heuristic\_analysis

# Gelei Chen

Part 1 - Planning problems

# Problem 1

### A.Solving Air Cargo Problem 1 using breadth\_first\_search...

Expansions   Goal Tests   New Nodes

    43          56         180

Plan length: 6  Time elapsed in seconds: 0.03546692700183485

Can find an optimal solution.

### B.Solving Air Cargo Problem 1 using depth\_first\_graph\_search...

Expansions   Goal Tests   New Nodes

    12          13          48

Plan length: 12  Time elapsed in seconds: 0.010756155999843031

Can’t find an optimal solution.

### C.Solving Air Cargo Problem 1 using uniform\_cost\_search...

Expansions   Goal Tests   New Nodes

    55          57         224

Plan length: 6  Time elapsed in seconds: 0.04722273099832819

Can find an optimal solution.

# Problem 2

### A.Solving Air Cargo Problem 2 using breadth\_first\_search...

Expansions   Goal Tests   New Nodes

   3346        4612       30534

Plan length: 9  Time elapsed in seconds: 13.357075322001037

Can find an optimal solution.

### B.Solving Air Cargo Problem 2 using depth\_first\_graph\_search...

Expansions   Goal Tests   New Nodes

   859         860         7745

Plan length: 846  Time elapsed in seconds: 5.017339144000289

Can’t find an optimal solution.

### C.Solving Air Cargo Problem 2 using uniform\_cost\_search...

Expansions   Goal Tests   New Nodes

   4714        4716       42806

Plan length: 9  Time elapsed in seconds: 50.23530941900026

Can find an optimal solution.

# Problem 3

### A.Solving Air Cargo Problem 3 using breadth\_first\_search...

Expansions   Goal Tests   New Nodes

  14120       17673       124926

Plan length: 12  Time elapsed in seconds: 108.30074011100078

Can find an optimal solution.

### B.Solving Air Cargo Problem 3 using depth\_first\_graph\_search...

Expansions   Goal Tests   New Nodes

   1401        1402       11649

Plan length: 1345  Time elapsed in seconds: 12.390295859000616

Can’t find an optimal solution.

### C.Solving Air Cargo Problem 3 using uniform\_cost\_search...

Expansions   Goal Tests   New Nodes

  17259       17261       151533

Plan length: 12  Time elapsed in seconds: 334.0809108530011

Can find an optimal solution.

### Part 2 - Domain-independent heuristics

# Problem 1

### A.Solving Air Cargo Problem 1 using astar\_search with h\_ignore\_preconditions...

Expansions   Goal Tests   New Nodes

    41          43         170

Plan length: 6  Time elapsed in seconds: 0.034630326998012606

Can find an optimal solution.

### B.Solving Air Cargo Problem 1 using astar\_search with h\_pg\_levelsum...

Expansions   Goal Tests   New Nodes

    11          13          50

Plan length: 6  Time elapsed in seconds: 2.9545704900010605

Can find an optimal solution.

# Problem 2

### A.Solving Air Cargo Problem 2 using astar\_search with h\_ignore\_preconditions...

Expansions   Goal Tests   New Nodes

   1447        1449       13271

Plan length: 9  Time elapsed in seconds: 11.095083012005489

Can find an optimal solution.

### B.Solving Air Cargo Problem 2 using astar\_search with h\_pg\_levelsum...

Expansions   Goal Tests   New Nodes

    82          84         803

Plan length: 9  Time elapsed in seconds: 838.8579387490026

Can find an optimal solution.

# Problem 3

### A.Solving Air Cargo Problem 3 using astar\_search with h\_ignore\_preconditions...

Expansions   Goal Tests   New Nodes

   4940        4942       43911

Plan length: 12  Time elapsed in seconds: 84.11933344900172

Can find an optimal solution.

### B.Takes longer than 10 minutes. No result

### Part 3 - Written Analysis

### Optimal plan:

* 1. Problem 1:

Load(C1, P1, SFO)

Load(C2, P2, JFK)

Fly(P1, SFO, JFK)

Fly(P2, JFK, SFO)

Unload(C1, P1, JFK)

Unload(C2, P2, SFO)

* 1. 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(C3, P3, SFO)

Unload(C1, P1, JFK)

Unload(C2, P2, SFO)

* 1. Problem 3:

Load(C1, P1, SFO)

Fly(P1, SFO, ATL)

Load(C3, P1, ATL)

Fly(P1, ATL, JFK)

Unload(C3, P1, JFK)

Unload(C1, P1, JFK)

Load(C2, P2, JFK)

Fly(P2, JFK, ORD)

Load(C4, P2, ORD)

Fly(P2, ORD, SFO)

Unload(C4, P2, SFO)

Unload(C2, P2, SFO)

1. Compare and contrast non-heuristic search result metrics

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Optimality | Goal test | Time | Node expansion |
| P1-BFS | YES | 56 | 0.035 | 43 |
| P1-DFS | NO | 13 | 0.010 | 12 |
| P1-Uniform | YES | 57 | 0.047 | 55 |
| P2-BFS | YES | 4612 | 13.357 | 3346 |
| P2-DFS | NO | 860 | 5.017 | 859 |
| P2-Uniform | YES | 4716 | 50.235 | 4714 |
| P3-BFS | YES | 17673 | 108.3 | 14120 |
| P3-DFS | NO | 1402 | 12.39 | 1401 |
| P3-Uniform | YES | 17261 | 334.08 | 17259 |

1. Compare and contrast heuristic search result metrics using A\* with the "ignore preconditions" and "level-sum"

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Optimality | Goal test | Time | Node expansion |
| P1-ignore | YES | 43 | 0.034 | 41 |
| P1-levelsum | YES | 13 | 2.954 | 11 |
| P2-ignore | YES | 1449 | 11.095 | 1447 |
| P2-levelsum | YES | 84 | 838.85 | 82 |
| P3-ignore | YES | 4942 | 84.119 | 4940 |

1. What was the best heuristic used in these problems? Was it better than non-heuristic search planning methods for all problems? Why or why not?

For problem 1, it looks like the best heuristics is ignore preconditions. The performance of P1-ignore is almost the same as P1-BFS.

For problem 2, it looks like the best heuristics is ignore preconditions. And it is better than non-heuristic search planning, because it is faster than P2-BFS, as well as fewer node expansion.

For problem 3, I failed to get results for level sum, because it takes more than 10 minutes to finish. P3-ignore takes much longer time than P3-DFS, but P3-DFS cannot find an optimal solution for problem 3.

* An accurate heuristic needs fewer node expansion than a less accurate one, but requires more time to compute. For example, based on the statistics above, ignore-precondition heuristics is less accurate than level-sum heuristics, takes fewer time to compute and requires more node expansion.
* DFS can’t guarantee an optimal plan
* BFS can find an optimal plan, but it also requires more time and node expansion.
* Uniform cost takes almost same amount of time and node expansion as BFS, but it can find an optimal plan.