Udacity Project - Build a Forward Planning Agent

Jeff Rix

Tables Include all data for questions:

Problem	Algorithm	Actions	Expansions	Goals	New Nodes	Plan Len	Time	% New Nodes to Actions	Time Per Action	% Plan Len to Actions
1	BFS	20	43	56	178	6	0.003	8.90%	0.000150	30.00%
1	DFS	20	21	22	84	20	0.002	4.20%	0.000100	100.00%
1	UCS	20	60	62	240	6	0.004	12.00%	0.000200	30.00%
1	Greedy Unmet Goal	20	7	9	29	6	0.00008	1.45%	0.000004	30.00%
1	Greedy Level Sum	20	6	8	28	6	0.0008	1.40%	0.000040	30.00%
1	Greedy Max Level	20	6	8	24	6	0.04	1.20%	0.002000	30.00%
1	Greedy Set Level	20	6	8	28	6	0.175	1.40%	0.008750	30.00%
1	A* Unmet Goal	20	50	52	206	6	0.005	10.30%	0.000250	30.00%
1	A* Level Sum	20	28	30	122	6	0.128	6.10%	0.006400	30.00%
1	A* Max Level	20	43	42	180	6	0.135	9.00%	0.006750	30.00%
1	A* Set Level	20	33	35	138	6	0.392	6.90%	0.019600	30.00%

Problem	Algorithm	Actions	Expansions	Goals	New Nodes	Plan Len	Time	% New Nodes to Actions	Time Per Action	% Plan Len to Actions
2	BFS	72	3343	4609	30503	9	0.7877	423.65%	0.010940	12.50%
2	DFS	72	624	625	5602	619	1.032	77.81%	0.014333	859.72%
2	UCS	72	5154	5156	46618	9	1.27	647.47%	0.017639	12.50%
2	Greedy Unmet Goal	72	17	19	170	9	0.009	2.36%	0.000125	12.50%
2	Greedy Level Sum	72	9	11	86	9	0.979	1.19%	0.013597	12.50%
2	Greedy Max Level	72	27	29	279	9	1.63	3.88%	0.022639	12.50%
2	Greedy Set Level	72	9	11	84	9	4.076	1.17%	0.056611	12.50%
2	A* Unmet Goal	72	2467	2469	22522	9	0.894	312.81%	0.012417	12.50%
2	A* Level Sum	72	357	359	3426	9	26.19	47.58%	0.363750	12.50%
2	A* Max Level	72	2887	2889	26594	9	148.345	369.36%	2.060347	12.50%

Problem	Algorithm	Actions	Expansions	Goals	New Nodes	Plan Len	Time	% New Nodes to Actions	Time Per Action	% Plan Len to Actions
2	A* Set Level	72	1037	1039	9605	9	361.6	133.40%	5.022222	12.50%

Problem	Algorithm	Actions	Expansions	Goals	New Nodes	Plan Len	Time	% New Nodes to Actions	Time Per Action	% Plan Len to Actions
3	BFS	88	14663	18098	129625	12	4.03	1473.01%	0.045795	13.64%
3	Greedy Unmet Goal	88	25	27	230	15	0.014	2.61%	0.000159	17.05%
3	Greedy Level Sum	88	14	16	126	14	2.24	1.43%	0.025455	15.91%
3	A* Unmet Goal	88	7388	7390	65711	12	3.34	746.72%	0.037955	13.64%
3	A* Level Sum	88	369	371	3403	12	50.39	38.67%	0.572614	13.64%

Problem	Algorithm	Actions	Expansions	Goals	New Nodes	Plan Len	Time	% New Nodes to Actions	Time Per Action	% Plan Len to Actions
4	BFS	104	99736	114953	944130	14	36.63	9078.17%	0.352212	13.46%
4	Greedy Unmet Goal	104	29	31	280	18	0.03	2.69%	0.000288	17.31%
4	Greedy Level Sum	104	17	19	165	17	4.15	1.59%	0.039904	16.35%
4	A* Unmet Goal	104	34330	34332	328509	14	21.06	3158.74%	0.202500	13.46%
4	A* Level Sum	104	1208	1210	12210	15	267.14	117.40%	2.568654	14.42%

Use a table or chart to analyze the number of nodes expanded against number of actions in the domain

Actions to Nodes

Legend: ▬ BFS ▬ Greedy Unmet Goal ▬ A* Unmet Goals
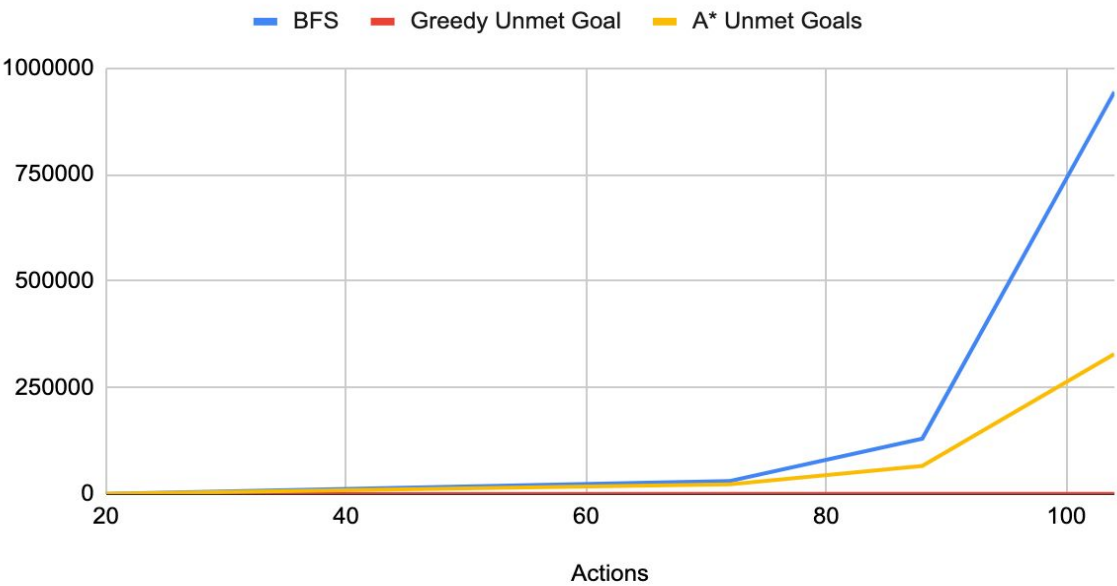
As the problem actions increase the uninformed searches have the greatest increases in new nodes expanded. On problem 3 and 4 BFS saw a 2x - 3x increase in new nodes expanded compared to the worst informed search. The greedy search algorithms consistently open the lowest amount of new nodes ranging from 1.2% to 3.88% (% new nodes to actions) on all problems. The A* search algorithms open a median amount of new nodes compared to the other problems. Depending on the type of heuristic A* search has the largest difference in percentage between other A* heuristics averaging 300% change (A* Unmet compared to A* Level sum problem 4).

Use a table or chart to analyze the search time against the number of actions in the domain

Search Time

Legend: ▬ A* Unmet Goals ▬ A* Level Sum ▬ Greedy Unmet Goal ▬ BFS
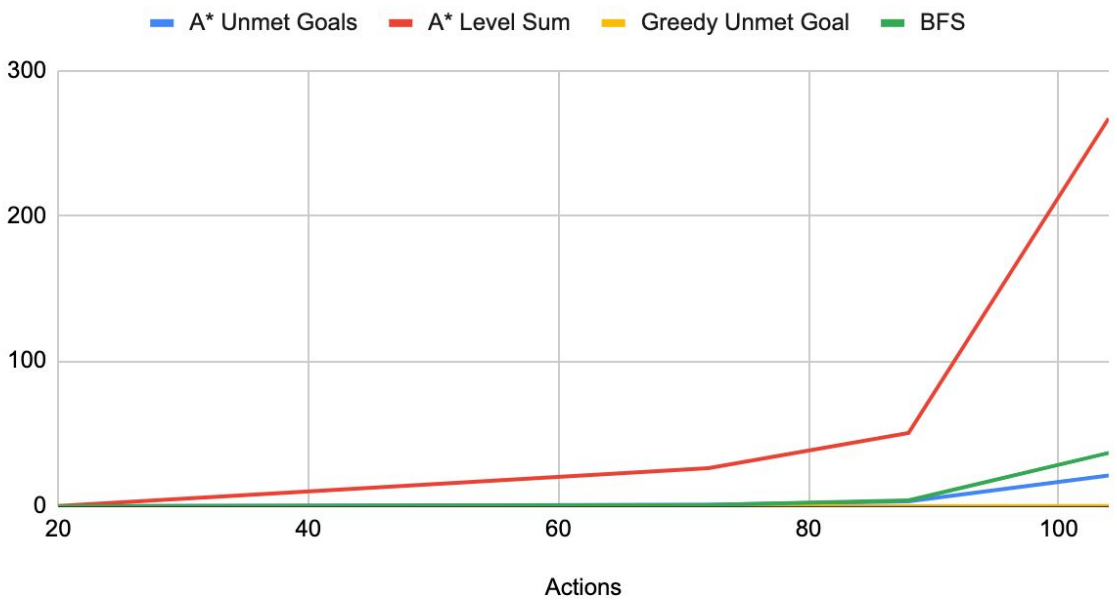
For problem 1 and 2 the search time is similar except in problem 2 the collective A* searches time increases dramatically taking 98% of all search time to actions for problems 1 and 2. This trend continues over the problem 3 for the algorithms I choose, with A* Level Sum consistently having the worst time to actions of 0.572614 seconds/action. The greedy search algorithms continued to achieve the best search time to total actions percentage, with greedy unmet performing the best 0.000159 seconds/action.

Use a table or chart to analyze the length of the plans returned by each algorithm on all search problems

See chart above

For the first two problems the plan lengths are all the same except for DFS, which performs horribly with a 619 plan length for problem 2 compared to a 9 plan length for all the other algorithms. For problems 3 and 4 the plan lengths are all similar, with the A* unmet goal algorithm performing the best with 14 plan length for 104 actions.

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?

Greedy Unmet Goal is the fastest search for problems with a very restricted domain.

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)

Greedy Unmet Goal - quickest but plan length is longer by 23%.
A* Unmet goal - Since this solution is not need in real time A* Unmet goal will find the shortest plan with an exhaustive search of the nodes and it is also the quickest of the A* searches.

Which algorithm or algorithms would be most appropriate for planning problems where it is important to find only optimal plans?

A * Unmet is fast and opens a tremendous amount of new nodes to find the optimal plans.