Heuristic Analysis - AIND Planning

Optimal Plan

	Plan
Problem 1	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	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(C2, P2, SFO) Unload(C1, P1, JFK)
Problem 3	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(P2, ORD, SFO) Fly(P1, ATL, JFK) Unload(C4, P2, SFO) Unload(C3, P1, JFK) Unload(C2, P2, SFO) Unload(C1, P1, JFK)

Non - Heuristic Search

Problem 1

	Expansions	Goal Tests	New Nodes	Plan length	Time elapsed in seconds:
Breadth First Search	43	56	180	6	0.1600014899
Depth First Graph Search	12	13	48	12	0.04615733211
Uniform Cost Search	55	57	224	6	0.1911994214

Problem 2

	Expansions	Goal Tests	New Nodes	Plan length	Time elapsed in seconds:
Breadth First Search	3343	4609	30509	9	45.4882749
Depth First Graph Search	582	583	5211	575	8.509073104
Uniform Cost Search	4852	4854	44030	9	58.32772805

Problem 3

	Expansions	Goal Tests	New Nodes	Plan length	Time elapsed in seconds:
Breadth First Search	14663	18098	129631	12	274.9904778
Depth First Graph Search	627	628	5176	596	9.877911799
Uniform Cost Search	18235	18237	159716	12	263.2118611

Heuristic Search

Problem 1

	Expansions	Goal Tests	New Nodes	Plan length	Time elapsed in seconds:
Astar Search H1	55	57	224	6	0.200338639
Astar Search H Ignore Preconditions	41	43	170	6	0.159827719
Astar Search H Pg Levelsum	31	33	126	6	1.070328582

Problem 2

	Expansions	Goal Tests	New Nodes	Plan length	Time elapsed in seconds:
Astar Search H1	4852	4854	44030	9	57.4273025
Astar Search H Ignore Preconditions	1450	1452	13303	9	17.5903540
Astar Search H Pg Levelsum	2004	2006	18626	9	510.7185525

Problem 3

	Expansions	Goal Tests	New Nodes	Plan length	Time elapsed in seconds:
Astar Search H1	18235	18237	159716	12	251.2535009
Astar Search H Ignore Preconditions	5040	5042	44944	12	73.4152019
Astar Search H Pg Levelsum	7425	7427	67757	12	2939.334652

What was the best heuristic used in these problems?

Ignore Preconditions

Was it better than non-heuristic search planning methods for all problems? Why or why not?

Yes, because it doesn't need to expand as many nodes, than the non-heuristic methods.

Why does DFS not provide an optimal plan for this case?

Because DFS is neither complete nor optimal. DFGS (Depth First Graph Search) is complete but not optimal. DFGS finds a solution but it doesn't need to be the optimal solution.
(Russell and Norvig 2010)

Which are the fastest and slowest uninformed algorithms? Why?

The fastest algorithm is DFGS, but this is due the fact that DFGS is not optimal, so it finds a solution very fast, but we can see that the path length is very big.
The slower algorithms were BFS and UCS, for the first two problems BFS were the faster algorithm and for the third problem UCS was faster than BFS. These two algorithms are optimal and complete, but it takes more time to find a solution.
(Russell and Norvig 2010)

Which are the fastest and slowest heuristic searches? Why?

The fastest algorithm is the *Ignore Precondition* heuristic, it is computationally more expensive than the *H1* but it reduces the amount of nodes to be expanded and so it reduces the search space. The *Pg Levelsum* is the slowest one, but this algorithm needs to build the entire planning graph for the problem which could also be computationally expensive.

Which algorithms (uninformed and heuristic) have the lowest node expansions and goal tests? Why? What does this mean for the performance in this problem?

DFGS has the lowest amount of node expansions and goal tests. As mentioned above DFGS is not optimal, it expands in breadth only if it gets stucked, so it only expands as many nodes as it needs to find a solution. Because of the lower amount of node expansions the memory demand is lower and so it is also faster.

References

Russell, Stuart Jonathan, and Peter Norvig. 2010. *Artificial Intelligence: A Modern Approach*. Prentice Hall.