Classical Planning

Name: Gabriel de Azevedo Camargo

Problem1	Actions	Expansions	Goal Tests	New Nodes	plan length	seconds
breadth_first_search	20	43	56	178	6	0,0059318010000311006
deepth_first_graph_search	20	21	22	84	20	0,00334765799999559
uniform_cost_search	20	60	62	240	6	0,0093032459998233
greedy_best_first_graph_search whith h_unmet_goals	20	7	9	29	6	0,0016420510000898503
greedy_best_first_graph_search whith h_pg_levelsum	20	6	8	28	6	0,28266267999993033
greedy_best_first_graph_search with h_pg_maxlevel	20	6	8	24	6	0,2136006369998995
greedy_best_first_graph_search with h_pg_setlevel	20	6	8	24	6	0,7150617499999101
astar_search with h_unmet_goals	20	50	52	206	6	0,009041450999802692
astar_search with h_pg_levelsum	20	28	30	122	6	0,6817009399999279
astar_search with h_pg_maxlevel	20	43	45	180	6	0,75008318999926
Astar_search with h_pg_setlevel	20	12	14	50	6	1,3110839299999952

Problem2	Actions	Expansions	Goal Tests	New Nodes	plan length	seconds

breadth_first_search	72	3043	4609	3503	9	1,9957514050000214
deepth_first_graph_search	72	624	625	5602	619	2,9882419940000204
uniform_cost_search	72	5154	5156	46618	9	3,134724344999995
greedy_best_first_graph_search whith h_unmet_goals	72	17	19	170	9	0,018505498999957126
greedy_best_first_graph_search whith h_pg_levelsum	72	9	11	86	9	6,099695438999788
greedy_best_first_graph_search with h_pg_maxlevel	72	27	29	249	9	12,354076578000093
greedy_best_first_graph_search with h_pg_setlevel	72	27	29	249	9	54,42036947299994
astar_search with h_unmet_goals	72	2467	2469	22522	9	2,1431403970000247
astar_search with h_pg_levelsum	72	357	359	3426	9	156,11447957099995
astar_search with h_pg_maxlevel	72	2887	2889	26594	9	911,3324272639998
Astar_search with h_pg_setlevel	72	996	998	9307	9	1606,64334339

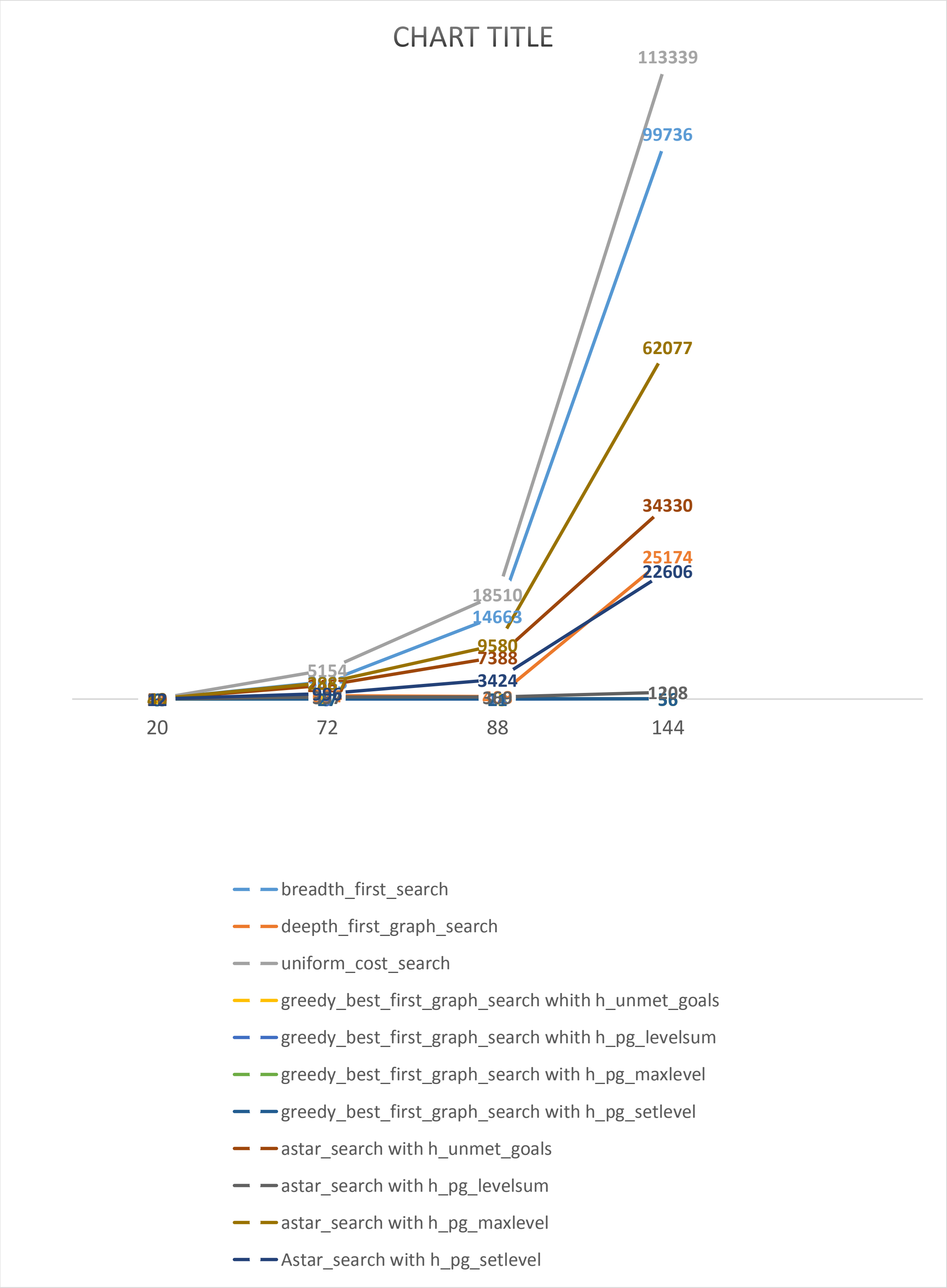
Problem3	Actions	Expansions	Goal Tests	New Nodes	plan length	seconds

breadth_first_search	88	14663	18098	129625	12	10,3917676299975
deepth_first_graph_search	88	408	409	3364	392	1,10222758800046
uniform_cost_search	88	18510	18512	161936	12	13,984674420999909
greedy_best_first_graph_search whith h_unmet_goals	88	25	27	230	15	0,03596143699996901
greedy_best_first_graph_search whith h_pg_levelsum	88	14	16	126	14	14,380704655000045
greedy_best_first_graph_search with h_pg_maxlevel	88	21	23	195	13	17,344076811999912
greedy_best_first_graph_search with h_pg_setlevel	88	21	23	195	13	68,69487972099978
astar_search with h_unmet_goals	88	7388	7390	65711	12	8,134848398999566
astar_search with h_pg_levelsum	88	369	371	3403	12	258,55322256100044
astar_search with h_pg_maxlevel	88	9580	9582	86312	12	1834,3324272639998
Astar_search with h_pg_setlevel	88	3423	3425	31596	12	3806,64334339

Problem4	Actions	Expansions	Goal Tests	New Nodes	plan length	seconds

breadth_first_search	104	99736	114953	944130	14	92,61904181
deepth_first_graph_search	104	25174	25175	228849	24132	2540,9882419940000204
uniform_cost_search	104	113339	113341	1066413	14	96,363681
greedy_best_first_graph_search whith h_unmet_goals	104	56	58	580	17	950,902983
greedy_best_first_graph_search whith h_pg_levelsum	104	56	58	580	17	26,314
greedy_best_first_graph_search with h_pg_maxlevel	104	56	58	580	17	63,20
greedy_best_first_graph_search with h_pg_setlevel	104	56	58	580	17	265,66
astar_search with h_unmet_goals	104	34330	34332	328509	15	57,26924
astar_search with h_pg_levelsum	104	1208	1210	12210	15	1494,79184
astar_search with h_pg_maxlevel	104	62077	62079	599376	14	3011,3324272639998
astar_search with h_pg_setlevel	104	22606	22608	224229	14	6046,64334339

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



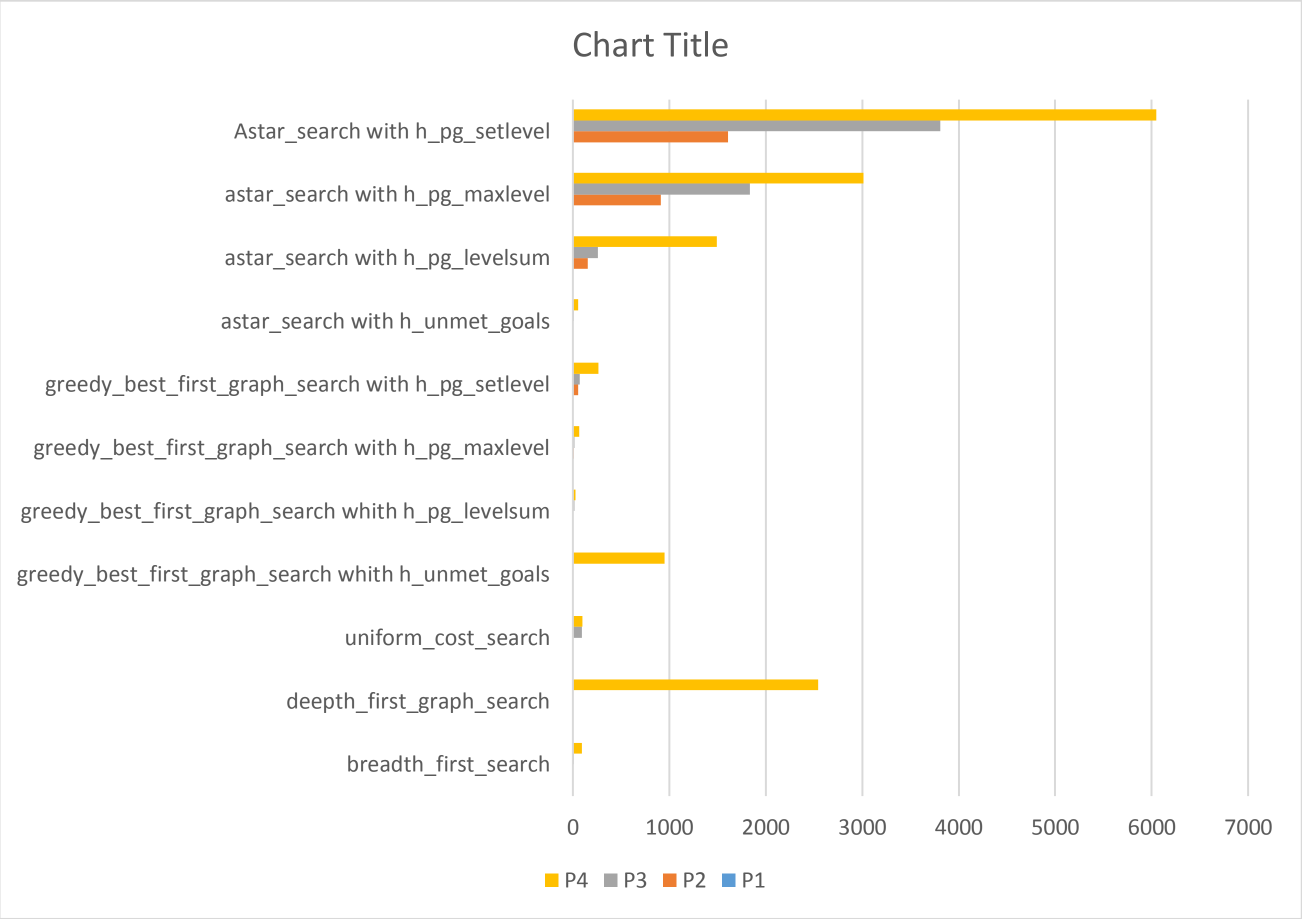
The best algorithm in relation of expands x action is:

- breadth_first_search
- uniform_cost_search
- astar_search with h_unmet_goals
- astar_search with h_pg_setlevel
- astar_search with h_pg_maxlevel

The worst algorithm in relation of time x action is:

- greedy_best_first_graph_search whith h_unmet_goals
- greedy_best_first_graph_search whith h_pg_levelsum
- greedy_best_first_graph_search with h_pg_maxlevel
- greedy_best_first_graph_search with h_pg_setlevel

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



The worst algorithm in relation of time x action is:

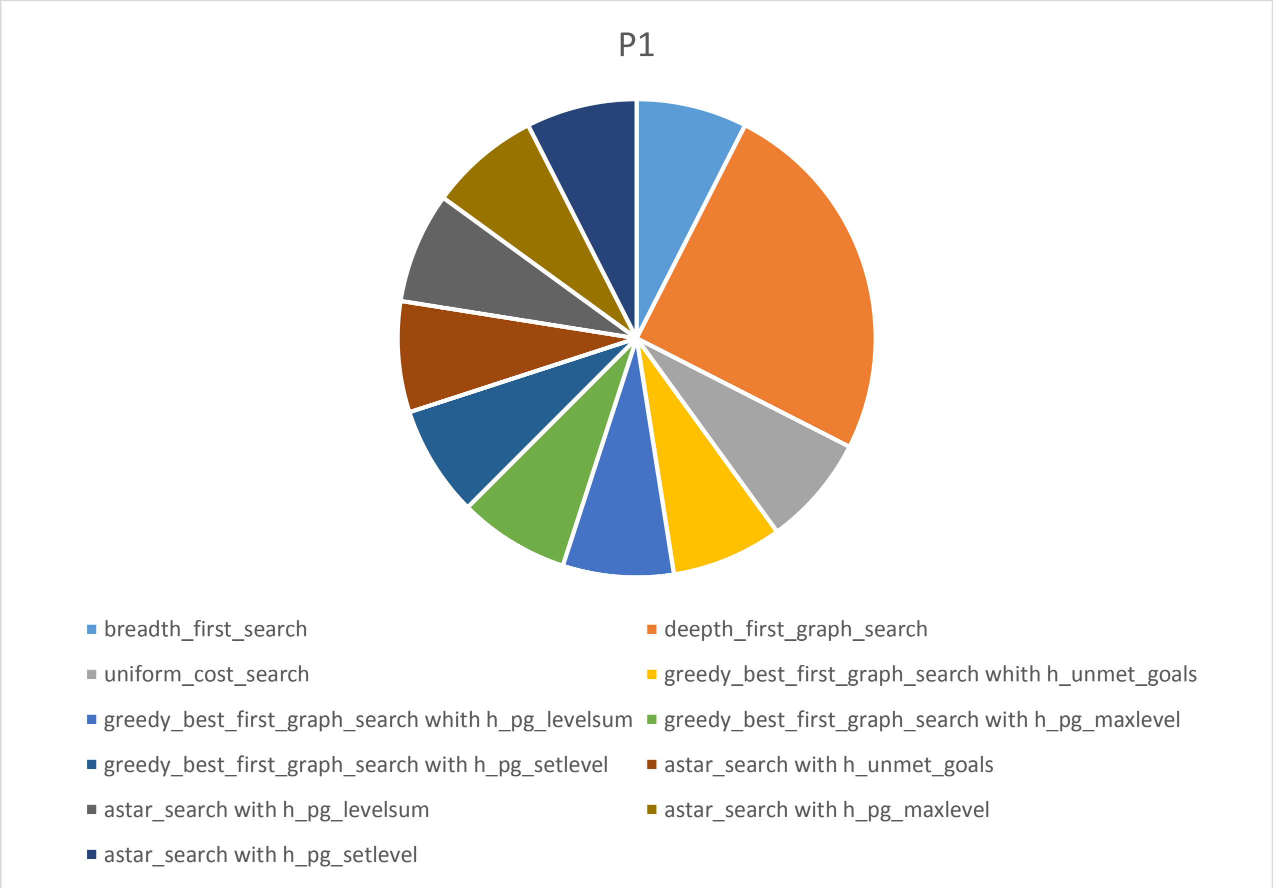
- astar_search with h_pg_levelsum
- astar_search with h_pg_maxlevel
- astar_search with h_pg_setlevel
- deepth_first_graph_search

The best algorithm in raltion of time x action is:

- greedy_best_first_graph_search with h_pg_maxlevel

- uniform_cost_search
- greedy_best_first_graph_search whith h_pg_levelsum
- astar_search with h_unmet_goals

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



Submission includes a short answer to each of the following questions. (A short answer should be at least 1-2 sentences at most a small paragraph.)

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?

R: Have two options:

First: deepth_first_graph_search = 0,003s that is not the best time but the number of expansions and other variables are good.

Second: greedy_best_first_graph_search whith h_unmet_goals = 0,0001 that is the best time to small number of actions like is possible to see in the problem 1

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)

R: astar_search with h_unmet_goals, because that algorithm have the best time x the number of actions.

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

R: uniform_cost_search is the best option but have to: breadth_first_search, astar_search with h_pg_maxlevel, astar_search with h_pg_setlevel