Heuristic Analysis

<u>Optimal sequence of actions for each problem</u>

Problem 1 initial state and goal:
Init(At(C1, SFO) ∧ At(C2, JFK)
∧ At(P1, SFO) ∧ At(P2, JFK)
∧ Cargo(C1) ∧ Cargo(C2)
∧ Plane(P1) ∧ Plane(P2)
∧ Airport(JFK) ∧ Airport(SFO))
Goal(At(C1, JFK) ∧ At(C2, SFO))

Problem 1 optimal sequence of actions (Plan Length is 6):
Load(C1, P1, SFO)
Load(C2, P2, JFK)
Fly(P1, SFO, JFK)
Fly(P2, JFK, SFO)
Unload(C1, P1, JFK)
Unload(C2, P2, SFO)

Problem 2 initial state and goal:
Init(At(C1, SFO) ∧ At(C2, JFK) ∧ At(C3, ATL)
∧ At(P1, SFO) ∧ At(P2, JFK) ∧ At(P3, ATL)
∧ Cargo(C1) ∧ Cargo(C2) ∧ Cargo(C3)
∧ Plane(P1) ∧ Plane(P2) ∧ Plane(P3)
∧ Airport(JFK) ∧ Airport(SFO) ∧ Airport(ATL))
Goal(At(C1, JFK) ∧ At(C2, SFO) ∧ At(C3, SFO))

Problem 2 optimal sequence of actions (Plan Length is 9):
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)

Problem 3 initial state and goal:

Init(At(C1, SFO) ∧ At(C2, JFK) ∧ At(C3, ATL) ∧ At(C4, ORD)
∧ At(P1, SFO) ∧ At(P2, JFK)
∧ Cargo(C1) ∧ Cargo(C2) ∧ Cargo(C3) ∧ Cargo(C4)
∧ Plane(P1) ∧ Plane(P2)
∧ Airport(JFK) ∧ Airport(SFO) ∧ Airport(ATL) ∧ Airport(ORD))
Goal(At(C1, JFK) ∧ At(C3, JFK) ∧ At(C2, SFO) ∧ At(C4, SFO))

Problem 3 optimal sequence of actions (Plan Length is 12):
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(C1, P1, JFK)

Unload(C2, P2, SFO)

Unload(C3, P1, JFK)

Unload(C4, P2, SFO)

Results of Uninformed (non-Heuristic) Planning Searches

Air Cargo Problem #1 Results:

Algorithm	Node Expansions Required	Goal Tests	Time Elapsed (seconds)	Path Length	Optimality of Solution
Breadth first search	43	56	0.029215988457881	**6**	**Yes**
Breadth first tree search	1458	1459	0.932160026800977	**6**	**Yes**
Depth first graph search	12	13	0.007999601777935	12	No
Depth limited search	101	271	0.08273260929082116	50	No
Uniform cost search	55	57	0.034877220789246044	**6**	**Yes**
Recursive best first search with h_1	4229	4230	2.5985483067463475	**6**	**Yes**
Greedy best first graph search with h_1	**7**	**9**	**0.0046004920096818225**	**6**	**Yes**

Best results in in bold.

Air Cargo Problem #2 Results:

Algorithm	Node Expansions Required	Goal Tests	Time Elapsed (seconds)	Path Length	Optimality of Solution
Breadth first search	3343	4609	13.246570964317645	**9**	**Yes**
Depth first graph search	**582**	**583**	**2.899518064387925**	575	No
Uniform cost search	4853	4855	39.8513308241656	**9**	**Yes**
Greedy best first graph search with h_1	998	1000	6.5471101048453875	17	No

Best results in in bold

Air Cargo Problem #3 Results:

Algorithm	Node Expansions Required	Goal Tests	Time Elapsed (seconds)	Path Length	Optimality of Solution
Breadth first search	14663	18098	93.84579087573694	**12**	**Yes**
Depth first graph search	**627**	**628**	**3.1331954041697117**	596	No
Uniform cost search	18151	18153	395.2587076508809	**12**	**Yes**
Greedy best first graph search with h_1	5398	5400	96.186572173206	26	No

Best results in in bold.

<u>Results of Heuristic Planning Searches</u>

Air Cargo Problem #1 Results:

Algorithm	Node Expansions Required	Goal Tests	Time Elapsed (seconds)	Path Length	Optimality of Solution
A* search with h_1	55	57	**0.03496136890415056**	**6**	**Yes**
A* search with h ignore preconditions	41	43	0.041971736504992994	**6**	**Yes**
A* search with h_pg levelsum	**11**	**13**	2.982791512971254	**6**	**Yes**

Best results in in bold.

Air Cargo Problem #2 Results:

Algorithm	Node Expansions Required	Goal Tests	Time Elapsed (seconds)	Path Length	Optimality of Solution
A* search with h_1	4853	4855	40.374580395227504	**9**	**Yes**
A* search with h ignore preconditions	1506	1508	**12.930300076918485**	**9**	**Yes**
A* search with h_pg levelsum	**86**	**88**	688.3105468402779	**9**	**Yes**

Best results in in bold.

Air Cargo Problem #3 Results:

Algorithm	Node Expansions Required	Goal Tests	Time Elapsed (seconds)	Path Length	Optimality of Solution
A* search with h_1	18151	18153	352.6416092280062	**12**	**Yes**
A* search with h ignore preconditions	**5118**	**5120**	**90.16661089714137**	**12**	**Yes**

Best results in in bold.

Analysis

A comparison of the metrics of the non-heuristic search algorithms regarding optimality, time elapse and number of node expansions, shows the following:

- Uniform Cost Search and Breadth First Search algorithms consistently obtain the optimal of the solution across all three problems. In contract to the other algorithms which do not consistently achieve optimality of the solution.
- Regarding the running time (time elapsed) of the algorithms, Depth First Graph Search performed be best in terms of running time when compared across the three problem domains.
- Regarding number of node expansions (memory usage), Depth First Graph Search preformed the best in terms of node expansions when compared across the three problem domains. The exception occurring in problem #1 where Greedy Best First Graph Search preformed best but not by a large factor.

These results are not surprising because in (2010, S. Russell and P. Norvig, page 108) it was reported that: "Depth-first search expands the deepest unexpanded node first. It is neither complete nor optimal, but has linear space complexity. Depth-limited search adds a depth bound."

A comparison of the metrics of the heuristic search using A* Search with "ignore preconditions" compared to A* Search with "level-sum" regarding optimality, time elapse and number of node expansions, shows the following:

- Both A* Search with "ignore preconditions" and A* Search with "levelsum" achieve the optimality of solution.
- Regarding time elapse (running time), A* Search with "ignore preconditions" had the best performance compared to A* Search with "levelsum" for problems 1 and 2. There were no metrics for problem 3 for A* Search with "levelsum".
- Regarding number of node expansions (memory usage), A* Search with "levelsum" preformed the best in terms of node expansions when compared with A* Search with "ignore preconditions" across the two of three problem domains. There were no metrics for problem 3 for A* Search with "levelsum".

What was the best heuristic used in these problems?

The best heuristic to use for this problem domain (air cargo) is A* Search with "ignore preconditions" because it is both optimal and uses less memory when compared to the other algorithms.

Was it better than non-heuristic search planning methods for all problems? Why or why not?

The A* Search with "ignore preconditions" algorithm was better than the non-heuristic searching planning algorithms when comparing metrics such as optimality, time elapse and number of node expansions for the problem domain of air cargo planning.

References:

Artificial Intelligence: A Modern Approach (2010, Third Edition), by S. Russell and P. Norvig