Assignment 3: CSP

Dan Blanchette

July 7, 2025

Abstract

For this assignment, a comparative analysis was conducted of two local search algorithms, Hill Climbing and Min-Conflicts, for solving the graph coloring problem. These algorithms are used to solve the Constraint Satisfaction Problem (CSP). Both algorithms were evaluated across a range of color constraints (2 to 6) using a consistent framework for conflict detection and graph representation. These metrics were selected for results reporting and formatting in this report. Experimental results show that while Hill Climbing required a minimum of four colors to produce valid solutions, Min-Conflicts achieved successful colorings with as few as three colors. Moreover, Min-Conflicts demonstrated superior performance in terms of computational efficiency, frequently converging to a solution in under one millisecond. These findings promote the effectiveness of Min-Conflicts in rapidly navigating large search spaces and suggest its practical utility for CSPs where solution quality and runtime are critical factors.

1 Algorithm Overview

1.1 Hill Climbing

Hill Climbing is a local search algorithm that incrementally improves a complete assignment by evaluating neighboring solutions. Hill Climbing's graph coloring starts with a random assignment and iteratively changes the color of a randomly selected node to reduce conflicts. The process continues until a conflict-free solution is found or a step limit is reached. While straightforward and often effective, Hill Climbing is susceptible to getting stuck in local minima due to its greedy nature.

1.2 Min-Conflicts

Min-Conflicts is a heuristic algorithm also designed for constraint satisfaction problems. It operates by repeatedly selecting a variable involved in a conflict and assigning it the color that minimizes the number of constraint violations. Unlike Hill Climbing, it focuses only on problematic variables, which allows it to navigate large or densely constrained search spaces efficiently. This makes it particularly effective for problems such as graph coloring and the N-Queens problem.

2 Results

Table 1: Hill Climbing Performance on Graph Coloring

Algorithm	Colors	Success	Time (s)	Steps
Hill Climbing	2	False	0.108645	10000
Hill Climbing	3	False	0.135229	10000
Hill Climbing	4	True	0.000533	$\leq$ 10000
Hill Climbing	5	True	0.000594	$\leq$ 10000
Hill Climbing	6	True	0.000098	$\leq$ 10000

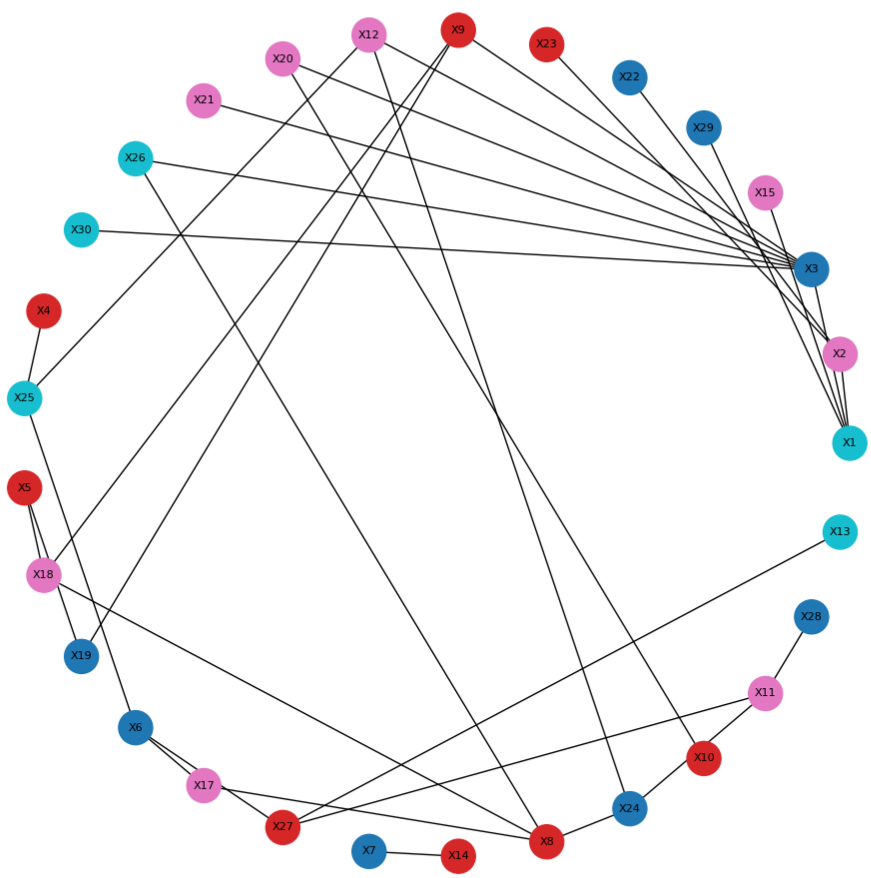

Figure 1: Hill Climbing Successful Graph 4 Colors

Table 2: Min-Conflicts Performance on Graph Coloring

Algorithm	Colors	Success	Time (s)	Steps
Min-Conflicts	2	False	0.137380	10000
Min-Conflicts	3	True	0.000126	$\leq$ 10000
Min-Conflicts	4	True	0.000218	$\leq$ 10000
Min-Conflicts	5	True	0.000156	$\leq$ 10000
Min-Conflicts	6	True	0.000068	$\leq$ 10000

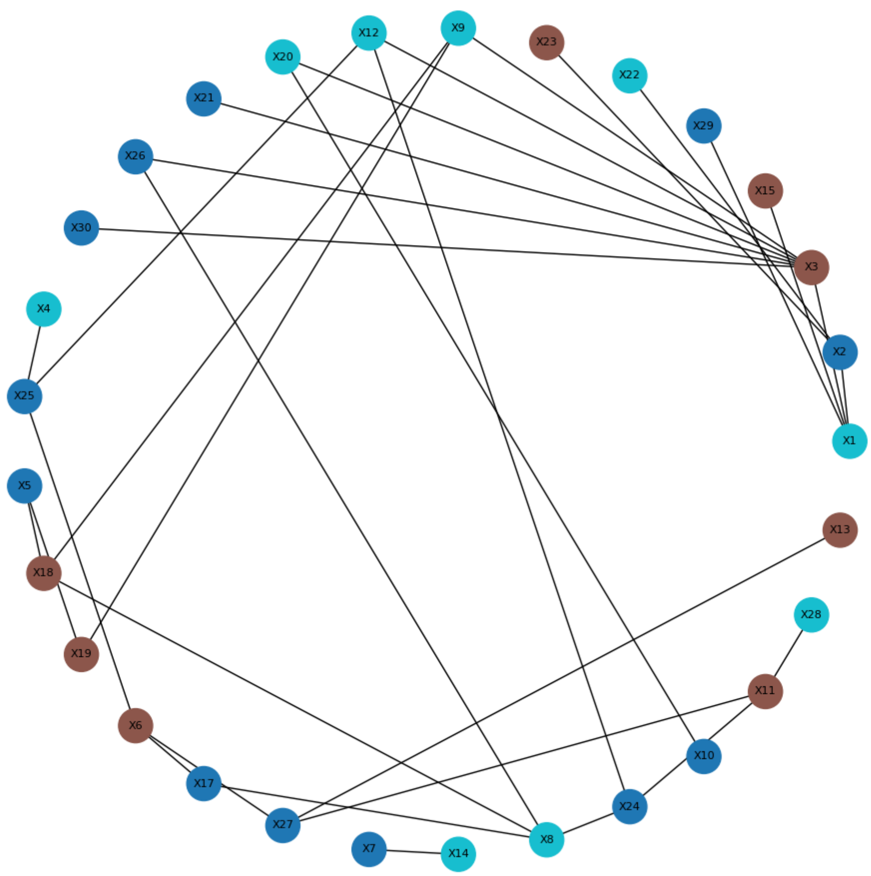

Figure 2: Min-Conflicts Successful Graph 3 Colors

3 AI Programming

We used ChatGPT to help visualize the data output for verification that the algorithms worked as intended and to generate some LaTeX code for the tables. I also used it for the min-conflicts algorithm to debug and make it functional. I found that it was pretty decent at writing the LaTeX code for this paper's formatting, except that it occasionally misinterpreted some Unicode symbols, and I needed to update them using Overleaf's editor AI debugger. For the debugging of min-conflicts, it took some precise prompting to get it to do what I wanted, but after a few attempts, I compared it to the text's pseudocode. We then added the number of colors variable to the function, allowing it to run multiple CSPs with an increasing number of colors.

4 Conclusions

This project evaluated the performance of two local search algorithms, Hill Climbing, and Min-Conflicts, in solving the graph coloring problem. Hill Climbing was effective only when the number of colors was sufficiently large (≤ 4), indicating an issue with escaping local minima in more constrained scenarios. In contrast, the Min-Conflicts algorithm consistently outperformed Hill Climbing both in terms of solution quality and runtime efficiency, successfully solving instances with fewer colors and significantly lower computational costs. These results reinforce the strength of Min-Conflicts as a robust heuristic for CSPs, particularly in large or densely connected search spaces. One final note: it appears to be impossible to achieve success with only two colors for both algorithms due to situations where an odd number of territories share a common boundary. Hill Climbing has the potential to improve its number of colors before a solution is found, but that is subject to the randomness of which variables are chosen to be changed.