IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⋇ Impact Factor 7.39 ⋇ Vol. 11, Issue 7, July 2022

DOI: 10.17148/IJARCCE.2022.11796

"Design and Development of High-Performance Algorithm to find brain tumor"

Sana Sheikh[1], Hirendra R. Hajare[2]

M.Tech Scholar, BIT , Ballarpur[1]

Assistant Professor, BIT , Ballarpur[2]

Abstract: This paper shows the performance of artificial intelligence mainly associated with the swarm intelligence concept to find minima and maxima of a set of benchmark functions. The implementation is basically carried to optimize Neural Networks using swarm intelligence to overcome the effect of training algorithms (Back propagation) which get stuck at local minima or local maxima in many applications, using swarm intelligence. This is a part of research work which is carried to show the performance of above algorithms in finding the maxima/minima of the benchmark functions.. The results shows that these swarm concepts have no tendency to get stuck at local minima or local maxima and proper parameter selection to these algorithms produces excellent results.

Keywords: Ant colony optimization, Artificial. Intelligence, Evolutionary algorithms, Genetic Algorithm, Particle swarm optimization.

1. INTRODUCTION

The use of genetic algorithms (GA) for problem solving is not new. The pioneering work of J. H. Holland in the 1970's proved to be a significant contribution for scientific and engineering applications. Since then, the output of research work in this field has grown exponentially although the contributions have been, and are largely initiated, from academic institutions world-wide. Those problems once considered being "hard" or even "impossible," in the past are no longer a problem as far as computation is concerned. Therefore, complex and conflicting problems that require simultaneous solutions, which in the past were considered deadlocked problems, can now be obtained with GA. Furthermore, the GA is not considered a mathematically guided algorithm. The optimum obtained is evolved from generation to generation without stringent mathematical formulation such as the traditional gradient-type of optimizing procedure. In fact, GA is much different in that context. It is merely a stochastic, discrete event and a nonlinear process. The obtained optimum is an end product containing the best elements of previous generations where the attributes of a stronger individual tend to be carried forward into the following generation. The rule of the game is "survival of the fittest will win" [11].

Ant colony algorithm (Colorni et al., 1991; Dorigo et al., 1996) is a kind of meta-heuristic algorithms and has been successfully applied to solve many combinatorial optimization problems, such as travelling salesman problem (Gambardella & Dorigo, 1995; Dorigo et al., 1996; Dorigo & Gambardella, 1997; Stützle & Hoos, 1997), sequential ordering problem (Gambardella & Dorigo, 2000), generalized assignment problem (Lourenço & Serra, 2002), scheduling problem (Stützle, 1998; Merkle et al., 2002; Merkle & Middendorf, 2003), network routing problem (Schoonderwoerd et al., 1996; Di Caro & Dorigo, 1998;), set covering problem (Hadji et al., 2000; Rahoual et al., 2002; Lessing et al., 2004), etc. Great achievements of ant colony algorithm have attracted lots of attentions from different disciplinary researchers, and its application fields have been expanded from combinatorial optimization to continuous optimization problems, single-objective problems to multi-objective problems, static problems to dynamic problems, etc [12].

2. GENETIC ALGORITHM: THE CONCEPT AND FLOWCHART

GA is inspired by the mechanism of natural selection [16], a biological process in which stronger individuals are likely be the winners in a competing environment, Here, GA uses a direct analogy of such natural evolution. It presumes that the potential solution of a problem is an individual and can be represented by a set of parameters. These parameters are regarded as the genes of a chromosome and can be structured by a string of values in binary form. A positive value, generally known as fitness value, is used to reflect the degree of "goodness" of the chromosome for solving the problem, and this value is closely related to its objective value.
Throughout a genetic evolution, a fitter chromosome has the tendency to yield good-quality offspring, which means a better solution to the problem. In a practical application of GA, a population pool of chromosomes has to be installed

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified ⁜ Impact Factor 7.39 ⁜ Vol. 11, Issue 7, July 2022
DOI: 10.17148/IJARCCE.2022.11796

and they can be randomly set initially. The size of this population varies from one problem to the other. In each cycle of genetic operation, termed an evolving process, a subsequent generation is created from the chromosomes in the current population. This can only be successful if a group of those chromosomes, generally called "parents" or a collection term "mating pool," are selected via a specific selection routine. The genes of the parents are to be mixed and recombined for the production of offspring in the next generation. It is expected that from this process of evolution (manipulation of genes), the "better" chromosome will create a larger number of offspring, and thus has a higher chance of surviving in the subsequent generation, emulating the survival-of-the-fittest mechanism in nature. A scheme called roulette wheel selection [17] is one of the most commonly used techniques in such a proportionate selection mechanism. The cycle of evolution is repeated until a desired termination criterion is reached. This criterion can also be set by the number of evolution cycles (computational runs), the amount of variation of individuals between different generations, or a predefined value of fitness. The flowchart for GA is shown in figure 1.

Flowchart-

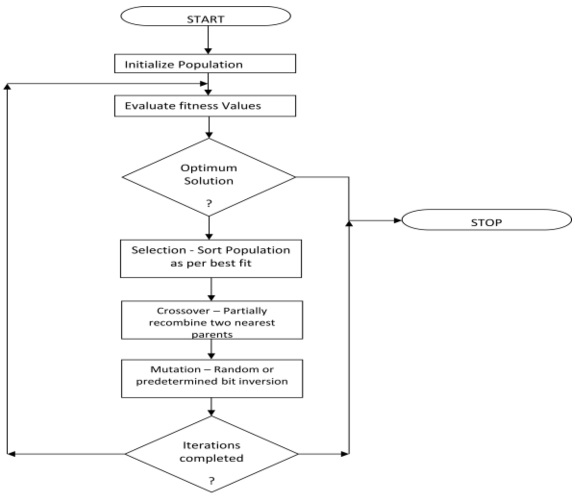


Figure 1 – Flowchart for genetic algorithm

3. ANT COLONY OPTIMIZATION: THE CONCEPT, ALGORITHM AND FLOWCHART

ACO algorithms make use of simple agents called ants which iteratively construct candidate solutions to a combinatorial optimization problem [13]. The ants' solution construction is guided by (artificial) pheromone trails and problem-dependent heuristic information. In principle, ACO algorithms can be applied to any combinatorial optimization problem by defining solution components which the ants use to iteratively construct candidate solutions and on which they may deposit pheromone. An individual ant constructs candidate solutions by starting with an empty solution and then iteratively adding solution components until a complete candidate solution is generated. We will call each point at which an ant has to decide which solution component to add to its current partial solution a choice point. After the solution construction is completed, the ants give feedback on the solutions they have constructed by depositing pheromone on solution components which they have used in their solution [18]. Typically, solution components which are part of better solutions or are used by many ants will receive a higher amount of pheromone, and hence, will more likely be used by the ants in future iterations of the algorithm. To avoid the search getting stuck, typically before the pheromone trails get reinforced, all pheromone trails are decreased by a factor ρ. The flowchart for ACO is shown in figure 2.
The pheromone τ_{ij} is given by,

$$\tau_{ij} = (1 - \rho).\tau_{ij} + \sum_{k=1}^{m} \Delta\,\tau_{ij}^{k} \qquad (1)$$

Where ρ is the evaporation rate, m is the number of ants, and
$\Delta\,\tau_{ij}^{k}$ is the quantity of pheromone laid by ant k Or the pheromone is updated as

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⸎ Impact Factor 7.39 ⸎ Vol. 11, Issue 7, July 2022

DOI: 10.17148/IJARCCE.2022.11796

$$\tau_{ij} = \left[(1-\rho).\tau_{ij} + \sum_{k=1}^{m} \Delta\,\tau_{ij}^{best}\right]_{\tau min}^{\tau max} \quad (2)$$

Where, τ_{max} and τ_{min} are respectively the

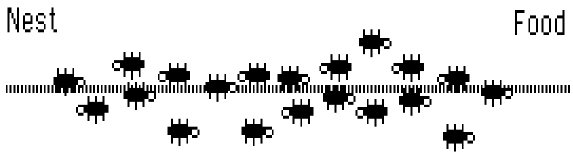
upper and lower bounds imposed on the pheromone

$$\Delta\,\tau_{ij}^{k} = \begin{cases} \frac{Q}{L_k} & \text{if ant k used} \\ 0 & \text{Otherwise} \end{cases} \quad (3)$$

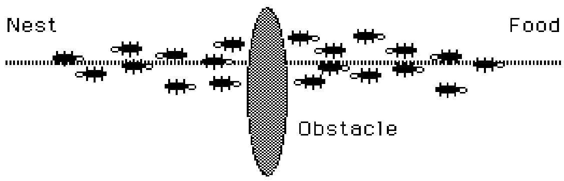
Where Q is a constant and L k is the length of the tour constructed by ant k.

Natural Behavior of Ant

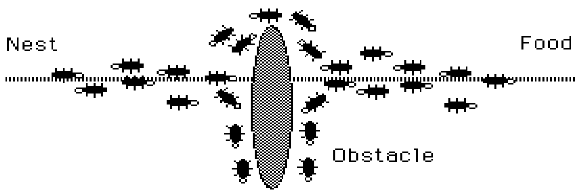
1) Real Ant follows a path between nest and food source.



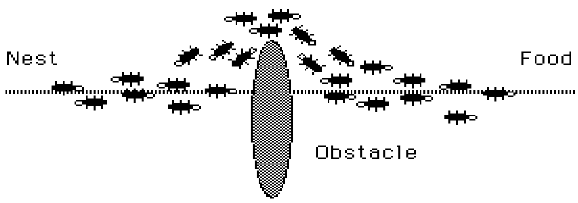
2) An obstacle appears on the path Ant chooses to turn left and right with equal probabilities.



3) Pheromone is deposited more quickly on the shorter path.



4) All ants have chosen the shorter path



Particle Swarm Intelligence

Particle swarm optimization (PSO) is a population based stochastic optimization technique developed by Dr. Eberhart and Dr. Kennedy in 1995, inspired by social behavior of bird flocking or fish schooling.PSO shares many similarities with evolutionary computation techniques such as Genetic Algorithms (GA). The system is initialized with a population of random solutions and searches for optima by updating generations. However, unlike GA, PSO has no evolution operators such as crossover and mutation [8].

In PSO, the potential solutions, called particles, fly through the problem space by following the current optimum particles. Each particle keeps track of its coordinates in the problem space which are associated with the best solution (fitness) it has achieved so far. (The fitness value is also stored.) This value is called pbest. Another "best" value that is

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified ⋇ Impact Factor 7.39 ⋇ Vol. 11, Issue 7, July 2022
DOI: 10.17148/IJARCCE.2022.11796

tracked by the particle swarm optimizer is the best value, obtained so far by any particle in the neighbors of the particle. This location is called lbest. When a particle takes all the population as its topological neighbors, the best value is a global best and is called gbest. Acceleration is weighted by a random term, with separate random numbers being generated for acceleration toward pbest and lbest locations.PSO is a metaheuristic as it makes few or no assumptions about the problem being optimized and can search very large spaces of Candidate solutions. However, metaheuristics such as PSO do not guarantee an optimal solution is ever found. More specifically, PSO does not use the gradient of the problem being optimized, which means PSO does not require that the optimization problem be differentiable as is required by classic optimization methods such as gradient descent and quasi-newton methods. PSO can therefore also be used on optimization problems that are partially irregular, noisy, change over time, etc.

System Architecture

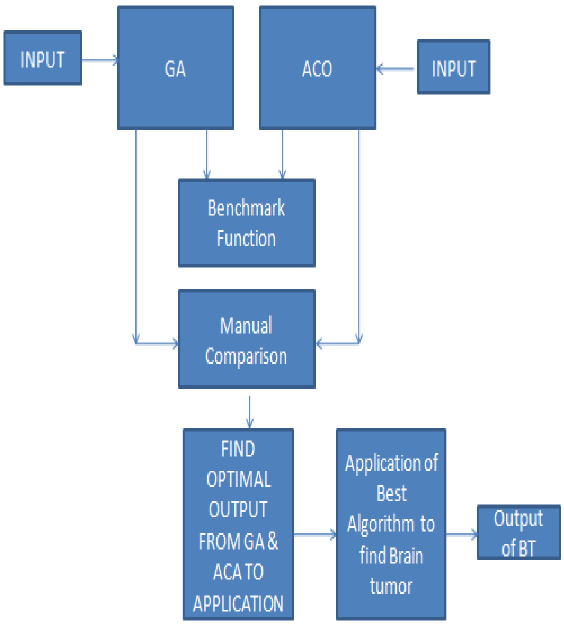


Figure No. 5.1 System Architecture

This is the flow diagram of the complete project that is how to process the implement the project required algorithm and how to process on the input data and how to getting output object of this project finally.

RESULT OF GA & ACO

The following table shows the output of Genetic Algorithm and Ant colony Algorithm

Sr. No.	Name of the function	Minima	GA – Single point Crossover	GA – Two point Crossover	ACO
			Iterations	Iterations	Iterations
1	GRIEWANK	[0 0]	225	29	1439
2	SPHERE	[0 0]	73	92	1021
3	ACKLEY	[0 0]	98	20	240
4	ROSENBROCK	[1 1]	39	692	762
5	RASTRIGIN	[0 0]	37	117	1015

Table No.1- Performance in terms of numbers of iteration required to find Minima.

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified ⧖ Impact Factor 7.39 ⧖ Vol. 11, Issue 7, July 2022
DOI: 10.17148/IJARCCE.2022.11796

Sr. No.	Name of the function	GA – Single point Crossover	GA – Two point Crossover	ACO
		Minima at	Minima at	Minima at
1	GRIEWANK	[0 0]	[0 0]	[-0.5165 0.0178] 10^{-5}
2	SPHERE	[0 0]	[0 0]	[0.885 0.4283] 10^{-5}
3	ACKLEY	[0 0]	[0 0]	[-0.0857 -0.1707] 10^{-4}
4	ROSENBROCK	[1 1]	[1 1]	[0.9685 0.9378]
5	RASTRIGIN	[0 0]	[0 0]	[-0.555 -0.1541] 10^{-4}

Table No.2 -The values obtained by different swarm algorithms.

Output of Brain Tumor by ACO

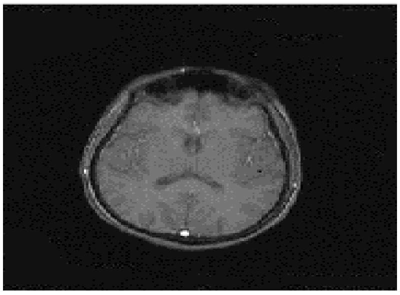


Figure :- Original test image

Figure :-No Tumor Detected in Test Image Normal

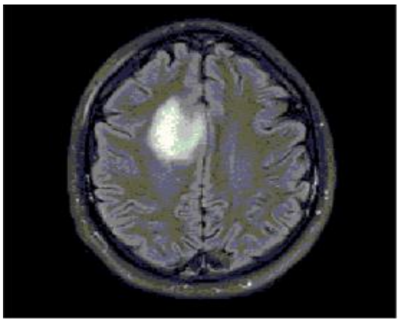


Figure :-Original test image

Figure :- Brain Tumor Detected in Test Image ABNormal

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified ∺ Impact Factor 7.39 ∺ Vol. 11, Issue 7, July 2022
DOI: 10.17148/IJARCCE.2022.11796

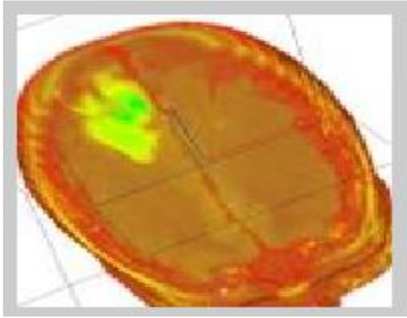


Figure :-Original test image

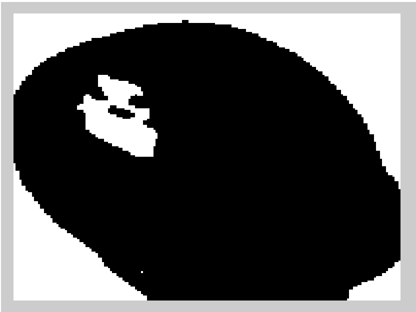


Figure :- Brain Tumor Detected in Test Image ABNormal

Future Scope

In the real word there are so many future scope of this project with the same logic or some modification on the project. Following are the same area where you can implement

1. Face detection,
2. Face Recognition,
3. Traffic control systems etc.
4. Content-based image retrieval
5. Medical imaging
6. Locate tumors and other pathologies
7. Locate objects in satellite images (roads, forests, crops, etc.)
8. Fingerprint recognition

CONCLUSIONS

'The Particle Swarm Intelligence', 'Ant C olony optimization' is the effectively algorithm over the Genetic algorithm, Therefore by using the Ant colony optimization we will find the brain tumor which will very helpful to the medical field by saving the time. It will also benefit to the application like face detection, recognition, Traffic control systems etc. and so many other application.

The project is to implement basic genetic algorithm and this algorithm can be combined with other optimization algorithms like 'The Particle Swarm Intelligence', 'Ant Colony optimization'. And effectively use it to find minima or maxima of benchmark function where other algorithms or methods stuck at local minima/ maxima. Latter on comparison the output depend on the performance table value. And then by the optimal algorithm find the tumor in the brain.

REFERENCES

[1] Russell Eberhart and James Kennedy, "A New Optimizer Using Particle Swarm Theory", sixth 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 Seventh Annual Conference on Evolutionary Programming, pp. 591-601, 1998.

DOI: 10.17148/IJARCCE.2022.11796

[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, member, IEEE, K. S. Tang, and S. Kwong, member, IEEE, "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 b´ezout number", seventh 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", fourth 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] F. 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', IEEE transactions on neural networks, vol-14, no.-2, pp. 337-350, March 2003.