Heuristic Analysis: AI Planning and Search

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Problem** | **Algorithm** | **Expansions** | **Goal Tests** | **New Nodes** | **Plan Length** | **Time Elapsed** |
| **1** | breadth\_first\_search... | 43 | 56 | 180 | **6** | 0.08 |
|  | breadth\_first\_tree\_search... | 1458 | 1459 | 5960 | **6** | 2.10 |
|  | depth\_first\_graph\_search... | 21 | 22 | 84 | **12** | 0.03 |
|  | depth\_limited\_search... | 101 | 271 | 414 | **50** | 0.19 |
|  | uniform\_cost\_search... | 55 | 57 | 224 | **6** | 0.08 |
|  | recursive\_best\_first\_search with h\_1... | 4229 | 4230 | 17023 | **6** | 5.86 |
|  | greedy\_best\_first\_graph\_search with h\_1... | 7 | 9 | 28 | **6** | 0.01 |
|  | astar\_search with h\_1... | 55 | 57 | 224 | **6** | 0.08 |
|  | astar\_search with h\_ignore\_preconditions... | 55 | 57 | 224 | **6** | 0.08 |
|  | astar\_search with h\_pg\_levelsum... | 11 | 13 | 50 | **6** | 2.75 |
| **2** | breadth\_first\_search... | 3343 | 4609 | 30509 | **9** | 78.00 |
|  | breadth\_first\_tree\_search... | - | - | - | **-** | - |
|  | depth\_first\_graph\_search... | 624 | 625 | 5602 | **619** | 20.81 |
|  | depth\_limited\_search... | 222719 | 2053741 | 2054119 | **50** | 4830.82 |
|  | uniform\_cost\_search... | 4853 | 4855 | 44041 | **9** | 152.89 |
|  | recursive\_best\_first\_search with h\_1... | - | - | - | **-** | - |
|  | greedy\_best\_first\_graph\_search with h\_1... | 998 | 1000 | 8982 | **21** | 36.28 |
|  | astar\_search with h\_1... | 4853 | 4855 | 44041 | **9** | 200.88 |
|  | astar\_search with h\_ignore\_preconditions... | 4853 | 4855 | 44041 | **9** | 183.72 |
|  | astar\_search with h\_pg\_levelsum... |  |  |  |  |  |
| **3** | breadth\_first\_search... | 14663 | 18098 | 129631 | **12** | 312.87 |
|  | breadth\_first\_tree\_search... | - | - | - | **-** | - |
|  | depth\_first\_graph\_search... | 408 | 409 | 3364 | **392** | 8.32 |
|  | depth\_limited\_search... | - | - | - | **-** | - |
|  | uniform\_cost\_search... | - | - | - | **-** | - |
|  | recursive\_best\_first\_search with h\_1... | - | - | - | **-** | - |
|  | greedy\_best\_first\_graph\_search with h\_1... | 4255 | 4257 | 37660 | **14** | 324.52 |
|  | astar\_search with h\_1... | - | - | - | **-** | - |
|  | astar\_search with h\_ignore\_preconditions... | 17851 | 17853 | 156483 | **12** | 1947.64 |
|  | astar\_search with h\_pg\_levelsum... | - | - | - | **-** | - |

Christopher Brian Currin

# Optimal Plans

|  |  |  |
| --- | --- | --- |
| Problem 1 (plan length 6) | Problem 2 (plan length 9) | Problem 3 (plan length 12) |
| Load(C1, P1, SFO)  Load(C2, P2, JFK)  Fly(P1, SFO, JFK)  Fly(P2, JFK, SFO)  Unload(C1, P1, JFK)  Unload(C2, P2, SFO) | 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) | 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(C4, P2, SFO)  Unload(C3, P1, JFK)  Unload(C2, P2, SFO)  Unload(C1, P1, JFK) |

# Discussion

For each problem, breadth-first search seems to be the most optimal heuristic and depth first (graph) search the worst. This was judged on accuracy (did it give the correct result) vs time elapsed vs new nodes (favoured less than time). While other heuristics gave the correct length, they were generally slower than a breadth-first search. The depth first search, while consistently the fastest to complete, also gave a sub-optimal plan length. The best automatic heuristics did not perform as well as the best uninformed heuristic; likely due to the relative simplicity of the problem; as automatic heuristics probably favour intuitive over exhaustive searches. Both breadth-first and A\* searches are *complete* and *optimal* and given no actual heuristic, A\* acts equivalent to breadth-first search. Due to the 10-min timeout (sometimes relaxed) many algorithms did not complete to give a result, but for the given results, problem 1 is indicative of the performance of an algorithm on the other problems. We can expect that A\* with h\_g\_levelsum heuristic will provide the optimal solution with the minimal number of new nodes, but in a substantial amount of time compared to another algorithm. Again, in a more elaborate planning problem, the results will likely be favoured away from a breadth-first search result. Further, it depends how good the heuristic is for a given problem. Of course timing is dependent on the simulation hardware, and a dual-core 2011 laptop probably does not provide the best results.