CSL-302 Artificial Intelligence
Project1 Report
Analysis of Search Techniques on 8-puzzle problem

Ans6-Solution depth to the 8-puzzle problem on which the evaluation was performed at SOLDEPTH=50.

Initial configuration of 8 puzzle problem:
6 8 3
5 1 4
9 7 2
Goal configuration of 8 puzzle problem:
9 1 2
3 4 5
6 7 8
where 9 is the blank tile.

Performance Evaluation of different search techniques using the following metrics:

Search Technique	Nodes Generated	Nodes expanded	Maximum Depth Reached	Computation Time	Goal found at depth
Breadth-First-Search-Tree	664702	373180	22	0.23s	22
Depth-First-Search-Tree(Limited Depth =SOLDEPTH)	128808	71539	50	0.03s	48
Greedy Best First Graph Search	2441	1423	76	0.15s	60
A*(misplaced tile heuristic)Tree Search	22585	13421	22	0.02s	22
A*(manhattan distance heuristic)Tree Search	1338	787	22	0.00s	22
Iterative Deepening A* Tree Search	29982	29976	22	0.02	22

A* (Manhattan heuristic) clearly performs the best in terms of computation time and number of nodes generated and explored.

Ans7-The Manhattan distances heuristic is clearly a much better heuristic than the misplaced tiles as can be seen from the above table. The misplaced tiles heuristic generates 20x more nodes and expands roughly twice the number of nodes as compared to Manhattan distances heuristic.

Ans8-Varied SOLDEPTH from 10 to 50 using 10 step increments. For the same SOLDEPTH and initial state recorded the metrics for each search technique.

In general, computation time,number of nodes created/expanded increases for each search technique with increase in SOLDEPTH.

The best search technique out of all is the A*(Manhattan) in terms of least computation time and least number of nodes created/expanded.

Performance Evaluation of Breadth-First-Tree -Search using the following metrics:

Solution Depth	**Nodes Generated**	**Nodes Expanded**	**Maximum Depth Reached**	**Computation Time**	**Goal found at depth**
10	592	340	8	0.00s	8
20	26588	14970	16	0.02s	16
30	97145	53927	18	0.04s	18
40	666150	374208	22	0.23s	22
50	298470	175576	20	0.11s	20

Performance Evaluation of Depth-First-Tree-Search(Limited Depth) using the following metrics:

Solution Depth	Nodes Generated	Nodes Expanded	Maximum Depth Reached	Computation Time	Goal found at depth
10	242	131	10	0.00s	8
20	77852	43227	20	0.02s	18
30	674370	375154	30	0.22s	30
40	850602	473346	40	0.25s	38
50	102680	56916	50	0.02s	50

Performance Evaluation of Greedy-Best-First-Graph-Search using the following metrics:

Solution Depth	Nodes Generated	Nodes Expanded	Maximum Depth Reached	Computation Time	Goal found at depth
10	195	113	22	0.00s	10
20	461	262	38	0.01s	38
30	789	444	44	0.01s	38
40	2213	1287	88	0.11s	84
50	3336	1933	65	0.28s	52

Performance Evaluation of A*-Tree-Search(Misplaced Tiles Heuristic) using the following metrics:

Solution Depth	Nodes Generated	Nodes Expanded	Maximum Depth Reached	Computation Time	Goal found at depth
10	35	19	8	0.00s	8
20	1016	611	16	0.01s	16
30	4100	2458	18	0.01s	18
40	24696	14714	22	0.03s	22
50	7333	4403	20	0.01s	20

Performance Evaluation of A*-Tree-Search(Manhattan Distance Heuristic) using the following metrics:

Solution Depth	**Nodes Generated**	**Nodes Expanded**	**Maximum Depth Reached**	**Computation Time**	**Goal found at depth**
10	33	18	8	0.00s	8
20	169	100	16	0.00s	16
30	1192	707	18	0.00s	18
40	1948	1150	22	0.00s	22
50	548	321	20	0.00s	20

Performance Evaluation of IDA*-Tree-Search using the following metrics:

Solution Depth	**Nodes Generated**	**Nodes Expanded**	**Maximum Depth Reached**	**Computation Time**	**Goal found at depth**
10	18	14	8	0.00s	8
20	1212	1209	16	0.00s	16
30	5704	5699	18	0.01s	18
40	22602	22592	22	0.02s	22
50	7611	7603	20	0.00s	20

Ans9-We can directly observe from the evaluation function that if w=0 then f=h i.e the A* search acts like a Greedy-Best-First-Search-Tree technique and if w=1 then f=g and A* search acts like a Uniform Cost Search technique. If w=0.5 then both the functions g(actual cost) and h(heuristic function) have the same weightage which is A*.Increasing w shifts A* towards Uniform Cost Search and decreasing it towards 0 shifts A* towards Greedy-Best-First-Tree-Search.

Performance Evaluation of A*-Tree-Search(Manhattan Distance Heuristic) using the following metrics:

Solution Depth	w	Nodes Generated	Nodes Expanded	Maximum Depth Reached	Computation Time	Goal found at depth
3	0.2	9	3	3	0.00s	3
	0.5	9	3	3	0.00s	3
	0.8	13	6	3	0.00s	3
4	0.2	12	5	4	0.00s	4
	0.5	12	5	4	0.00s	4
	0.8	19	9	4	0.00s	4
5	0.2	13	6	5	0.00s	5
	0.5	13	6	5	0.00s	5
	0.8	29	15	5	0.00s	5
6	0.2	21	10	6	0.00s	6
	0.5	21	10	6	0.00s	6
	0.8	54	30	6	0.00s	6
7	0.2	37	19	8	0.00s	7
	0.5	25	13	7	0.00s	7
	0.8	80	45	7	0.00s	7
8	0.2	38	20	9	0.00s	8
	0.5	27	14	8	0.00s	8
	0.8	106	59	8	0.00s	8
9	0.2	40	21	10	0.00s	9
	0.5	42	22	9	0.00s	9
	0.8	206	119	9	0.00s	9
10	0.2	22	11	8	0.00s	8
	0.5	33	18	8	0.00s	8
	0.8	137	78	8	0.00s	8

From the above table we can clearly conclude that in general A*(w=0.5) performs the best among the different values for w for any solution depth whereas A*(w=0.8) performs the worst out of all 3 choices for w since increasing w shifts more weightage towards g(n) and less towards h(n) thus,shifting A* more towards uniform search which is an uninformed search and thus ignores the heuristic value and hence performs the worst. A*(w=0.2) performs better than (w=0.8) because it depends more on the heuristic value which is the information which helps in reaching towards the goal in less number of steps.

By-Jaskaran Singh
2011CS1012