Performance Evolution of Optimization Techniques for Mathematical Benchmark Functions

SUDHIR G.AKOJWAR[1], PRAVIN R. KSHIRSAGAR[2]
[1]Department of Electronics and Telecommunication Engineering.
[1]Gondwana University
[1]Government College of Engineering,
[1]Chandrapur (M.S)-442403,
[1]INDIA
[1]sudhirakojwar@gmail.com
[2]Department of Electronics Engineering.
[2]Gondwana University
[2]Rajiv Gandhi College of Engineering, Research and Technology, Chandrapur (M.S)-442403,
[2]INDIA
[2]pravinrk88@yahoo.com

Abstract: - This paper demonstrates several optimization techniques which comprise Genetic algorithm (GA), Ant Colony optimization (ACO) and Particle swarm optimization (PSO). The proposed paper enforces the concept of artificial intelligence to detect minima / maxima by applying set of mathematical benchmark functions. For Optimization technique, artificial intelligence used which comprise of several algorithms like Particle swarm optimization, genetic algorithm, ant colony optimization, neural network and fuzzy system. The proposed work will use particle swarm intelligence and genetic intelligence. This paper bestows comparison between PSO and GA according to performance. In random search algorithm, GA cannot detect the global optimization solution. The benchmark functions used under these algorithms are Rosenbrock, Griewank, Ackley and Sphere. They have multiple local minima / maxima and single global minima / maxima. Neural network has propensity to strikes at local minima / maxima. Result demonstrates that discomfort of neural network is thoroughly segregated by particle swarm intelligence and genetic algorithm.

Key-Words: - Particle swarm optimization, Ant colony optimization, Genetic Algorithm, Artificial Intelligence, Evolutionary Algorithms

1. Introduction

Optimization is integral part of day-to-day life. Optimization problems pullulating in many Biomedical Engineering where inference makes is arduous in complex situations. The search space may be so enormous that global optimum cannot found in a reasonable time. The extant linear or nonlinear methods may not be dexterous or computationally inexpensive for solving such problems. Therefore, optimization is effective tool in inference making where inference has to be taking to optimize one or more objectives in complex circumstances. Therefore, there is urging of efficient algorithm that effectively transact optimization problem. In the past decade several optimization algorithms has been developed which includes Genetic algorithm (GA), Ant colony optimization (ACO) and Particle swarm optimization (PSO).

2. Optimization techniques

Different optimization techniques are as follows

2.1 Particle swarm optimization (PSO)

The particle swarm optimization is Population search algorithm based on the simulation of the bird flocking, developed by Reynolds. PSO is a robust stochastic optimization technique based on the movement of intelligence of swarms. For solving the sophisticated problem, it enforces the concept of social interaction. It uses a number of particles, which called as agents that constitute a swarm moving around the space in the search for looking the best solution. PSO algorithm is inducing by social behaviour of bird horde and fish schooling.

In PSO, the particles are initially moving in at random directions with different velocities, in the search-space. The direction of a particle is gradually modify according to the prior best positions of itself and its best neighbour, detecting in their target and hopefully retracing even better positions. The inertia weight controls the amount of reiteration in the particle's velocity so that no two particles moving in the search space are at the same position at any instant of time. Each particle in PSO algorithm preserve the best fitness value achieved among all particles in the swarm, called the global best fitness , and the Velocity of particle is given by

v[ ] = w*v[ ] + c_1 * rand() * (pbest[ ] - present[ ]) + c_2 * rand () * (gbest[ ] - present[ ]) (1)

Where

Present [ ] = present [ ] + v [ ] (2)

Here, v [ ] is particle velocity, present [ ] is current particle (solution). pbest [ ] is best solution (fitness) of particle from past. gbest [ ] is best solution obtained by any other particle. $c_1$and c_2 are learning factors, w is the inertia factor and rand () is a random number between (0, 1).

2.2 Ant colony optimization (ACO)

Ant Colony Optimization is a experimental approach induced by the real ant system proposed by Marco Dorigo in 1992. This algorithm comprises of mainly four main components, which are ant, pheromone, daemon action, and decentralized control that contribute to the overall system. Ants are imaginary agents that used in order to mimic the exploration in search space. In real life pheromone is a chemical material dispersed by ants over the pathway they travel and its severity changes over period due to evaporation. In ACO the ants spill out pheromones when travelling in the search space over the pathway and the quantities of these pheromones demonstrate the intensity of the trail. The ants prefer the direction based on pathway marked by the high intensity of the trail. The intensity of the trail can presumed as a global memory of the system. Daemon action is use to gather global information, which cannot be done by a single ant and uses the information to determine whether it is necessary to add extra pheromone in order to help the convergence. The decentralized control is use in order to make the algorithm robust and malleable within a dynamic environment. As ACO, system encounter the problem of ant lost or ant failure so in order to overcome this inconvenience decentralized system used. These basic components contribute to a cooperative interaction that leads to the emergence of shortest paths depict the initial phase, mid-range status of any system, and the outcomes of the ACO algorithm respectively.

The pheromone τ_{ij} given by,

$$\tau_{ij} = (1-\rho).\tau_{ij} + \sum_{k=1}^{m} \tau_{ij}^{k} \quad (3)$$

Where ρ is the evaporation rate, m is the number of ants, and τ_{ij}^{k} is the quantity of pheromone laid by ant k or the pheromone is updated as

$$\tau_{ij} = \left[(1-\rho).\tau_{ij} + \sum_{k=1}^{m} \tau_{ij}^{best}\right]_{\tau min}^{\tau max} \quad (4)$$

Where, τ_{max} and τ_{min} are respectively the upper and lower bounds imposed on the pheromone

$$\tau_{ij}^{k} = \begin{cases} \dfrac{Q}{L_k} ifantkused \\ 0 Otherwise \end{cases} \quad (5)$$

Where Q is constant and L_k is the length of the tour constructed by ant k.

2.3 Genetic Algorithm (GA)

The Genetic Algorithm introduced by John Holland in 1975 based on the concept of the natural search algorithm. It applies principle of Darwinian evolution theory i.e. survival of fitness. According to theory, only the fittest in each linage will sustain, reproduce, and proliferate, and consecutive lineage will become better and better compared to prior lineage. So genetic algorithms is, it simulates the process of evolution. If evolution is an optimizing, process, and consecutive linage are becoming better and better, each linage is like iteration. So, with each iteration, there is a progressive improvement of the objective function. Each of these feasible solutions is encode as a chromosome, also referred to as genotype, and each of these chromosomes will get a measure of fitness through fitness function (evaluation or objective function). The value of fitness function of a chromosome determines its competency to withstand and procreate descendant. The high fitness value implies the better solution for maximization and the low fitness value implies the superior solution for minimization problems. A basic GA has five main components: a random number generator, a fitness evaluation unit, a reproduction process, a crossover process, and a mutation operation. Reproduction prefer the fittest candidates of the populace, while crossover is the mechanism of modulating the fittest chromosomes and entrusting preferable genes to the coming linage, and mutation alters some of the genes in a chromosome

3. The Benchmark Mathematical Functions

The various benchmark mathematical functions as follows

3.1 Rosenbrock function

It is mathematical benchmark function introduced by Howard H. Rosenbrock in 1960, also known as Rosenbrock's vally or Rosenbrock's banana function. It pertains to the non-convex function and it used for testing optimization algorithms.

The global minima is kept within a long narrow parabolic shaped flat valley and to detect the global minima, that is for any optimizing algorithm to converge at this point, the job is not a simple.

It defined by

$$f(x) = (1-x)^2 + 200(y - x^2)^2 \qquad (6)$$

It has a global minimum at (x, y) = (1, 1) where f (x, y) = 0. A different coefficient of the second term is sometimes given, but this does not affect the position of the global minimum.

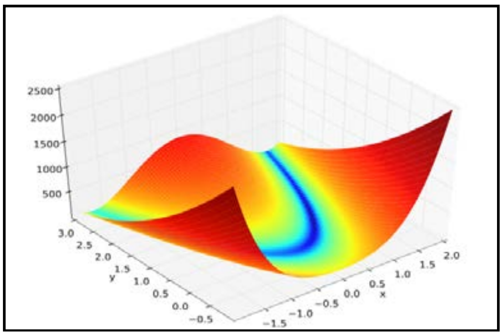

Fig. 1 - Plot of the mathematical benchmark Rosenbrock function of two variables

3.2 Rastrigin function

Rastrigin as a two dimensional optimization problem first introduced this mathematical function. It is a non-convex function and is the example of non-linear multimodal function. It formed for testing the optimization algorithms. Afterwards the Rastrigin function made generalized that is multidimensional by Muhlenbeinet. It has a large search space and large numbers of local minima's.

It defined by:

$$f(x) = An + \sum_{i=1}^{n} [Xi^2 - A\cos(2\pi Xi)] \qquad (7)$$

Here, A=10 and $x_i \in$ [-5.12, 5.12] It has a global minimum at x=0 where f(x) =0

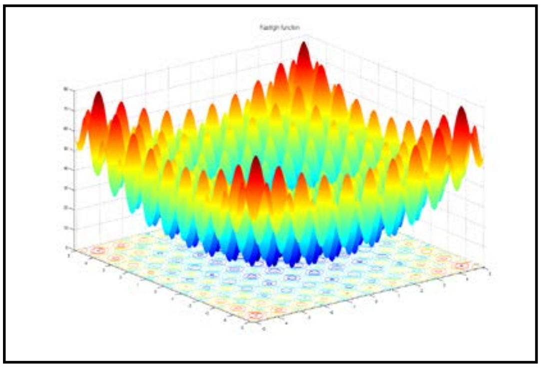

Fig. 2 - Plot of the Rastrigin equation of two variables

3.3 Ackley Function

Ackley function proposed by Ackley and generalized by Back has numerous local minima's and algorithms as gradient steepest descent fails to swim across the valleys among the optima. The exponential term comprised in the function shroud its surface assist to comprise numerous local minima's. The complexity of the Ackley function is somewhat moderated. It belongs to the minimization problem. Originally, this function was design for two variables; afterwards it was generalize for N variables or dimensions.

Formally, this problem can described as detecting a string that minimizes the following equation:

$$f(x) = -30 * \exp\left[-0.2\sqrt{\left(\frac{1}{n} * \sum_{i=1}^{n} x_i^2\right)}\right.$$
$$\left. - \exp\left[\sum_{i=1}^{n} \cos(2\pi * x_i)\right]\right] + 30 + e \qquad (7)$$

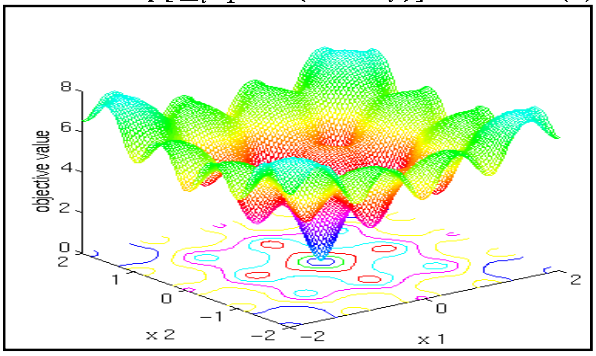

Fig. 3 - Plot of the mathematical benchmark Ackley function of two variables

3.4 Griewank function

It used for the optimization purpose, Grievance function comprises a product term that introduces interdependence among the variables in the function. The main aim of the function is failure of the techniques that optimize each variable selfsufficient. The optimum points of Griewank function are perpetually distributed. The function defined as follows:

$$f_1 = \frac{1}{4000}\sum_{i=1}^{n}(x_i)^2 - \prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \qquad (8)$$

$x \in [-400,400]^n$, $\min(f_1(x^*)) = f_1(0) = 0$

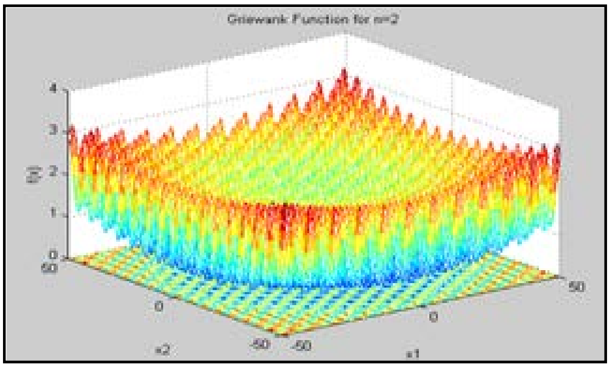

Fig. 4 - Plot of the mathematical benchmark Griewank function of two variables

3.5 Sphere function

Sphere function also commonly known as De Jong's function f_{Sph}(x) is a simple and strong convex function, which has been widely used in the improvement of the appraisal of genetic algorithms and the principle of evolutionary strategies as part of the test set.

The function defined as follows:

$$f_{Sph}(x) = \sum_{i=1}^{p} x_i^2 \qquad (9)$$

$x_i \in [-6.12, 6.12]$

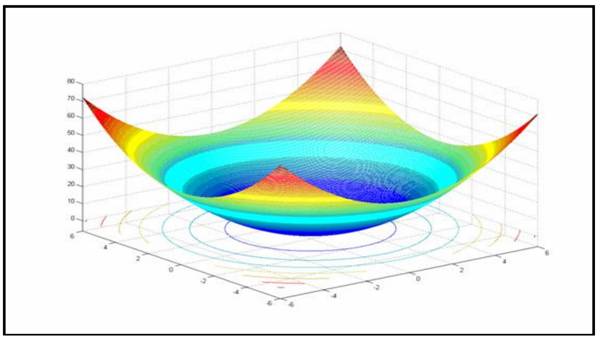

Fig. 5 - Plot of the mathematical benchmark for sphere function

4. Parameters Selected

Following parameters are preferred for results

- Index performer parameter => Mean Square Error
- Maximum iterations Allowed (Epoch) =>1600
- Initial Random Population=>Random Real for PSO/Random Integer for GA/ Random Real for ACO
- Initial Population in Numbers = 30 for PSO/20 for GA/300 for ACO
- Velocity Limit for PSO => V_{min} = -30 and V_{max} = 30/ Min. limit = -20 & Max. Limit = 20
- Number of bits/chromosome =>10
- Crossover – single point and double point
- Mutation – single bit random
- Particle Position Limit for PSO => X_{min} = -20 and X_{max} = 20
- Target =>Minima
- Execution Repetition =>About 25 times
- $\rho = 0.8$, $\rho_o = 0.2$ for ACO
- c_1 & c_2 = 2 for PSO
- Search space limit for ACO => [-20 20]

5. Results

The table demonstrate the performance of PSO/GA and ACO for the benchmark functions prefer with multiple local minima and maxima. The entire benchmark functions exclude Rosenbrock had global minima at [1, 1] while others had global minima at [0, 0]. Prior swarm initialized was random for PSO, Ant, and random integer for GA. Every algorithm was independently execute with maximum 1600 iterations and repeatedly considering squared error and the number of iterations was note down corresponding to the nearest/accurate roots. The squared error was further detracting until the algorithm was incapable to converge and the roots were note down.

Following tables 1, 2 and 3 lists the performance of PSO, GA with single point crossover and double point crossover and ACO in terms of minimum values obtained by swarm intelligence, least squared error value for mathematical benchmark function, and the iteration required to find minimum.

Sr. No.	Function Name	PSO	GA – Single point Cross sover	GA – Two point Crosso ver	ACO
		Minima	Minima	Minima	Minima
1	GRIEWANK	[-0.2969 -0.8446] 20^{-3}	[0 0]	[0 0]	[-0.5165 0.0178] 20^{-5}
2	SPHERE	[0.1590 -0.0467] 20^{-3}	[0 0]	[0 0]	[0.885 0.4283] 20^{-5}
3	ACKLEY	[-0.2294 -0.7133] 20^{-3}	[0 0]	[0 0]	[-0.0857 -0.1707] 20^{-4}
4	RASTRIGIN	[0.77 -0.7863] 20^{-4}	[0 0]	[0 0]	[-0.555 -0.1541] 20^{-4}
5	ROSENBROCK	[0.9996 0.9991]	[1 1]	[1 1]	[0.9685 0.9378]

Table 1- The values obtained by PSO, ACO & GA

Sr. No.	Function Name	PSO	GA – Single point Crossover	GA – Two point Crossover	ACO
		Mean Squared Error(MSE)			
1	GRIEWANK	20^{-13}	0	0	20^{-30}
2	SPHERE	20^{-13}	0	0	20^{-19}
3	ACKLEY	20^{-5}	0	0	20^{-8}
4	RASTRIGIN	20^{-11}	0	0	20^{-14}
5	ROSENBROCK	20^{-11}	0	0	20^{-6}

Table 2 - The Mean squared error for benchmark functions

Sr. No.	Function name	Minima	PSO	GA – Single point Cross Over	GA – Two point Cross over	ACO
			Iterations	Iterations	Iterations	Iterations
1	GRIEWANK	[0 0]	918	237	32	1467
2	SPHERE	[0 0]	1296	78	87	1521
3	ACKLEY	[0 0]	867	97	31	242
4	RASTRIGIN	[0 0]	1456	47	127	1517
5	ROSENBROCK	[1 1]	956	41	697	778

Table 3 – Iteration required to find Minima

6. Conclusions

The performance results demonstrate that the swarm intelligence concept when used for detecting minima and maxima of mathematical benchmark functions having multiple minima and maxima have no propensity to get stuck at local minima and maxima and they converge at the global minima and maxima with least error

The mean squared error form above table has a minimum value of the order 20^{-5}. The back propagation Neural Network has propensity to be stuck at local minima and maxima is overcome by incorporating the PSO. When feed forward artificial neural network was rehearsed using back propagation rehearsed for some points with respect to the above benchmark functions using brute search force method the artificial neural network demonstrate early convergence to sphere function only while for other functions after many training it stuck at local minima for even MSE=20^{-2}. The solution to the back propagation is to use these swarm concepts for better and early convergence. A preferable outlook to feed forward artificial neural network with back propagation is to use artificial neural network with artificial Intelligence. The results demonstrate that PSO converge early to all mathematical benchmark functions for detecting the minima and maxima.

References:

[1] Russell Eberhart and James Kennedy, A New Optimizer Using Particle Swarm Theory, *6th international symposium on Micro Machine and Human Science, IEEE*, 1995.

[2] Kennedy, I., Eberhart, R., Particle Swarm Optimization, Proc. IEEE International Conference on Neural Networks (Perth, Australia), IEEE Service Center, Piscataway, NJ, Iv: 1942- 1948, I995.

[3] Shi, Y., Eberhart, R., Parameter Selection in Particle Swarm Optimization, *Proceedings of the 7th Annual Conference on Evolutionary Programming*, pp. 591-601, 1998.

[4] R.C. Eberhart and Y. Shi. Comparing inertia weights and constriction factors in particle swarm optimization. *Proceedings of the 2000 Congress on Evolutionary Computation*, 1:84 - 88, 2000.

[5] M. Clerc and J. Kennedy. The particle swarm stability, and convergence in a multidimensional complex space. *IEEE Transactions on Evolutionary Computation*, 6:58 - 73, 2002.

[6] S. Ujjin and P. J. Bentley, Particle Swarm Optimization Recommender System, IEEE *Swarm Intelligence Symposium*, 2003, 24-26 April 2003, pp. 124-131.

[7] Valdez, F. and Melin P. Parallel Evolutionary Computing using a cluster for Mathematical Function Optimization, *Nafips. San Diego CA, USA*, 598-602. June 2007.

[8] K. F. Man, K. S. Tang, and S. Kwong, Genetic algorithms: Concepts and applications, *IEEE transactions on industrial electronics*, vol. 43, no. 5, October 1996.

[9] Dongshu yan, jintao zhang, bo Yuma, Genetic algorithm for finding minimal multi-homogeneous bezout number, *7th IEEE/ACIS international conference on computer and information science.*

[10] H. Wright, Genetic algorithms for real parameter optimization, Foundations of Genetic Algorithms, *J. E. Rawlins, Ed. San Mateo, CA: Morgan Kaufmdnn,* 1991, pp. 205-218.

[11] John A. Miller, Walter D. Potter, Ravi V. Gandham and Chinto N. Lapena, An evaluation of local improvement operators for genetic algorithms, IEEE transaction on systems, man and cybernetics, vol. 23. No. 5 Sept. /Oct. 1993.

[12] Zhiyong li, Wei Zhou, Bo Xu, Kenli li, An ant colony genetic algorithm based on pheromone diffusion, *4th international conference on natural computation,* 2008 IEEE.

[13] Thomas Stutzle and Holger H. Hoos, Min-Max Ant System, *Future generation computer systems, Elsevier,* 2000.

[14] Van den Bergh, Particle Swarm Weight Initialization in Multi-Layer Perceptron Artificial Neural Networks, *Accepted for ICAI. Durban, South Africa,* 1999, pp. 41-45.

[15] R. Mendes, P. Cone, M. Rocha, Particle swarm for feed forward neural network training, *Proc. International Joint Conference on Neural Networks,* pp. 1895-1899.2002.

[16] Goldberg D. E. Genetic algorithms in search optimization, and machine learning. *Addison-Wesley,* 1989.

[17] Handbook of Genetic Algorithms. New York: Van Nostrand Reinhold, 1991.

[18] *Ant colony optimization,* Marco Dorigo and Thomas Stutzle, A Bradford Book, The MIT Press, Cambridge, Massachusetts, London, England.

[19] *Ant colony optimization, Artificial ants as a computational Intelligence techniques,* Marco Dorigo, Mauro Birattari, and Thomas Stutzle, Universite Libre de Bruxelles, Belgium.

[20] Li Ting Wu Li, An Enhanced Parallel Back propagation Learning Algorithm for Multilayer Perceptron, *Proceedings of the 7th World Congress on Intelligent Control and Automation*, June 25 - 27, 2008, Chongqing, China.

[21] En Hui Zheng, Min Yang, Tuning of Neural networks based on Genetic Algorithm and statistical learning theory, *3rd international conference on machine learning and cybernetics, Shanghai*, 26-29 August, 2004.

[22] Nikolay Y. Nikolaev and Hitoshi Iba, Learning polynomial feed forward neural networks by genetic programming and back propagation.

[23] Bin Gao, Jing-Hua Zhu and Wen-chang Lang A Novel Hybrid Optimization Algorithm Based on GA and ACO for Solving Complex Problem, *International Journal of Multimedia and Ubiquitous Engineering,* Vol.10, No.8 (2015), pp.243-252

[24] Radha Thangaraj, Millie Pant, Ajith Abraham, Pascal Bouvry, Particle swarm optimization: *Hybridization perspectives and experimental illustrations, Elsevier*, 2010, pp 1-19

[25] Rania Hassan, Babak Cohanim, Olivier de Weck, A Comparison of particle swarm optimization and the genetic algorithm, *American Institute of Aeronautics and Astronautics,* pp 1 – 13

[26] V. Saishanmuga Raja, S.P. Rajagopalan A Comparative analysis of optimization techniques for artificial neural network in biomedical applications, *Journal of Computer Science* 10 (1): 106-114, 2014 ISSN: 1549-3636, 2014 Science Publications, pp. 106 -119

[27] Mohd Nadhir Ab Wahab, Samia Nefti-Meziani, Adham Atyab, A Comprehensive Review of Swarm Optimization Algorithms, *PLOS ONE DOI:10.1371/journal.pone.*0122827 May 18, 2015, pp. 1-36,

[28] V.Selvi , Dr.R.Umarani, Comparative Analysis of Ant Colony and Particle Swarm Optimization Techniques, *International Journal of Computer Applications (0975 – 8887),* Volume 5– No.4, August 2010, pp. 1-6

[29] Ginnu George, Kumudha Raimond, A Survey on Optimization Algorithms for Optimizing the Numerical Functions, *International Journal of Computer Applications* (0975 – 8887) Volume 61– No.6, January 2013, pp. 41-46

[30] Riccardo Poli, An Analysis of Publications on Particle Swarm Optimisation Applications, *University of Essex, UK, Technical Report CSM*-469, ISSN: 1744-8050, May 2007, pp. 1-57