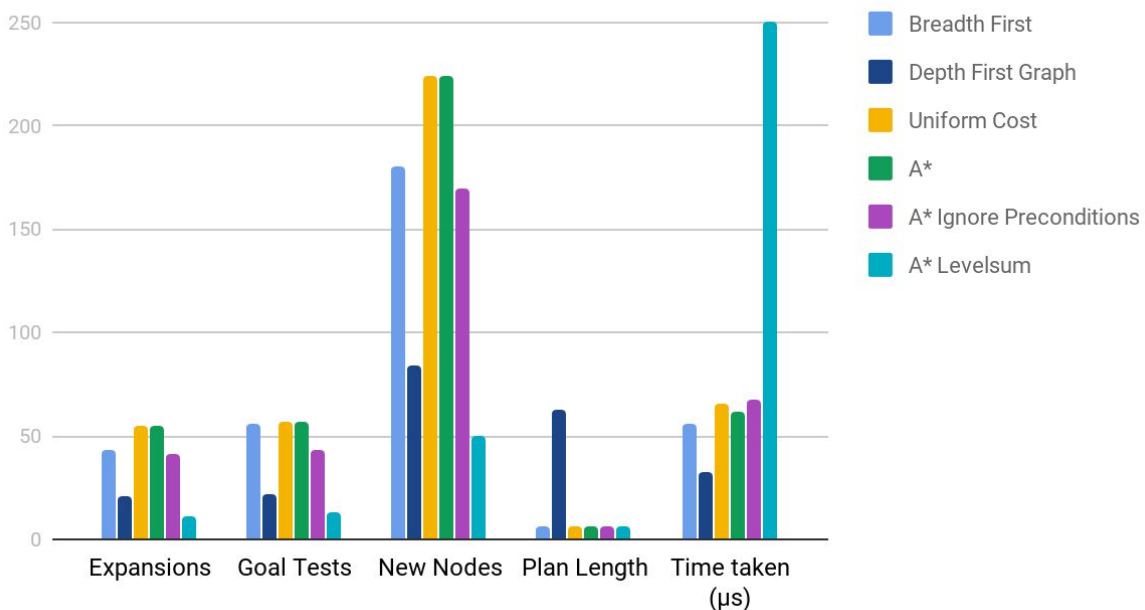


AIND-PLANNIG Written Analysis

Pulasthi Bandara

Analysis

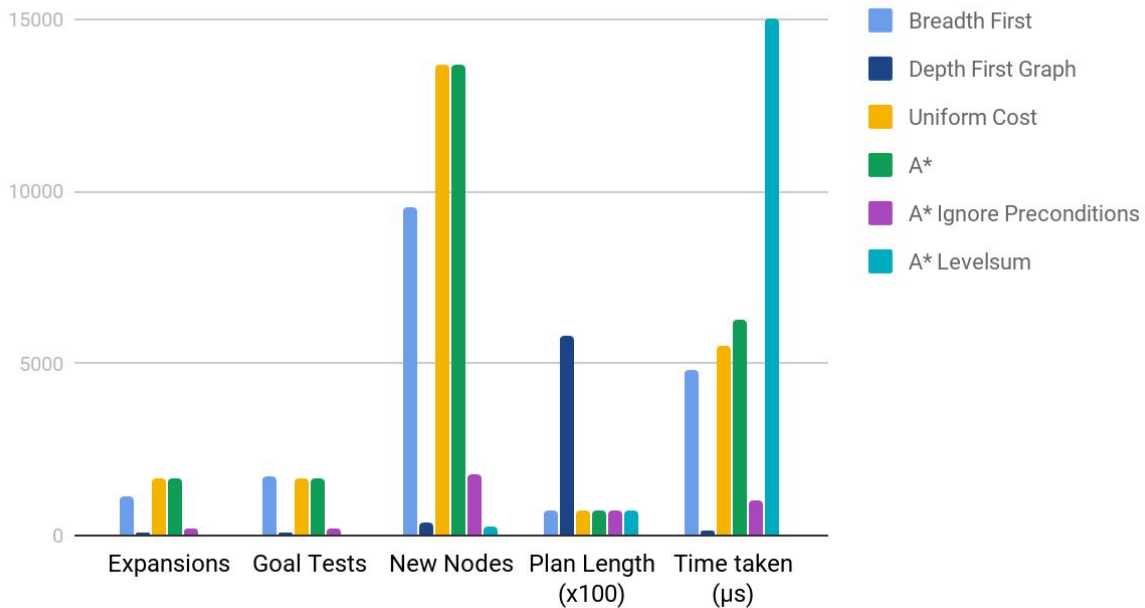
Problem 1



(Levelsum search took 2116.9063770212233 μs)

All algorithms considered, managed to find the optimal solution (6 steps) except for depth-first search(20 steps) which was 4 times more expensive. However when considering the time taken for execution, depth-first search executed the fastest just taking 32μs. Most of the other algorithms took as much as twice the the to finish with exception being the A* levelsome search. It took more than 2000μs to return a result. which is more than 40 times longer when compared to the average execution time of other services. When considering the number of expansions and new nodes, A*, A* ignore preconditions and breadth first search traversed twice as much nodes when compared with depth-first or level some algorithms. One notable observation is that for problem 1 uniform cost search and A*h1 search returned identical results for all metrics.

Problem 2



(Levelsum search took 107899.55452084541 μs)

In problem 2 the results are consistent with the problem 1 result only more evident than the earlier observations. Even in problem 2 depth-first search failed to achieve the optimal search (Optimal being 7 steps and the depth-first result being 58 steps) which was 8 time more expensive (Problem 1 was only 4 times expensive for depth-first search). Similar to problem 1 depth-first search executed the fastest (.1s). One notable difference to observe is A* ignore preconditions executed faster (1s) than rest of the algorithms. A* Levelsum took 100s which is 40 times the average of other algorithms. When considering the nodes processed, breadth-first and A* processed around 40-times more nodes than the rest, where A* level-some processed the least nodes. Similar to Problem 1 Uniform Cost and A* h1 search has identical results.

Problem 3

I couldn't evaluate problem 3 since every execution kept returning this error

```
Expansions      Goal Tests      New Nodes
20736           20737           177984
```

The selected planner did not find a solution for this problem. Make sure you have completed the AirCargoProblem implementation and pass all unit tests first.

I have all unit tests passing. hence, I assume this is due to lack of available memory.

Non-heuristic search Analysis

Considering the three non-heuristic searches tested (breadth-first, depth-first and uniform cost), uniform cost search performed the worst processing most number of nodes and taking the longest time to process. Comparing the differences of breadth-first and depth-first, depth-first search executed the faster while processing the least amount of nodes. However the solution it achieved was not optimal. Breadth-first search managed to achieve an optimal solution but consumed more memory while processing nodes and took longer to return a result.

Heuristic Search Analysis

The most notable difference between levelsum search and ignore-preconditions is that levelsum search is considerably slower than ignore-preconditions. In problem 2 levelsum took 100 times longer (100s vs 1s) to complete. However when considering the number node traversals and goal checks ignore-preconditions consumed more resources (10x more).

Conclusion

Out of all the algorithms considered above ignore-preconditions and levelsum algorithms performed the best. However for very complex problems it is possible that levelsum may take a long time to complete. On the other hand ignore-preconditions has the may consume more memory when required to search deeper. Therefore the actual best algorithm is dependent on the resource availability. If not constrained by available memory ignore-preconditions is more desirable since it delivers optimal results fastest. However, when constrained by available memory or if execution time is not a constraint, a levelsum search is more desirable.

Raw Results

Solving Air Cargo Problem 1 using breadth_first_search...

Expansions	Goal Tests	New Nodes
43	56	180

Plan length: 6 Time elapsed in seconds: 0.055929404916241765
Load(C1, P1, SFO)
Load(C2, P2, JFK)
Fly(P2, JFK, SFO)
Load(C2, P2, SFO)
Fly(P1, SFO, JFK)
Load(C1, P1, JFK)

Solving Air Cargo Problem 1 using depth_first_graph_search...

Expansions	Goal Tests	New Nodes
21	22	84

Plan length: 20 Time elapsed in seconds: 0.032889571972191334

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

Solving Air Cargo Problem 2 using breadth_first_search...

Expansions	Goal Tests	New Nodes
1125	1704	9548

Plan length: 7 Time elapsed in seconds: 4.828249841928482

Load(C2, P2, JFK)
Load(C3, P2, ATL)
Fly(P2, ATL, SFO)
Load(C1, P2, SFO)
Load(C1, P2, JFK)
Load(C2, P2, SFO)
Load(C3, P2, SFO)

Solving Air Cargo Problem 2 using depth_first_graph_search...

Expansions	Goal Tests	New Nodes
62	63	397

Plan length: 58 Time elapsed in seconds: 0.1587238151114434

Fly(P2, ATL, SFO)
Fly(P1, SFO, ATL)
Fly(P2, SFO, JFK)
Fly(P1, ATL, JFK)
Fly(P2, JFK, ATL)
Load(C3, P2, ATL)
Fly(P2, ATL, SFO)
Fly(P1, JFK, ATL)
Fly(P2, SFO, JFK)
Fly(P1, ATL, SFO)
Load(C3, P2, JFK)
Fly(P1, SFO, JFK)
Fly(P2, JFK, ATL)
Fly(P1, JFK, ATL)
Fly(P2, ATL, SFO)
Load(C1, P2, SFO)
Fly(P1, ATL, SFO)
Fly(P2, SFO, ATL)
Fly(P1, SFO, JFK)
Fly(P2, ATL, JFK)

```

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

```

Solving Air Cargo Problem 1 using astar_search with h_1...

Expansions	Goal Tests	New Nodes
55	57	224

Plan length: 6 Time elapsed in seconds: 0.06140073388814926

```

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

```

Solving Air Cargo Problem 2 using astar_search with h_1...

Expansions	Goal Tests	New Nodes
1643	1645	13692

Plan length: 7 Time elapsed in seconds: 6.246298179961741

```

Load(C3, P2, ATL)
Fly(P2, ATL, SFO)
Load(C1, P2, SFO)
Load(C2, P2, JFK)
Load(C3, P2, SFO)
Load(C1, P2, JFK)
Load(C2, P2, SFO)

```

Solving Air Cargo Problem 1 using astar_search with h_ignore_preconditions...

Expansions	Goal Tests	New Nodes
41	43	170

Plan length: 6 Time elapsed in seconds: 0.06736779795028269
Load(C1, P1, SFO)
Fly(P1, SFO, JFK)
Load(C1, P1, JFK)
Load(C2, P2, JFK)
Fly(P2, JFK, SFO)
Load(C2, P2, SFO)

Solving Air Cargo Problem 2 using astar_search with h_ignore_preconditions...

Expansions	Goal Tests	New Nodes
201	203	1774

Plan length: 7 Time elapsed in seconds: 1.0436042819637805
Load(C3, P2, ATL)
Fly(P2, ATL, SFO)
Load(C1, P2, SFO)
Load(C1, P2, JFK)
Load(C3, P2, SFO)
Load(C2, P2, JFK)
Load(C2, P2, SFO)

Solving Air Cargo Problem 1 using astar_search with h_pg_levelsum...

Expansions	Goal Tests	New Nodes
11	13	50

Plan length: 6 Time elapsed in seconds: 2.1169063770212233
Load(C1, P1, SFO)
Fly(P1, SFO, JFK)
Load(C2, P2, JFK)
Fly(P2, JFK, SFO)
Load(C1, P1, JFK)
Load(C2, P2, SFO)

Solving Air Cargo Problem 2 using astar_search with h_pg_levelsum...

Expansions	Goal Tests	New Nodes
21	23	232

Plan length: 7 Time elapsed in seconds: 107.89955452084541
Load(C3, P2, ATL)
Fly(P2, ATL, SFO)
Load(C1, P2, SFO)
Load(C2, P2, JFK)
Load(C3, P2, SFO)
Load(C1, P2, JFK)
Load(C2, P2, SFO)

Solving Air Cargo Problem 1 using uniform_cost_search...

Expansions	Goal Tests	New Nodes
55	57	224

Plan length: 6 Time elapsed in seconds: 0.06532905716449022
Load(C1, P1, SFO)
Load(C2, P2, JFK)
Fly(P1, SFO, JFK)
Fly(P2, JFK, SFO)
Load(C1, P1, JFK)
Load(C2, P2, SFO)

Solving Air Cargo Problem 2 using uniform_cost_search...

Expansions	Goal Tests	New Nodes
1643	1645	13692

Plan length: 7 Time elapsed in seconds: 5.509934653993696
Load(C3, P2, ATL)
Fly(P2, ATL, SFO)
Load(C1, P2, SFO)
Load(C2, P2, JFK)
Load(C3, P2, SFO)
Load(C1, P2, JFK)
Load(C2, P2, SFO)