AIND-Panning Metrics Analysis

Uninformed/Informed Search Analysis

Objective:

To find the optimal plan considering less memory usage and less search time for the three problems given.

Metrics Considered:

Path length – The number of steps required to find the plan.

Expansions – Number of nodes expanded to find the optimal plan. More number of node expansions requires more memory to be used.

Time(s) – Number of seconds taken by the search to find the plan.

N/A - Refers no data has been collected due to the search running more than 10 mins.

Problem 1

Optimal Solution:

Load(C2, P2, JFK)

Load(C1, P1, SFO)

Fly(P2, JFK, SFO)

Unload(C2, P2, SFO)

Fly(P1, SFO, JFK)

Unload(C1, P1, JFK)

S.No	Search	Path Length	Expansions	Goal Tests	Time(s)
1	Breadth First Search	6	43	56	0.048
2	Breadth First Tree Search	6	1458	1459	1.497
3	Depth First Graph Search	12	12	13	0.010
4	Depth Limited Search	50	101	271	0.119
5	Uniform Cost Search	6	55	57	0.059
6	Recursive Best First Search + h_1	6	4229	4230	4.097
7	Greedy Best First Graph Search + h_1	6	7	9	0.006
8	A star + h_1	6	55	57	0.048

9	A star + h_ignore_preconditions	6	41	43	0.054
10	A star + h_pg_levelsum	6	55	57	1.353

Uninformed Search:

Searches Considered: Breadth First Search, Depth First Graph Search and Uniform Cost Search

Breadth First and Uniform Cost searches find the optimal plan with path length of 6 whereas Depth First Graph Search finds the plan with path length of 12 which is not optimal. Breadth First search out performs Uniform Cost Search in terms of number of node expansions and time. Hence for the problem 1, the best uninformed search for Problem-1 is Breadth First Search.

Informed Search:

Searches Considered: A star + h_ignore_preconditions and A star + h_pg_levelsum

Both A star searches with h_ignore_precondistions and h_pg_levelsum finds the optimal plan with the path length of 6. But the h_ignore_preconditions outperform h_pg_levelsum in terms-of node expansions and search time. Hence, the best informed search for Problem-1 is A star + h_ignore_preconditions.

Problem 2

Optimal Solution:

Load(C1, P1, SFO)

Load(C2, P2, JFK)

Load(C3, P3, ATL)

Fly(P1, SFO, JFK)

Unload(C1, P1, JFK)

Fly(P2, JFK, SFO)

Unload(C2, P2, SFO)

Fly(P3, ATL, SFO)

Unload(C3, P3, SFO)

S.No	Search	Path Length	Expansions	Goal Tests	Time(s)
1	Breadth First Search	9	3346	4612	20.684
2	Breadth First Tree Search	N/A	N/A	N/A	N/A
3	Depth First Graph Search	515	520	521	3.876
4	Depth Limited Search	N/A	N/A	N/A	N/A
5	Uniform Cost Search	9	4853	4855	17.226
6	Recursive Best First Search + h_1	N/A	N/A	N/A	N/A
7	Greedy Best First Graph Search+ h_1	21	998	1000	3.36
8	A star + h_1	9	4853	4855	17.206
9	A star + h_ignore_preconditions	9	1450	1452	7.034
10	A star + h_pg_levelsum	9	4853	4855	794.70

Uninformed Search:

Searches Considered: Breadth First Search, Depth First Graph Search and Uniform Cost Search

Breadth First and Uniform Cost searches find the optimal plan with path length of 9 whereas Depth First Graph Search finds the plan with path length of 515 which is not optimal. Breadth First search out performs Uniform cost search in terms of number of node expansions and but Uniform Cost Search outperforms Breadth First Search in time. Hence, the best uninformed search for Problem-2 is decided based on whether we need less memory or less search time.

Informed Search:

Searches Considered: A star + h_ignore_preconditions and A star _ h_pg_levelsum

Both A star searches with h_ignore_precondistions and h_pg_levelsum finds the optimal plan with the path length of 9. But the h_ignore_preconditions outperform h_pg_levelsum in terms-of node expansions and search time. Especially h_ignore_preconditions is 100 times better than h_pg_levelsum in terms of search time. Hence, the best informed search for Problem-1 is A star + h_ignore_preconditions.

Problem 3

Optimal Solution:

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)

Unload(C1, P1, JFK)

Unload(C3, P1, JFK)

Fly(P2, ORD, SFO)

Unload(C2, P2, SFO)

Unload(C4, P2, SFO)

S.No	Search	Path Length	Expansions	Goal Tests	Time(s)
1	Breadth First Search	12	14120	17673	157.191
2	Breadth First Tree Search	N/A	N/A	N/A	N/A
3	Depth First Graph Search	660	677	678	6.065
4	Depth Limited Search	N/A	N/A	N/A	N/A
5	Uniform Cost Search	12	18223	18225	82.07
6	Recursive Best First Search + h_1	N/A	N/A	N/A	N/A
7	Greedy Best First Graph Search + h_1	22	5578	5580	30.066
8	A star + h_1	12	18223	18225	86.079
9	A star + h_ignore_preconditions	12	5040	5042	27.253
10	A star + h_pg_levelsum	N/A	N/A	N/A	N/A

Uninformed Search:

Searches Considered: Breadth First Search, Depth First Graph Search and Uniform Cost Search

Breadth First and Uniform Cost searches find the optimal plan with path length of 12 whereas Depth First Graph Search finds the plan with path length of 660 which is not optimal. Breadth First search out performs Uniform cost search in terms of number of node expansions and but Breadth First Search takes almost double the time as Uniform Cost Search for the search. Hence, the best uninformed search for Problem-3 is Uniform Cost Search as it takes little more memory and less search time compared to Breadth First Search.

Informed Search:

Searches Considered: A star + h_ignore_preconditions and A star _ h_pg_levelsum

A star search with h_ignore_precondistions finds the optimal plan with the path length of 12. But h_pg_levelsum runs more than 10 mins. Considering h_pg_levelsum running more time, its definitely going to take more search time than h_ignore_preconditions and more time will cause more number of nodes to be expanded further to find the optimal plan. Hence, the best informed search for Problem-3 is A star + h_ignore_preconditions.

Conclusion: For Problem 2 and 3, the informed heuristic search A star search with h_ignore_preconditions is the clear winner in terms of finding the optimal plan with less memory and in less search time. For Problem 1, A star search with h_ignore_preconditions outperforms Uninformed searches in terms of node expansions and slightly taking more time than Breadth First Search. As Problem 1 is the easiest one among the 3 problems, the Uninformed Breadth First Search has little advantage in finding the optimal plan in less time compared to A star with h_ignore_preconditions. Considering the complexity of problems 2 and 3, A star with h_ignore_preconditions is the best search that outperforms other searches in terms of memory and time. So the overall winner is A star with h_ignore_preconditions. The

reason behind informed search is better than the uninformed search is, the informed search has the advantage of knowing the extra information (heuristic) in every state where as the uninformed search does only have the initial problem state and only trying to find the goal by expanding all the possibilities with the goal test. The h_ignore_preconditions heuristic is a relaxed admissible heuristic performs better than h_levelsum which is sub goal independence and inadmissible heuristic.