Ambadipudi, Vidya

Aritificial Intelligence NanoDegree  Udacity

Heuristic Analysis AIND Planning

Table of Contents

[Air Cargo Problem Heuristic Analysis 1](#_Toc509221644)

[Uninformed planning search results 1](#_Toc509221645)

[A\* search with heuristics 1](#_Toc509221646)

# Air Cargo Problem Heuristic Analysis

Following tables show all the results obtained by run\_search.py on each of the problems and the analysis for each of the problems. The goal is to find an optimal solution for each of the problems.

## Uninformed planning search results

After running the uninformed searches for each of the problems, here are the results.

1. All searches return results within 10 mins.
2. From the following table we see that **DFGS** can find the solution in shortest time but it has a **larger plan length** which also increases with problem complexity.
3. Even though **BFS** takes longer time, it always finds the shortest possible plan length. Hence it seems to be an **optimal solution**.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Problem | Search | Expansions | Goal Tests | New Nodes | Time elapsed | Plan length |
| P1 | breadth\_first\_search | 43 | 56 | 180 | 0.02927613532056832 | 6 |
| P1 | depth\_first\_graph\_search | 21 | 22 | 84 | 0.020789620444058136 | 20 |
| P1 | uniform\_cost\_search | 55 | 57 | 224 | 0.03278008878444154 | 6 |
| P2 | breadth\_first\_search | 3401 | 4672 | 31049 | 7.923552268753231 | 9 |
| P2 | depth\_first\_graph\_search | 1192 | 1193 | 10606 | 8.026055367750866 | 1138 |
| P2 | uniform\_cost\_search | 4761 | 4763 | 43206 | 11.078699247896022 | 9 |
| P3 | breadth\_first\_search | 3401 | 4672 | 31049 | 7.923552268753231 | 9 |
| P3 | depth\_first\_graph\_search | 1192 | 1193 | 10606 | 8.026055367750866 | 1138 |
| P3 | uniform\_cost\_search | 4761 | 4763 | 43206 | 11.078699247896022 | 9 |

## A\* search with heuristics

After running the A\* search for each problem with heuristics we see that

1. Search with heuristic ignore\_preconditions always returns results within 10 minutes but for pg\_levelsum it takes more than 10 minutes for Problem 2 and 3.
2. Looking at node expansions and goal test for pg\_levelsum it is a good heuristic and performs better for a more complex problem P3, but overall it is very slow due to complex calculations involved.
3. Ignore\_preconditions performance is optimal as it is considerably faster than levelsum even with large node expansions and goal tests.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Problem | Search | Expansions | Goal Tests | New Nodes | Time elapsed | Plan length |
| P1 | astar\_search h\_ignore\_preconditions | 41 | 43 | 170 | 0.03652682695491718 | 6 |
| P1 | astar\_search h\_pg\_levelsum | 11 | 13 | 50 | 0.8604882804461651 | 6 |
| P2 | astar\_search h\_ignore\_preconditions | 1450 | 1452 | 13303 | 4.071257500178407 | 9 |
| P2 | astar\_search h\_pg\_levelsum | 86 | 88 | 841 | 161.20797474132857 | 9 |
| P3 | astar\_search h\_ignore\_preconditions | 5003 | 5005 | 44586 | 15.181570110974887 | 12 |
| P3 | astar\_search h\_pg\_levelsum | 311 | 313 | 2863 | 800.3601679028045 | 12 |