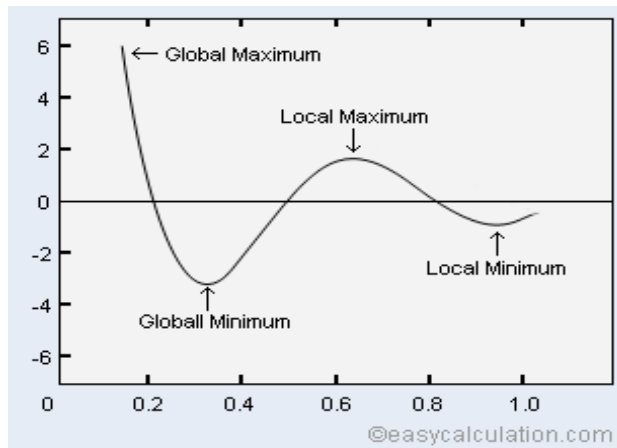


Nontraditional Optimization: GA

- To find a global minimum, users normally try a heuristic approach where several local minima are found by repeated trials with different starting values or by using different techniques
- The smallest of all known local minima is then assumed to be global minima. Certainly not very reliable.
- Traditional techniques are mostly derivative based



- **Features of Nontraditional Techniques**

- Do not require derivative information (derivative free methods)
- Search wide domains of the objective/cost surface
- Can efficiently deal with a large number of variables
- Suitable for extremely complex cost surfaces; multiobjective problems
- Suggests possible other sub-optimal solutions which may be more suitable for implementation
- Highly flexible and versatile

- Computationally expensive (more suitable for off-line applications)
- Intuitive (no strong mathematical ground; black box nature) and inspired from some natural/biological process
- Stochastic in nature (relying heavily on random number generation)
- Large number of hyper parameters (problem specific tuning)

- **Some Nontraditional Optimization Techniques**

- Genetic Algorithm (GA)
- Differential Evolution (DE)
- Simulated Annealing (SA)
- Particle Swarm Optimization (PSO)
- Ant Colony Optimization (ACO)
- Artificial Bee Colony (ABC)

- Artificial Immune System (AIS)
- Gray Wolf Algorithm (GWA)
- Firefly Algorithm (FA)
- Bacterial Foraging Optimization (BFO)
- Cuckoo Search Algorithm (CSA)
- Harmony Search (HS)

and the list goes on ...

Genetic Algorithm (GA)

- Introduced by John Holland in 1975
(University of Michigan) and
popularized by his student David Goldberg
- Philosophically, GA is based on Darwin's
theory of survival of the fittest
- Basic idea is to start from a randomly chosen set of probable solutions
and then allow intermixing of these candidate solutions over many
generations and gradually evolve towards the optimal solution



Steps for Binary Coded GA (BCGA)

- In principle, suitable for maximization problems
- Let us first consider the unconstrained optimization problem

- find $\underline{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$ to

$$\text{maximize } f(\underline{x})$$

$$\text{where } x_i^{\min} \leq x_i \leq x_i^{\max} \quad i = 1, 2, \dots, n$$

$$\text{i.e. } \underline{x} \in \Omega \subset \mathcal{R}^n$$

Initialization

Encoding

Fitness Evaluation

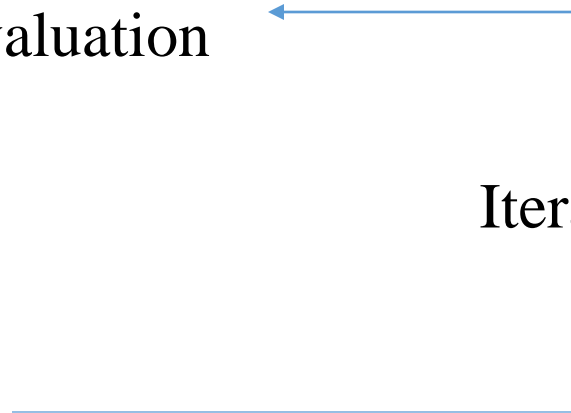
Selection

Crossover

Mutation

Decoding

Iteration



i) Initialization

- A population size N is chosen and then N random points are picked from the phenotype space (search space) Ω
- N is even and usually a few 10's to a few 100's (not too high!)

ii) Encoding

- N points from the phenotype space are mapped to N corresponding points in the genotype space
- Each variable x_i is represented by a binary substring of length l_i
- Such binary substrings are concatenated for all $i=1,2,\dots,n$
- Performed for all N points

- l_i depends on the desired accuracy level of x_i

- $\epsilon_i = \frac{x_i^{max} - x_i^{min}}{2^{l_i} - 1}$

- $l_i = \text{ceiling} \left[\log_2 \left(1 + \frac{x_i^{max} - x_i^{min}}{\epsilon_i} \right) \right]$

- Mapping from the range $[x_i^{min}, x_i^{max}]$ to $[0, 2^{l_i} - 1]$ is done as follows:

$$x_i^d = \text{round} \left\{ \frac{x_i - x_i^{min}}{x_i^{max} - x_i^{min}} (2^{l_i} - 1) \right\} \text{ or } x_i = x_i^{min} + \epsilon_i' \cdot x_i^d$$

(ϵ_i' is the actual resolution)

- x_i^d is then converted to binary substring D_i

Numerical Example:

- Two design variables $x_1 \in [-2, 2]$ and $x_2 \in [1, 12]$
- To represent $\underline{x} = \begin{pmatrix} -0.85 \\ 4.21 \end{pmatrix}$ as a binary string or chromosome.
- Desired accuracy level of 0.2 for both x_1 and x_2 .

Solution:

resolution for x_1 ,
$$\epsilon_1 = \frac{x_1^{\max} - x_1^{\min}}{2^{l_1} - 1}$$

solving for l_1 , $2^{l_1} = 21$ i.e. $l_1 = 5$

similarly,
$$\epsilon_2 = \frac{x_2^{\max} - x_2^{\min}}{2^{l_2} - 1}$$

solving for l_2 , $l_2 = 6$

Now, $x_1 = x_1^{min} + x_1^d \cdot \epsilon'_1$ where ϵ'_1 is the actual resolution of x_1

$$\epsilon'_1 = \frac{x_1^{max} - x_1^{min}}{2^{l_1} - 1} = 0.13$$

$$\therefore -0.85 = -2 + 0.13 \times x_1^d$$

$$\Rightarrow x_1^d = 8.9 \approx 9$$

Hence, binary representation of -0.85 is '01001'

similarly,

$$x_2 = x_2^{min} + x_2^d \cdot \epsilon'_2 \quad \text{where } \epsilon'_2 \text{ is the actual resolution of } x_2$$

$$\epsilon'_2 = \frac{x_2^{max} - x_2^{min}}{2^{l_2} - 1} = 0.1746$$

$$\therefore 4.21 = 1 + 0.1746 \times x_2^d$$

$$\Rightarrow x_2^d = 18.38 \approx 18$$

Hence, binary representation of 4.21 is '010010'

Hence, '01001010010' maps the point $\underline{x} = \begin{pmatrix} -0.85 \\ 4.21 \end{pmatrix}$

iii) Decoding and Fitness Evaluation

- Next the fitness values at all the N points in the genotype space are computed
- Extract the decimal integer x_i^d from the GA string and then map back to the phenotype space as

$$x_i = x_i^{min} + \epsilon'_i \cdot x_i^d$$

- Compute the objective function value once all x_i (i=1, 2, ... n) are obtained for a particular chromosome
- This is usually the fitness value of that chromosome

Exercise:

Let an optimization problem have two design variables x_1 and x_2 where both are varying in the range $[-5, 5]$. If they are represented by 6 bit binary substrings each then what is the point in the phenotype space corresponding to the chromosome 100110 011100 ?

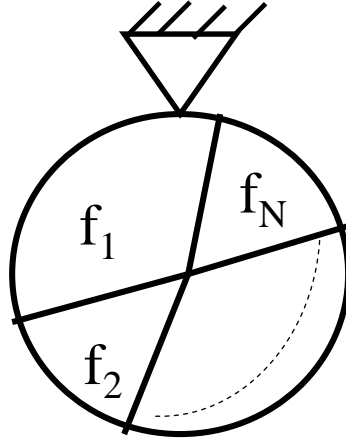
Answer: $\begin{pmatrix} 1.03 \\ -0.55 \end{pmatrix}$

iv) Selection

A mating pool (size N) is selected from the population

(a) Proportionate selection/Roulette-Wheel selection

- Probability of getting selected in a mating pool \propto fitness
- Implemented with the help of a Roulette-Wheel

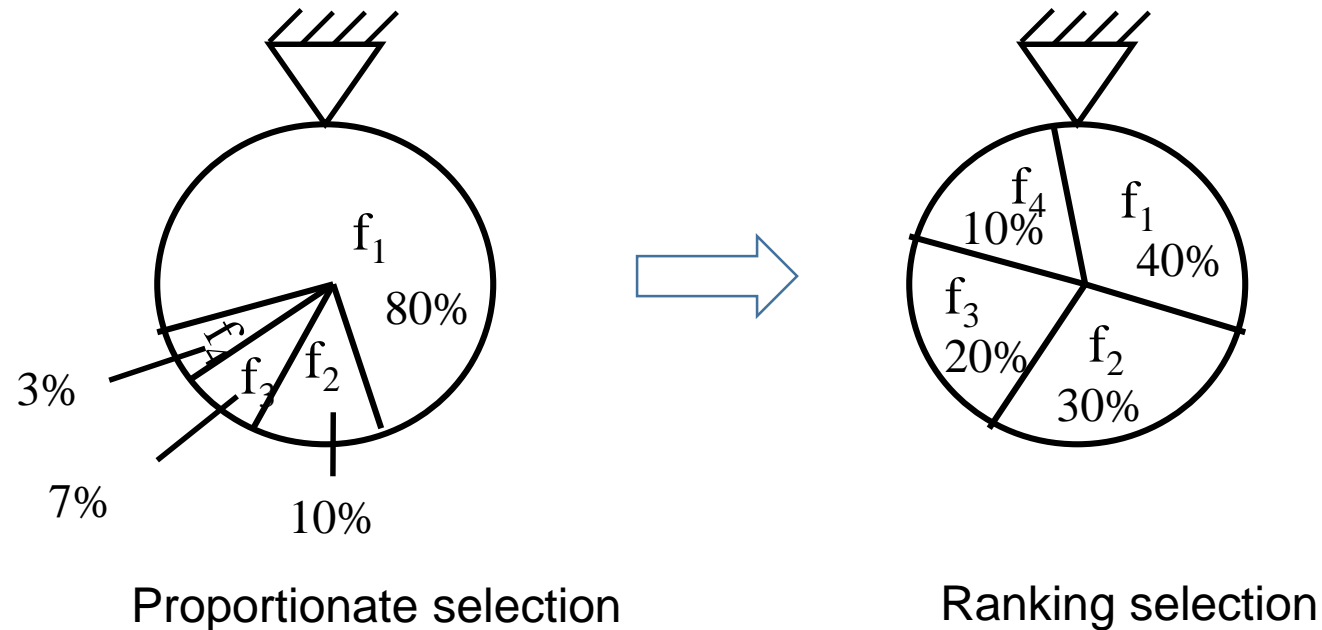


$$P_i = \frac{f_i}{\sum_{i=1}^N f_i}$$

- Balancing Population Diversity and Selection Pressure (Exploration/Exploitation balance)

(b) Ranking Selection

- Least fit string is given a rank of 1, next one rank 2 etc.
- Thus the fittest string gets the highest rank i.e. N
- Then Roulette wheel selection is performed based on the ranks (rather than on the fitness)



(c) Tournament Selection

- Small number (2-3) of chromosomes are randomly picked from the population and the fittest is put in the mating pool
- All the candidates are returned to the original population
- The process is repeated N times
- The least fit candidate is eliminated