Introduction to Genetic Algorithm

<u>Cheat Sheet:</u>

1. **Objective function, fitness function, variables and limits.**
   These are some of the terms you should be well versed with before you start attempting any optimization problem.
   Let us consider an optimization problem given by

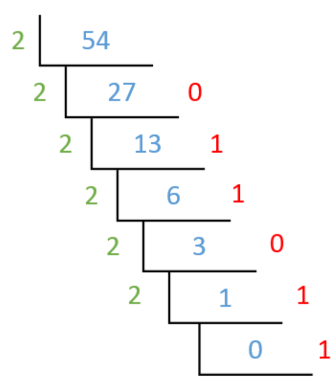
$$y = x_1^3 + x_1^2 + 5x_2^2 + 4x_3^4$$
$$0 \le x_1 \le 6$$
$$-1 \le x_2 \le 100$$
$$-54 \le x_3 \le 546$$

   Here,

   - $y = x_1^3 + x_1^2 + 5x_2^2 + 4x_3^4$ is the objective function or the fitness function (the function you wish to maximize or minimize)
   - x_1, x_2 and x_3 are the variables
   - The limits or the domain of the variables or the search space are

$$0 \le x_1 \le 6$$
$$-1 \le x_2 \le 100$$
$$-54 \le x_3 \le 546$$

2. **Converting decimal to binary**
   Successively divide the decimal number by 2, the remainders of the process taken in reverse order gives the binary form of the number. In the following example we try to find the binary equivalent of the decimal number 54. The process of successive division is shown below.



   The Binary equivalent is 110110

3. **Converting binary to decimal**

The following formula can be used to convert the binary form of a number into its decimal form

$$D = (b_0 \times 2^0) + (b_1 \times 2^1) + \cdots + (b_n \times 2^{n-1})$$

To convert 110110 to decimal form, first we identify the bs and n

n+1 = number of bits = 6 => n = 5
$b_0 = 0$
$b_1 = 1$
$b_2 = 1$
$b_3 = 0$
$b_4 = 1$
$b_5 = 1$

Applying the formula,

$$D = (b_0 \times 2^0) + (b_1 \times 2^1) + \cdots + (b_n \times 2^{n-1})$$
$$D = (0 \times 2^0) + (1 \times 2^1) + (1 \times 2^2) + (0 \times 2^3) + (1 \times 2^4) + (1 \times 2^5)$$
$$D = 54$$

4. How to select the number of bits to represent a variable?

$$n = \log_2\left(\frac{x_{\max} - x_{\min}}{p}\right)$$
$$p = \text{precision required}$$

5. Finding the variable value from the decimal value

Use the following formula to find the variable value x from the decimal value D

$$x = x_{\min} + \frac{x_{\max} - x_{\min}}{2^n - 1} \times D$$

6. Some mutation and cross over schemes

Cross- over

a. Single Point cross over <- Discussed in the talk
b. Multi point cross over <- Similar but splicing happens at more than one location on the chromosome

Mutation

a. Fixed bit mutation <- Mutation always happens on a fixed bit position
b. Random bit mutation <- Discussed in the talk

7. How do I create random numbers?

Random number generators are already available in the literature. Based on the programming language you are using, you can find pre-written codes for random number generation online.

I recommend that you use a random number generator which takes **Time** as the seeding input.

8. Some test functions that you can use to test your GA code

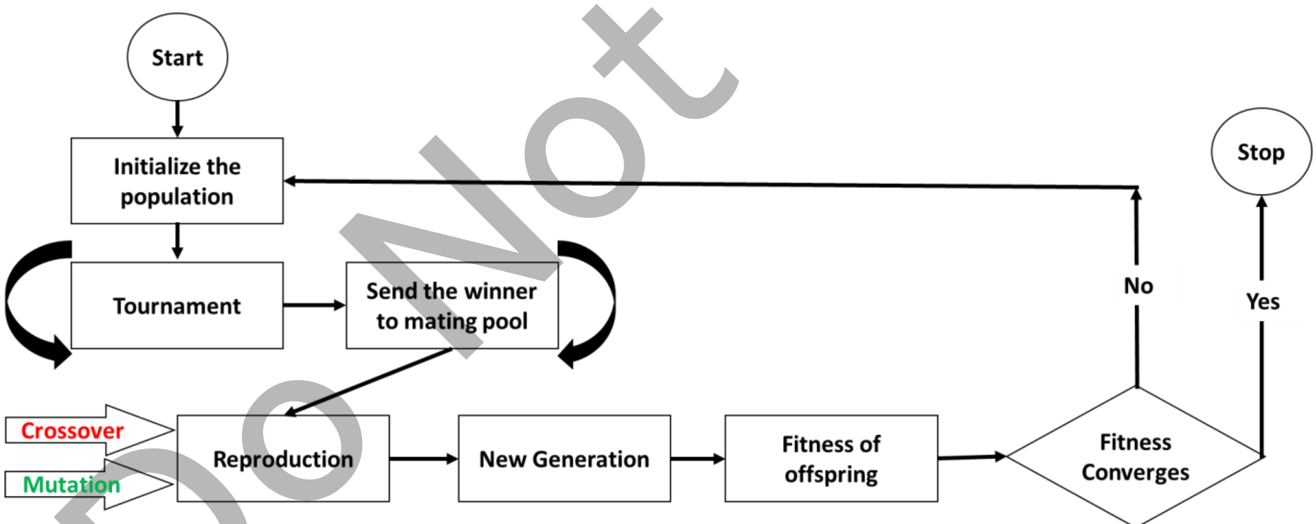
a. Kursawe function
b. Mishra's Bird function – constrained
c. Rosenbrock function constrained with a cubic and a line
d. Rosenbrock function constrained to a disk
e. Zitzler–Deb–Thiele's function

These are some of the standard functions available in the literature, you can google the names of the functions and you will get the functionality and the constraints.

9. How can I use GA in my field?

a. Identify an optimization problem
b. Write down the problem in mathematical form
c. Identify the objectives and the variables
d. Find the constraints of the variables
e. Fix the precision level you want for your answer, find the number of bits required to achieve this precision
f. Solve your problem using GA

10. Overall flow of the process.



11. Converting a minimization problem to a maximization problem.

By definition, GA is a maximization procedure, meaning, you can only find the maxima of a function using GA. In case your problem is a minimization problem, you need to convert it to a maximization problem and then proceed. The following equalities may be used to convert any minimization problem into a corresponding maximization problem.

a) $\min[f(x)] = \max[-f(x)]$
b) $\min[f(x)] = \max[1/f(x)]$
c) $\min[f(x)] = \max[1/(1 + f(x)^2]$

Using option (c) is recommended for obvious reasons.

12. Solving multi-variable problems.

Most of the problems that require optimization are multi-variable in nature. Let us take an example and see how these may be tackled using GA. I am giving the same problem as what was discussed in the talk.

Q. Minimize $y(x_1, x_2) = x_1^2 + x_2^2$; $0 \le x_1 \le 1$ and $0 \le x_2 \le 1$

A. We should first identify the variables as x_1 and x_2

Suppose we use four bits to represent each variable, we now populate the initial population with individuals of $4n$ bits where n is the number of variables.

x_1	x_2		**Individual**
1001	1111	→	10011111
1100	1011	→	11001011
…	…	…	…
1011	1101	→	10111101

We do the crossover, mutation etc. on the individual strings but we find the decimal values of $x_1$and x_2 separately.

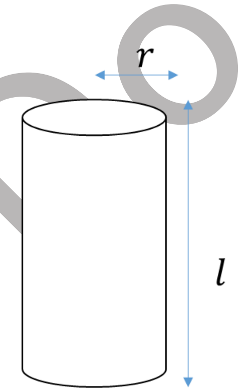
Note: The number of bits which represent the variables need not be the same, meaning you can represent x_1 with 4 bits and x_2 with 6 number of bits. The more the number of bits, the better will be your resolution. (You can use this once you have mastered the basic algorithm).

13. Solving multi-objective problems.

Many a times, you may have to solve certain multi-objective problems. Let us look at an example. For clarity and continuity, I am giving the same problem as what was discussed in the talk.

Q. Find the dimensions of the cylinder that has minimum mass and maximum surface area.

A.



Mass of the cylinder $= m = \rho\pi r^2 l$
Surface area of the cylinder $= S = 2\pi r l + 2\pi r^2$

We have two objectives $m(r, l)$ and $S(r, l)$

Note: We have to minimize m, hence we apply the transformation as given in the preceding heading.

$$\text{Min } [m] = \text{Max } [1/(1+m^2)]$$

$$\text{Let, } 1/(1+m^2) = m^*$$

Now our objective functions are m^* and S

We need to form a combined objective function containing the effects of both these objective functions. Let us call this combined objective function as O

Now,

$$O = w_1 m^* + w_2 S$$

Where, w_1 and w_2 are the weights that you wish to provide to each of the objectives. If suppose your main aim is to minimize the mass then $w_1 > w_2$. However, if you are unsure of the weights you have to provide, you can keep them same. But, the summation of the weights should be unity. $w_1 + w_2 = 1$, keeping the weights equal will give you $w_1 = 0.5$ and $w_2 = 0.5$.

The same procedure can be followed for three or four or more objectives.

14. Some References:

[1] Genetic algorithms in search, optimization, and machine learning; David E. Goldberg; 1989
[2] Multi-objective optimization using evolutionary algorithms; Kalyanmoy Deb; 2001
[3] Soft Computing; D. K. Pratihar; 2008

Note: This document is updated once every couple of months. You can download the latest version from www.tbnitin.tk/GeneticAlgorithm.pdf . For previous versions of this document, please contact the author on the email address given in www.tbnitin.tk