Department of CSE-H

MATHEMATICAL PROGRAMMING
22MT2004

GENETIC ALGORITHM

CO - 4

Session - 22

P. Murali Krishna
Asst. Professor

AIM OF THE SESSION

To familiarize students with the basic concept of genetic algorithm with example problem.

INSTRUCTIONAL OBJECTIVES

This Session is designed to:

1. Describe the Genetic algorithm.
2. Describe the Importance of Infinite Dimensional Optimization

LEARNING OUTCOMES

At the end of this session, student should be able to:

1. understand key terms such as chromosome, gene, population, fitness function, crossover, mutation, and selection..
1. Understand the Genetic algorithms with example problem

Mathematical Optimization

Mathematical optimization is the process of finding the best set of inputs that maximizes (or minimizes) the output of a function.

In the field of optimization, the function being optimized is called the objective function.

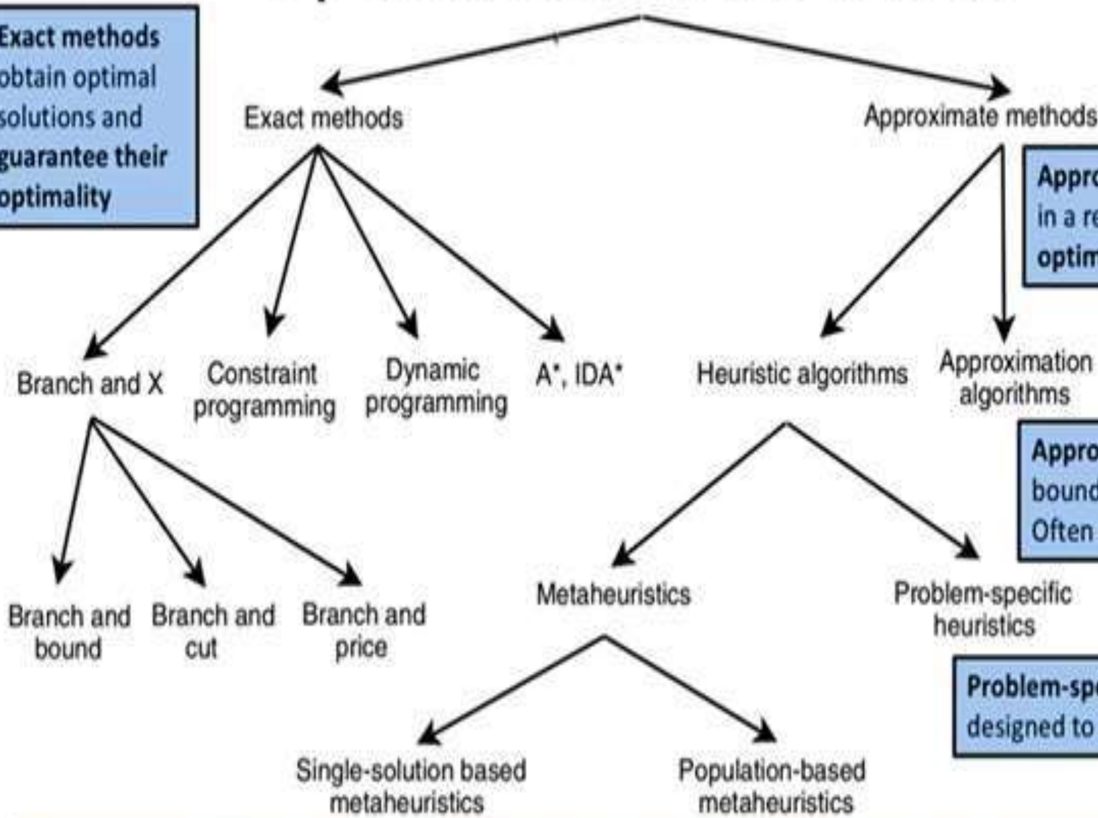
$max\ z = 4x_1x_2 + 2x_2 + 8x_3$	*Objective Function*
$s.t.\ x_1 + x_2 + x_3 \geq 26$	*Inequality Constraint*
$x_1^2 + x_2^2 + x_3^2 + x_4^2 = 40$	*Equality constraint*
$1 \leq x_1x_2x_3 \geq 25$	*bounds on Variables*
$x_1 = (10,12,46)$	*Initial Values*

Solution strategies for Optimization Problems

Methods to solve Optimization Problems	**Nature of Solution**
Linear or Non Linear programming	Exact Solution
Branch and Bound	Exact Solution
Heuristic Method	Inexact, Near optimal Solution
Meta-heuristic Method	Inexact, Near optimal Solution

Optimization methods

Exact methods obtain optimal solutions and **guarantee their optimality**



Approximate methods generate high-quality solutions in a reasonable time, but they **do not guarantee the optimality of the obtained solutions**.

Approximation algorithms ⊂ Approximate methods

Approximation algorithms provide provable bounds on solution quality and on run-time. Often obtained by relaxation.

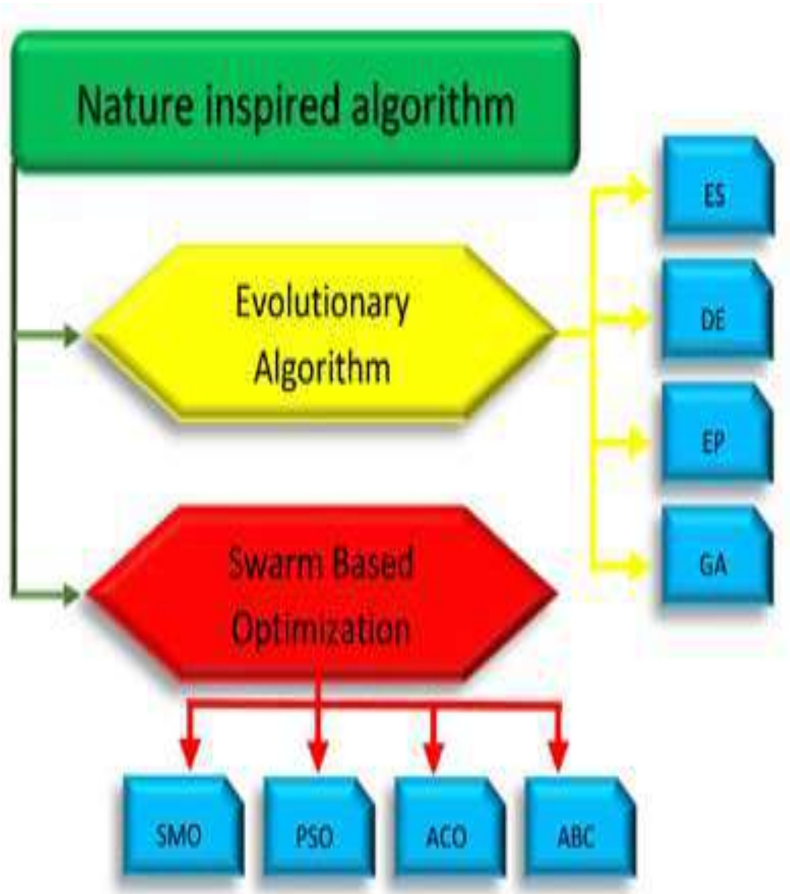
Problem-specific heuristics are tailored and designed to solve a specific problem and/or instance.

Metaheuristics are **general-purpose algorithms** that can be applied to solve almost any optimization problem. Unlike approximation algorithms, metaheuristics **do not provide any bound** on how close the obtained solutions is to the optimal one. Unlike exact methods, metaheuristics **allow to tackle large-size problem instances by delivering satisfactory solutions in a reasonable time**

Traditional Optimization Techniques – Problems!

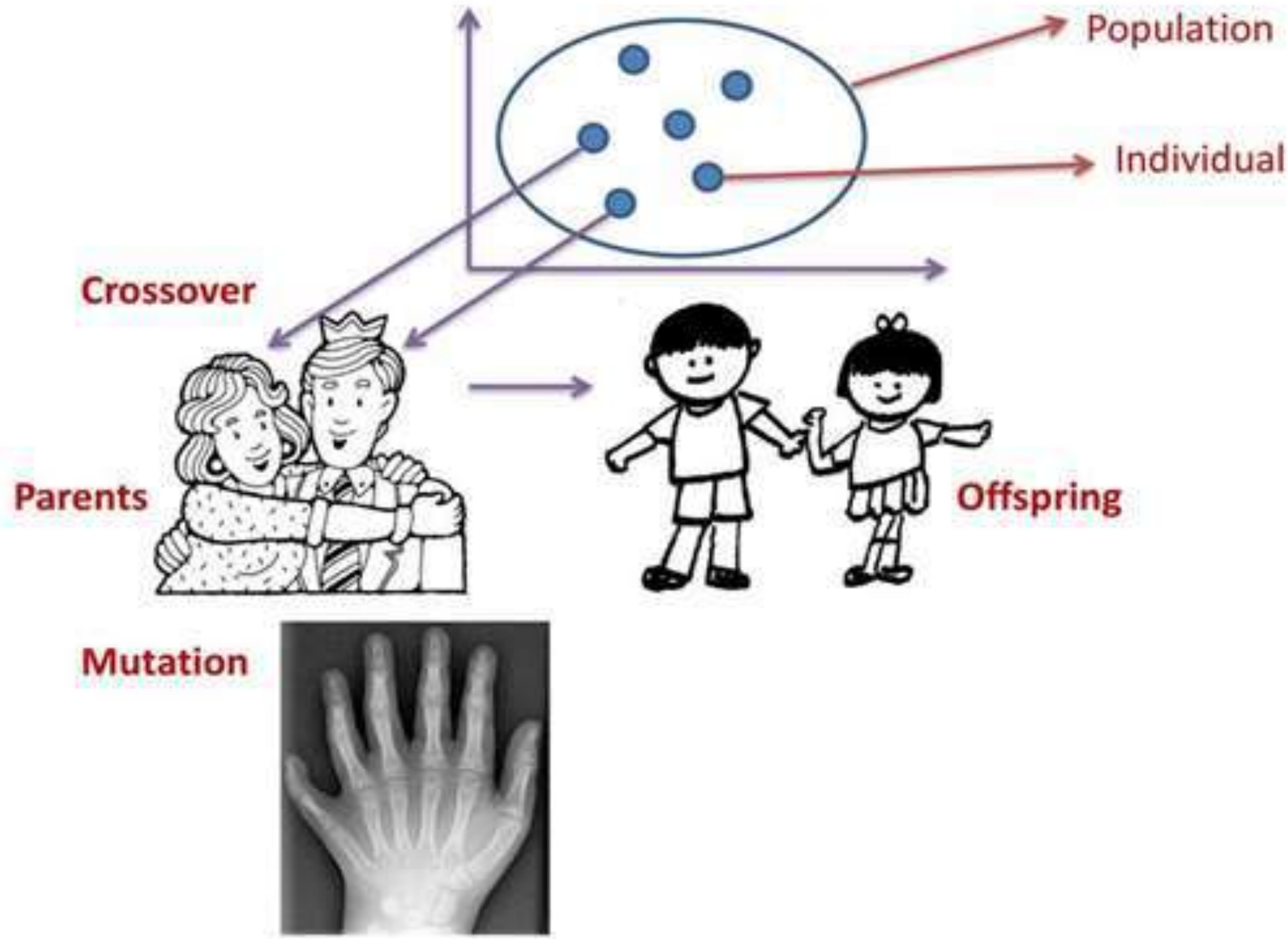
- Different methods for different types of problems.
- Constraint handling e.g. using panalty method is sensitive to penalty parameters.
- Often get stuck in local optima (lack global perspective).
- Usually need knowledge of first/second order derivatives of objective functions and constraints.

Nature Inspired Algorithms for Optimization



GA→ Genetic Algorithm
ES → Evolution Strategy
DE → Differential Evolution
EP → Evolution Programming
ABC → Artificial Bee Colony
ACO→ Ant Colony Optimization
PSO → Particle Swarm Optimization
SMO→ Sequential Minimal Optimization

Evolutionary Algorithms



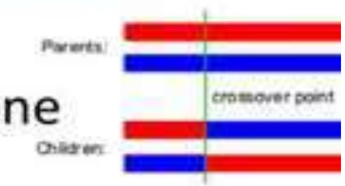
Evolutionary Algorithms

- **Terminologies**
  1. **Individual** - carrier of the genetic information (chromosome). It is characterized by its state in the search space, its fitness (objective function value).
  2. **Population** - pool of individuals which allows the application of genetic operators.
  3. **Fitness function** - The term "fitness function" is often used as a synonym for objective function.
  4. **Generation** - (natural) time unit of the EA, an iteration step of an evolutionary algorithm.

Features of the EAs

Evolutionary Algorithms

- Selection - Roulette wheel, Tournement, steady state, etc.
  - Motivation is to preserve the best (make multiple copies) and eliminate the worst
- Crossover – simulated binary crossover, Linear crossover, blend crossover, etc.
  - Create new solutions by considering more than one individual
  - Global search for new and hopefully better solutions



- Mutation – Polynomial mutation, random mutation, etc.
  - Keep diversity in the population
  - 010110 →010100 (bit wise mutation)

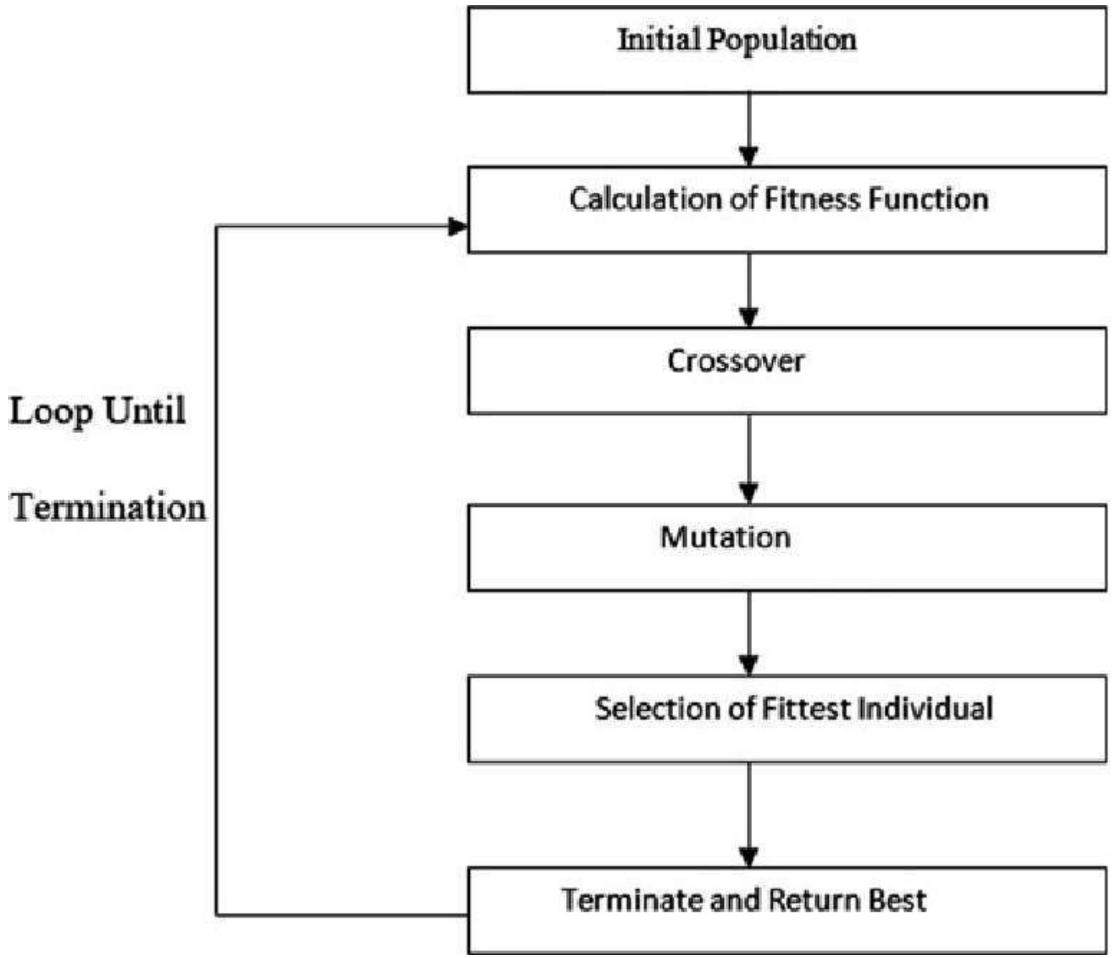
Evolutionary Algorithms

- Concept of **exploration vs exploitation.**
- Exploration – Search for promising solutions
  - Crossover and mutation operators
- Exploitation – preferring the good solutions
  - Selection operator
- **Excessive** exploration – Random search.
- **Excessive** exploitation – Premature convergence.

What is GA

- A **genetic algorithm** (or **GA**) is a search technique used in computing to find true or approximate solutions to optimization and search problems.
- Genetic algorithms are categorized as global search heuristics.
- Genetic algorithms are a particular class of evolutionary algorithms that use techniques inspired by evolutionary biology such as inheritance, mutation, selection, and crossover (also called recombination).

Basic Structure of GA



Example

- Using Genetic algorithm maximize the function

$$\boldsymbol{f(x) = x^2}$$

with x in interval [0, 31] i.e., x = 0,1,2,…..30,31.

Select Encoding Technique: **Binary encoding Technique**

The minimum value is **0** and maximum value is **31**

To represent the values, use 5-digit binary code numbers between 0 to 31

0 (00000) to 31 (11111) is obtained

The objective function is to be maximized $f(x) = x^2$

String No.	Initial Population (Randomly selected)	X value	Fitness value $f(x) = x^2$	Prob. $\frac{f(x)}{\sum f(x)}$	%prob	Expected count $\frac{f(x)}{Ave\sum f(x)}$	Actual count
1	01100	12	144	0.1247	12.47	0.4987	1
2	11001	25	625	0.5411	54.11	2.1645	2
3	00101	5	25	0.0216	2.16	0.0866	0
4	10011	19	181	0.3126	31.26	1.2502	1
Sum			1155	1.0	100	4	4
Average			288.75	0.25	25	1	1
Max.			625	0.5411	51.11	2.1645	2

String No.	Mating pool	Crossover point	Offspring after crossover	X value	Fitness value f(x) = x^2
1	0110|0	4	01101	13	169
2	1100|1	4	11000	24	576
3	11|001	2	11011	27	729
4	10|011	2	10001	17	289
Sum					1763
Average					440.75
Max.					729

Genetic Algorithm Solved Example – Mutation

String No.	Offspring after crossover	Mutation Chromosome for flipping	Offspring after mutation	X Value	Fitness $f(x) = x^2$
1	01101	10000	11101	29	841
2	11000	00000	11000	24	576
3	11011	00000	11011	27	729
4	10001	00101	10100	20	400
Sum					2546
Average					636.5
Maximum					841

String No.	Initial Population (Randomly selected)	X value	Fitness value $f(x) = x^2$	Prob. $f(x)/\sum f(x)$	%prob.	Expected count $f(x)/Avg(\sum f(x))$	Actual count
1	01100	12	144	0.1247	12.47	0.4987	1
2	11001	25	625	0.5411	54.11	2.1645	2
3	00101	5	25	0.0216	2.16	0.0866	0
4	10011	19	181	0.3126	31.26	1.2502	1
Sum			1155	1.0	100	4	4
Average			288.75	0.25	25	1	1
Max.			625	0.5411	51.11	2.1645	2

String No.	Mating pool	Crossover point	Offspring after crossover	X value	Fitness value $f(x) = x^2$
1	0110|0	4	01101	13	169
2	1100|1	4	11000	24	576
3	11|001	2	11011	27	729
4	10|011	2	10001	17	289
Sum					1763
Average					440.75
Max.					729

String No.	Mating pool	Crossover point	Offspring after crossover	X value	Fitness value $f(x) = x^2$
1	0110|0	4	01101	13	169
2	1100|1	4	11000	24	576
3	11|001	2	11011	27	729
4	10|011	2	10001	17	289
Sum					1763
Average					440.75
Max.					729

SELF-ASSESSMENT QUESTIONS

1. What are the genotypes of the parents?

A. Both are homozygous dominant.
B. Both are heterozygous dominant.
C. Both are homozygous recessive.
D. The male is homozygous dominant; the female is homozygous recessive.

2. Consider the following **a. Evaluation b. Selection c. reproduction d. Mutation.** Which of the following are found in genetic algorithms?

A. (b), (c) and (d) only.
B. (b), and (d) only.
C. (a) , (b), (c) and (d) .
D. (a) , (b), and (d) only .

references

- https://slideplayer.com/slide/12678549/
- https://www.slideshare.net/MayankJain430/genetic-algorithm-ppt
- https://www.geeksforgeeks.org/genetic-algorithms/
- https://en.wikipedia.org/wiki/List_of_genetic_algorithm_applications
- https://en.wikipedia.org/wiki/Genetic_algorithm
- https://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol1/hmw/article1.html


Thank you