Natacha Fernandez
AIND Project 3: Planning Search
March 27, 2017

# Heuristic Analysis

This analysis compares the performance of the algorithms in the *Air Cargo Problems,* including optimality, time elapsed, and number of node expansions required. See results on Fig. 1, 2 & 3.

### Fig. 1. Air Cargo Problem 1 Results Table

| Search | Length | Expansions | Goal Tests | New Nodes | Time (sec) | Optimal |
|---|---|---|---|---|---|---|
| BREATH FIRST | 6 | 43 | 56 | 180 | 0.048 | Yes |
| DEPTH FIRST GRAPH | 20 | 21 | 22 | 84 | 0.019 | No |
| UNIFORM COST | 6 | 55 | 57 | 224 | 0.059 | Yes |
| A* H1 | 6 | 55 | 57 | 224 | 0.060 | Yes |
| A* IGNORE PRECOND | 6 | 41 | 43 | 170 | 0.063 | Yes |
| A* LEVEL SUM | 6 | 11 | 13 | 50 | 5.352 | Yes |

### Fig. 2. Air Cargo Problem 2 Results Table

| Search | Length | Expansions | Goal Tests | New Nodes | Time (sec) | Optimal |
|---|---|---|---|---|---|---|
| BREATH FIRST | 9 | 3343 | 4609 | 30509 | 18.501 | Yes |
| DEPTH FIRST GRAPH | 619 | 624 | 625 | 5602 | 4.524 | No |
| UNIFORM COST | 9 | 4852 | 4854 | 44030 | 59.717 | Yes |
| A* H1 | 9 | 4852 | 4854 | 44030 | 63.208 | Yes |
| A* IGNORE PRECOND | 9 | 1506 | 1508 | 13820 | 20.498 | Yes |
| A* LEVEL SUM | 9 | 86 | 88 | 841 | 1196.907 | Yes |

### Fig. 3. Air Cargo Problem 3 Results Table

| Search | Length | Expansions | Goal Tests | New Nodes | Time (sec) | Optimal |
|---|---|---|---|---|---|---|
| BREATH FIRST | 12 | 14663 | 18098 | 129631 | 140.925 | Yes |
| DEPTH FIRST GRAPH | 392 | 408 | 409 | 3364 | 2.355 | No |
| UNIFORM COST | 12 | 18235 | 18237 | 159716 | 530.175 | Yes |
| A* H1 | 12 | 18235 | 18237 | 159716 | 642.133 | Yes |
| A* IGNORE PRECOND | 12 | 5118 | 5120 | 45650 | 121.807 | Yes |
| A* LEVEL SUM | 12 | 404 | 406 | 3718 | 8736.479 | Yes |

**Non-Heuristic Search Strategies**

Even though all three non-heuristic strategies (BFS[1], DFGS[2] and UCS[3]) do find a solution for the air cargo problems, only BFS and UCS are optimal strategies and produce the shortest path. With DFGS out of contention due to lack of optimality, our results show that BFS is the best non-heuristic strategy to solve all three problems (based on the time elapsed).

It's worth noting that DFGS have small memory requirements and is considerably faster than BFS and UCS, making it suitable for solving problems where optimality is not critical, as long as there's no need to keep track of explored nodes.

**Heuristic Search Strategies**

All three heuristic strategies are optimal and complete. Based on the time elapsed, we can conclude that *A*[4] *Search with ignore preconditions* is the best strategy, as it relaxes the problem making it easier to solve. However, note that it only outperforms BFS in the third problem.

The time complexity of *A* Search with level sum* is too high, making it a poor strategy.

## Conclusion:

Considering problems 1 and 2 are quite simple, with small state spaces, a non-heuristic approach such as BFS is all there's needed to reach a satisfactory solution. However, for more complex problems, finding a good plan can be too expensive without adequate heuristics.

Fig. 4. Best Search Strategies for each Air Cargo Problem

| PROBLEM 1 | PROBLEM 2 | PROBLEM 3 |
|---|---|---|
| Best Strategy - length 6:<br>**BREATH FIRST SEARCH** | Best Strategy - length 9:<br>**BREATH FIRST SEARCH** | Best Strategy - length 12:<br>**A* W/ IGNORE PRECOND** |
| Load(C1, P1, SFO)<br>Load(C2, P2, JFK)<br>Fly(P2, JFK, SFO)<br>Unload(C2, P2, SFO)<br>Fly(P1, SFO, JFK)<br>Unload(C1, P1, JFK) | Load(C1, P1, SFO)<br>Load(C2, P2, JFK)<br>Load(C3, P3, ATL)<br>Fly(P2, JFK, SFO)<br>Unload(C2, P2, SFO)<br>Fly(P1, SFO, JFK)<br>Unload(C1, P1, JFK)<br>Fly(P3, ATL, SFO)<br>Unload(C3, P3, SFO) | Load(C2, P2, JFK)<br>Fly(P2, JFK, ORD)<br>Load(C4, P2, ORD)<br>Fly(P2, ORD, SFO)<br>Unload(C4, P2, SFO)<br>Load(C1, P1, SFO)<br>Fly(P1, SFO, ATL)<br>Load(C3, P1, ATL)<br>Fly(P1, ATL, JFK)<br>Unload(C3, P1, JFK)<br>Unload(C2, P2, SFO)<br>Unload(C1, P1, JFK) |

---

[1] BFS - Breath First Search    [2] DFGS - Depth First Graph Search    [3] UCS - Uniform Cost Search    [4] A* - A Star