HEURISTIC ANALYSIS

In the following sections, I will compare the performance of heuristic and non-heuristic search algorithms for the Air Cargo search problem. The algorithsm compared are:

- Non-heuristic: Breadth First Search (BFS), Depth First Graph Search (DFGS), Uniform Cost Search (UCS)
- Heuristic: A* with constant heuristic, A*with ignore preconditions, A* with level sum heuristic.

PROBLEM 1

The table below illustrate the performance of non-heuristic (blue shaded) and heuristic (yellow shaded) search algorithms on Problem 1. Among the non-heuristic methods, BFS and UCS perform similarly since BFS and UCS are equivalent in the setting of uniform, fixed cost for all edges. DFGS expands the fewest nodes, but obtains a plan that is unreasonably long. This is consistent with the depth-first search strategy which returns the first solution found but not necessarily the shallowest solution.

The heuristic method A* with "level-sum" performs the least amount of node expansions and is still able to return an optimal plan of length 6. However, A* with "level-sum" takes longer to arrive at this optimal solution than A* with "ignore preconditions" heuristic (1.0 seconds vs 0.03 seconds). This suggests that the "level-sum" heuristic is more accurate than the "ignore precondition" heuristic (i.e. can better guide the search) but is more expensive to compute.

This is the smallest of the three search problems when considering the size of the state space to be search (number of fluents is 12). Consequently, one expects heuristic and non-heuristic methods to perform comparably. Note how the BFS algorithm and A* algorithm with "ignore preconditions" heuristic perform similarly in terms of number of nodes expanded, plan length, and elapsed time.

	Expansions	Goal Tests	New Nodes	Plan Length	Time Elapsed
BFS	43	56	180	6	0.023 sec
DFGS	21	22	84	20	0.011 sec
UCS	55	57	224	6	0.030 sec
$A^*_{h\_1}$	55	57	224	6	0.029 sec
$A^*_{h\_ignore\_preconditions}$	41	43	170	6	0.034 sec
$A^*_{h\_pg\_level\_sum}$	11	13	50	6	1.009 sec

The optimal plan is of length 6 and corresponds to
Load(C1, P1, SFO)
Load(C2, P2, JFK)
Fly(P2, JFK, SFO)
Unload(C2, P2, SFO)
Fly(P1, SFO, JFK)
Unload(C1, P1, JFK)

PROBLEM 2

The table below illustrate the performance of non-heuristic (blue shaded) and heuristic (yellow shaded) search algorithms on Problem 2. Once again, DFGS expands the fewest nodes, but obtains a plan that is unreasonably long (619 actions) due to favoring the first solution found but not necessarily the shallowest solution. BFS is the best performing non-heuristic method in terms of arriving at an optimal plan of length 9.

Among the heuristic methods, A* with "ignore preconditions" arrives the quickest at an optimal plan of length 9 but expands many more nodes than the "level-sum" heuristic. The level-sum heuristic would have been the best in terms of minimizing work and arriving at the shortest plan length, but it required ~1.5 minutes to return an answer (versus 10 seconds for "ignore preconditions"). So, while the complexity of evaluating the "level-sum" heuristic is high and thus slows down the search, the "level-sum" heuristic does the best job at guiding the search since it is the most accurate heuristic in this set.

Problem 2 is considerably larger than Problem (number of fluents 27 as opposed 12). Consequently, one expects heuristic and non-heuristic method performance to diverge at least in the amount of work done (number of nodes expanded). Note how the BFS algorithm expands nearly twice as many nodes as A* algorithm with "ignore preconditions" heuristic and more than 35 times the number of nodes expanded by A* with "level-sum".

	Expansions	Goal Tests	New Nodes	Plan Length	Time Elapsed
BFS	3343	4609	30509	9	10.01 sec
DFGS	624	625	5602	619	2.5 sec
UCS	4852	4854	44030	9	32.68 sec
$A^*_{h\_1}$	4852	4854	44030	9	32.11 sec
$A^*_{h\_ignore\_preconditions}$	1506	1508	13280	9	10.35 sec
$A^*_{h\_pg\_level\_sum}$	86	88	841	9	100.67 sec

The optimal plan is of length 9 and corresponds to
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)

PROBLEM 3

The table below illustrate the performance of non-heuristic (blue shaded) and heuristic (yellow shaded) search algorithms on Problem 3. As usual, DFGS expands the fewest nodes, but obtains a plan that is unreasonably long (392 actions). BFS is again the best performing non-heuristic method in terms of arriving at an optimal plan of length 12.

Among the heuristic methods, A* with "ignore preconditions" arrives the quickest at an optimal plan of length 12, but expands nearly 12 times as many nodes as the "level-sum" heuristic. However, the level-sum heuristic required ~10 minutes to return an answer (10 times longer than "ignore preconditions"). So as in problem 2, the "level-sum" heuristic does a better job of guiding the search (leads to fewer expanded nodes and an optimal plan) since it is a more accurate heuristic, but is expensive to compute.

Problem 3 is the largest of all the problems (32 fluents). Once again, one expects a big divergence in the performance of heuristic and non-heuristic method. Note how the BFS algorithm expands nearly three times as many nodes as A* algorithm with "ignore preconditions" and 35 times the number of nodes expanded by A* with "level-sum". Furthermore, BFS requires a longer amount of time to return an answer than A* with "ignore preconditions"

	Expansions	Goal Tests	New Nodes	Plan Length	Time Elapsed
BFS	14663	18098	129631	12	75.22 sec
DFGS	408	409	3364	392	1.31 sec
UCS	18235	18237	159716	12	278.23 sec
$A^*_{h\_1}$	18235	18237	159716	12	278.5 sec
$A^*_{h\_ignore\_preconditions}$	5118	5120	45650	12	62.85 sec
$A^*_{h\_pg\_level\_sum}$	408	410	3758	12	670.4 sec

The optimal plan is of length 12 and corresponds to
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)

CONCLUSION

For small problems (Problem 1) non-heuristic approaches are competitive with heuristic based search algorithms. However, for large problems (Problem 3) the divergence between non-heuristic and heuristic based algorithms becomes apparent both in terms of search time and number of nodes expanded.

Accurate heuristics (e.g. level-sum) can drastically reduce the amount of work done by an algorithm relative to an inaccurate heuristic (e.g. ignore precondition). However, if that accurate heuristic is expensive to compute then the search algorithm will take a longer time to arrive at an answer (e.g Time Elapsed for “ignore precondition” and “level sum” on Problem 3 differed by a factor of 10).