Javier A. Suarez Rivas

CSCI 523 Artificial Intelligence

Dr. Wilkins

Fall 2015

**8 Puzzle Results Report**

I created a python script (printsolutions.py) which generates 50 different puzzles by randomly performing 100 different moves to the initial state, and pass the same puzzle to the 5 different algorithms. The results of each algorithm are printed into a csv file named (astar.csv), which then was transformed into an excel document (astar.xlsx) where the data was analyzed into three different categories: A\*, Hill Climbing, and A\* versus Random Restart. Each of this categories have their own tab on the excel workbook. In each tab the information is grouped in two different categories, number of states created and time (in seconds). Per category the lowest values are highlighted with a green background and the higher values are with a red one. The more intense the color is, the better the value is in it respective range.

1. **A\*: Manhattan Distance versus Misplaced**

Based on the heuristics comparison, it is clear that Manhattan distances dominates the misplaced titles one, which was also reflected in our results. In order to make the comparison closer to the average I deleted the three highest number of generated states by Misplaced and the three lowest by Manhattan.

In average Manhattan added 279.52 states to the queue, while misplaced added a total of 1576.50. Representing in average a 408% average increase (following the average increase formula). In terms of time, the Manhattan heuristic took 0.041 seconds in average to complete, while the misplaced heuristic completed in an average of 0.939 seconds. In some specific cases, the complexity for solving a problem was higher for both algorithms, having a direct impact in both the number of states created and the time required to solve the problem. However in this cases the impact was much higher on the misplaced heuristic, especially on time, in this cases the average increase between the two algorithms stayed in the range of being four times the states than Manhattan. In terms of time, solving this harder problems took the misplaced heuristic about twenty times more than its counterpart (represented as more than a 2000% average increase).

The data distribution can be observed in the following graphs:

1. **Hill Climbing Steepest Vs First Choice**

In hill climbing the number of generated states was incremented once a new state was tested. For instance due to the nature of the algorithms, steepest ascend will always generate more or equal states than First Choice.

* 1. **Effectiveness**

Both Steepest and First Choice reached a solution in 6 out of the 50 puzzles which is equivalent to 12%.

* 1. **Analysis**

In order to analyze hill climbing, the data set was divided into two different categories problems solved and the ones that stopped at a local maxima.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Results Table Hill Climbing | | | | | | |
| Solved | **hc\_S: States** | **hc\_FC: states** | Avg Increase | **hc\_S time:** | **hc\_FC Time** | **Avg Increase** |
| No | 10.11 | 7.77 | 25% | 0.0002482 | 0.0002272 | 11% |
| Yes | 31.50 | 22.50 | 40% | 0.0006987 | 0.00068517 | 2% |

In both cases the Steepest generates more nodes, however when a solution is reached the time difference between both is minimum, only two percent.

1. **A\* versus Random Restart**

As explained in the state declaration function, I set up my random restart in such a way to be always complete and always reach a solution using steepest hill climbing. Which makes this comparison the most interesting of all, based on the fact that we already knew the expected results from the previous sections and those were just confirmed with the experimentation.

Since we are using Hill Climbing, which only cares about getting a solution rather that the path obtained, our main goal for this comparison will be how fast we reach our goal and the number of states that are tested to reach such goal. Space wise, hill climbing will always be better than A\* because it never saved the current state and just keeps track of the current and the possible next, while A\* keeps all possible state on memory.

In average A\* evaluated 449.44 states while RR hill climbing 3097.56. So RR hill climbing needed to test about 5 more times states than A\* to reach a solution. RR Hill Climbing tested fewer nodes in 8 of the 50 puzzles, representing 16% of our total attempts. Where things get interesting is when we analyze our time results, which are displayed in seconds. RR Hill Climbing finished faster in 36% of the occasions and was in average .03 seconds faster per attempt and 1.46 seconds faster if we add all the values together. By just reading this values seems like RR Hill climbing could be better if we want a fast solution. However let’s analyze the data more individually by looking at the following graph.

Random Restart has an average of 0.09 seconds and how it can be observed remains relatively between its parameters through all the solved puzzles. A\* dominates, except for certain puzzle that create some higher points on our graph, which elevate its total time. So A\* is generally faster but could tends to have some worst performance in specific occasions, while RR hill climbing could be considered to always stay in within a certain range in its executions.

Now let us examine deeper that puzzle 47 that created that high pick on our graph.



This puzzle had the biggest depth from all of generated puzzles from start to goal, being 25. It was also the puzzle that generated more states for the Misplaced heuristic with 40891 and had the worst time performance for Misplaced with 183.77 seconds.

From this observation we can conclude that the greater is the depth of a solution for A\* the longer the algorithm will run, however the impact it has in the performance of the algorithm is exponential. So one side A\* will be faster in more than 60% percent of the occasions than RR Hill Climbing, however you have the risk of encountering a problem that will give you a worst performance and delay your solution, while time-wise RR Hill Climbing is more stable to reach the goal and for instance more reliable if time is all that matters.