Report

Build a Forward Planning Agent

Analyze the search complexity as a function of domain size, search algorithm, and heuristic

	Air Cargo 1	**Air Cargo 2**	**Air Cargo 3**	**Air Cargo 4**
Actions	20	72	88	104
Breath First Search	43	3343	14663	99736
Depth First Search	21	624	408	25174
Uniform Cost Search	60	5154	18510	113339
Greedy with h unmet goals	7	17	25	29
Greedy with h pg level sum	6	9	14	17
Greedy h pg max level	6	27	21	56
Greedy with pg set level	6	9	35	107
A* h unmet goals	50	2467	7388	34330
A* h pg level sum	28	357	369	1208
A*h pg max level	43	2787	9580	62077
A* h pg set level	33	1037	3423	22606

Table 1: Nodes expanded vs. number of actions for different algorithms and heuristics

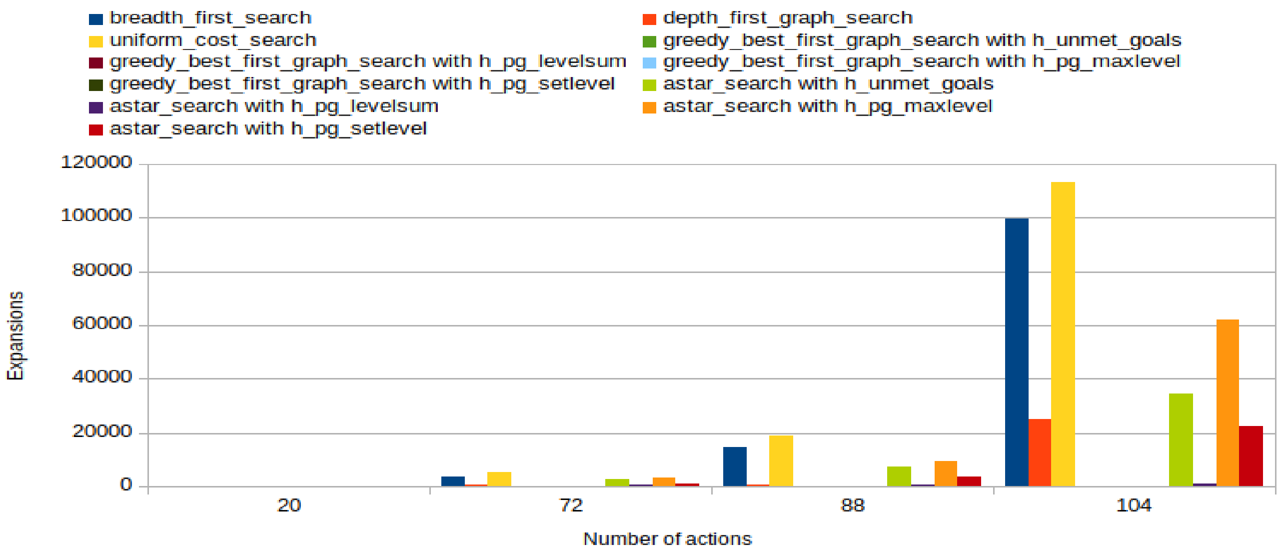

Figure 1: Nodes expanded vs. number of actions for different algorithms and heuristics

It appears from Table 1, Fig. 1, with increased domain size, this number of expanded nodes increases. Uniform cost search was the maximum number of nodes expanded. Greedy Best First Graph Search with different heuristics, in particular the heuristic function of LEVELSUM, had the minimum number of nodes extended.

Analyze search time as a function of domain size, search algorithm, and heuristic

Problem	Air Cargo Problem 1	Air Cargo Problem 2	Air Cargo Problem 3	Air Cargo Problem 4	
Actions	**20**	**72**	**88**	**104**	**Algorithm**
Time (Sec)	0.233186958000033	9.32208232999994	16.7975480700001	94.7388729539998	breadth first search
	0.122530521000044	52.990589304	307.966252391	3156.024047635	depth first graph search
	0.30208527000002	120.21959184	693.805046097	7389.849309276	uniform cost search
	0.015042947000097	0.570422437000047	0.770595014000037	3.88743919199987	greedy best first graph search with h unmet goals
	0.015943917999948	0.270107745000018	0.669564173000026	5.00919776700016	greedy best first graph search with h pg levelsum
	0.00516289799998	0.450499178999962	0.146837311000013	965.863128274	greedy best first graph search with h pg maxlevel
	0.218474410999988	0.369129243999964	0.863370798999995	1.40851894200023	greedy best first graph search with h pg setlevel
	0.103391274000046	0.634742369000037	0.835418521999941	2.59617345700008	astar search with h unmet goals
	0.551640850000013	1.42796521399998	7.44061273200009	34.665185457	astar search with h pg levelsum
	0.00138369299998	0.008520957999963	0.013813171000038	0.023433104000105	astar search with h pg maxlevel
	0.013565766000056	0.545184608999989	1.10464959000001	7.64540633199977	astar search with h pg setlevel

Table 2: Search time vs. number of actions for different algorithms and heuristics

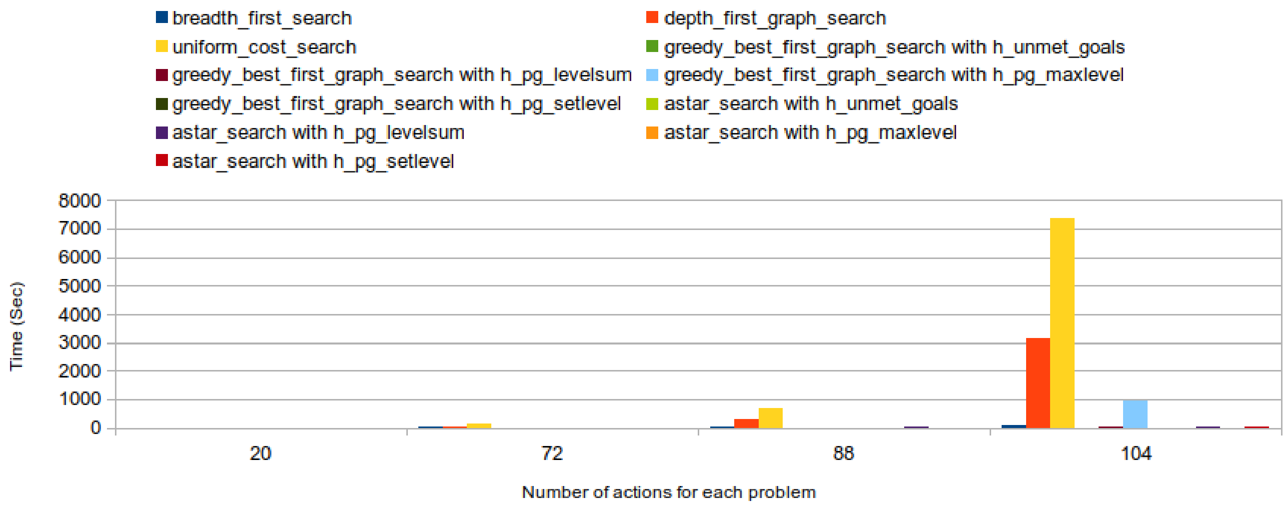

Figure 2: Search time vs. number of actions for different algorithms and heuristics

It is displayed in Table 2, Fig. 2, with increasing domain size, the search time increases. Problem 1, with 20 actions, consumed the shortest search time in all algorithms, while problem 4 took the longest search time with 104 actions. For every problem, A* consumed the shortest search time with MAXLEVEL heuristic, while Uniform cost search consumed the longest search time with problems 2, 3.

Analyze the optimality of solution as a function of domain size, search algorithm, and heuristic

	Air Cargo 1	**Air Cargo 2**	**Air Cargo 3**	**Air Cargo 4**
Actions	20	72	88	104
Breath First Search	6	9	12	14
Depth First Search	20	619	392	24132
Uniform Cost Search	6	9	12	14
Greedy with h unmet goals	6	9	15	18
Greedy with h pg level sum	6	9	14	17
Greedy h pg max level	6	9	13	17
Greedy with pg set level	6	9	17	23
A* h unmet goals	6	9	12	14

A* h pg level sum	6	9	12	15
A*h pg max level	6	9	12	14
A* h pg set level	6	9	12	14

Table 3: Plan length vs. number of actions for different algorithms and heuristics

Table 3 shows that the length of the plan increases with the increase of the domain size. Depth-First has also been shown to have produced the longest plan for all problems. Other than Depth-First algorithms have produced the same plan length for problems 1 and 2. With increasing domain sizes (i.e. problems 3 and 4), the shortest plans were produced by Breadth-First, Uniform-cost and A*.

1. **Which algorithm or algorithms would be most appropriate for planning in a very restricted domain (i.e., one that has only a few actions) and needs to operate in real time?**
   A* with MAXLEVEL heuristics for all problems consumed the shortest time between all algorithms. In addition, A* with MAXLEVEL is one of the shortest plan algorithms. A* with MAXLEVEL would therefore be suitable for the planning of a very limited domain in real time.
2. **Which algorithm or algorithms would be most appropriate for planning in very large domains (e.g., planning delivery routes for all UPS drivers in the U.S. on a given day)?**
   A* with MAXLEVELheuristic consumed the shortest time between all algorithms for all problems, despite A* with MAXLEVEL, many expansions and many new nodes were generated. In addition, A* is one of the algorithms with MAXLEVEL that produced the shortest plans for all problems. A* with MAXLEVEL would therefore be suitable for planning very large domains. Greedy Best First search can also be used in large domains because Greedy Best First Search(with different heuristics) produced minimum expansions and new nodes. With MAXLEVEL heuristic, however, Greedy 's best first search(with different heuristics) took longer than A*.

3. **Which algorithm or algorithms would be most appropriate for planning problems where it is important to find only optimal plans?**
   Uniform Cost and Breadth First (all are identical as step costs). A* is used as a graph, so that consistent (not only acceptable) heuristics are required to be optimal.