Air Cargo Planning Search: Heuristic Analysis

Vamshidhar Pandrapagada

June 18, 2017

1. Introduction

The goal of this project is to solve deterministic logistics planning problems for an Air Cargo transport system using a planning search agent. We define a group of problems in classical PDDL (Planning Domain Definition Language) for the air cargo domain example as described in Artificial Intelligence: A Modern Approach textbook -3rd Edition, Chapter 10.

The project sets up the problems for search, experiment with various automatically generated heuristics, including planning graph heuristics, to solve the problems, and then provide an analysis of the results.
.

Our Analysis will focus on obtaining the metrics/results for each search type to determine an optimal solution for transporting cargo between the airports.
At first, Let's run uninformed planning searches for 3 different air cargo problems and compute the metrics on number of node expansions required, number of goal tests, time elapsed, and optimality of solution for each search algorithm.

Uninformed Search: Analysis

The following tables show the metrics for problem-search type combination.

Air Cargo Problem 1

Search_Type	Expansions	Goal.Tests	New.Nodes	Time.Seconds	Plan.Length	Optimal
Breadth First Search	44	57	184	0.021	6	Yes
Breadth First Tree Search	1440	1441	5880	0.620	6	Yes
Depth First Graph Search	21	22	84	0.009	20	No
Depth Limited Search	96	248	391	0.055	50	No
Uniform Cost Search	55	57	224	0.025	6	Yes
Recursive Best First Search	4206	4207	16926	1.825	6	Yes
Greedy Best First Search	7	9	28	0.003	6	Yes

Air Cargo Problem 2

Search_Type	Expansions	Goal.Tests	New.Nodes	Time.Seconds	Plan.Length	Optimal
Breadth First Search	3346	4612	30534	9.25	9	Yes
Breadth First Tree Search	-	-	-	> 10 Minutes	-	-
Depth First Graph Search	1124	1125	10017	5.358	1085	No
Depth Limited Search	213491	1967093	1967471	574.151	50	No
Uniform Cost Search	4605	4607	41839	7.603	9	Yes
Recursive Best First Search	-	-	-	> 10 Minutes	-	-
Greedy Best First Search	465	467	4185	0.767	20	No

Air Cargo Problem 3

Search_Type	Expansions	Goal.Tests	New.Nodes	Time.Seconds	Plan.Length	Optimal
Breadth First Search	14120	17673	124926	65.246	12	Yes
Breadth First Tree Search	-	-	-	> 10 Minutes	-	-
Depth First Graph Search	677	678	5608	2.42	660	No
Depth Limited Search	-	-	-	> 10 Minutes	-	-
Uniform Cost Search	16955	16857	149060	33.84	12	Yes
Recursive Best First Search	-	-	-	> 10 Minutes	-	-
Greedy Best First Search	4024	4026	35226	7.861	30	No

At first glance, Breadth First Search and Uniform Cost Search always perform better in reaching the goal with optimal plan.
Depth first search on the other hand is clearly the winner in terms of performance and memory utilization, with fewer node expansions and plan length. This is very obvious because the path to the end goal while traversing down the depth of the tree does not require to store pointers at each level.

If memory utilization is not a constraint (will not be because of its abundance and lower price to obtain) **Breadth First Search** is the recommended strategy. This is because of the following reasons:

1. The main advantages of this technique is it being Optimal and Complete.
2. Even though **Uniform Cost Search** has similar plan length, the number of expansions and goal tests are more when compared to **Breadth First Search**.
3. **Depth First Search** is not optimal and plan lengths are longer.

Informed Heuristic Search: Analysis

A Planned graph is used to construct better heuristic estimates. In this exercise, these heuristics are applied to A-Star search technique.Below we compare informed search result metrics using A* with the “ignore preconditions” and “level-sum” heuristics.

.

Air Cargo Problem 1

Search_Type	Expansions	Goal.Tests	New.Nodes	Time.Seconds	Plan.Length	Optimal
A-Star Search H1	55	57	224	0.025	6	Yes
A-Star Search Ignore Preconditions	41	43	170	0.025	6	Yes
A-Star Search Level Sum	39	41	158	0.786	6	Yes

Air Cargo Problem 2

Search_Type	Expansions	Goal.Tests	New.Nodes	Time.Seconds	Plan.Length	Optimal
A-Star Search H1	4605	4607	41839	7.609	9	Yes
A-Star Search Ignore Preconditions	1311	1313	11989	2.564	9	Yes
A-Star Search Level Sum	997	999	8994	283.350	9	Yes

Air Cargo Problem 3

Search_Type	Expansions	Goal.Tests	New.Nodes	Time.Seconds	Plan.Length	Optimal
A-Star Search H1	16955	16957	149060	32.601	12	Yes
A-Star Search Ignore Preconditions	4444	4446	39227	9.856	12	Yes
A-Star Search Level Sum	1751	1753	15309	984.405	12	Yes

All the three informed heuristic search results yield optimal plan. **A-Star Search Level Sum** heuristic runs for a long time (especially for Problem 2 and 3) but use very less memory. This is still not a desired heuristic due to time complexity.
Compare between **A-Star H1 and A-Star Ignore Preconditions**, Ignore pre-conditions is a clear winner with fewer node expansions and faster performance in returning the goal state.

Comparison: Non-heuristic Uninformed Search vs Heuristic Informed Search

If the problem domain is simpler, the complicated A-Star heuristic search does not add any value either in-terms of optimality or performance. But as the problem gets complicated, using a A-Star search with Ignore-Pre Conditions heuristic is a very evident better performer. For Problem 3, below is the gist of comparison.

Search_Type	Expansions	Goal.Tests	New.Nodes	Time.Seconds	Plan.Length	Optimal
Breadth First Search	14120	17673	124926	65.246	12	Yes
A-Star Search Ignore Preconditions	4444	4446	39227	9.856	12	Yes

Conclusion

Custom heuristics (calculated using planning graphs) when applied to search techniques like A-Star definitely yield better results than Uninformed searches. As we have observed, these informed search strategies are better both in terms of speed and optimality.

References

1. Artifical Intelligence: A Modern Approach (3rd Edition) by Stuart.J.Russell and Peter Norvig