Uninformed Search Analysis

The Optimal Path for Air Cargo Problem 1 has length 6 and an example is shown below

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 1:

Search Strategy	Expansions	Path Length	Time elapsed
breadth first search	43	6	0.0325786
breadth first tree search	1458	6	0.9657307
depth first graph search	12	12	0.0075322
depth limited search	101	50	0.0951324
uniform cost search	55	6	0.0387649
recursive best first search	4229	6	2.6420850
greedy best first graph search	7	6	0.0050427

The Optimal Path for Air Cargo Problem 2 has length 9 and an example is shown below

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(C2, P2, SFO)
Unload(C1, P1, JFK)

Problem 2:

Search Strategy	Expansions	Path Length	Time elapsed
breadth_first_search	3401	9	13.9891
depth_first_graph_search	187	187	0.70826
uniform_cost_search	4761	9	12.47857
greedy_best_first_graph_search	550	9	1.375950

The Optimal Path for Air Cargo Problem 3 has length 12 and an example is shown below

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

Problem 3:

Search Strategy	Expansions	Path Length	Time elapsed
breadth_first_search	14491	12	107.40521584003
depth_first_graph_search	3300	3179	51.725192012032494
uniform_cost_search	17783	12	54.70459768897854
greedy_best_first_graph_search	4031	22	12.423212600988336

As we can see above, Breadth First Search and Uniform Cost were the only two search strategies that were able to find the optimal path length. However, Depth First Search was consistently the fastest implementations. The strategy that I would recommend to find the optimal path length would be Uniform Cost, because consistently found the optima path length, much like Breadth First Search, however Uniform Cost performed faster than BFS.

Informed Search Analysis

Problem 1:

Search Strategy	Expansions	Path Length	Time elapsed
A* with h1	55	6	0.04544836300192401
A* with h_ignore_preconditions	41	6	0.039756847952958196
A* with h_pg_levelsum	11	6	1.4439715900225565

Problem 2:

Search Strategy	Expansions	Path Length	Time elapsed
A* with h1	4761	9	15.567265372024849
A* with h_ignore_preconditions	1450	9	6.394095312978607
A* with h_pg_levelsum	86	9	1201.15900225565

Problem 3:

Search Strategy	Expansions	Path Length	Time elapsed
A* with h1	17783	12	75.28117345599458
A* with h_ignore_preconditions	5003	12	23.969229681999423

While all heuristics were able to find the optimal path length only the ignore preconditions heuristic was able to return the answer in a timely manner. This leads me to the conclusion that the recommended heuristic is A* search with the Ignore Preconditions heuristic.

Explanations:
DFS is unable to find the optimal path length because if it chooses the incorrect path then it has to traverse the entire path before it moves on to the next path. BFS is a non-optimal search in general case while UCS is. In our case though where the graph is acyclic and all the edges have equal weights it finds a shortest path, which happens to be an optimal solution in the problem settings. BFS stops as soon as it finds the first path to the goal node. UCS on the other hand explores several alternative paths and chooses the shortest out of them. That being said because BFS uses a queue instead of a stack it is not as efficient as UCS cause it to have a slower execution.
The ignore preconditions heuristic has a faster execution than the level sum because it does not have to worry about satisfying the preconditions of an action, this can cause problems as some action may undo other actions and some operations can achieve multiple goals, but it is still quicker than checking all the possible actions at a level to see if you have reached a goal state.

References:
Artificial Intelligence: A Modern Approach (2010, 3rd Ed.), by S. Russell & P. Norvig