Heuristic Analysis

Ilya Nikokoshev

February 25, 2018

Abstract

As part of the Project 3 in the Artificial Intelligence Nanodegree Program, we implement a planning search agent to solve deterministic logistics planning problems using an example of an Air Cargo transport system.

1 Problem Solutions

Problem 1 Solution

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

Problem 2 Solution

```
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)
```

Problem 3 Solution

```
Load(C2, P2, JFK)
Load(C1, P1, SFO)
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)
```

2 Non-heuristic Searches

We present results for the runs of three uninformed planning strategies in the Table 1. The selected strategies are Breadth First Search, Uniform Cost Search and Depth First Graph Search. Those strategies do not use any additional information about the structure of the problem states.

In line with the results in the Chapter 3 (all references are to the AIMA book), Breadth First Search and Uniform Cost Search always produce the optimal plan.

Figure 3.21 in Section 3.4.7 goes into more detail regarding the time and space complexity of selected methods. In particular, Depth First Graph Search has its own advantages as it uses less memory and is practically faster in our runs. However, the solutions it finds are typically very far from optimal.

Table 1: Results for Non-heuristic planinng methods

	Search Method	Length	Optimal?	Time Elapsed	Expansions	Goal Tests
1	`breadth_first_search`	6	Yes	0.15	43	56
1	`depth_first_graph_search`	12	No	0.04	12	13
1	`uniform_cost_search`	6	Yes	0.19	55	57
2	`breadth_first_search`	9	Yes	45.23	3343	4609
2	`depth_first_graph_search`	575	No	10.36	582	583
2	`uniform_cost_search`	9	Yes	69.89	4853	4855
3	`breadth_first_search`	12	Yes	255.13	14663	18098
3	`depth_first_graph_search`	596	No	7.58	627	628
3	`uniform_cost_search`	12	Yes	184.44	18223	18225

3 Heuristic Searches

We present results for the runs of an A* search with two heuristics, Ignore Preconditions (`h_ignore_preconditions`) and Level Sum (`h_pg_levelsum`), in the Table 2.

As proven in Section 3.5.2, A* search always finds optional solution, provided that the heuristics is consistent. This is confirmed by our results.

Clearly, using A* search with Level Sum heuristics requires less Node Expansions and Goal Tests to achieve optionality than any other method. However, this heuristics is computationally expensive and Ignore Preconditions performs a search much faster in practice. For problems 2 and 3 this heuristics also dominates Breadth First Search and Uniform Cost Search. The problem 1 is simple enough that it can be solved slightly faster by Breadth First Search.

Table 2: Results for Heuristic planinng methods

	Heuristic	Length	Optimal?	Time Elapsed	Expansions	Goal Tests
1	`h_ignore_preconditions`	6	Yes	0.19	41	43
1	`h_pg_levelsum`	6	Yes	0.68	11	13
2	`h_ignore_preconditions`	9	Yes	31.40	1450	1452
2	`h_pg_levelsum`	9	Yes	61.17	86	88
3	`h_ignore_preconditions`	12	Yes	127.46	5040	5042
3	`h_pg_levelsum`	12	Yes	333.36	325	327

4 Conclusion

We have compared five different strategies of finding a solution to Air Cargo problem. Using informed search we were able to achieve a better balance by finding an optional solution faster using A* with domain-independent Ignore Preconditions heuristic.