Assignment 3 – CS4300

Arc Consistency Algorithms

Braden Scothern & Kyle Heaton

# Introduction

In order to explore Arc Consistency, we have measured the AC-1 and AC-3 algorithms when applied to the *N*-queens problem using a series of tests. The tests cover the range of *N* = 4:10 with 200 randomly generated *N* x *N* boards for each percentage *p*, of ones which varies from 0 to 1 in steps of 0.2. This results in 1200 boards being tested for each *N*. As these tests are run we have answered the following questions

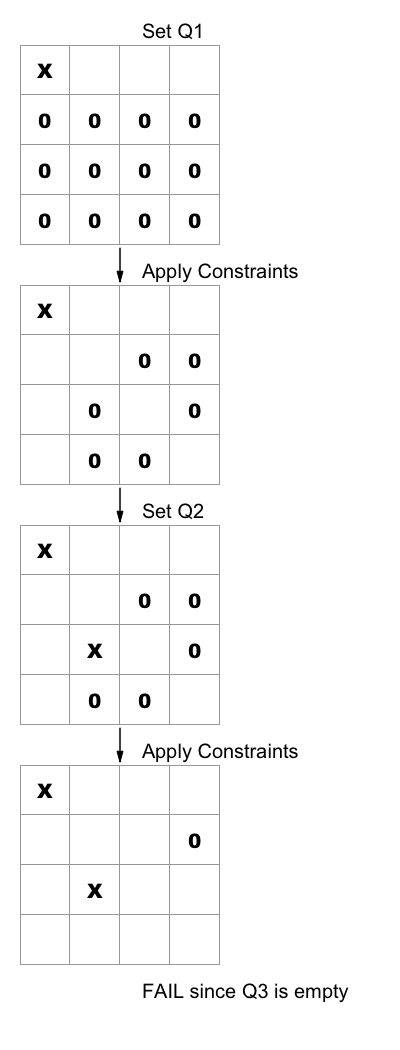
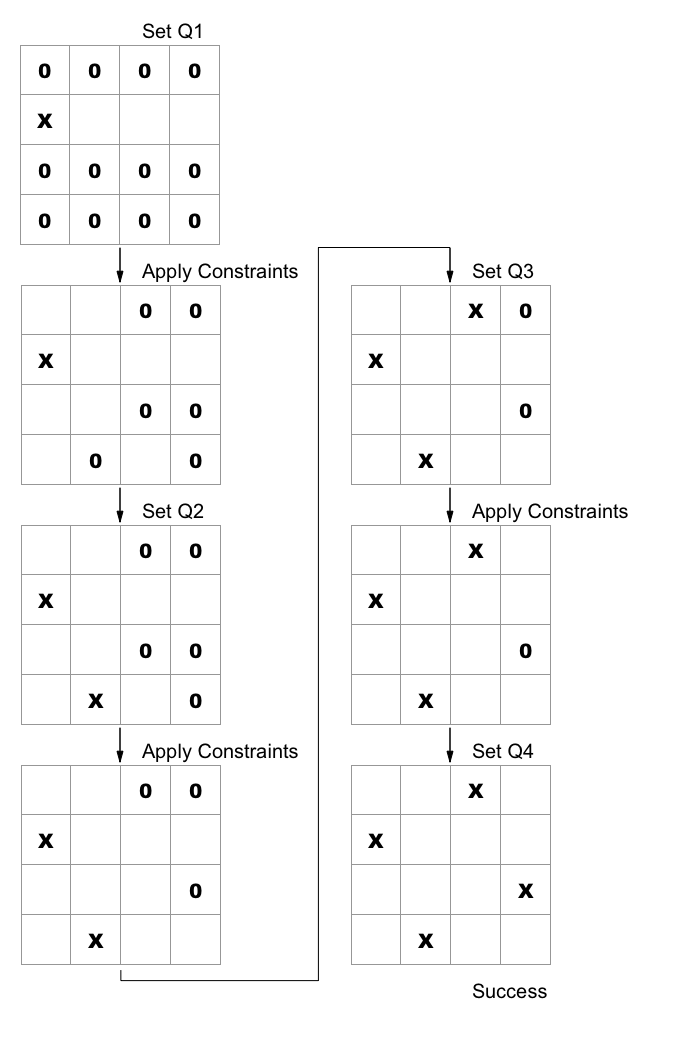
* What is the number of ones before and after the application of the constraint algorithms?
* What is the execution time of each algorithm for each trial (using tic and toc)?

# Method

Our method on this assignment was wholly unsuccessful. The plan however was to first, randomly generate a domain matrix according to all of the values of n and p. Second track how many ones were in that domain before and after running the Arc Consistency functions on them. Since the results of Ac1 and Ac3 should produce the same reductions, only Ac3 was recorded and used for reduction calculations. Third, time how long it would take AC1 and AC3 to run and experimentally determine then compare their complexities.

# Verification of Program

In order to make sure that the logic of checking that a queen has been placed in the correct location we will calculate some examples by hand.



**Example 2**

**Example 1**

Results from Matlab.

* Example 1
* Example 2

# Data and Analysis

I would love to put some information in here, but I wasn’t ever able to get the data into some manageable form. I found myself having lots and lots of matrices that were mostly 0s, and a little bit of data in them. I spent most of my time on this assignment setting up a structure to be able to easily extract results from our tests and present them, but I failed miserably.

# Interpretation

# Critique

The hardest part of this assignment for me was Matlab. I can’t blame my lack of progress on the language, but that is where I struggled the most. When trying to do averages of reduction results, I had so many for loops that I got half way through and had to start over because it was such a mess I didn’t know what was going on. I am sure there is a Matlab way of doing a lot of the things, but they evaded me. I really would have liked to a better explanation on what kind of data was expected. I liked the graphs, but I had questions that I didn’t ask soon enough. This assignment for me was a difficult balance between trying to figure out things on my own, and feeling like I am asking for a handout from the professor.

# Log

Braden Scothern

* 6 hours – Writing and debugging code
* 3 hours – Writing report sections 1, 3, & 5

Kyle Heaton

* 5 hours getting framework surrounding AC1 and AC3 to be able to generate domain matrices and pull the data we need from tests
* 8 hours trying to get the data that we are pulling from AC1 and AC3 into some manageable form to plot it and actually be able to glean some knowledge from the graphs
* 1 hour writing sections 2, 4, 6 in lab report

# Appendix

MATLAB Code Files with Brief description: