Heuristic Analysis

Adelene Sim
16th July 2017

Optimal plans for air cargo problems
Optimal plans identified for the air cargo problems are listed below.

Air Cargo Problem 1:

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

Air Cargo Problem 2:

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

Air Cargo Problem 3:

```
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(P1, ATL, JFK)
Unload(C1, P1, JFK)
Unload(C3, P1, JFK)
Fly(P2, ORD, SFO)
Unload(C2, P2, SFO)
Unload(C4, P2, SFO)
```

Compare and contrast non-heuristic search
Figure 1 illustrates the efficacy of three different non-heuristic searches: breadth first search (BFS), depth first search (DFS) and uniform cost search (UCS). While DFS is consistently faster and routinely expands the fewest nodes, it also is unable to identify the optimal paths for all three air cargo problems. Inspecting the output of DFS suggests that DFS tends to lead to unnecessary moves (hence increasing the overall path length). This is likely attributed to first traversing the down the graph rather than across it as in BFS or UCS.

BFS and UCS are more comparable and lead to the same optimal path lengths for all three problems. While BFS consistently expands fewer nodes, it sometimes takes longer than UCS to complete its search. The efficacy of BFS compared to UCS is problem-dependent, as the search expansion depends on the nature of the search tree.

Compare and contrast heuristic search

Figure 2 shows that both A* search heuristics ("ignore preconditions" and "level-sum") led to the optimal path lengths, with the former heuristic consistently faster than the latter. This is because the level-sum approach first builds a planning graph before search, and this planning graph generation takes a considerable amount of time. However, the discrepancy in time lapsed could partially be due to inefficient coding, rather than purely a consequence of a more expensive heuristic and data structure. The "level-sum" heuristic always expands fewer nodes to reach the optimal plan, compared with the "ignore precondition" heuristic.

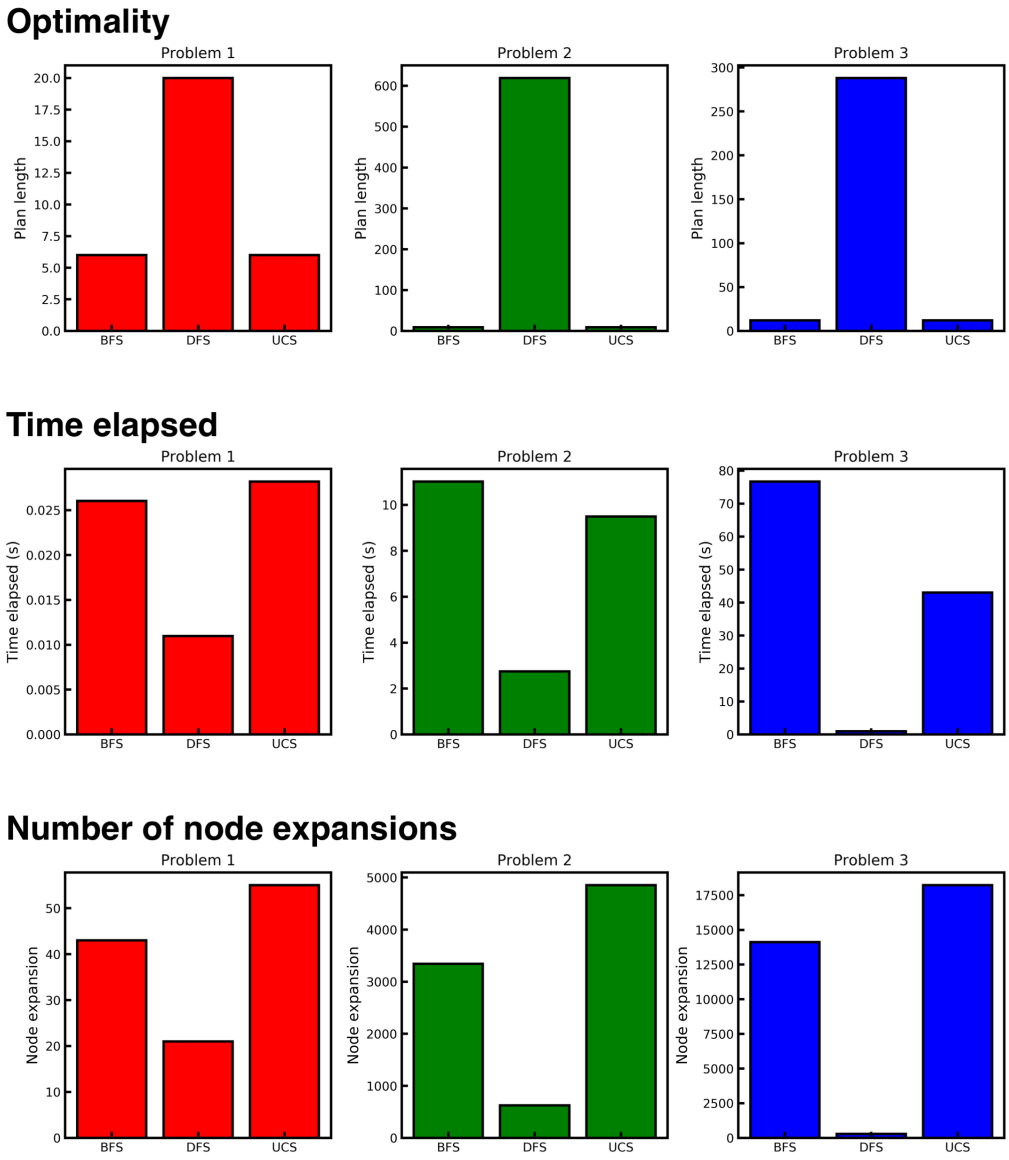

Figure 1: The optimality (path length), time elapsed and number of node expansions for the different air cargo problems using non-heuristic searches BFS, DFS and UCS respectively. While DFS is fast, and expands the least number of nodes, it is consistently unable to yield the optimal path.

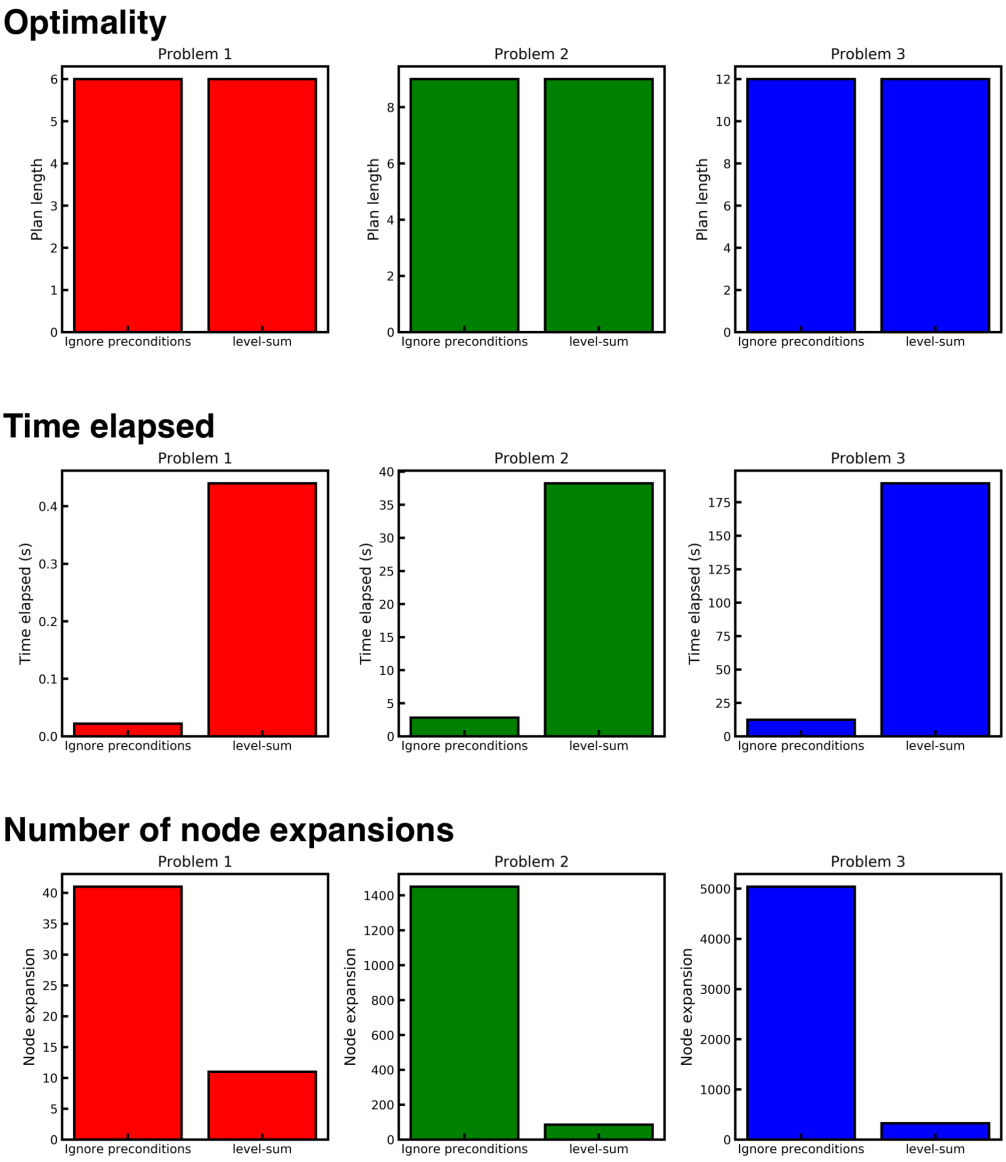

Figure 2: The optimality (path length), time elapsed and number of node expansions for the different air cargo problems using searches with the "ignore preconditions" and "level-sum" heuristics respectively. Both heuristics lead to the same optimal path lengths, but the "ignore preconditions" heuristics consistently run faster. The "level-sum" heuristics however leads to fewer node expansions.

Compare and contrast heuristic and non-heuristic search

Excluding DFS, which led to non-optimal plans, the heuristic searches, as expected, were more efficient in that they explored fewer nodes than the non-heuristic searches. By guiding the search with an admissible heuristic, the search is more directed, and hence fewer nodes are explored. For heuristic search, the "limit-sum" heuristic is most ideal. If run-time is most important, then avoiding the generation of the planning graph is preferred. If so, the "ignore preconditions" heuristic is better.