INTERNATIONAL JOURNAL OF ADVANCE RESEARCH AND DEVELOPMENT

(Volume2, Issue4)

Available online at: www.ijarnd.com

Correlation of GA and PSO for Analysis of Efficient Optimization

Mr. B. Chandrashaker Reddy [1], P. Venkat Prasad Reddy[2], M. Rajeshwari[3], Y. Sai Kavya[4]

[1] *Assistant Professor, Electronics and Communication Engineering, NNRG, Telangana, India*
[2] *Student, Electronics and Communication Engineering, NNRG, Telangana, India*
[3] *Student, Electronics and Communication Engineering, NNRG, Telangana, India*
[4] *Student, Electronics and Communication Engineering, NNRG, Telangana, India*

ABSTRACT

This paper aims to claim the correlation between PSO and GA to analyze best optimization technique. We have opted Particle Swarm Optimization (PSO) from Swarm based and Genetic Algorithm (GA) from Evolution based, it claims that PSO & GA produces the same effectiveness and moreover PSO is more computationally efficient than GA. Griewangks function is taken as input test function to compare PSO and GA to find out best optimized value. Evolution is the change in gene pool of population from generation to generation by processes such as mutation, selection and crossover. Swarm intelligence is based on nature-inspired behavior and is successfully applied to optimization problems in a variety of fields. In this system interaction between individuals and simple behavior between population and environment usually lead to detection of aggregate behavior, which is typical for whole colony. This could be observe by ants, bees, birds or bacteria in the nature which inspired to develop the algorithm called Swarm-based intelligence and are successfully applied for solving complicated optimization problem. PSO and GA are similar in the sense that these two heuristics are population based search methods. GA has been popular in academia and industry because of its intuitiveness and ability to effectively solve higher non-linear optimization problems. It is based on principles of Genetics and Natural Selection. The main limitation of GA is its effective computational cost. PSO works for flock of birds and is applied to so many areas such as function optimization, artificial neural network training, fuzzy system control and other areas where GA can be applied.

***Keywords*: *Particle Swarm Optimization, Genetic Algorithm, Fitness Value, Iteration, Best Optimization, Behavior, Griewangks Test Function*.**

---

1. INTRODUCTION

Optimization is about finding out the best solution from the appropriate solutions. Optimization refers to finding the input values such that we get the "best" output values. The definition of "best" varies from problem to problem.

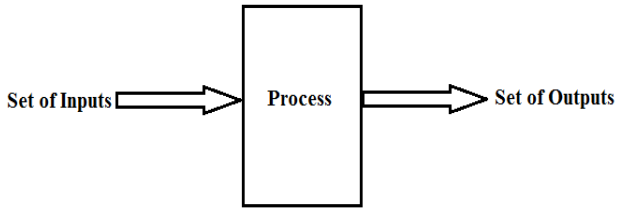


Figure 1. Optimization Block Diagram

Optimization is divided into two categories depending on whether variables are continuous or discrete. An optimization problem with discrete variables is known as combinatorial optimization problem. Combinatorial optimization is about finding an optimal object from finite set of objects. It operates on the domain of those optimization problems, in which the set of feasible solutions is discrete, in which the goal is to find the best solution.

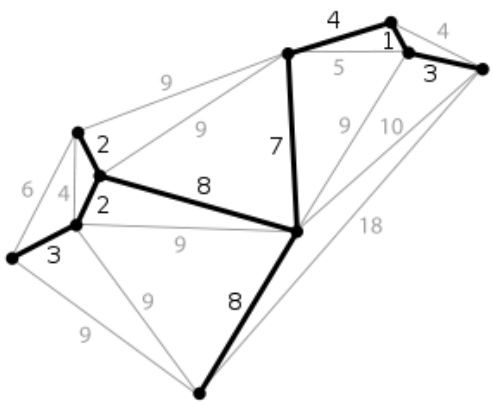
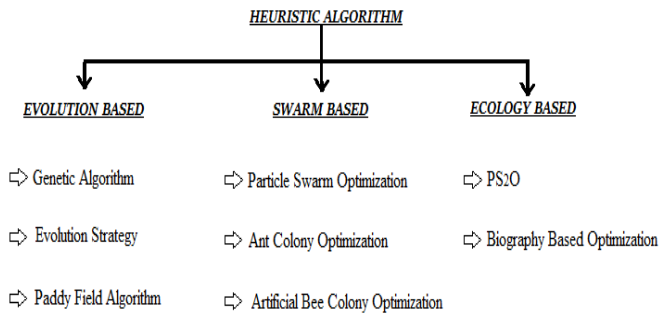


Figure 2. Random Optimization path

The optimization problems with continuous variables include constrained problems. It is the process of optimizing a specific objective function with respect to some variables in the presence of constraints on those variables. The objective function is an energy function which is to be minimized, or utility function, which is to be maximized.

1.1 Optimization Techniques

Classical optimization technique is the first optimization technique. It is used in finding optimal solutions of continuous and differential functions. It has limited scope in practical applications, and also it is not applicable for non-linear equations. To overcome the above limitations we go for heuristic methods. Heuristics means discoveries. These are classified into 3 types as given below.



Recently, genetic algorithms (GA) and particle swarm optimization (PSO) have attracted considerable attention among various heuristic optimization techniques. GA has been popular in academia and industry because of its intuitiveness and ability to effectively solve higher non-linear optimization problems. PSO is population-based global optimization technique which is inspired by the social behavior of bird flocking in search for food. It is inspired by the swarming or collaborative behavior of biological populations. Since these two approaches are supposed to find a solution to a given objective function but employ different strategies and computational effort, it is appropriate to estimate their performance.

2. METHODOLOGY

GA is a search based optimization technique based on principles of Genetics and Natural selection. Genetics is the study of heredity. Heredity is a biological process where a parent passes certain genes onto their children. Every child inherits genes from both of their biological parents and these genes in turn express specific distinguishing quality. Natural selection is a process whereby organisms better adapted to their environment tend to survive and produce more child or offspring.

PSO is population based optimization technique. It deals with group of random variables. All the variables are defined in some pre-defined pattern and the behavior of the particles is unknown. PSO works for flock of birds that are continuously moving in some direction & we decide some pattern for them, estimating the values & analyzing is formed accordingly. The input test function is taken as unconstrained global test function where PSO and GA applied on Griewangks test function to check efficiency.

3. GENETC ALGORITHM

Genetic Algorithm was developed by Prof. John Holland and his students at the University of Michigan during 1960 – 1970. Genetic Algorithm (GA) is a search-based optimization technique based on the principles of Genetics and Natural Selection. In GAs, we have a population of possible solutions to the given problem. These solutions then undergo recombination and mutation, producing new children, and the process is repeated over various generations.

Each individual is assigned a fitness value (based on its objective function value) and the fittest individuals are given a higher chance to mate and yield more "fitter" individuals. This is in line with the Darwinian Theory of "Survival of the Fittest". In this way we keep "evolving" better individuals or solutions over generations, till we reach a stopping criterion.

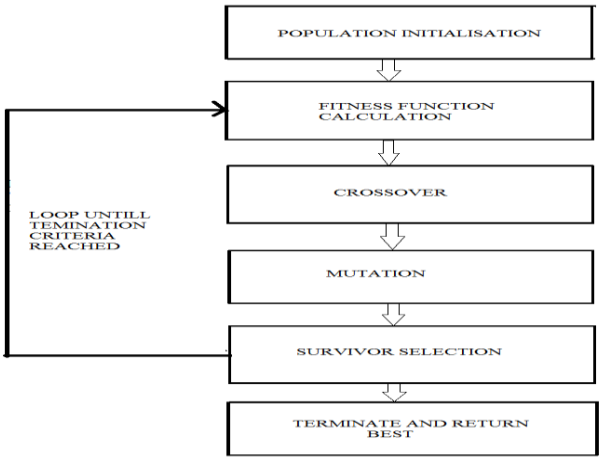
3.1. Working Principle

The GA is an iterative optimization procedure. Instead of working with a single solution in each iteration, a GA works with a number of solutions in each iteration. In the absence of any knowledge of the problem domain, a GA begins its search from a random population of solutions.

But now notice how a GA processes strings in an iteration. If a termination criterion is not satisfied, three different operators – reproduction, crossover and mutation – are applied to update the population of strings. One iteration of these three operators is known as a generation in the parlance of GAs. Since rendering of a solution in a GA is similar to a natural chromosome and GA operators are similar to genetic operators, the above procedure is called a genetic algorithm.

3.2. Flow Chart

We start with an initial population (which may be generated at random or seeded by other heuristics), select parents from the generated population for mating. Apply crossover and mutation operators on the parents to generate new child. And finally these child replace the existing individuals in the population and the process continues. In this way GAs actually try to imitate the human evolution to some extent.



3.3. Steps involved in GA procedure

1) Initialization

Create an initial population. The population size depends on the nature of the problem, this population is usually randomly generated and can be of any desired size, from only few individuals to thousands.

2) Evaluation

Each member of the population is then evaluated and we calculate the fitness for an individual. A fitness function produces next generation of states. The fitness value is calculated by how well it fits with our desired requirements. A good fitness function should return better states and give score to each state. The probability of being chosen for reproduction based on fitness score.

3) Selection

Selection methods rate the fitness of each solution and preferentially select the best solutions. Two pairs are selected at random to reproduce. They are selected based on fitness function score. Depending on fitness function score, one may be selected more than once whereas one may not be selected at all.

4) Crossover

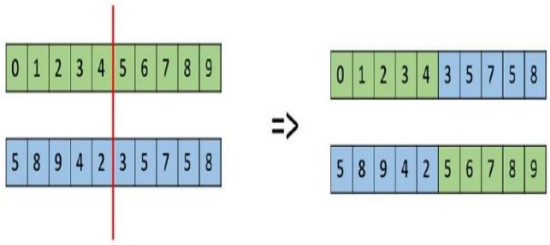
The crossover operator is analogous to reproduction and biological crossover. In this more than one biological parents are selected and one or more off-springs are produced using the genetic material of the parents. For each pair to be mated, a crossover point is chosen at random from within the bit string. Offspring are created by exchanges between parents and the Crossover point.

Crossover Operators

Some of the most popularly used crossover operators are

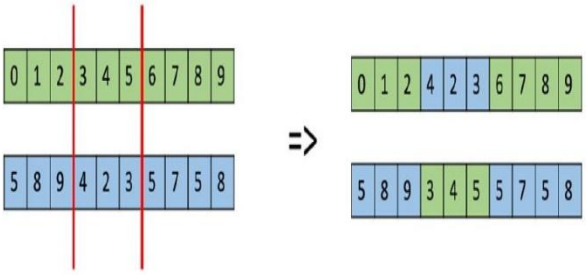
One Point Crossover

In this one-point crossover, a random crossover point is selected and the tails of its two parents are swapped to get new off-springs.



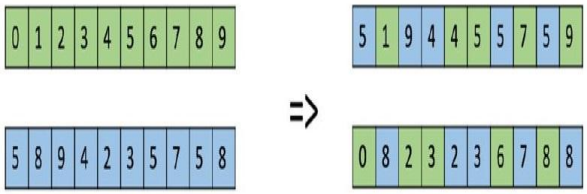
Multi Point Crossover

In Multi point crossover, the chromosome is divided into segments and the alternating segments are swapped to get new off-springs.



Uniform Crossover

In a uniform crossover, we don't divide the chromosome into segments, rather we treat each gene separately.

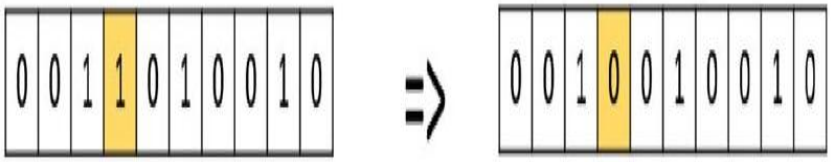


5) Mutation

Each location in bit string, can be subject to a mutation with a small random probability. In simple terms, mutation may be defined as a small random twist or pull in the chromosome, to get a new solution.

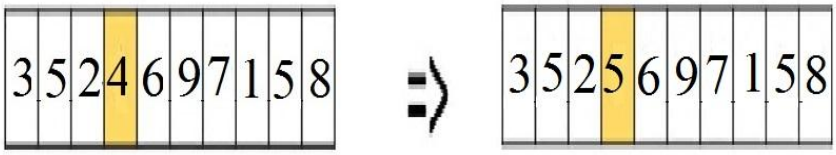
Bit Flip Mutation

In Bit flip mutation, from an entire chromosome we select one or more random bits and flip them. This is used for binary encoded GAs.



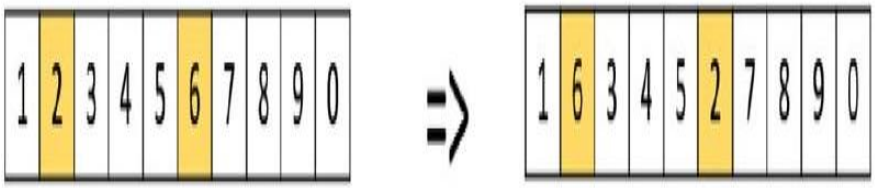
Random Resetting

Random Resetting is same as Bit flip mutation. Instead of binary codes we assign integers to a chromosome. From an entire chromosome, we select random integer and flip them.



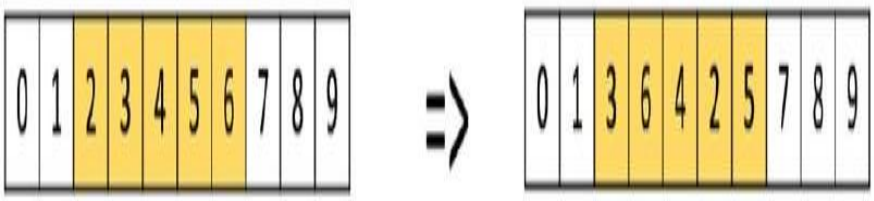
Swap Mutation

In swap mutation, we select two positions on the chromosome at random, and exchange the values. This is common in permutation based encodings.



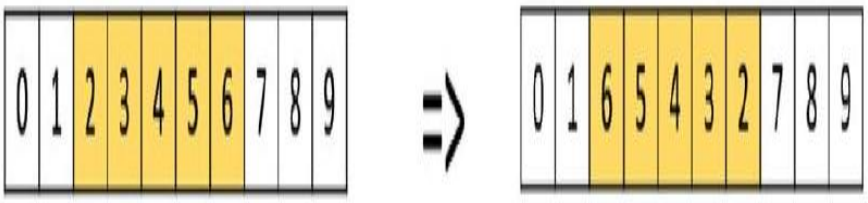
Scramble Mutation

Scramble mutation is also popular with permutation representations. In this, from the entire chromosome, a subset of genes is chosen and their values are scrambled randomly.



Inversion Mutation

In inversion mutation, we select a subset of genes like in scramble mutation, but instead of shuffling the subset, we merely invert the entire string in the subset.



3.4. Algorithm

1) Produce an initial population of individuals.
2) Evaluate the fitness of all individuals.
3) While
   Stopping criteria not met
   do
   Select fittest individuals for reproduction
   Recombine between individuals

Mutate individuals
Evaluate the fitness of the modified individuals
Generate a new population
End While

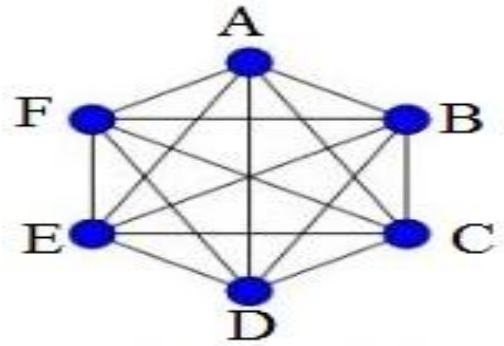
4. PARTICLE SWARM OPTIMIZATION

PSO was developed by Dr. Eberhart and Dr. Kennedy in 1995. PSO is population based stochastic optimization technique to find out optimized solution. PSO is inspired from the natural social behavior as well as dynamic movements with communication of insects and birds and fishes. PSO is more suitable for flock of birds. The birds have some techniques to share their experiences. They usually talk to each other and share information about their food, their inhabitants and some other information. By this some behavior is utilized for some analysis for some applications.

4.1. Swarms in search of food

The sequence which birds flow and this is used in various applications for optimization.

1) Imagine a flock of birds that are continuously moving in some direction in search of hidden source of food. Each bird in the search space is considered as a ''particle''.
2) The particle in the search space which is closest to the food source, chirps or sounds loudest so that the other birds can swing around in his direction. Here the pattern of communication used is fully connected topology. In a fully connected topology, each particle shares it information with all the other particles.



FULLY CONNECTED TOPOLOGY

3) If any of the circling birds comes closer to the target than the first. The one which is coming much nearer to the food, it chirps much louder as compared to the previous particle, So that the complete group of birds can notice that bird and now follow it to form a much optimized pattern.
4) This tightening pattern continues till any of the particles obtain that food source.

4.2. Process behind PSO

PSO is heuristic search method whose mechanics are inspired by swarming or collaborative behavior of biological populations. In PSO, each single solution is a bird in search space. We call it ''Particle''. These particles are randomly distributed in search space. Each particle is having its position and velocity in current population, which shows current solution available in search space. At each iteration, the particles update their particle position and velocity that are weighted through fitness function. Fitness function is a function which takes the solution as input and produces stability of the solution as output.

Fitness function should follow following characteristics:

1) A fitness function should be sufficiently fast to compute.
2) It must quantitatively measure how fit individuals can be produced from the given solution.

Two optimum values defines the fitness of objective function.

1) Present Best: It is the best solution of each particle that is achieved so far. It is represented as ''Pbest''.
2) Global Best: It is the best solution tracked by any particle among the whole population. It is represented as ''Gbest''

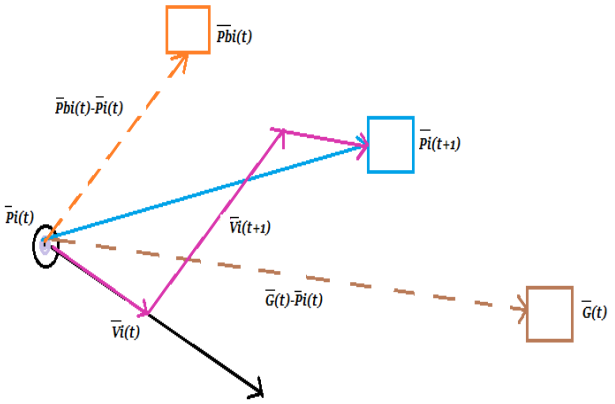


Figure 3. Optimized value for a particle

Vi(t+1) = w Vi(t) + a1.r1(Pbest,i(t)–Pi(t)) + a2.r2 (Gbest,i(t) - Pi(t))
Pi (t+1) = Pi (t) + Vi (t+1)
Where Pi (t) indicate the position of ith particle.
Vi(t) indicate the velocity of ith particle.
Pbest, i indicate the personal best position of ith particle.
Gbest, i indicate the global best position of ith particle.
a1 and a2 are position acceleration constant.
r1 and r2 are random values generated between [0, 1].
w is inertia weight used to produce balance between local and global search.
PSO provides smoother response and provides much optimization for more number of particles.

4.3. Example to find Global Minimum

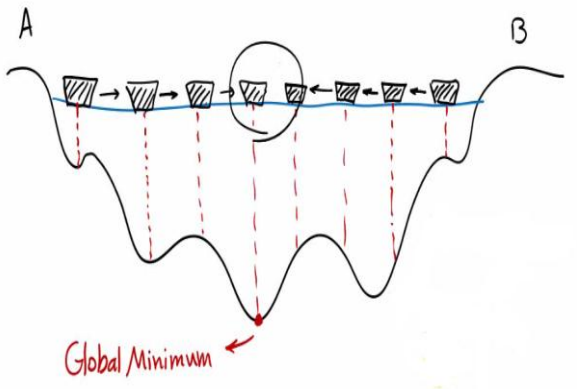


Figure 4.Example of PSO

Consider two persons A and B travelling on surface of a lake in boats X and Y to find out the deepest location. This is one of the optimization technique. They measure the depths at each and every point comparing each other's depths they find out the Global Minimum.

4.4. Algorithm

Steps in PSO algorithm can be briefed as below

1) Initialize the particles by assigning a random position in the problem space to each particle.
2) Evaluate the fitness function for each particle.
3) For each individual particle, compare the particle's fitness value with its Present best (i.e., pbest). If the current value is better than the pbest value, then set this value as the pbest and the current particle's position, Pi.
4) Identify the particle that has the best fitness value. The value of its fitness function is identified as Global best (i.e., gBest) and its position as G.
5) Update the velocities and positions of all the particles using (1) and (2).
6) Repeat steps 2–5 until a stopping criterion is met.

4.5. Flow chart

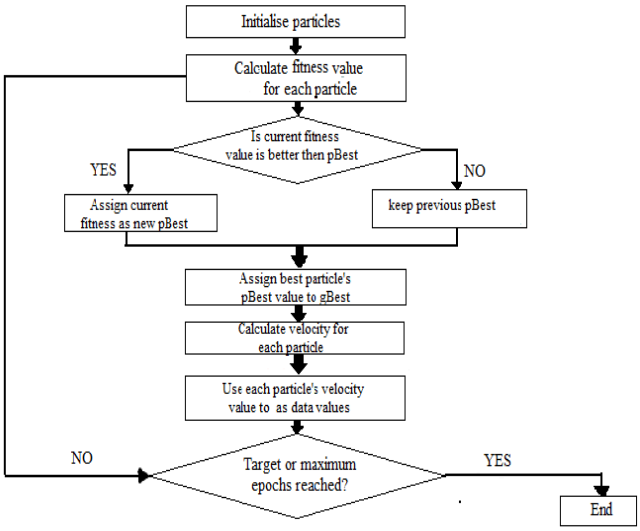


Figure 5. Flowchart of the PSO algorithm

4.6. Result

The input function is taken as Griewangks test function. We apply this test function as an input to GA and PSO and find the best optimization technique.

Griewangks Function:

It is given by equation

$$f(\mathbf{x}) = \sum_{i=1}^{d} \frac{x_i^2}{4000} - \prod_{i=1}^{d} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$$

It has many widespread local minima, which are regularly distributed. The complexity is shown in the zoomed in plots.

Input Domain: The function usually evaluated on the hyper cube. The input limits range from -600 to 600.

When a Griewangks function is given as input to Genetic Algorithm, the best value obtained is 1.66713 and the mean value obtained is 1.66729.

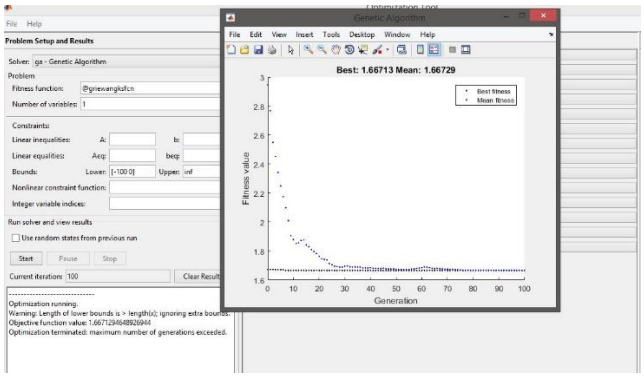


Figure 7. Output of GA

When a Griewangks function is given as input to Particle Swarm Optimization, the best value obtained is 1.73572e-12 and the mean value obtained is 0.128919.

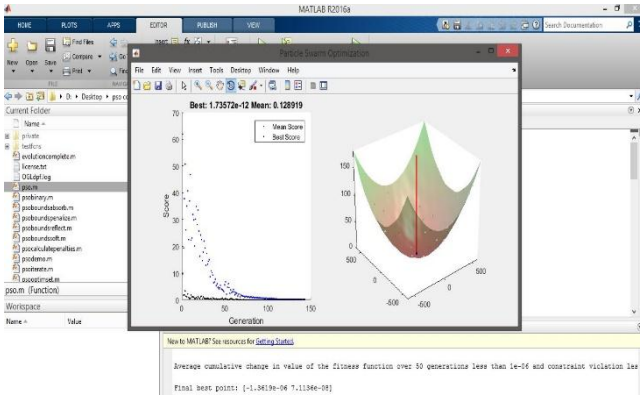


Figure 8. Output of PSO

Comparison of different unconstrained global Test function outputs using GA and PSO

S. No.	Test Function	GA	PSO
1	Griewangks function	1.66713	1.73572e-12
2	Rosenbrock function	1	2.11423e-08
3	Schwefel function	-719.527	-837.966

5. CONCLUSION

PSO is a relatively recent heuristic search method whose mechanics are inspired by swarming or collaborative behavior of biological populations. PSO is similar to the GA in the sense that they are both population-based search methods and they both depend on information sharing among their population members to enhance their search processes using a combination of deterministic and probabilistic rules.

Conversely, the GA is a well-established algorithm with many versions and applications. The objective of this research is to test the hypothesis that states that although PSO and the GA on average yield the same effectiveness, PSO is more computationally efficient than the GA. To investigate this, one statistical test is set to examine the two elements of this claim, equal effectiveness but superior efficiency for PSO over the GA.

6. REFERENCES

1. Kennedy, J. and Eberhart, R., “Particle Swarm Optimization,” Proceedings of the IEEE International Conference on Neural Networks, Perth, Australia 1995, pp. 1942-1945.
2. Kennedy, J. and Eberhart, R., Swarm Intelligence, Academic Press, 1st ed., San Diego, CA, 2001.
3. https://www.youtube.com/watch?v=sB1n9a9yxJk
4. https://www.youtube.com/watch?v=CqvtscWwJSw
5. Goldberg, D., Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley, Reading, MA, 1989, pp. 1-25.
6. Venter, G. and Sobieski, J., “Particle Swarm Optimization,” AIAA 2002-1235, 43rd AIAA/ASME/ASCE/ AHS/ASC Structures, Structural Dynamics, and Materials Conference, Denver, CO., April 2002.
7. C. Andrés, S. Lozanob. A particle swarm optimization algorithm for part–machine grouping. 2005. Robotics and Computer-Integrated Manufacturing 22 (2006) 468–474.
8. https://en.wikipedia.org/wiki/Genetic_algorithm, 2015.
9. https://en.wikipedia.org/wiki/Particle_swarm_optimization, 2015.
10. Mohammad Khajehzadeh, Mohd Raihan Taha, Ahmed El-Shafie and Mahdiyeh Eslami, 2011." A Survey on Meta-Heuristic Global Optimization Algorithms" Research Journal of Applied Sciences, Engineering and Technology 3(6): 569-578, 2011, ISSN: 2040-7467 © Maxwell Scientific Organization, 2011.
11. Clerc M., “Particle Swarm Optimization”. ISTE Ltd., London, UK, 2006.
12. Engelbrecht A. P., “Fundamentals of Computational Swarm Intelligence”. John Wiley & Sons, 2006.