**Abstract**: In this paper, we try to find the performance parameters of the PID controller for better optimization. We use Integral of Time-Weighted Error as the Criteria for the controller. Our approach is to use a nested form of Genetic Algorithm and Ant Colony Optimization, which helps to deliver better outcomes than both of them individually.

**I Introduction**

A Proportional-Integral-Derivative controller basically consists of 5 basic coefficients for a Fractional Order Transfer Function. They are Kp, Ki, Kd, Lamda, Mu (Will be using symbols here.). Using the proposed algorithms, we would try to tune these coefficients efficiently, so that they can be used for industrial control.

There have been many algorithms proposed for the given purpose. Some of them are Genetic Algorithm, Ant Colony Optimization, Fuzzy Algorithm, Evolutionary Programming and Particle Swarm Optimization. Genetic Algorithm is an artificial version of Darwinian theory. It simulates the process of evolutionand revolves around three important notions. They are Reproduction, Crossover and Mutation. Genetic Algorithm gives a well optimized output for a wide span of problems. It is governed by the probabilistic conversion rules, making it less efficient for a given set of variables.

In the real world, ants use pheromone, which gets evaporated with time, to find the best route for themselves. The evaporation of pheromone is done because the ants keep on following the first trail with highest intensity of pheromone, thus diverging from the optimal solution.

The repetitive set of iterations are used to calculate the input parameters for ACO, till a desired output is reached. The output of the nth iteration is referred for the calculation of the input of the (n+1) th input parameters.

The following sections II and III gives an in depth knowledge of Genetic Algorithm and Ant Colony Optimization. Section 3 is about our proposed “Enhanced Genetic Algorithm”. Section IV states the Multi-Objective function used in our algorithm. Section V contains all our simulations in accordance to our algorithm

II Genetic Algorithm

“*Charles Darwin proposed the idea of natural selection, which states that only the fittest would flourish and the weaker population would eventually stagnate.*

Genetic Algorithms are meta-heuristic algorithms, based on the process of Natural Evolution and Natural Genetics. Developed by John Holland, the goal of his research has been:

* To abstract and rigorously explain the adaptive processes of natural systems.
* To design artificial systems software that returns the important mechanisms of natural systems. [1]

As the initial set of instructions are completely randomized, Genetic Algorithm adapts according to the given environment by reproducing with each iteration. The fittest offspring are kept for the next iteration, where they act as the parents. After a certain number of cycles, mutation takes place. Mutation is done because their might me some useful genetic resources that may be skipped. Termination step only happens after an optimal solution is achieved for the given objective function.

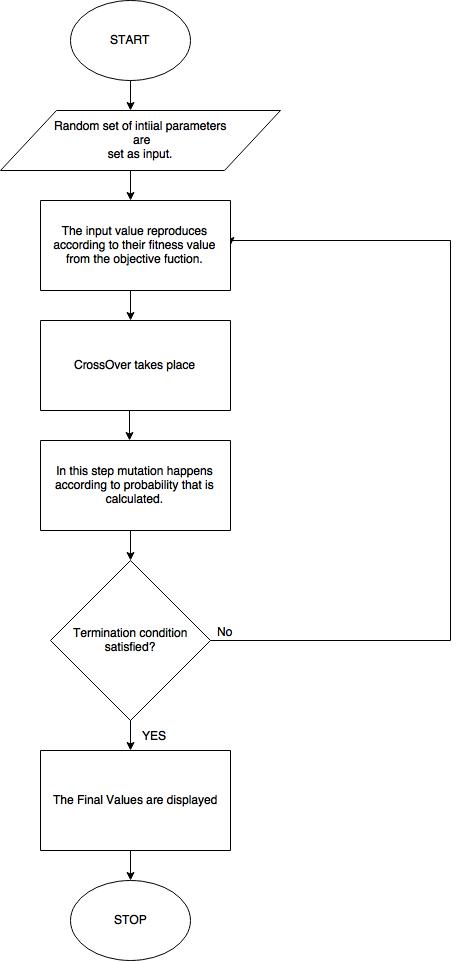


Fig.1 Flow Chart for fundamental Genetic Algorithm

Normal search techniques are more deterministic. Hence, for a small change in the fitness function, they have to be modified. This contributes to a more expensive redesign, as they are less flexible. Genetic algorithm has the property of self-restoration and self-guidance.

But Genetic Algorithm has an infinite sample space. Thus, when it is implemented for a finite region, the output of the system may not be accurate. Hence, further modifications are required for this purpose.

References

1. David E Goldberg