AIND Project 3: Implementing a Planning Search
Timotheus Gmeiner

Air Cargo Planning Heuristic Analysis

The three tables below contract compare and contrast non-heuristic search result metrics. The goal is to find the optimal solution for each cargo problem - the search algorithm that finds the shortest path among all possible paths from start to goal.

Uninformed Search Strategy

Uninformed search strategies have no additional information beyond that provided in the problem definition. They can only generate successors and distinguish a goal state from a non-goal state. This section compares four strategies in terms of speed (time elapsed), memory used (node expansions) and optimality (if solution of optimal length is found).

Table 1: Problem 1 Results - Uninformed

Search Type	Optimal (Y/N)	Plan Length	Time Elapsed (s)	Node Expansions	Goal Tests	New Nodes
Breadth First Search (1)	Y	**6**	**0.035**	**43**	56	180
Breadth First Tree Search (2)	Y	**6**	1.05	1458	1459	5960
Depth First Graph Search (3)	N	20	0.175	21	22	84
Uniform Cost Search (5)	Y	**6**	**0.04**	55	57	224

Table 2: Problem 2 Results - Uninformed

Search Type	Optimal (Y/N)	Plan Length	Time Elapsed (s)	Node Expansions	Goal Tests	New Nodes
Breadth First Search (1)	Y	**6**	**1.34**	**477**	1110	4317
Breadth First Tree Search (2)	Y	**6**	173.46	65477	65478	582028
Depth First Graph Search (3)	N	183	**0.71**	**184**	185	1663
Uniform Cost Search (5)	Y	**6**	**3.95**	1462	1464	13408

Table 3: Problem 3 Results - Uninformed

Search Type	Optimal (Y/N)	Plan Length	Time Elapsed (s)	Node Expansions	Goal Tests	New Nodes
Breadth First Search (1)	-	-	-	-	-	-
Breadth First Tree Search (2)	-	-	-	-	-	-
Depth First Graph Search	-	-	-	-	-	-

(3)						
Uniform Cost Search (5)	-	-	-	-	-	-

Analysis of Uninformed Search

Breadth First Search and *Uniform Cost Search* are yield an optimal search strategy under 10 minutes. Depth First Search is the fastest strategy and uses the least memory - but does so sub-optimally. Breadth First Search is optimal, memory efficient, and fast.

What about for problem 3?

Informed (or Heuristic) Search Strategies

Table 1: Problem 1 Results - Informed

Search Type	Optimal (Y/N)	Plan Length	Time Elapsed (s)	Node Expansions	Goal Tests	New Nodes
A*Search & h1 (8)	Y	6	0.43	55	57	224
A*Search & Ignore Preconditions (9)	Y	6	**0.41**	41	43	170
A* Search & Level Sum (10)	Y	6	.75	**11**	13	50

Table 2: Problem 2 Results - Informed

Search Type	Optimal (Y/N)	Plan Length	Time Elapsed (s)	Node Expansions	Goal Tests	New Nodes
A*Search & h1 (8)	Y	6	4.18	1462	1464	13408
A*Search & Ignore Preconditions (9)	Y	6	0.25	67	69	615
A* Search & Level Sum (10)	N	6	11.6	10	12	98

Table 1: Problem 3 Results - Informed

Search Type	Optimal (Y/N)	Plan Length	Time Elapsed (s)	Node Expansions	Goal Tests	New Nodes
A*Search & h1 (8)	-	-	-	-	-	-
A*Search & Ignore Preconditions (9)	-	-	-	-	-	-
A* Search & Level Sum (10)	-	-	-	-	-	-

Analysis of Informed Search

Comparing the heuristic search result metrics using A* with the "ignore preconditions" and "level-sum" heuristics for Problems 1, 2, and 3 base on the same criteria used for the uninformed search strategies (speed, memory, optimality) we find that …. Is the best...

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?