CS 4253/6613: Artificial Intelligence

Project 1b: Constraint Satisfaction Techniques

1   Introduction

In this project you will write Python functions as part of a program to develop a general constraint satisfaction problem solver and apply it to two well-known domains:

Map coloring: Given a set of regions on a map, the goal is to assign colors to regions so that no two adjacent regions have the same color. You are provided a generator for producing random map-coloring problem instances, using the algorithm given in Exercise 6.10 of the Russell & Norvig textbook (3rd Edition). The generator asks for the number of nodes in the problem and outputs the regions and their neighborhood mappings in a JSON file. You should experiment with at least 10 instances of different problem sizes by varying the number of nodes from 10, 50, and 100.

Sudoku: A Sudoku puzzle piece consists of a partially filled 9-by-9 matrix which is further divided into nine 3-by-3 regions. Each cell can contain an integer between 1 and 9. The puzzle constraints are that integers in each row, column, and region must be distinct. To solve a puzzle the player has to fill in all empty cells while satisfying the constraints. You will be provided a Sudoku problem generator for use in your experiments. Your results should be averaged over at least 100 problem instances.

2   Algorithms and heuristics

Constraint satisfaction problems can be solved by various techniques. In this project you will first implement *depth first search with backtracking as provided in Figure 6.5 of the texbook*. You will then first add *forward checking* and thereafter the *arc consistency (AC3 algorithm) as provided in Figure 6.3 of the texbook*. You also need to implement and experiment with the following variable ordering heuristics:

- random

- minimum-remaining value,

- minimum-remaining value together with degree.

What to hand in: Your code must be clear, concise, and well documented. Your grade will principally depend on a report for the project that contains your analysis of the relative advantages and cost of forward checking and AC3, as well as various variable ordering heuristic, for each of the problem domains.