# Constraint Satisfaction Problems

by

Nwodo Nenne

(Matriculation Number: 130805063)

*Department of Computer Sciences*

*University of Lagos*

(Ref: CSC 514/2016-17)

***This paper provides information about Constraint Satisfaction problems***

Constraint satisfaction problems (CSPs) are mathematical problems defined as a set of objects whose state must satisfy a number of constraints or limitations.

## THE FORMAL DEFINITION

Formally, a constraint satisfaction problem is defined as a triple **<X, D, C >,** where

X is a set of variables,

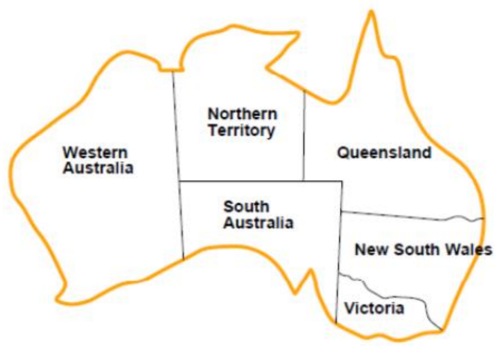
D is a set of the respective domains for each variable

C is a set of constraints.

The aim is to choose a value for each variable so that the resulting possible world satisfies the constraints; we want a model of the constraints.
A finite CSP has a finite set of variables and a finite domain for each variable. Many of the methods considered in this chapter only work for finite CSPs, although some are designed for infinite, even continuous, domains.

## EXAMPLE

### Graph Coloring as a CSP




***Variables:*** Western Australia, Northern Territory, South Australia, Queensland, New South Wales , Victoria

***Domains:*** $D_i$ = {red, green, blue}

***Constraints:*** adjacent regions must have different colors

***Solutions*** are assignments satisfying all constraints, e.g.

{*Western Australia* = **red**, *Northern Territory* = **green**, *Queensland* = **red**, *New South Wales* = **green**, *South Australia* = **blue**, *Victoria* = **red**}

An evaluation is *consistent* if it does not violate any of the constraints.

An evaluation is *complete* if it includes all variables.

An evaluation is a *solution* if it is consistent and complete; such an evaluation is said to *solve* the constraints satisfaction problem.

## RESOLUTION OF CSPS

Constraint satisfaction problems on finite domains are typically solved using a form of search. The most used techniques are variants of **Backtracking**, **Constraint propagation**, and **Local search.**

**Backtracking** is a recursive algorithm. It maintains a partial assignment of the variables. At first, all variables are unassigned. At each progression, a variable is picked, and every single conceivable value is appointed to it thus. For each value, the consistency of the partial assignment with the constraints is checked; if there should arise an occurrence of consistency, a recursive call is performed

**Constraint propagation** techniques are techniques used to modify a constraint satisfaction problem. More precisely, they are methods that implement a type of local consistency, which are conditions related to the consistency of a group of variables and/or constraints.

**Local-search algorithms** turn out to be very effective in solving many CSPs. They use a complete-state formulation: the initial state assigns a value to every variable, and the successor function usually works by changing the value of one variable at a time

## THEORETICAL ASPECTS OF CSPS

1. *Decision Problems:* An important question is whether for each set of relations, the set of all CSPs that can be represented using only relations chosen from that set is either in P or NP-complete.
2. *Function Problems*

## VARIANTS OF CSPS

1. *Dynamic CSPs*: (*DCSP*s) are useful when the original formulation of a problem is adjusted somehow, typically because the set of constraints to consider evolves because of the environment. DCSPs are viewed as a sequence of static CSPs, each one a transformation of the previous one in which variables and constraints can be added or removed. Information found in the initial formulations of the problem can be used to refine the next ones. DCSPs can be solved using **Oracles**, **Local repair** and **Constraint recording.**
2. *Flexible CSPs:* Classic CSPs treat constraints as hard, meaning that they are *imperative* (each solution must satisfy all of them) and *inflexible* (in the sense that they must be completely satisfied or else they are completely violated). **Flexible CSP**s relax those assumptions, partially *relaxing* the constraints and allowing the solution to not comply with all of them. E.g. MAX-CSP, Weighted CSP, Fuzzy CSP.

3. *Decentralized CSPs:* In DCSPs each constraint variable is thought of as having a separate geographic location. Strong constraints are placed on information exchange between variables, requiring the use of fully distributed algorithms to solve the constraint satisfaction problem.

**BIBLIOGRAPHY**

"Artificial Intelligence: A Modern Approach." *Artificial Intelligence: A Modern Approach*. N.p., n.d. Web. 10 June 2017.

"Constraint satisfaction problem." *Wikipedia*. Wikimedia Foundation, 09 June 2017. Web. 10 June 2017.