University of Puerto Rico
Mayagüez Campus
Department of Electrical and Computer Engineering

8-Queens & 8-puzzle

ICOM 5015 - 001D
Prof. J. Fernando Vega Rivero

David A. Castillo Martínez
Christian J. Perez Escobales
Ramón J. Rosario Recci
April 8, 2024

Abstract

The 8-puzzle and the 8-queens problems are classic combinatorial optimization problems that require finding optimal solutions in a broad search space. A number of instances are generated for both problems, and variations of the hill climbing algorithm are used to solve them. The search cost and the percentage of problems solved for each algorithm are measured and plotted, and the results are compared to gain insight into the performance of these algorithms. The results provide valuable insights into the trade-offs between solution quality and search cost for different algorithms.

Introduccion

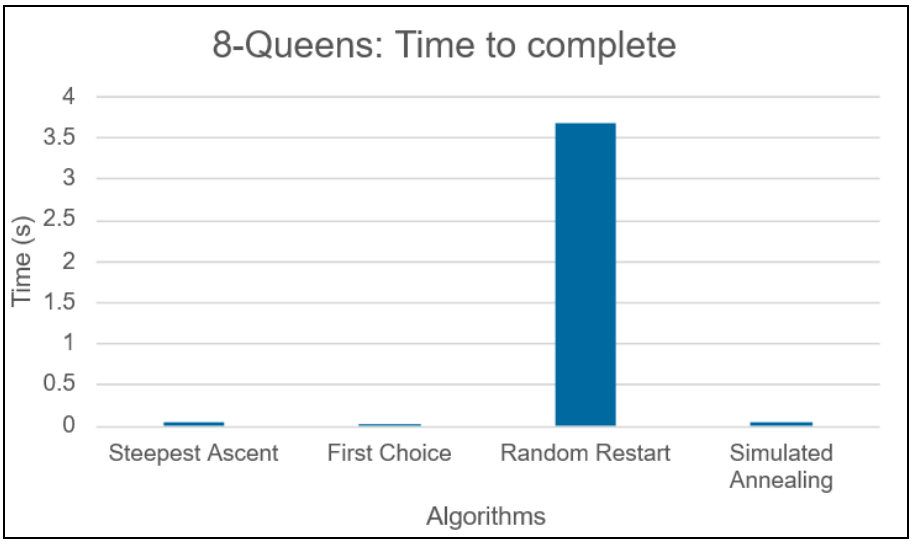
Exercise 4.4 of the “Artificial Intelligence: A modern Approach” text book asks for the following, to generate a large number of instances of 8-queens and 8-puzzle and solve them (where possible) using hill-climbing (variants of steepest-ascent and first-choice), random restart hill climbing, and simulated annealing. Measure the search cost and the percentage of problems solved, and graph and comment on the results [1]. To solve this exercise we must understand each of the required algorithms. Hill Climbing is a local search algorithm that starts with an initial solution and tries to improve it by making small changes to the solution. In the context of the 8-puzzle problem, hill climbing can be used to move tiles on the board in a way that approaches the goal state. Random restart hill climbing iteratively implements hill climbing with the condition that if a certain amount of time has passed or a local maximum has been reached, the configuration is restarted. The theory behind random restart is that it is computationally more efficient to spend time exploring the state space rather than trying to carefully navigate through what may be an alley. The steepest-ascent hill climbing algorithm starts from an initial state and then moves to the best neighboring state at each step until it reaches a goal state or a local maximum. Simulated annealing is a probabilistic approach that allows downhill moves in the state space and uses randomness to generate states. Using a probability function that takes temperature as one of its parameters, a high temperature allows a high probability of accepting a non-optimal random move. As the temperature cools, the state configuration should be approaching a goal state; when the temperature is close to 0, only positive moves are accepted in cost change [2].

Discussion

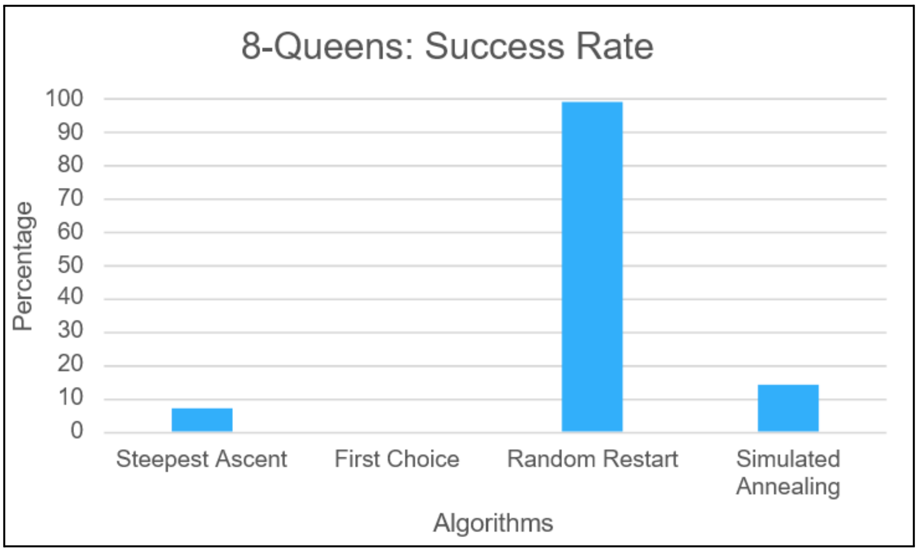
8 Queens:

The 8 queens problem involves placing eight queens on a chessboard in such a way that none can attack another. There are two distinct approaches to solving this problem: the first is to start with 0 queens on the board and add one with each step; the second is to start with the 8 queens already on the board and move them until reaching the goal state. For this report, we will use the second approach. As a performance measure, the time it takes for each algorithm to solve 100 instances of the queens problem was used.

Results: Time taken by each algorithm to complete 100 instances of the 8 queens problem. Success percentage of each algorithm by instance size for the 8 queens problem.



Graph 1: Time taken to complete 100 run of each algorithm for the 8-Queens problem

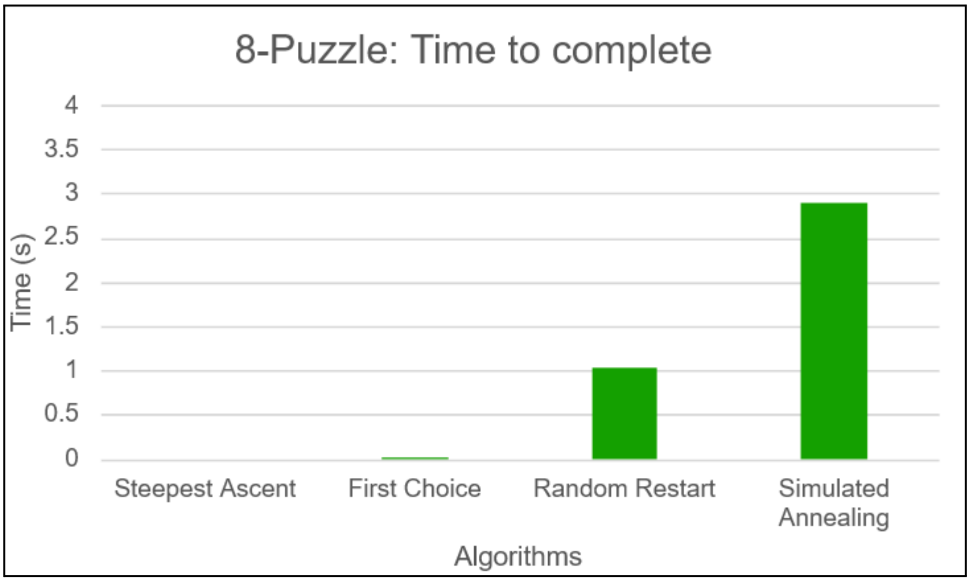


Graph 2: Success rate of each algorithm for the 8-Queens problem

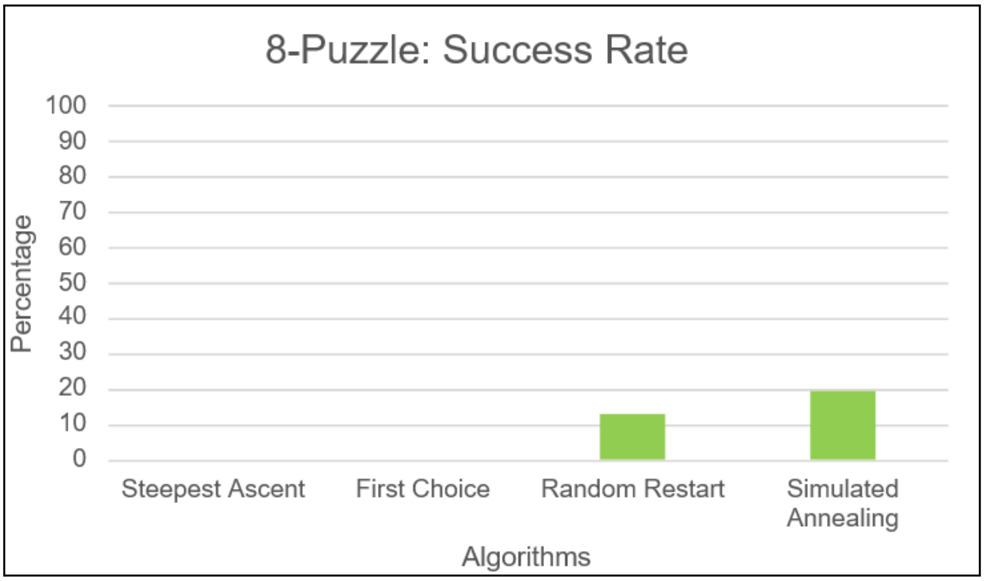
In Graph 1, it can be seen that consistently, the first choice hill climbing algorithm is much faster than steepest ascent hill climbing, random restart hill climbing and simulated annealing. This is because first-choice hill climbing immediately chooses a better state than the current one, while random restart and simulated annealing introduce a random element that extends the time it takes to choose the next state [3]. Observing the success percentages in Graph 2, it is observed that, in the case of the eight queens, the random restart hill climb algorithm has a significantly high success rate. The data for both of these graphs was obtained from the output of the eight-queens code, which can be seen in Appendix A-1

8-Puzzle:
The 8-puzzle problem involves arranging tiles with numbers from 1 to 8 in a 3X3 square in order. There is an empty space, and tiles can only be moved adjacent to this space. In the case of the 8-puzzle problem, the state space has a size of 9! possible configurations. However, many of these are unreachable from the initial state, and the branching factor effect is much lower than 4 (the number of possible moves on each tile). As a performance measure, the time it takes for each algorithm to solve 100 instances of the 8-puzzle problem was used.

Results: Time taken by each algorithm to complete 100 instances of the 8-puzzle problem. Success percentage of each algorithm by instance size for the 8-puzzle problem.



Graph 3: Time taken to complete 100 runs of each algorithm for the 8-Puzzle problem



Graph 4: Success rate of each algorithm for the 8-Puzzle problem

For the 8-puzzle problem, in Graph 3, it is observed that similar to the case of the eight queens, first-choice hill climbing, and steepest ascent run much faster than random restart and simulated annealing. However, a discrepancy is seen in the success percentages in Graph 4. In this figure, it is seen how random restart and simulated annealing are much more effective at solving problems than first-choice hill climbing and steepest hill climbing. It can also be observed that the performance of simulated annealing is similar. The data for both of these graphs was obtained from the output of the eight-puzzles code, which can be seen in Appendix A-2.

Conclusion

The 8-puzzle and 8-queens problems are classic combinatorial optimization problems that have been widely studied in artificial intelligence. In this report, we generated a large number of instances for both problems and solved them using 4 different algorithms: steepest ascent hill climbing, first-choice hill climbing, random restart hill climbing, and simulated annealing. We measured search time, the percentage of problems solved, and compared them. Our results showed that the first-choice hill climbing and steepest ascent algorithms had differing performance on the 8-puzzle and 8-queens problems. They were able to find solutions with a high success rate optimally for the 8-queens problem but struggled with the 8-puzzle problem. Random restart and simulated annealing showed good performance on both problems, as they can escape local optima and find better solutions. It can be inferred that for the 8-queens problem, it would be better to use first-choice hill climbing or steepest ascent to solve them as they run much faster but achieve a similar success rate to random restart and simulated annealing. However, for the 8-puzzle case, random restart and simulated annealing seem to be a better choice; especially random restart hill climbing because it takes much less time to execute than simulated annealing but has a similar success rate. Our study provides information on the performance of hill climbing, and simulated annealing algorithms on the 8-puzzle and 8-queens problems. Our results highlight the importance of selecting appropriate optimization algorithms for different types of problems and complexities. Further research could explore additional optimization techniques or problem variants to improve solution quality and search efficiency.

References

[1] S. Russell and P. Norvig, “Beyond Classical Search,” Artificial Intelligence: A Modern Approach, 3rd ed. Upper Saddle River, NJ, USA: Pearson Education, Inc., 2010, ch. 4, pp. 120-160.

[2] J. Pearl. Heuristics. Intelligent Search Strategies for Computer Problem Solving. Addison Wesley, Reading, MA, 1984.

[3] Piltaver, R. Lustrek, Mitja & G. Matjaz. “The Pathology of Heuristic Search,” Journal of Experimental & Theoretical Artificial Intelligence, 24, pp. 65-94, 2012.

Appendix

Appendix A: Code Outputs

Appendix A-1: Output for 8-Queens Exercise

```
Out of 100 random 8-queens
Steepest Hill Climbing
     Successes: 7
     Fails: 93
     Time Taken: 0.033499717712402344

First Hill Climbing
     Successes: 0
     Fails: 100
     Time Taken: 0.016420841217041016

Random Restart Hill Climbing
     Successes: 99
     Fails: 1
     Time Taken: 3.66892671585083

Simulated Annealing
     Successes: 14
     Fails: 86
     Time Taken: 0.03503108024597168
```

Appendix A-2: Output for 8-Puzzles Exercise

```
Out of 100 random 8-puzzles
Steepest Hill Climbing
     Successes: 0
     Fails: 100
     Time Taken: 0.0

First Hill Climbing
     Successes: 0
     Fails: 100
     Time Taken: 0.0020005702972412111

Random Restart Hill Climbing
     Successes: 13
     Fails: 87
     Time Taken: 1.0205903053283691

Simulated Annealing
     Successes: 19
     Fails: 81
     Time Taken: 2.8906519412994385
```

Task Distribution

- Ramon J Rosario Recci - Collaborated on the Python code, analysis, interpretation of results, and collaborated on writing the assignment’s formal report.
- DAVID A CASTILLO-MARTINEZ - Implemented Python code for the exercises using the Aimacode repository, and collaborated on writing the assignment’s formal report.
- CHRISTIAN J PEREZ-ESCOBALES - Collaborated on the analysis, interpretation, writing the assignment’s formal report, created the presentation, and edited the video presentation.