Constraint Satisfaction Problem in Map Coloring

Ayush Agrawal (B14CS010)
Varun Kumar (B14CS039)

The graph for map coloring problem is planar. Although the below algorithms can also run for non-planar graphs, but for map coloring we restrict the graphs to be planar.

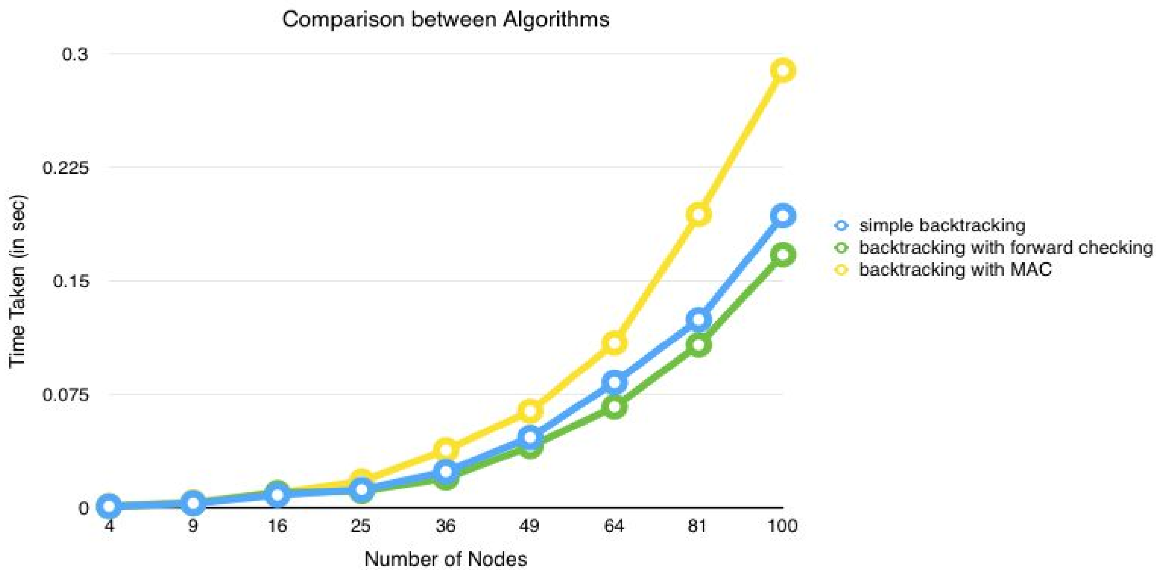
<u>ALGORITHMS IMPLEMENTED :-</u>

1. **Simple Backtracking** - Here depth-first search is used which assigns a value to a variable at a time and backtracks if no value can be assigned to a variable.
2. **Backtracking with forward checking** - Here forward checking is an additional inference to simple backtracking in which we establish arc consistency for an assigned variable. That is, for every unassigned neighbour A of the current variable, we reduce domain of A which is consistent with current assignment.
3. **Backtracking with MAC** - Here MAC is an additional inference to simple backtracking in which after assignment, arc consistency is established for all the arcs containing the neighbours of currently assigned variable. For arc consistency, AC-3 algorithm is used.
4. **Min-conflicts algorithm** - Here a complete assignment is made to all the variables. Then for a fixed number of steps, a random variable is selected and value is assigned to that variable which results in minimum conflicts with the constraints. If, at any time, a solution is found with no conflicts, then it is returned.

HEURISTICS IMPLEMENTED :-

1. **Most constrained variable** - This is a heuristic for the variable to be picked next in the algorithm for assignment. Here, the variable which has least number of values remaining in its domain is selected. That is, it is the most constrained variable.
2. **Degree Heuristic** - This is a heuristic combined with the above heuristic to select the variable with most degree amongst the variables with most constraints as returned by the above heuristic.
3. **Least constraining value** - This is a heuristic for assigning value to a selected variable. It chooses such value which results in least reduction of the domain of neighbouring variables to the current variable.

OBSERVATION :-



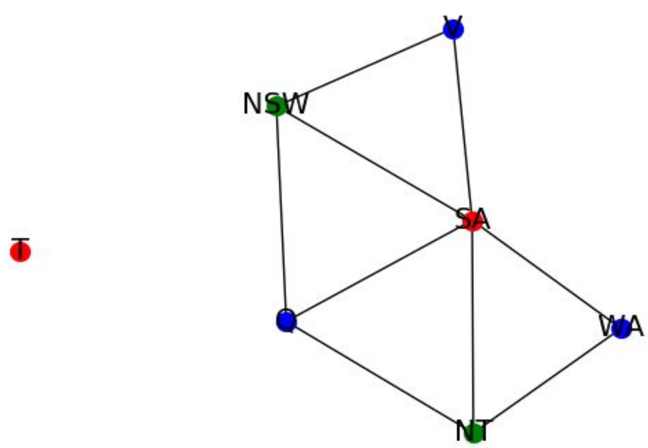
Here we iterate different graphs with n nodes and n*1.5 (approx) edges for each algorithm.

1. Forward checking is faster than simple backtracking because it prunes the further possibilities of assignment which are not consistent with the solution.
2. But MAC, which is an devised for optimization in backtracking algorithm, takes more time than simple backtracking in case of map coloring problem. It is because though it also prunes the possibilities, it takes time for AC-3 algorithm for maintaining arc consistency in map coloring problem.
3. Min-conflicts algorithm is based on random selection of variable to be assigned value. Thus, it can find the solution even quicker than all backtracking variants or it can be unsuccessful to find a solution in the given number of steps (mostly in large graphs because of too many random inconsistent assignments possible).

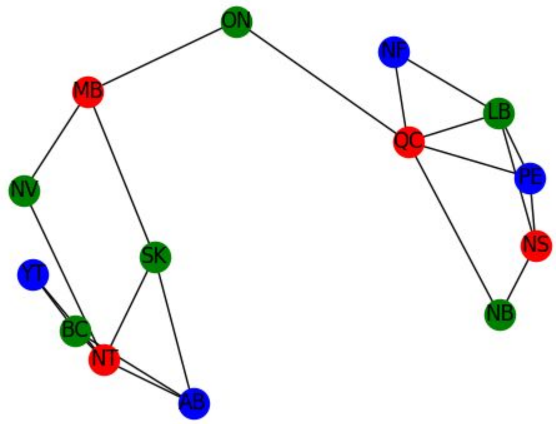
OUTPUT EXAMPLES :-

Following examples implement both the heuristics shown above for all variants of the backtracking algorithm. For the examples, we have used the map of australia, the map of canada and a random map with fixed number of vertices and edges.

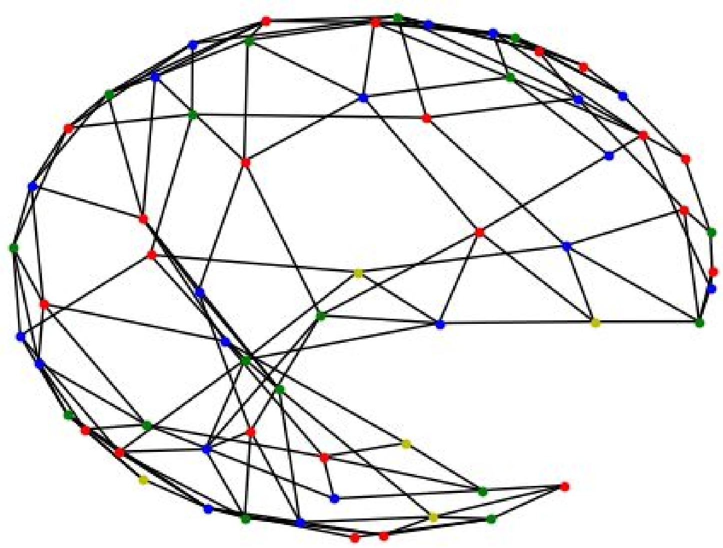
Map coloring for Australia :



Map coloring for Canada :



Map coloring for Random Graph (40 vertices, 80 edges, 4 colors) :



Example 1 -

Australia (3 colors) :
simple backtracking time (found) : 0.003086 secs
backtracking with FC time (found) : 0.002733 secs
backtracking with MAC time (found) : 0.004974 secs
min-conflicts time (found) : 0.001386 secs

Canada (3 colors) :

simple backtracking time (found) : 0.011982 secs
backtracking with FC time (found) : 0.009427 secs
backtracking with MAC time (found) : 0.016954 secs
min-conflicts time (found) : 0.003991 secs

Random Graph (40 vertices, 80 edges, 4 colors):
simple backtracking time (found) : 5.49859 secs
backtracking with FC time (found) : 0.05254 secs
backtracking with MAC time (found) : 6.20188 secs
min-conflicts time (not found) : 36.6748 secs

From the above example, it is evident that simple backtracking takes much time than backtracking with forward checking. While backtracking with MAC takes slightly more time than simple backtracking.

Example 2 -

Australia :
simple backtracking time (found) : 0.001213 secs
backtracking with FC time (found) : 0.000916 secs
backtracking with MAC time (found) : 0.001326 secs
min-conflicts time (found) : 0.000571 secs

Canada :
simple backtracking time (found) : 0.004521 secs
backtracking with FC time (found) : 0.003054 secs
backtracking with MAC time (found) : 0.005034 secs
min-conflicts time (found) : 0.001253 secs

Random Graph2(40 vertices, 80 edges, 4 colors):
simple backtracking time (found) : 0.049858 secs
backtracking with FC time (found) : 0.036369 secs
backtracking with MAC time (found) : 0.066791 secs
min-conflicts time (found) : 0.010521 secs

Min-conflicts algorithm can find solution in very less time for some graphs, but for some graphs, it takes much time and even then it can't find the solution for limited number of iterations.