Heuristic Analysis – Planning

By Ashvin Someshwar

This report presents the results of different search strategies in the Air Cargo Planning problems. This report is presented by problem listing out the best search strategy for each.

Problem 1

Result Table

Strategy	Informed	Plan length (Optimal)	Expansions	Goal Tests	New Nodes	Execution Time (s)
Breadth First Search	No	6 (Yes)	43	56	180	0.03
Breadth First Tree Search	No	6 (Yes)	1458	1459	5960	1.032
Depth First Search	No	20 (No)	21	22	84	0.016
Depth Limited Search	No	50 (No)	101	271	414	0.098
Uniform Cost Search	No	6 (Yes)	55	57	224	0.0403
Recursive Best First Search (h - 1)	Yes	6 (Yes)	4229	4230	17023	2.96
Greedy Best First Graph Search (h - 1)	Yes	6 (Yes)	7	9	28	0.005
A* search (h - 1)	Yes	6 (Yes)	55	57	224	0.0392
A* search (h – ignore preconditions)	Yes	6 (Yes)	41	43	170	0.0391
A* search (h – pg levelsum)	Yes	6 (Yes)	11	13	50	1.058

Optimal Sequence

Plan Length = 6

Load(C1, P1, SFO)
Load(C2, P2, JFK)
Fly(P2, JFK, SFO)
Unload(C2, P2, SFO)
Fly(P1, SFO, JFK)
Unload(C1, P1, JFK)

Uninformed Searches

Among the uninformed searches the execution time for depth first search algorithm is the best coming up with a plan in 0.016s. However the plan is not optimal.

The best uninformed search strategy seems to be the Breadth First Search algorithm which executes in 0.03s and comes up with an optimal plan.

Informed Searches

Surprisingly for this problem the Greedy Best First Search algorithm comes up with an optimal plan in 0.005s with minimum number of node expansions and goal tests. However, this cannot be considered optimal since the algorithm itself is not optimal and the heuristic used is a mock heuristic.

The A* search with ignore preconditions seems to be the best algorithm which executes in 0.0391s. Although, it explores more paths than the levelsum heuristic the search completes much faster because of the simple heuristic function.

Problem 2

Result Table

Strategy	Informed	Plan length (Optimal)	Expansions	Goal Tests	New Nodes	Execution Time (s)
Breadth First Search	No	9 (Yes)	3343	4609	30509	15.636
Depth First Search	No	619 (No)	624	625	5602	4.073
Uniform Cost Search	No	9 (Yes)	4853	4855	44041	14.155
Greedy Best First Graph Search (h - 1)	Yes	21 (No)	998	1000	8982	3.119
A* search (h - 1)	Yes	9 (Yes)	4853	4855	44041	13.312
A* search (h – ignore preconditions)	Yes	9 (Yes)	1450	1452	13303	5.039
A* search (h – pg levelsum)	Yes	9 (Yes)	86	88	841	251.44

Optimal Sequence

Plan Length = 9

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)

Uninformed Searches

Uniform Cost Search performs the best amongst the uninformed searches with an execution time of 14.155s however expands more nodes than the Breadth First Search.

Informed Searches

A* with ignore preconditions performs best with an execution time of 5.039s.

Problem 3

Result Table

Strategy	Informed	Plan length (Optimal)	Expansions	Goal Tests	New Nodes	Execution Time (s)
Breadth First Search	No	12 (Yes)	14663	18098	129631	156.153
Depth First Search	No	392 (No)	408	409	3364	2.662
Uniform Cost Search	No	12 (Yes)	18223	18225	159618	79.88
Greedy Best First Graph Search (h - 1)	Yes	22 (No)	5578	5580	49150	26.138
A* search (h - 1)	Yes	12 (Yes)	18223	18225	159618	87.874
A* search (h – ignore preconditions)	Yes	12 (Yes)	5040	5042	44944	26.616
A* search (h – pg levelsum)	Yes	12 (Yes)	325	327	3002	1635.80

Optimal Sequence

Plan Length = 12

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)

Uninformed Searches

The fastest optimal search is the Uniform Cost Search although it expands more nodes than Breadth First Search.

Informed Searches

The most optimal search is the A* with ignore preconditions heuristic.

Analysis

Among the uninformed search strategies the fastest is the depth first search algorithm. It does not have to expand many nodes and take up much working memory either. However, as seen from the AIMA book on the comparison of algorithms Depth First Search is not guaranteed to complete and the search is not optimal either (Norvig and Rusell 2009). From our analysis **Uniform Cost Search** seems to be the best among the uninformed search strategies in case we need the optimal solution in the fastest execution time.

Criterion	Breadth-First	Uniform-Cost	Depth-First	Depth-Limited	Iterative Deepening	Bidirectional (if applicable)
Complete?	Yes^{a}	Yes^{a,b}	No	No	Yes^{a}	Yes^{a,d}
Time	$O(b^{d+1})$	$O(b^{1+\lfloor C^*/\epsilon \rfloor})$	$O(b^m)$	$O(b^\ell)$	$O(b^d)$	$O(b^{d/2})$
Space	$O(b^{d+1})$	$O(b^{1+\lfloor C^*/\epsilon \rfloor})$	$O(bm)$	$O(b\ell)$	$O(bd)$	$O(b^{d/2})$
Optimal?	Yes^{c}	Yes	No	No	Yes^{c}	Yes^{c,d}

Figure 3.17 Evaluation of search strategies. b is the branching factor; d is the depth of the shallowest solution; m is the maximum depth of the search tree; l is the depth limit. Superscript caveats are as follows: ^{a} complete if b is finite; ^{b} complete if step costs $\geq \epsilon$ for positive ϵ; ^{c} optimal if step costs are all identical; ^{d} if both directions use breadth-first search.

Amongst the informed search strategies the heuristic used made the difference on execution time. The level sum heuristic expands lesser nodes than ignore preconditions. This heuristic assumes the goals are independent and calculates the sum of level costs of individual goals. Whereas the ignore preconditions heuristic calculates the minimum number of actions to be performed to reach the goal state. Ignore preconditions performs much better than level sum because this is simpler than level sum and the cost of calculating the heuristic is much lesser. Hence **A* search with the ignore preconditions** seemed to be the best informed search strategy.

Bibliography

Norvig, Peter, and Stuart Rusell. *Artificial Intelligence : A mordern approach.* Prentice Hall, 2009.