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

Space complexity for each algorithm is:
breadth_first_search: O(b**d), where d is the depth of the search tree and b is the branch factor.

depth_first_graph_search: O(b*m), where b is the branch factor and m is the maximum depth of the tree.

uniform_cost_search: O(b**(1+C/e)), where b is the branch factor, C is the cost of the optimal solution and e is the least cost of every action.

greedy_best_first_graph_search: O(b**m), where b is the branch factor and m is the maximum depth of the tree.

A* search: O(b**d), where d is the depth of the search tree and b is the branch factor.

Table1: Nodes–Action relation table of different Search algorithms.

Problem	Search algorithm	Actions	New Nodes
1	breadth_first_search	20	178
1	depth_first_graph_search	20	84
1	uniform_cost_search	20	240
1	greedy_best_first_graph_search with h_unmet_goals	20	29
1	greedy_best_first_graph_search with h_pg_levelsum	20	28
1	greedy_best_first_graph_search with h_pg_maxlevel	20	24
1	greedy_best_first_graph_search with h_pg_setlevel	20	28
1	astar_search with h_unmet_goals	20	206
1	astar_search with h_pg_levelsum	20	122
1	astar_search with h_pg_maxlevel	20	180
1	astar_search with h_pg_setlevel	20	138
2	breadth_first_search	72	30503

2	depth_first_graph_search	72	5602
2	uniform_cost_search	72	46618
2	greedy_best_first_graph_search with h_unmet_goals	72	170
2	greedy_best_first_graph_search with h_pg_levelsum	72	86
2	greedy_best_first_graph_search with h_pg_maxlevel	72	249
2	greedy_best_first_graph_search with h_pg_setlevel	72	84
2	astar_search with h_unmet_goals	72	22522
2	astar_search with h_pg_levelsum	72	3426
2	astar_search with h_pg_maxlevel	72	26594
2	astar_search with h_pg_setlevel	72	9605
3	depth_first_graph_search	88	3364
3	uniform_cost_search	88	161936
3	greedy_best_first_graph_search with h_unmet_goals	88	230
3	greedy_best_first_graph_search with h_pg_levelsum	88	126
3	astar_search with h_unmet_goals	88	65711
3	astar_search with h_pg_levelsum	88	3403
4	breadth_first_search	104	944130
4	uniform_cost_search	104	1066413
4	greedy_best_first_graph_search with h_unmet_goals	104	280
4	greedy_best_first_graph_search with h_pg_levelsum	104	165
4	astar_search with h_unmet_goals	104	328509
4	astar_search with h_pg_levelsum	104	12210

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

Time complexity for each algorithm is:
breadth_first_search: *O*(b**d), where d is the depth of the search tree and b is the branch factor.

depth_first_graph_search: *O*(b**m), where b is the branch factor and m is the maximum depth of the tree.

uniform_cost_search: *O*(b**(1+C/e)), where b is the branch factor, C is the cost of the optimal solution and e is the least cost of every action.

greedy_best_first_graph_search: *O*(b**m), where b is the branch factor and m is the maximum depth of the tree.

A* search: *O*(b**d), where d is the depth of the search tree and b is the branch factor.

Table2: Time-Action relation table of different Search algorithms.

Problem	Search algorithm	Actions	Time (seconds)
1	breadth_first_search	20	0.0051
1	depth_first_graph_search	20	0.0027
1	uniform_cost_search	20	0.0077
1	greedy_best_first_graph_search with h_unmet_goals	20	0.0013
1	greedy_best_first_graph_search with h_pg_levelsum	20	0.3795
1	greedy_best_first_graph_search with h_pg_maxlevel	20	0.2858
1	greedy_best_first_graph_search with h_pg_setlevel	20	0.5078
1	astar_search with h_unmet_goals	20	0.0078
1	astar_search with h_pg_levelsum	20	0.9685
1	astar_search with h_pg_maxlevel	20	0.9997
1	astar_search with h_pg_setlevel	20	1.1913
2	breadth_first_search	72	1.7103

2	depth_first_graph_search	72	2.5623
2	uniform_cost_search	72	2.8736
2	greedy_best_first_graph_search with h_unmet_goals	72	0.0160
2	greedy_best_first_graph_search with h_pg_levelsum	72	8.4906
2	greedy_best_first_graph_search with h_pg_maxlevel	72	18.067
2	greedy_best_first_graph_search with h_pg_setlevel	72	12.770
2	astar_search with h_unmet_goals	72	1.9607
2	astar_search with h_pg_levelsum	72	223.39
2	astar_search with h_pg_maxlevel	72	1285.0
2	astar_search with h_pg_setlevel	72	1142.0
3	depth_first_graph_search	88	1.0213
3	uniform_cost_search	88	13.319
3	greedy_best_first_graph_search with h_unmet_goals	88	0.0346
3	greedy_best_first_graph_search with h_pg_levelsum	88	19.436
3	astar_search with h_unmet_goals	88	8.1278
3	astar_search with h_pg_levelsum	88	357.28
4	breadth_first_search	104	86.838
4	uniform_cost_search	104	100.67
4	greedy_best_first_graph_search with h_unmet_goals	104	0.0519
4	greedy_best_first_graph_search with h_pg_levelsum	104	34.853
4	astar_search with h_unmet_goals	104	49.114
4	astar_search with h_pg_levelsum	104	2028.4

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

Table3: Plan_length-Action relation table of different Search algorithms.

Problem	Search algorithm	Actions	Plan length
1	breadth_first_search	20	6
1	depth_first_graph_search	20	20
1	uniform_cost_search	20	6
1	greedy_best_first_graph_search with h_unmet_goals	20	6
1	greedy_best_first_graph_search with h_pg_levelsum	20	6
1	greedy_best_first_graph_search with h_pg_maxlevel	20	6
1	greedy_best_first_graph_search with h_pg_setlevel	20	6
1	astar_search with h_unmet_goals	20	6
1	astar_search with h_pg_levelsum	20	6
1	astar_search with h_pg_maxlevel	20	6
1	astar_search with h_pg_setlevel	20	6
2	breadth_first_search	72	9
2	depth_first_graph_search	72	619
2	uniform_cost_search	72	9
2	greedy_best_first_graph_search with h_unmet_goals	72	9
2	greedy_best_first_graph_search with h_pg_levelsum	72	9
2	greedy_best_first_graph_search with h_pg_maxlevel	72	9
2	greedy_best_first_graph_search with h_pg_setlevel	72	9
2	astar_search with h_unmet_goals	72	9

2	astar_search with h_pg_levelsum	72	9
2	astar_search with h_pg_maxlevel	72	9
2	astar_search with h_pg_setlevel	72	9
3	depth_first_graph_search	88	392
3	uniform_cost_search	88	12
3	greedy_best_first_graph_search with h_unmet_goals	88	15
3	greedy_best_first_graph_search with h_pg_levelsum	88	14
3	astar_search with h_unmet_goals	88	12
3	astar_search with h_pg_levelsum	88	12
4	breadth_first_search	104	14
4	uniform_cost_search	104	14
4	greedy_best_first_graph_search with h_unmet_goals	104	18
4	greedy_best_first_graph_search with h_pg_levelsum	104	17
4	astar_search with h_unmet_goals	104	14
4	astar_search with h_pg_levelsum	104	15

Q: 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: To plan in real time operation, minimize response would be more appreciated, and whether the plan is most optimized would be less important. And in real time operation, the environment is usually uninformed, therefore, the uninformed cost search might be most suitable. With a restricted domain, breadth first search and depth first search may be also efficient because the depth of search tree will be small. According to the above results, depth first search showed very good time cost when domain is restricted. Therefore, in the real time planning problem, it can use a combination of breadth first and depth first algorithm and also include the uninformed cost search algorithm to ensure a solution in shortest time.

Q: 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: The greedy best first graph search is best. In most cases, it would find the solution in shortest time although it might not be the best solution. According to the result above, it showed the best time and space complexity when applied to large domain cases.

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

A: The A* algorithm is best because it ensured to find the optimal plan, although it might have large time and space complexity because it need take every possibility into consideration to ensure the best plan.