**University of Dhaka**

**Department of Computer Science and Engineering**

**CSE-4111: Artificial Intelligence Lab**

**4th Year 1st Semester**

**Session**: 2018 -19

**Assignment No:** 02

**Assignment Topic:**

Experimental Settings for arc consistency algorithm.

**Submitted by:**

Amit Roy, Roll: JH- 400

**Submitted to:**

Dr. Md. Mosaddek Khan, Lecturer

Md. Mofijul Islam, Lecturer

Department of Computer Science and Engineering,

University of Dhaka

**Date of Submission:** 07 February, 2019

**Problem Definition:**

**Constraint Satisfaction Problem:**

A constraint satisfaction problem**(CSP)** consists of three components, **X**, **D**, and **C**:

**X** is a set of variables, **{X1 , . . . , Xn }**.

**D** is a set of domains, **{D1 , . . . , Dn }**, one for each variable.

**C** is a set of constraints that specify allowable combinations of values.

Each domain **Di** consists of a set of allowable values, **{v1 , . . . , vk}** for variable **Xi**. Each constraint **Ci** consists of a relation between two variables. This type of constraint is called binary constraint

An assignment that does not violate any constraints is called a consistent or legal assignment. A **complete assignment** is one in which every variable is assigned, and a solution to a **CSP** is a **consistent**, **complete assignment**.

A variable in a CSP is **arc-consistent** if every value in its domain satisfies the variable’s binary constraints.

***Xi*** is arc-consistent with respect to another variable ***Xj*** if for every value in the current domain ***Di*** there is some value in the domain ***Dj*** that satisfies the binary constraint on the arc ***(X, Xj)***. A network is **arc-consistent** if every variable is arc-consistent with every other variable.

In this problem we need to implement the most popular algorithm for arc consistency named AC-1, AC-2, AC-3, AC4.

**Experimental Setting(Plan):**

1. A set of constraints which will contain some relations between two variables of the consistency graph. The relations could be,

**x<=y**

**x>=y**

**y=x2**

**gcd(x,y) = 1**

**y%x = 0**

2. We will generate random graph using some python library named NetworkX for different number of nodes (e.g. 10,20,30,40,50)

3. For each node, we will consider it as a variable and randomly choose it’s domain. For each pair of variable we will choose one or more binary constraint from our constraint list(1).

4. Now, we will run the AC-1, AC-2, AC-3,AC-4 algorithm on the randomly generated graphs. For the same number of nodes we will generate different random graphs and run the algorithms several times(10-20) and take the average value of running time.

5. We will plot a graph where the **X axis** will be the number of nodes and the **Y axis** will be the performance measure. We will get 4 different lines for four different algorithms and then compare amongst the graphs to infer some significant knowledge of the performance difference of the four arc consistency algorithms.