Heuristic Analysis

July 16, 2017

1 Heuristic Analysis

This document is an analysis that collects results obtained at the end of the implementation of the planning air cargo problem. It is divided into two parts. First part compares and contrasts non-heuristic search result metrics (optimality, time elapsed, number of node expansions) for the three given problems. In the second part the anlysis compares and constrasts heuristic search result metrics using A* with the *ignore preconditions* and *level-sum* heuristics for the three given problem.

1.1 Non-heuristic search results

The following tables shows the results obtained by three non-heuristic search: breadth first search, depth first graph search and uniform cost search. Each table represents test overview for the three Air Cargo problems.

1.1.1 Air Cargo Problem 1

Both **Breadth First Search** and **Uniform Cost Search** are optimals with a plan composed by 6 actions, but **Breadth First Search** is the best choice in terms of time elapsed (0.03 secs), expansions (43) and new nodes (180). We show below the optimal plan elaborated by the **Breadth First Search (BFS)**.

Search	Expansions	Goal Tests	New Nodes	Plan Length	Time elapsed in seconds
Breadth First Search	43	56	180	6	0.039
Depth First Graph Search	21	22	84	20	0.030
Uniform Cost Search	55	57	224	6	0.042

OptimalPlan(BDF) = *Load(C1, P1, SFO) -> Load(C2, P2, JFK) -> Fly(P2, JFK, SFO) -> Unload(C2, P2, SFO) -> Fly(P1, SFO, JFK) -> Unload(C1, P1, JFK)*

1.1.2 Air Cargo Problem 2

Both **Breadth First Search** and **Uniform Cost Search** are optimals with a plan composed by 9 actions. **Uniform Cost Search** is better in terms of time elapsed (11.02 secs), while **Breadth First search** still remain the search algorithm with the best choice for number of expansions (3343) and new nodes (30509). We show below the optimal plan elaborated by the **Uniform Cost Search**

(UCS).

Search	Expansions	Goal Tests	New Nodes	Plan Length	Time elapsed in seconds
Breadth First Search	3343	4609	30509	9	11.919
Depth First Graph Search	624	625	5602	619	3.052
Uniform Cost Search	4853	4855	44041	9	11.028

OptimalPlan(UCS) = *Load(C1, P1, SFO) -> Load(C2, P2, JFK) -> Load(C3, P3, ATL) -> Fly(P1, SFO, JFK) -> Fly(P2, JFK, SFO) -> Fly(P3, ATL, SFO) -> Unload(C3, P3, SFO) -> Unload(C1, P1, JFK) -> Unload(C2, P2, SFO)*

1.1.3 Air Cargo Problem 3

Both **Breadth First Search** and **Uniform Cost Search** are optimals with a plan composed by 12 actions. **Uniform Cost Search** is the best choice in term of time elapsed (46.92 secs), while **Breadth First search** is the best choice in terms of number of expansions (14663) and new nodes (129631). We show below the optimal plan elaborated by the **Uniform Cost Search (UCS)**.

Search	Expansions	Goal Tests	New Nodes	Plan Length	Time elapsed in seconds
Breadth First Search	14663	18098	129631	12	88.568
Depth First Graph Search	408	409	3364	392	1.620
Uniform Cost Search	18223	18225	159618	12	46.929

OptimalPlan(UCS) = *Load(C1, P1, SFO) -> Load(C2, P2, JFK) -> Fly(P1, SFO, ATL) -> Load(C3, P1, ATL) -> Fly(P2, JFK, ORD) -> Load(C4, P2, ORD) -> Fly(P2, ORD, SFO) -> Fly(P1, ATL, JFK) -> Unload(C4, P2, SFO) -> Unload(C3, P1, JFK) -> Unload(C1, P1, JFK) -> Unload(C2, P2, SFO)*

1.2 Heuristic search results

The following tables shows the results obtained by two heuristic search: **A*** with the *ignore preconditions* heurisic and **A*** with the *level-sum* heuristic. Each table represents test overview applied on the three Air Cargo problems.

1.2.1 Air Cargo Problem 1

Both **A* with ignore preconditions** and **A* with the level-sum** are optimals with a plan composed by 6 actions. **A* with ignore preconditions** is the best choice in term of time elapsed (0.04 secs), while **A* with the level-sum** is the best choice in terms of number of expansions (39) and new nodes (158). We show below the optimal plan elaborated by the **A* with ignore preconditions (AIP)**.

Search	Expansions	Goal Tests	New Nodes	Plan Length	Time elapsed in seconds
A* with ignore preconditions	41	43	170	6	0.044

Search	Expansions	Goal Tests	New Nodes	Plan Length	Time elapsed in seconds
A* with the level-sum	39	41	158	6	0.967

OptimalPlan(AIP) = *Load(C1, P1, SFO) -> Fly(P1, SFO, JFK) -> Unload(C1, P1, JFK) -> Load(C2, P2, JFK) -> Fly(P2, JFK, SFO) -> Unload(C2, P2, SFO)*

1.2.2 Air Cargo Problem 2

Both **A* with ignore preconditions** and **A* with the level-sum** are optimals with a plan composed by 9 actions. **A* with ignore preconditions** is the best choice in term of time elapsed (3.63 secs), while **A* with the level-sum** is the best choice in terms of number of expansions (1129) and new nodes (10232). We show below the optimal plan elaborated by the **A* with ignore preconditions (AIP)**.

Search	Expansions	Goal Tests	New Nodes	Plan Length	Time elapsed in seconds
A* with ignore preconditions	1450	1452	13303	9	3.632
A* with the level-sum	1129	1131	10232	9	411.453

OptimalPlan(AIP) = *Load(C3, P3, ATL) -> Fly(P3, ATL, SFO) -> Unload(C3, P3, SFO) -> Load(C1, P1, SFO) -> Fly(P1, SFO, JFK) -> Unload(C1, P1, JFK) -> Load(C2, P2, JFK) -> Fly(P2, JFK, SFO) -> Unload(C2, P2, SFO)*

1.2.3 Air Cargo Problem 3

In this case **A* with ignore preconditions** is the only optimal choice with a plan composed by 9 actions. **A* with ignore preconditions** is the best choice in term of time elapsed (15.00 secs), while **A* with the level-sum** takes a very long time (> 10 min). We show below the optimal plan elaborated by the **A* with ignore preconditions (AIP)**.

Search	Expansions	Goal Tests	New Nodes	Plan Length	Time elapsed in seconds
A* with ignore preconditions	5040	5042	44944	12	15.006
A* with the level-sum	-	-	-	-	>10 min

OptimalPlan(AIP) = *Load(C2, P2, JFK) -> Fly(P2, JFK, ORD) -> Load(C4, P2, ORD) -> Fly(P2, ORD, SFO) -> Unload(C4, P2, SFO) -> Load(C1, P1, SFO) -> Fly(P1, SFO, ATL) -> Load(C3, P1, ATL) -> Fly(P1, ATL, JFK) -> Unload(C3, P1, JFK) -> Unload(C1, P1, JFK) -> Unload(C2, P2, SFO)*

1.3 Considerations

All three chosen *Non-Heuristic search* find a solution to all given problems. **Breadth First Search** finds a solution in a reasonable amount of time and it's always optimal. **Uniform Cost Search** is similar to **Breadth First Search** except that for more complex problem it's the best choice in terms

of time elapsed. Both **Breadth First Search** and **Uniform Cost Search** requires a good amount of memory for the tree expansions. As per [1] **Breadth First Search** is optimal because it always expands the shallowest unexpanded node, while **Uniform Cost Search** is optimal because it expands the node with the lowest path cost.

Depth First Search requires less memory and gets a very quick solution but it is never the optimal one, because it doesn't consider if a node is better than another. In general **Depth First Search** doesn't find the optimal solution because it explores every subtree even if it crosses an optimal node with the result that the final solution could not be optimal (as per [1] section 3.4).

Heuristic Search performs well in *Problem 1*. The best heuristic choice is **A* with ignore preconditions**. In fact it performs better than *Non-heuristic* approaches in terms of required memory, optimality and time elapsed because it's a simple calculus on goals set. In general **A* with the level-sum** performs poorly because it's a more complex heuristic function performing the level cost for each goal and then add them together. On Problem 3 it required more than 10 minutes to stop elaboration, hence this is a symtom of unscalable algorithm. So *Heuristic Search* is not the best approach to solve more complex problems.

In conclusion, the **Breadth First Search** is the best choice that can solve planning problems in a fast and optimal way. It's a robust algorithm that can be used also for complex problem.

1.4 References

- [1] Stuart J. Russel, Peter Norvig (2010). Artificial Intelligence: A Modern Approach (3rd edition).