# AIND Planning – Written Analysis

## *Overall performance of search algorithms*

| Non-heuristic search | | | | | | |
|---|---|---|---|---|---|---|
| Problem | Search Algorithm | Node Expansions | Goal tests | New nodes | Time taken (s) | Plan length |
| Air_cargo_p1 | Breadth-first | 43 | 56 | 180 | 0.1199 | 6 |
| | Depth-first graph | 12 | 13 | 48 | 0.03399 | 12 |
| | Uniform | 55 | 57 | 224 | 0.1389 | 6 |
| Air_cargo_p2 | Breadth-first | 3346 | 4612 | 30534 | 46.34 | 9 |
| | Depth-first graph | 859 | 860 | 7745 | 14.11 | 846 |
| | Uniform | 4853 | 4855 | 4401 | 58.24 | 9 |
| Air_cargo_p3 | Breadth-first | 14120 | 17673 | 124926 | 262.29 | 12 |
| | Depth-first graph | 1401 | 1402 | 11649 | 24.45 | 1345 |
| | Uniform | 18223 | 18225 | 159618 | 253.23 | 12 |
| Heuristic search | | | | | | |
| Air_cargo_p1 | AStar with h_1 | 55 | 57 | 224 | 0.144 | 6 |
| | AStar with ignore_precondition | 41 | 43 | 170 | 0.1163 | 6 |
| | AStar with level_sum | 11 | 13 | 50 | 0.5 | 6 |
| Air_cargo_p2 | AStar with h_1 | 4853 | 4855 | 44041 | 57.25 | 9 |
| | AStar with ignore_precondition | 1450 | 1452 | 13303 | 18.07 | 9 |
| | AStar with level_sum | 86 | 88 | 841 | 37.75 | 9 |
| Air_cargo_p3 | AStar with h_1 | 18223 | 18225 | 159618 | 259.07 | 12 |
| | AStar with ignore_precondition | 5040 | 5042 | 44944 | 80.29 | 12 |
| | AStar with level_sum | 325 | 327 | 3002 | 184.29 | 12 |

## *Optimal Plans for Each Problem:*

### *Problem 1*

```
Plan length: 6  Time elapsed in seconds: 0.4995792380000239
Load(C1, P1, SFO)
Fly(P1, SFO, JFK)
Load(C2, P2, JFK)
Fly(P2, JFK, SFO)
Unload(C1, P1, JFK)
Unload(C2, P2, SFO)
```

### *Problem 2*

```
Plan length: 9  Time elapsed in seconds: 37.747036611
Load(C1, P1, SFO)
Fly(P1, SFO, JFK)
Load(C2, P2, JFK)
```

```
Fly(P2, JFK, SFO)
Load(C3, P3, ATL)
Fly(P3, ATL, SFO)
Unload(C3, P3, SFO)
Unload(C2, P2, SFO)
Unload(C1, P1, JFK)
```

*Problem 3*

```
Plan length: 12  Time elapsed in seconds: 184.28801293100003
Load(C2, P2, JFK)
Fly(P2, JFK, ORD)
Load(C4, P2, ORD)
Fly(P2, ORD, SFO)
Load(C1, P1, SFO)
Fly(P1, SFO, ATL)
Load(C3, P1, ATL)
Fly(P1, ATL, JFK)
Unload(C4, P2, SFO)
Unload(C3, P1, JFK)
Unload(C2, P2, SFO)
Unload(C1, P1, JFK)
```

As a whole, the best heuristic search schemes outperform the best non-heuristic ones, with the level sum heuristic giving the overall best performance, especially in the more complicated search problems. However, the worst performing heuristic search (AStar with h1) performs worse than some of the non-heuristic searches.

For non-heuristic search, we can see that overall, a depth-first search returns us the results very quickly compared to breadth-first or uniform search. This could be due to the fact that going deep down a branch is much faster than expanding the network at each level, as with breadth-first search. However, the overall plan length is extremely long as not always the best move is made.

As suggested by Russell & Norvig in Chapter 10.3, planning graphs provide a data structure that could have better heuristics [1]. Additionally, these heuristics are generic and hence applicable across any problem. For example, in using heuristic search with planning graphs, we can see that all schemes return the shortest possible plan length. The level sum heuristic explores the least number of nodes compared to all other schemes, suggesting that the estimation is a rather good estimation of the problem. However, it takes a considerable amount of time compared to the ignore preconditions heuristic. The ignore preconditions heuristic, however, explores much more nodes compared to the level sum. This could be due to the fact that we essentially “add” edges to the graphical model, meaning that we need to visit more nodes in order to find the optimal solution.

The best heuristic search is the AStar with level sum heuristic search. It performs much better than the best performing non-heuristic search, the depth-first graph. It also manages to return an optimal plan length. It appears as since each move is independent from each other, the level sum gives a good estimate of the cost to the overall goal. However, in terms of search speed, it is unable to match the depth-first search. Overall, AStar with level sum heuristic search is optimal in terms of memory consumption and overall plan length; it searches longer but each search appears to be more informative in that it does not need to explore so many options.

## Works Cited

[1] S. Russell and P. Norvig, Artificial Intelligence: A Modern Approach, New Jersey: Pearson, 2010.