Problem 1						
Search Type	Expansions	Goal Tests	New Nodes	Plan Length	Time (s)	Optimal
Breadth First Search	43	56	180	6	0.0362	Yes
Depth First Graph Search	12	13	48	12	0.0095	No
Depth Limited Search	101	271	414	50	0.1093	No
A* Search, h1	55	57	224	6	0.0443	Yes
A* Search, ignore preconditions	41	43	170	6	0.0449	Yes
A* Search, pg levelsum	55	57	224	6	1.5385	Yes

Problem 2						
Search Type	Expansions	Goal Tests	New Nodes	Plan Length	Time (s)	Optimal
Breadth First Search	3343	4609	30509	9	15.546	Yes
Depth First Graph Search	582	583	5211	575	3.2251	No
Depth Limited Search	222719	2053741	2054119	50	1086.7	No
A* Search, h1	4834	4836	43866	9	12.680	Yes
A* Search, ignore preconditions	1450	1452	13303	9	4.7726	Yes
A* Search, pg levelsum	4834	4836	43866	9	1088.2	Yes

Intro

The following tables show the results gathered after solving the air cargo problems for this project with both informed and uninformed searches. The informed searches use `A*` as a basis for the informed searches, with each using a different heuristic to inform its search.

Analysis

With the exception of Depth Limited Search for Problem 3, all search algorithms found a path whereby the goal state could be reached. The failures of Depth-limited Search are expected given the limitations of DFGS are exacerbated by introducing depth limits in finite state space, the contrapositive to the benefits mentioned by Russell, Norvig (2003). As it is inherent to the algorithm, breadth first search managed to find the optimal path in each situation. Conversely, the other two uninformed searches did not. Depth First Graph Search, expectedly, managed shorter search times but its plan lengths are the longest by a significant amount for problems 2 and 3. Problem 1, it appears,

Problem 3						
Search Type	Expansions	Goal Tests	New Nodes	Plan Length	Time (s)	Optimal
Breadth First Search	14663	18098	129631	12	105.94	Yes
Depth First Graph Search	627	628	5176	596	3.3300	No
Depth Limited Search	——	——	——	—	>10 min	No
A* Search, h1	17695	17697	155138	12	54.540	Yes
A* Search, ignore preconditions	5026	5028	44804	12	18.600	Yes
A* Search, pg levelsum	17695	17697	155138	12	6575.6	Yes

had a solution amicable to DFGS – it manages to find a solution faster than any other search algorithm for that particular problem.

Problem 3 is a great example of the cost and time savings of an informed search, but also to choose heuristics wisely. Based on the testing results, the `pg levelsum` heuristic took longer than all other approaches for each problem. This is an example of a case where the f-costs along paths are increasing, and any contours (or bands) from the initial state to the goal state would less concentric, circular circles – more akin to Uniform Cost Search, which is understandable given the only thing that separates them is the heuristic function. An effective heuristic is exampled by `ignore preconditions`, which exhibits the concenctric, expanding circles from the initial state to the goal state – a result of it minimalizing complexity of the problem, as described by Russell, Norvig (2003).

From these results, we can glean that no one search solution is swiss army knife. Uninformed searches can be just as effective as informed searches, and can sometimes outperform them. For informed searches, a heuristic that is overly complex, or just otherwise wildly inaccurate or inconsistent, can bring even the mighty `A*` search to its knees.

```
Optimal Path - Problem 1

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

```
Optimal Path - Problem 2

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

```
Optimal Path - Problem 3

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

References

Russell Stuart J., Norvig Peter. Artificial Intelligence: A Modern Approach. 2003. 2.