**Genetic Algorithms**

Charles Darwin's idea of natural evolution served as the foundation for the search heuristic known as the genetic algorithm.

The genetic algorithm is a method that utilises natural selection, the mechanism that propels biological evolution, for resolving both constrained and unconstrained optimization problems. A population of unique solutions is repeatedly modified by the genetic algorithm. The genetic algorithm chooses members of the present population to serve as parents at each stage and utilises them to produce the offspring that will make up the following generation. The population "evolves" toward the best option over the course of subsequent generations. The genetic algorithm can be used to tackle several optimization problems, including those where the objective function is discontinuous, nondifferentiable, stochastic, or highly nonlinear and are not well suited for typical optimization algorithms. When some components must only have integer values, mixed integer programming problems can be solved using the evolutionary algorithm.

A genetic algorithm uses three main rules when creating the next generation from the current population, which are selection, crossover, and mutation.

* Selection rules choose the parents, who will contribute to the population of the following generation.
* Crossover rules combine two parents to create the next generation's offspring.
* Mutation rules subject each parent to random modifications.

Genetic algorithms are good at taking large search spaces and navigating through them looking for the most optimal combinations. Ordinarily these problems are very difficult to solve and can be very exhaustive.

**Constraint Satisfaction Problems:**

Problem definition

A CSP consists of:

* A set if variables X = {X1,…,Xn} ;
* For each variable xi, a finite set Di of possible values (domain)
* Set of constraints restricting values that the variables can take

The following form can be used to define a constraint satisfaction problem (CSP) in a (finite domain). Find values for the variables that satisfy each constraint given a collection of variables, a finite set of possible values for each variable, and a list of constraints. An example of this occurs in production scheduling. To ensure that each work is finished by the specified deadline, jobs must be processed on machines that can only handle one job at a time. Additional examples follow from the notion that an optimization problem can be stated as a series of CSPs. The solution to a CSP includes consistent and complete assignment. Where a consistent assignment dictates that an assignment does not violate any constraints and a complete assignment is where every variable is assigned.

**Current bibliography**

<https://www.mathworks.com/help/gads/what-is-the-genetic-algorithm.html>

<https://www.oxfordreference.com/view/10.1093/oi/authority.20110803095620476>

<https://www.cs.mcgill.ca/~dprecup/courses/AI/Lectures/ai-lecture05.pdf>