Jerry Chienyu Chen

October31/2016

CMPT310 Assignment 2 Report

Discussion About Algorithm & Implementation

At the beginning of implementing this assignment question, I read over the slides and also plenty resources up online about how to roughly set up the problem and mainly about the constraints with heuristics. It is obvious that we need to implement the question using backtracking with DFS to finish the task. While traversing through the graph, we also need heuristics to help us decide the optimal node we should choose at every step, and the best color it should be assigned in order to not affect the coloring on the rest of the node as much. Let me roughly talk about the order of heuristics I applied in my program, and I will discuss about them in details in the later sections. Basically we need to apply Minimum Remaining Value(MRV) heuristic to obtain the node that has the least amount of domain colors left over and choose it as the node we are going to color at this step! If there happens to be several nodes with the same amount of MRV, then we use degree heuristic to choose a node from them again! Degree heuristic chooses the node from the chosen nodes after MRV that has the greatest degree at the step and chooses it as the node we are going to color at the step. However, if there are nodes existed that have the same value of MRV and degree, then we can choose either of them. In my program, it will choose the node which ever comes first in the order of the dictionary data structure.

After the node is chosen, we now need to assign a color to it. Here we apply the Least Constraining Value(LCV) heuristic. LCV tests with a color choice of the color domain of the chosen node's assigning to the chosen node itself, and it calculates the sum of total constraints using this color on this chosen node. The sum of total constraints is calculated through adding the remaining color choices of all the nodes after eliminating the chosen color from the chosen node's neighbours' color domain. LCV chooses which ever color that results the largest sum of total constraints throughout the graph.

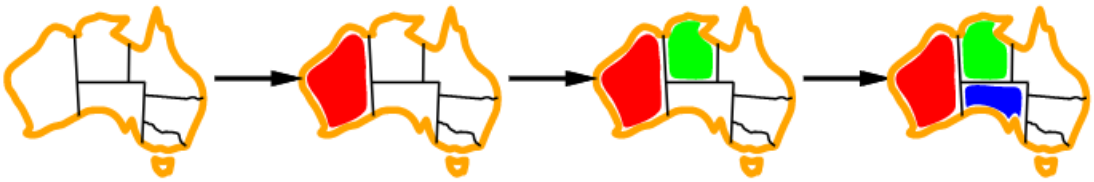
As the node is chosen and its assigned color is chosen as well, then we store the key and the color value to a result dictionary for result print out purposes. Afterwards, we update the data structures for the next iteration. If in the process the program fails to find a node or a color, then we backtrack. After all, we iterate through the result dictionary and print out the result to standard output onto the screen. Now let's talk about a high level overview of how each the heuristics mentioned previously are implemented in details.

Jerry Chienyu Chen

October31/2016

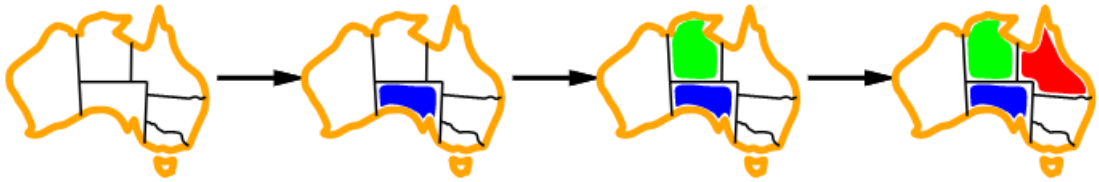
Minimum Remaining Value (MRV)

- Choose the variable with the fewest legal value
- We choose the variable that seems most likely to fail
- Minimum Remaining Value(MRV) heuristic obtains the node that has the least amount of domain colors left over



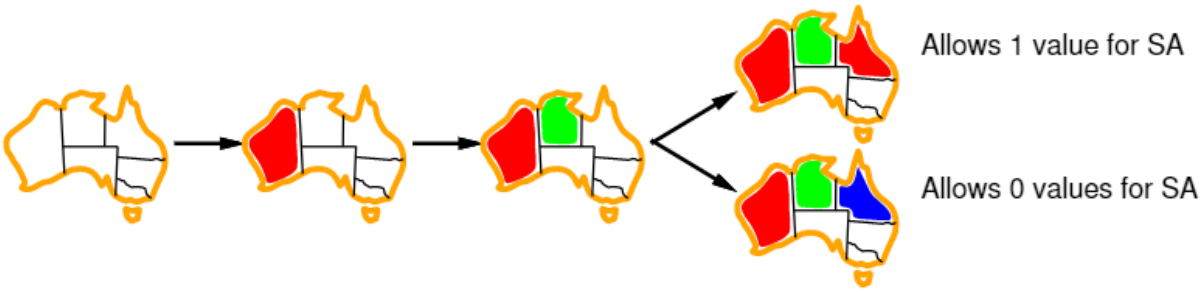
Degree Heuristic

- The tie-breaker among MRV variables
- Choose the variable with the most constraints on other unassigned variables
- In another words, degree heuristic chooses the MRV variable with the greatest degree value



Least Constraining Value

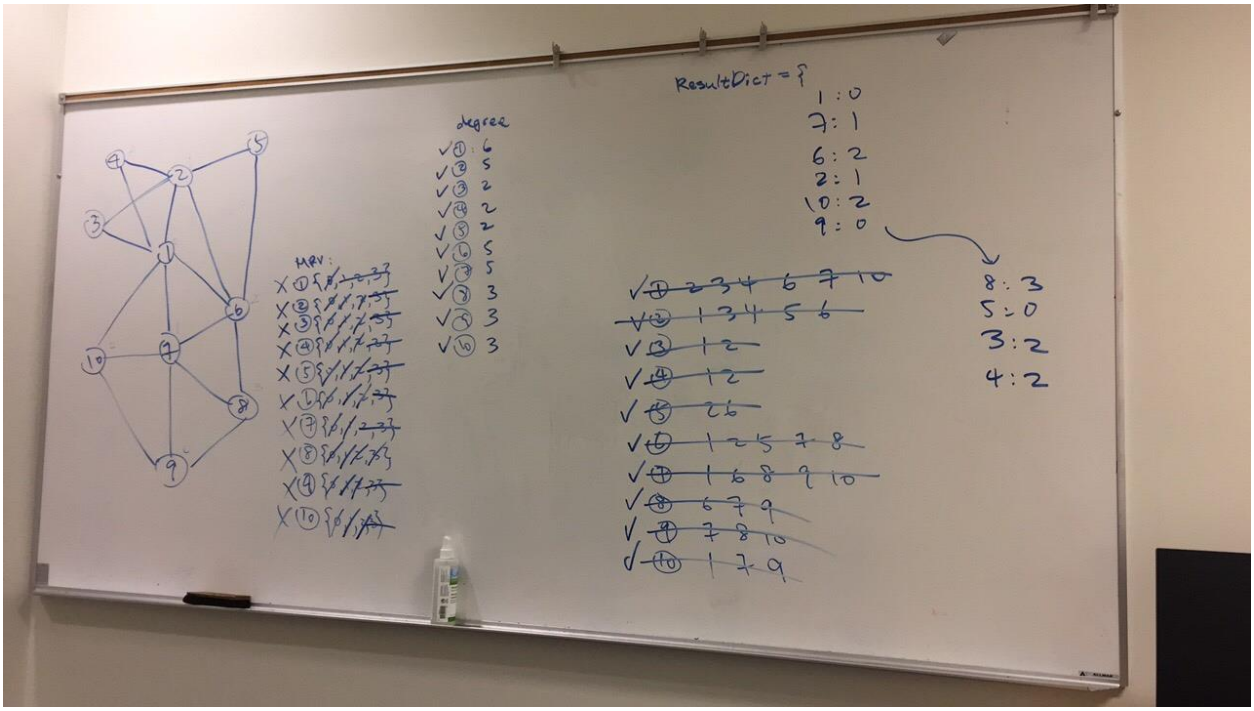
- Given a variable, have to decide which value to assign
- Choose the least constraining value
- Choose the one that rules out the fewest values in the remaining variables
- Leaves maximal flexibility for a solution



Jerry Chienyu Chen

October31/2016

Testing with Given Data Sets



Taking the given data set and testing it by hand in one of the CSIL rooms to test out if my program is outputting the right result was quite fun! As shown on the top right hand corner of the picture, the result dictionary has all the nodes of the graph assigned with a color displaying as numbers from 0 to 3. Drawing them out on the left hand side of the picture, the result does follows the constraint given from the assignment question, where adjacent nodes of a node are not assigned with the same color as the node itself. Moreover, it does complete assigning the given number of colors to color all the nodes of the graph which means that the result of my program is correct!

Testing with and without Heuristics

As mentioned in the optional section of the assignment, we are asked to compare the result with and without using the 3 heuristics introduced above. Using the 2 test cases given from the assignment, due to the small size of the test cases, the result of using the heuristics and without the heuristics both take about 0.1 – 0.2 seconds to complete which cannot really observe that the algorithm with heuristics is more optimal under this circumstance. But when tested with bigger data sets, the algorithm without heuristics is not even able to output a result at all because of the large data set. On the other hand, using the algorithm with heuristics that guides the program to select optimal node with optimal color assigned at each step gives us an output result even with a data set consist of almost 10000 nodes. Therefore, we can observe that algorithm using the heuristics does guide us to a solution optimally.

Jerry Chienyu Chen

October31/2016

More Testing