Heuristic Analysis

Introduction

In scope of this project was defined and solved a group of deterministic logistics planning problems for an Air Cargo transport system using a planning search agent. With progression search algorithms, optimal plans for each problem have been computed.

The search algorithms were improved with several domain-independent heuristics, including planning graph -based heuristics.

Optimal Plan for Problems 1, 2, and 3

Optimal plans for the three problems are provided in the table below.

Problem 1	Problem 2	Problem 3
Length 6	Length 9	Length 12

Load(C1, P1, SFO)
Load(C2, P2, JFK)
Fly(P2, JFK, SFO)
Unload(C2, P2, SFO)
Fly(P1, SFO, JFK)
Unload(C1, P1, JFK)
	Load(C1, P1, SFO)
	Load(C2, P2, JFK)
	Load(C3, P3, ATL)
	Fly(P2, JFK, SFO)
	Unload(C2, P2, SFO)
	Fly(P1, SFO, JFK)
	Unload(C1, P1, JFK)
	Fly(P3, ATL, SFO)
	Unload(C3, P3, SFO)
		Load(C1, P1, SFO)
		Load(C2, P2, JFK)
		Fly(P2, JFK, ORD)
		Load(C4, P2, ORD)
		Fly(P1, SFO, ATL)
		Load(C3, P1, ATL)
		Fly(P1, ATL, JFK)
		Unload(C1, P1, JFK)
		Unload(C3, P1, JFK)
		Fly(P2, ORD, SFO)
		Unload(C2, P2, SFO)
		Unload(C4, P2, SFO)

Non-heuristic Search Results Comparison

In this experiment, we have applied breadth-first (BFS), depth-first (DFS), and uniform cost (UCS) search algorithms to find a solution to the three planning problems.

Results are provided in the table below:

Algorithm	Problem	Plan Length	Time, s	Expansions	Goal Tests	New Nodes
BFS	1	6	0.0238571	43	56	180

DFS	1	20	0.0092573	21	22	84
UCS	1	6	0.0255910	55	57	224
BFS	2	9	5.8861582	3343	4609	30509
DFS	2	619	2.7190663	624	625	5602
UCS	2	9	8.1714758	4840	4842	43918
BFS	3	12	28.667148	14663	18098	129631
DFS	3	392	1.2989768	408	409	3364
UCS	3	12	37.944338	18193	18195	159379

From these results we see, that optimal plans were found when using **breath-first** and unified cost search. The former shows better performance in terms of speed and number of nodes expansion.

Plans produced by depth-first search are non-optimal. It shown fastest performance, which, however, doesn't justify very low solution optimality.

Heuristic A* Search Results

A* search with the "ignore preconditions" and "level-sum" heuristics for Problems 1, 2, and 3 shown the following results:

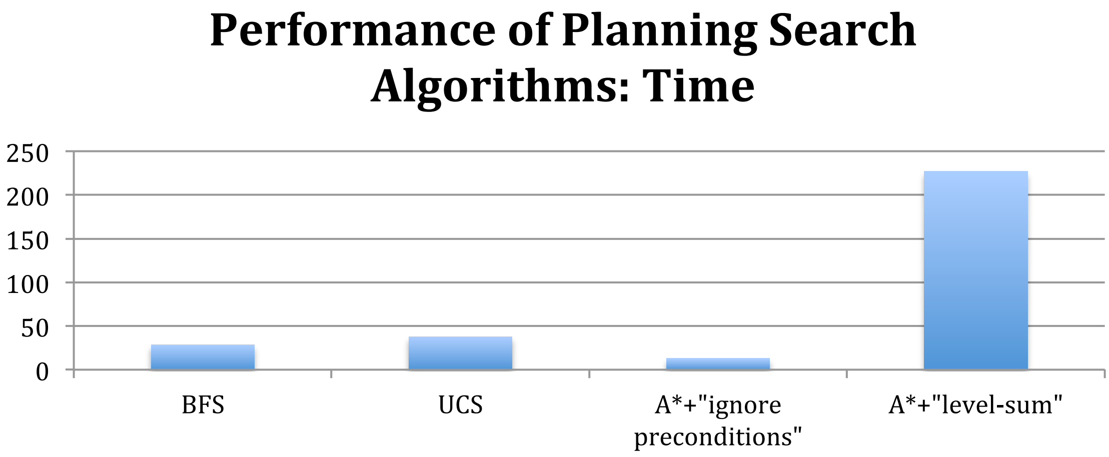
Search	Problem	Plan Length	Time, s	Expansions
A*+"ignore preconditions"	1	6	0.032593337	41
A*+"level-sum"	1	6	0.613103457	11
A*+"ignore preconditions"	2	9	3.204912981	1438
A*+"level-sum"	2	9	55.79514302	85
A*+"ignore preconditions"	3	12	12.82345364	4931
A*+"level-sum"	3	12	226.9863322	255

Both heuristics have delivered an optimal plan.

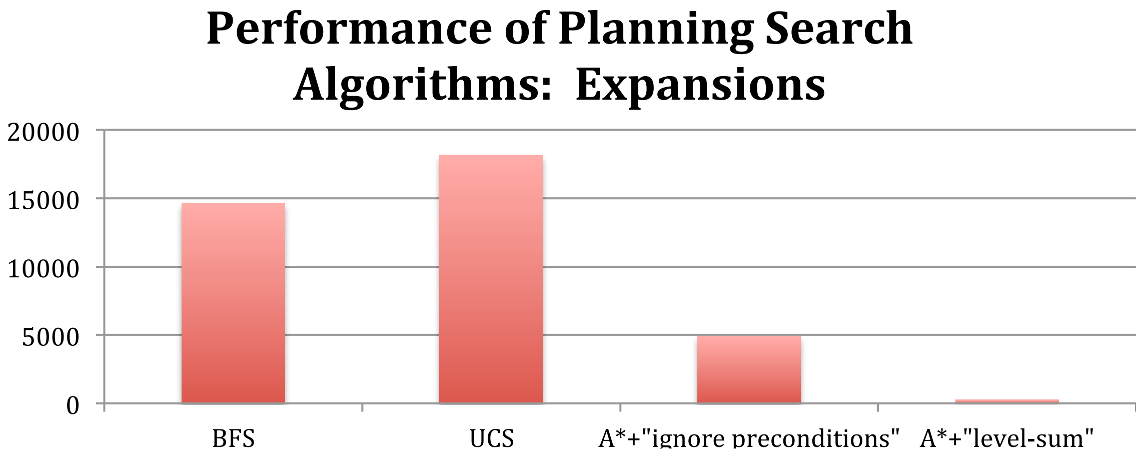
“Ignore pre-conditions” appeared to be much faster than Plan Graph -based “level-sum” heuristic. In turn, “level-sum” heuristic provided significantly better estimation of the path selection, which is clearly visible from the nodes expansion counts.

Conclusion

Overall, among tested approaches that delivered an optimal plan, A* search enriched with “ignore preconditions” heuristics shown best performance in terms of both speed, and search expansion.



Performance of Planning Search Algorithms: Time



Performance of Planning Search Algorithms: Expansions

A* search with “level-sum” heuristic appeared to be best in estimating search path cost and choosing a right expansion node, at the cost of speed, that explained by the need to build a plan graph and compute heuristic estimate.