COS314 Assignment 1 Report

Caleb Groeneveld u19340631

Heuristic

The Heuristic that I used was the number of tiles out of place. I use two for loops to go through the two-dimensional array that I keep my puzzles in and compare them to the end state. For the puzzles that require heuristics, I complete the available moves and then chose the puzzle with the least number of tiles that are out of place. In the case of the A* algorithm, I also add the level of the tree to the heuristic in order to try find the shortest path. The algorithm will pick the puzzle with the lowest heuristic value.

Performance

Algorithms \ Puzzles	Breadth First Search	Best First Search	Hill Climbing Search	A* Search
1 Goal: 5	143 Moves, Optimum: 5	19 Moves, Optimum: 5	Failed	8 062 Moves, Optimum: 5
2 Goal: 9	1 138 Moves, Optimum: 9	642 Moves, Optimum: 31	Failed	1 167 Moves, Optimum: 9
3 Goal: 12	4 880 Moves, Optimum: 12	2 268 Moves, Optimum: 44	Failed	1 710 Moves, Optimum: 12
4 Goal: 6	218 Moves, Optimum: 6	17 Moves, Optimum: 6	Failed	926 Moves, Optimum: 16
5 Goal: 14	15 802 Moves, Optimum: 14	3 034 Moves, Optimum: 38	Failed	938 Moves, Optimum: 14
6 Goal: 16	30 464 Moves, Optimum: 16	721 Moves, Optimum: 16	Failed	1722 Moves, Optimum: 16
7 Goal: 16	32 156 Moves, Optimum: 16	721 Moves, Optimum: 16	Failed	1717 Moves, Optimum: 16
8 Goal: 30	483 646 Moves, Optimum: 30	5 342 Moves, Optimum: 66	Failed	Failed
9 Goal: 28	478 740 Moves, Optimum: 28	1 608 Moves, Optimum: 50	Failed	Failed
10 Goal: 31	483 836 Moves, Optimum: 31	879 Moves, Optimum: 63	244 Moves, Optimum: 82	Failed

Key: The "Moves" refers to the total amount of moves that each algorithm did. A "Move" is considered as a move when the algorithm moves a tile into a new placement. Moves that are found to produce that have already been seen and moves that are rejected by the heuristic are counted. The "Optimum" is **not** the optimum number of moves required to solve the problem, but rather the shortest path found by the algorithm. The "Goal" is the shortest amount of moves as provided by the spec.

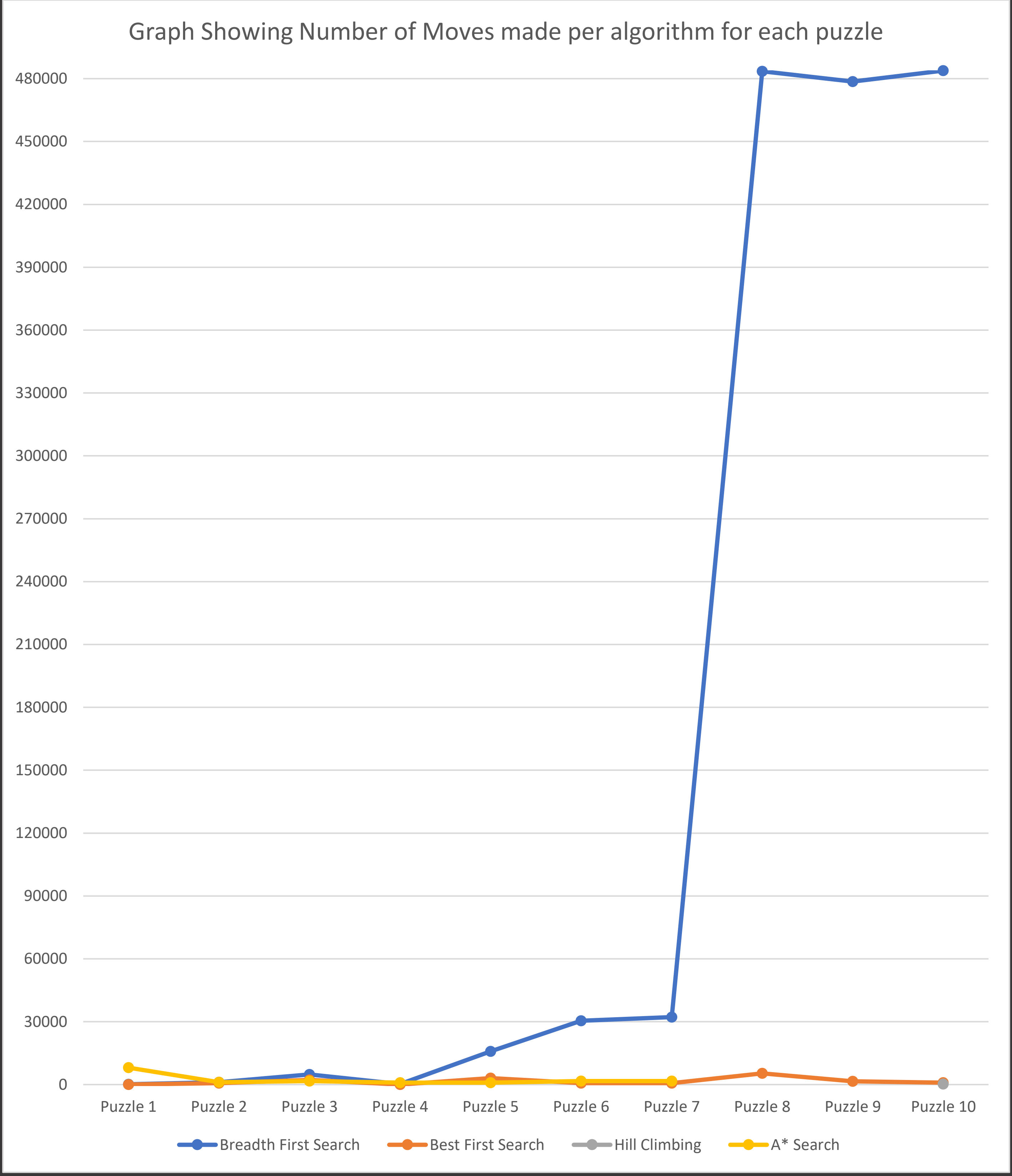
Discussion of Performance

Let us discuss the Breadth First Search first. This algorithm ran for the longest time and took an incredible number of moves. I understand that this is because the algorithm literally looks at every possible variance of the puzzle and makes every single move. As reflected in the results, the higher the Known Optimum to solve the puzzle is, the increasingly longer the puzzle ran and the more moves it produced. One could say that the relationship between the Known Optimum and the amount of moves required for the Breadth First Search to find the end state is logarithmic. The amount of time required also increases, the last three puzzles all took upwards of 15 minutes to solve. However, the Breadth First Search always yielded the shortest path, and was the most accurate out of all the algorithms in this manner.

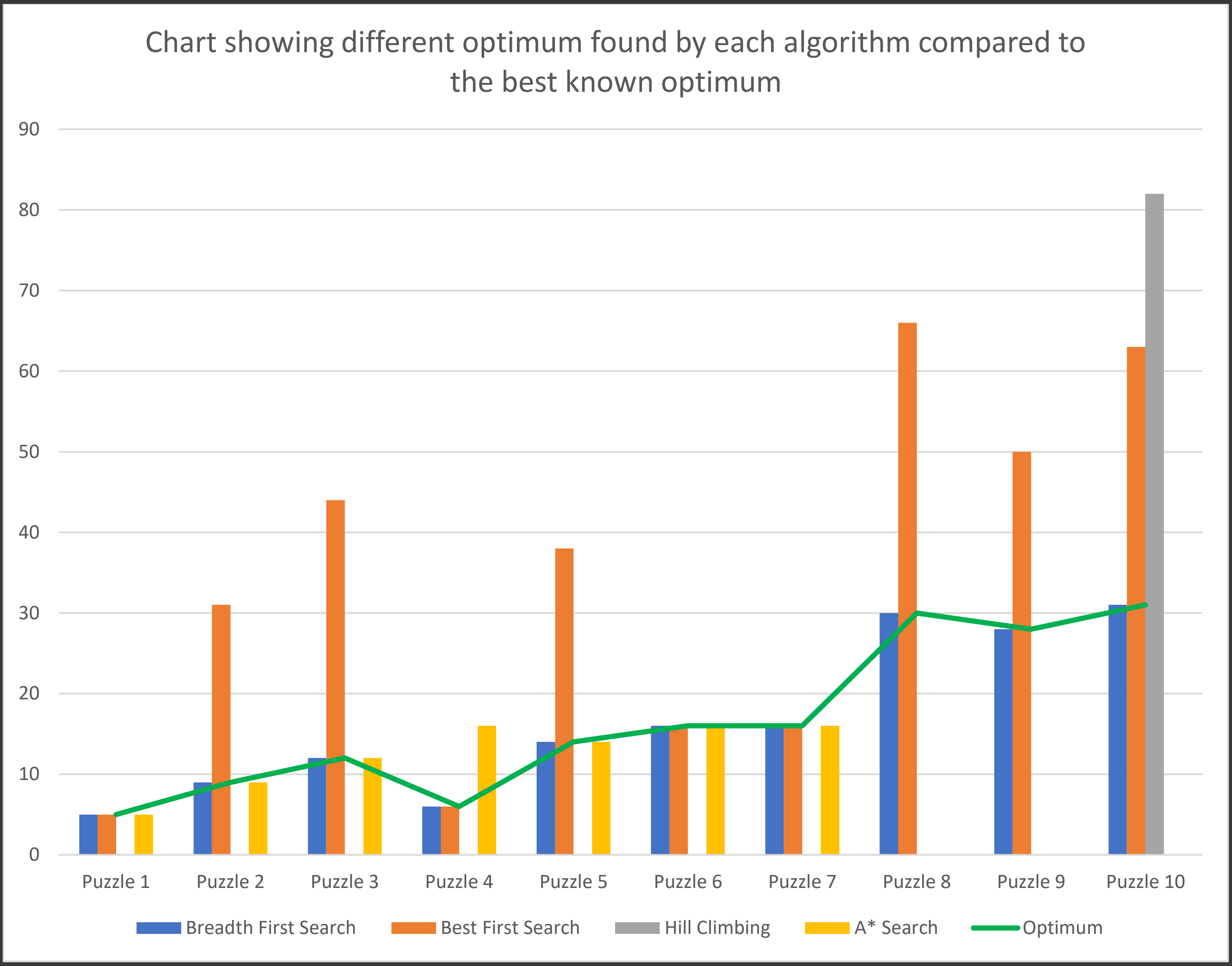
The Best First Search is probably my favourite out of these algorithms due to how speedy it is and how few moves are required to find the goal state. It also seems to be a very reliable algorithm, as Puzzle 6 and Puzzle 7 are reverses of each other and the Best First takes the same amount of steps on both. The one problem with the Best First Search is that it does not always find the best optimum. Unnecessary steps are sometimes taken, resulting in the algorithm sometimes running longer than necessary. A good example of this algorithm struggling is in Puzzle 8, where it took 5 342 Moves and found an optimum of 66. Similarly, the algorithm seemed to struggle with Puzzle 3. However, the Best First highly outperformed the Breadth First Search in terms of time and total amount of moves.

The Hill Climbing algorithm does not seem to be a useful algorithm. In almost every case, the algorithm got itself to a place where it could no longer make any moves, as all the states it wanted to move in had already been seen. This is likely due to the greedy nature of the algorithm, always looking at the next best move and not the best move overall, as the Best First Search does. The one exception was the final puzzle, and this surprised me greatly. I think that the board must have lined up so that the next best move never led to a “seen lock”. Even more surprisingly, the Hill Climbing algorithm solved this puzzle in an astonishing 244 moves, albeit finding the worst optimum. I would hypothesise that, given the opportunity to backtrack, the Hill Climbing algorithm would be even faster than the Best First. Sadly, the simple Hill Climbing algorithm seems to be very situational.

The A* Search is a very interesting and frustrating algorithm. It is essentially a modified Best First Search that has been altered to better find the optimum, to the detriment of it’s speed and low amount of moves. It makes even more unnecessary moves than the Best First Search does, as seen in Puzzle 1, only occasionally beating the amount of moves made by the Best First Search in other places (and not beating it by many moves). While it does semi-reliably provide an accurate optimum, there are places where it’s Best First Nature holds it back, as in puzzle 4. It is remarkable that simply adding the current level to the heuristic can alter an algorithm like this. However, the algorithm did fail to solve the longer puzzles. The adding the level to the heuristic makes this algorithm solve more and more puzzles the more levels it has, like the Breadth First Search. It seems that the Array container I was using overflowed because of the amount of moves this algorithm starts to make. This makes the A* algorithm very good for medium puzzles where the optimum needs to be found, but not good for short or long puzzles. It also has middling speed in solving these puzzles.



Here you can see the incredible amount of moves that Breadth First Search makes. Best First Search and A* Search have the same general amount of moves, with A* spiking in certain places. Hill Climbing has one value at the very end.



This chart shows how accurate to the Known Optimum Breadth First Search was. A* is fairly accurate in most places, while Best First is far above the Optimum. Hill Climbing has the worst optimum.