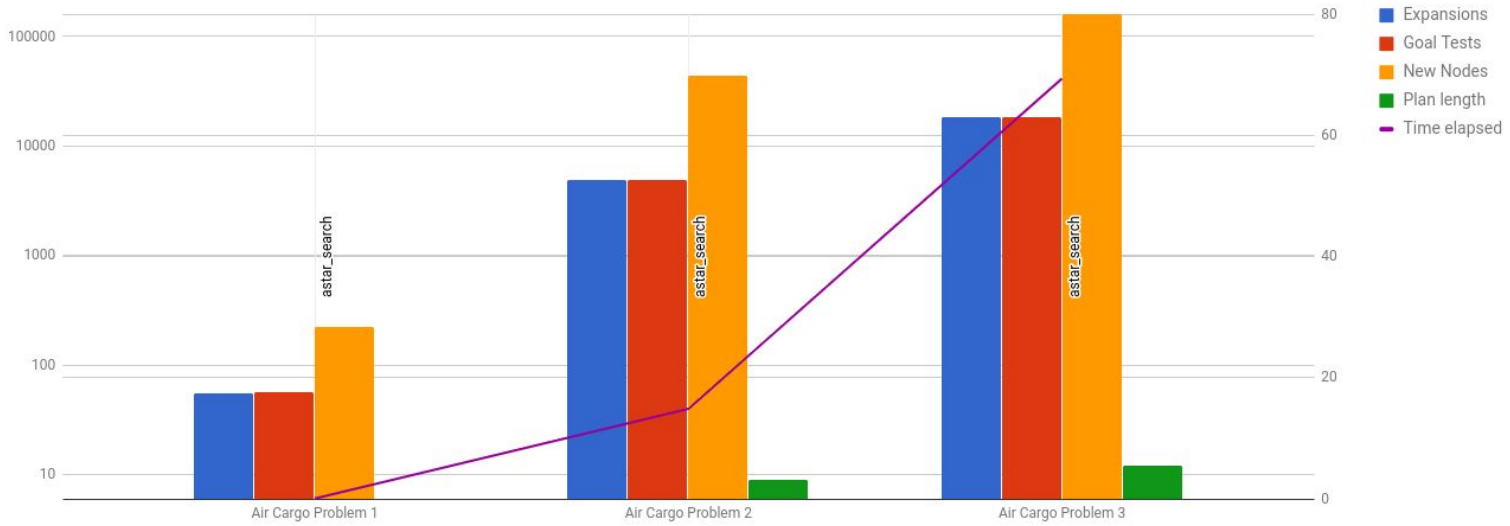
Breadth first search



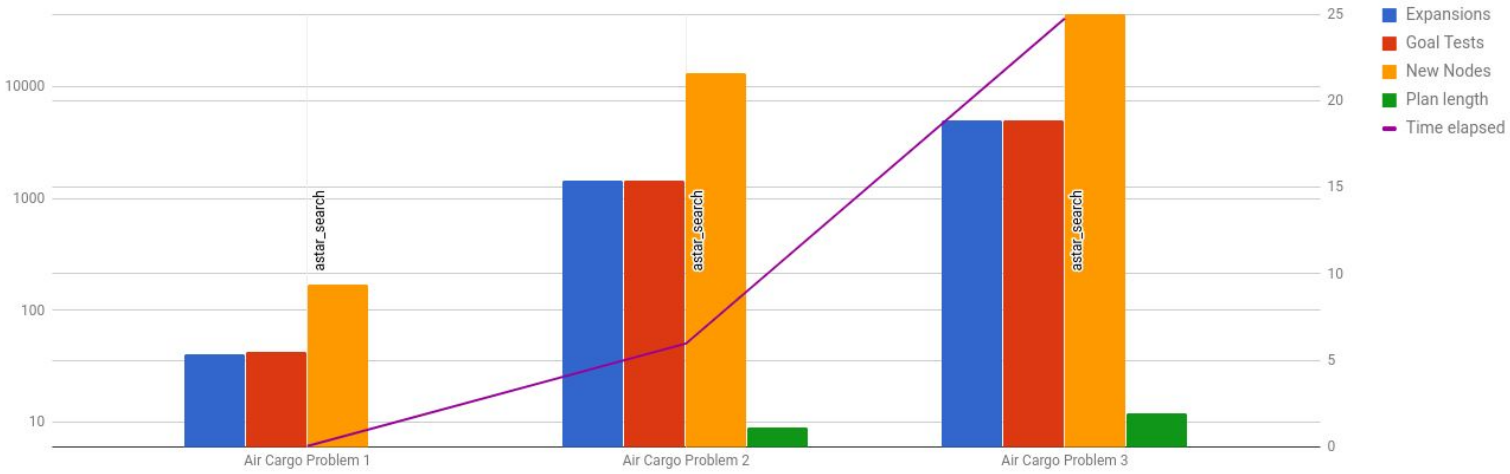
Depth first graph search



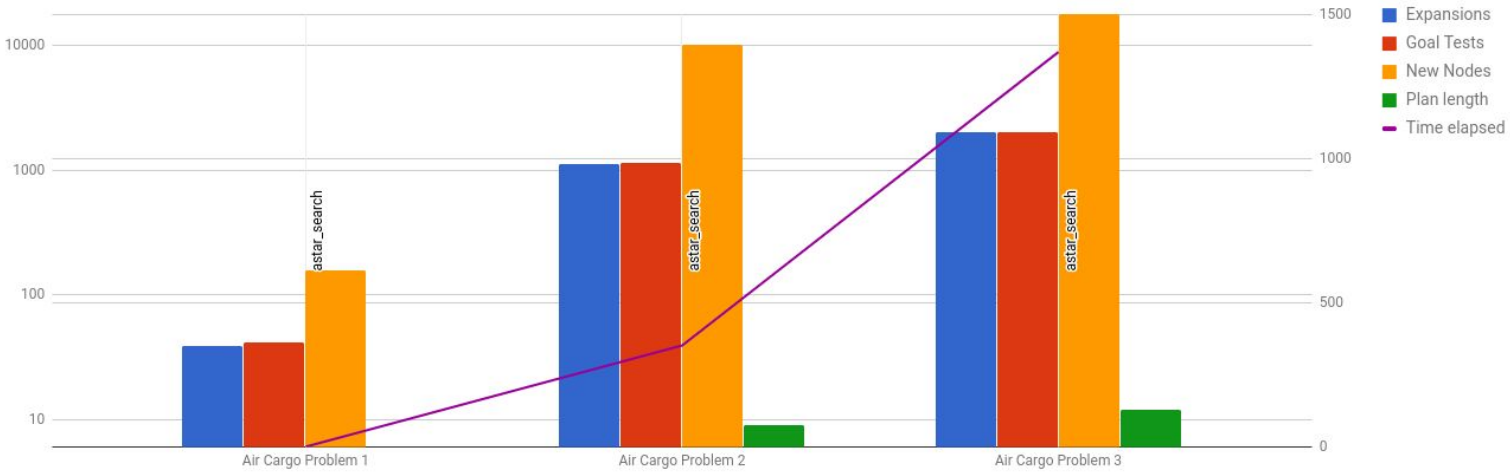
A* search (h_1 heuristic)



A* search (h_ignore_preconditions heuristic)



A* search (pg_levelsum heuristic)



Greedy best first graph search (h_1 heuristic)