As per UGC guidelines an electronic bar code is provided to seure your paper

*International Journal for Modern Trends in Science and Technology*, 8(01): 153-157, 2022

ISSN: 2455-3778 online
DOI: https://doi.org/10.46501/IJMTST0801026
Available online at: http://www.ijmtst.com/vol8issue01.html

# Particle Swarm Optimization (PSO) Model and Its Application in ANN Controller

**Priya Godbole[1] | Dr. Manjiri Pathak[2]**

[1]Assistant Professor, Department of Computer Science, Prerna College of Commerce, Nagpur, MS, India.
[2]Assistant Professor, Department of Computer Application, Prerna College of Commerce, Nagpur, MS, India.

**To Cite this Article**
Priya Godbole and Dr. Manjiri Pathak. Particle Swarm Optimization (PSO) Model and Its Application in ANN Controller. *International Journal for Modern Trends in Science and Technology* 2022, *8* pp. 153-157. https://doi.org/10.46501/IJMTST0801026

**Article Info**
Received: 02 December 2021; Accepted: 03 January 2022; Published: 08 January 2022.

## ABSTRACT

***Particle swarm optimization (PSO) is a population-based stochastic optimization algorithm inspired by intelligent collective behaviour from local interaction of some animals such as flocks of birds or schools of fish in which individual are self-organized and co-ordinate from the decentralized control. This type of model falls under Swarm Intelligence. It is drawn from the intelligent, experience-sharing, social flocking behaviour of birds. Artificial neural networks (ANNs) have excellent features with their ability for generalization, self-organization and self-learning. Particle Swarm Optimization (PSO) can be applied in ANN to avoid any possibilities from local extreme condition.***

***KEYWORDS:Boid, Particle swarm optimization (PSO), Social Cognitive Theory, Genetic Algorithms (GA), Artificial Neural Network (ANN) controller, Social component, cognitive component.***

## 1.INTRODUCTION

Particle Swarm Optimization (PSO), which was introduced by Russell Eberhart, an electrical engineer, and James Kennedy, a social psychologist, in 1995 [1] [2]. PSO was originally developed to solve non-linear continuous optimization problems. It is observed that birds can fly in large groups without collision for extended long distances, also maintain an optimum distance between themselves and their neighbours.

The Social Cognitive Theory, used in psychology, education, and communication,

suggest that part of knowledge acquired by humans can be directly influenced by

their neighbours within the context of social interactions and experiences [3][4]. The birds flocking behaviour is analogous to the human social behaviour, since the concept of bird's movement is generally similar to the concept of psychological behaviour changes in humans [1][6].

PSO is used to solve problems in an n-dimensional solution space [5]. The term "particles" refers the members (bird-like objects) of population, which basically denotes swarm positions in then-dimensional solution space. Each particle is initialized such that it moves through the solution space with a velocity vector representing the particle's speed in each dimension [7]. Each particle stores its historically best solution in its memory.

PSO exhibits the experience-sharing behaviour of each particle by which particles communicate with its neighbours or the whole swarm and directs the overall motion towards the most promising areas detected in

search space. In each iteration, a moving particle updates their experience in memory; if its current position is better than its historically best solution.

The key elements to design and improve the accuracy of an ANN are the architecture (or topology), the set of transfer functions (TF), and the set of synaptic weights and bias. These elements should be codified into the individual that represents the solution of our problem. The solutions generated by the bio-inspired algorithms will be measured by the fitness function with the aim to select the best particle that represents the best ANN [8]. Here we use Particle Swarm Optimization (PSO) Algorithm as a optimization technique to improve working of ANNs.

The simple mathematical model of swarms given by Craig Reynolds follows three distinct rules: separation, alignment and cohesion.

## 2. Working Mechanism of Original PSO

The PSO algorithm employs a swarm of particles which traverse a multidimensional search space to seek out optimal solution for real-value continuous problems [2]. In this algorithm, each particle globally compares its fitness to the entire swarm population and adjusts its velocity towards the swarm's global best particle.

The n-dimensional problem space has a number of dimensions that equals to the numbers of variables of the desired fitness function to be optimized. Let us consider a search space of dimension d. A particle iof the swarm is modelled by a position vector

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \qquad (1)$$

and a velocity vector denoted

$$v_{id}(t+1) = v_{id}(t) + c_1R_1(p_{id}(t) - x_{id}(t)) + c_2R_2(p_{gd}(t) - x_{id}(t)) \qquad (2)$$

where,

- $v_{id}$ represents the rate of position change of $i^{th}$particle in $d^{th}$dimention.
- t denotes the iteration counter.
- $x_{id}$ represents the position of the $i^{th}$ particle in the $d^{th}$ dimension.
- $p_{id}$ represents the historically best position of the ith particle in the dth dimension.
- $p_{gd}$ represents the position of the swarm's global best particle ($x_g$) in the $d^{th}$ dimension.
- R1 and R2 are two n-dimensional vectors with random numbers uniformly selected in the range of [0.0, 1.0], which introduce useful randomness for the search strategy.
- c1 and c2 are positive constant weighting parameters, also called the cognitive and social parameters, respectively, which control the relative importance of particle's personal experience versus swarm's social experience.

The aforementioned equations of position vector and velocity vector can be modelled to describe the basic flow of Particle Swarm Optimization (PSO) algorithm as shown below.

*Algorithm for Basic PSO [9]*

1. Initialize Position and Velocity for each particle.
2. Calculate desired fitness function.
3. Loop
4. Update value of historically best position (If current position is comparativelybetter than previously stored best position for the same particle).
5. Update particle's globally best position based on value of its desired fitnessfunction.
6. Update Velocity of each particle and move particle to new position.
7. End Loop

## 3. Extensions to Original PSO

Although local version of PSO algorithm shows good performance, it comes with some limitations when applied to complex problems having larger search space. Later on, numbers of parameter extensions are provided to original PSO that significantly had improved performance of an algorithm[5]. These parameters are given as follows:

i. *Limiting Particle's Velocities*- In local version of PSO, instant increase or instant decrease in particle's maximum velocity results in swarm explosion where value of velocity is not under control.This leads to swarm divergence in large search space when particles are still moving and takes large shifts from its current location. In order to govern uniform velocity for each particle while moving through all dimensions, Abido [10] had given an equation to calculate maximum velocity as follows:

$V_{max} = (x_d^{max} - x_d^{min})/K$ (3)

Where,

$x_d^{max}$ and $x_d^{min}$ are the maximum and minimum position values found so far bythe particles in the dthdimension.

K is a user-defined parameter that controls the shift intervals.

ii. *Defining Topology of Neighbours*- The search space of the problem is explored by each particle in the group after assigning indices to each one of them. The information related to swarms best global position is instantly shared among several local neighbourhoods of the particle. The particle's immediate neighbours are decided by the network topology in which they are arranged. There are 5 topologies of neighbours possible in particle group:The aforementioned idea is summarized as following:

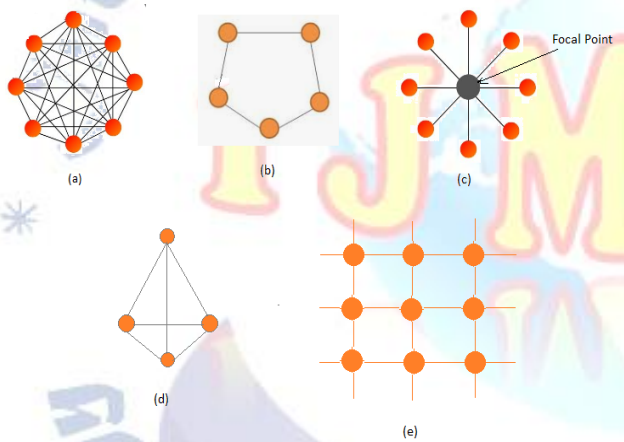


Figure1- Common regular shaped neighbourhood topologies: (a) the fully connected

network topology, (b) the ring (circle) topology, (c) the star (wheel) topology,

(d) the pyramid topology, and (e) the Von Neumann topology (the generally

recommended neighbourhood topology). Adapted from [11].

- Mesh/fully connected network topology; in this all particles can share information to all other particles in the group (Figure1 (a)).
- Ring/circle topology; in this each particle can share and access information to and from two immediate neighbours (one to its left and other to its right)(Figure1 (b)).
- Star topology; in which two particles can't share information directly. All particles are connected to central selected one particle commonly known as focal point which communicates information shared from all other particles (Figure1 (c)).
- Pyramid topology; in which each particle can share its information to three immediate neighbours arranged in pyramid (Figure1 (d)).
- Von Neumann topology structure; in which particle forms grid-like structure and each particle can share its information to four immediate neighbours of the grid/lattice structure (Figure1 (e)).Performance of specific topology depends upon type of the problem and size of search space to be explored.

iii. *Inertia/momentum Weight*- To simplify the convergence of particle in complex optimization problems the inertia weight is introduced. Hence, the proportion of local and global search abilities of the particle is balanced significantly.

## 4. Basic Working of Artificial Neural Networks (ANNs)

ANNs are system composed of neurons organized in input, output, and hidden layers. The neurons are connected to each other by multiple synaptic weights. The system learns information during each iteration in training phase. On completion of training phase, the system classifies the new information and generates behaviour based upon learned information. Set of interconnected neurons are responsible for summation of input information which is transformed with respect to their synaptic weights. In this summation phase, another input commonly known as bias is used and its value is considered as 1.

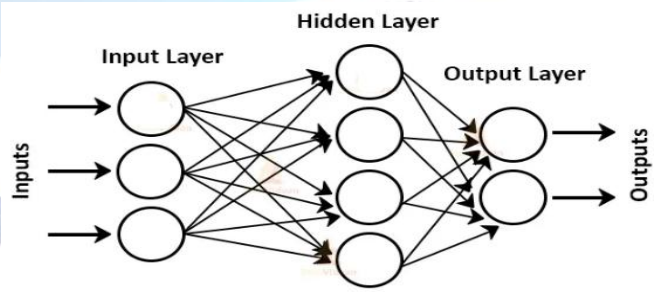


Figure2- working of ANNs. Adapted from Google images

As soon as the summation is performed, the result is assigned to transfer function and sent to output neuron; until they reach the last layer.

## 5. TUNING THE ANN USING PSO ALGORITHMS

PSO is widely used in neural network applications. PSO is assisted by the movement of the best particle from the group, known as the social compound, and their own experience, known as the cognitive compound [1][12]. The algorithm derives a complete set of solutions and estimates the best solution among them. In ANNs, PSO is applied to a multi-layered perceptron where the position of each particle, in a swarm, represents the set of synaptic weights of the neural network for the current iteration. Each particle can explore the search space in multiple dimensions based on synaptic weights[13].

Suppose a search space having dimension d. The position vector for the particle i can be expressed as:

$$x_{ij} = [x_{i1}, x_{i2}, \ldots, x_{id}]^T \quad (4)$$

and a velocity vector can be expressed as:

$$v_{ij} = [v_{i1}, v_{i2}, \ldots, v_{id}]^T \quad (5)$$

Each particle moves in the weighting space trying to minimize the learning error during training phase also keeps in memory the historically best position through its path of exploration and can be expressed as:

$$Pbest_{ij} = [pbest_{i1}, pbest_{i2}, \ldots, pbest_{id}]^T \quad (6)$$

Also the best position reached by the swarm can be expressed as:

$$Gbest_{ij} = [gbest_{i1}, gbest_{i2}, \ldots, gbest_{id}]^T \quad (7)$$

When the particle changes the position, it is analogous to updating the synaptic weights of the neural network controller to generate the proper control law to reduce tracking error. In each iteration k, the particles update their position by calculating the new velocity and move to the new position.

At the iteration (k+1), the velocity vector is calculated as follows:

$$v_{ij}(k+1) = wy_{ij}(k) + c_1r_1[pbest_{ij}(k) - x_{ij}(k)] + c_2r_2[gbest_{ij}(k) - x_{ij}(k)] \quad (8)$$

where,

- w being a variable parameter making it possible to control the changing of theparticle at the next iteration,
- $wv_{ij}$ is a physical changing component,
- $c_1r_1[pbest_{ij}(k) - x_{ij}(k)]$ is a cognitive changing component,
- $c_2r_2[gbest_{ij}(k) - x_{ij}(k)]$ is a social component of changing,
- c1 and c2 are cognitive and social parameters respectively, which presents the degree of attraction towards the best position of a particle and that of these informants,
- r1 and r2 are two random numbers drawn uniformly in the interval [0, 1] represent the proper exploration of particles in the search space.

The smallest learning error of each particle Pbesti and the smallest learning error found in the whole learning process Gbesti are applied to produce a fit of the positions towards the best solution or the targeted tracking error.

The position, at the iteration (k+1), of particle i is then defined as follows:

$$x_{ij}(k+1) = x_{ij}(k) + v_{ij}(k) \quad (9)$$

As the position changes, the values of Pbesti and Gbesti vectors are also get updated at (k+1)th iteration.

The following algorithm summarizes the PSO algorithm in ANN controller having N particles with k number of iterations:

*PSO Algorithm for ANN controller [14]*

1. Initialize Position and Velocity for each particle.
2. Loop
3. Calculate the control law from the controller input vector.
4. Calculate the outputs of the system.
5. Evaluate the positions of particles in the search space .
6. Update particle's local best position based on their history.
7. Update particle's global best position with respect to other particles.
8. Update Velocity of each particle and move particle to new position.
9. End Loop

## 6. FUTURE SCOPE AND CONCLUSION

Swarm Intelligence-based optimization techniques are wide-spread in many domains, and have a wide-range of successful applications in different areas. PSO shows collective natural behaviour of individuals in group and are somehow similar to Genetic Algorithms (GA).

In order to improve the performance of Artificial Neural Networks (ANNs) the SI optimization techniques like PSO is more useful. This paper shows how PSO can be applied to design and improve the performance of ANNs. ANN controller uses PSO as an optimization technique to simulate neural networks and to accelerate the speed of the learning phase of the neural network model and neural controller.

## REFERENCES


[1]

[2] J. Kennedy and R. C. Eberhart. "Particle Swarm Optimization". In Proceedings of IEEE. International Conference on Neural Networks, Perth, Australia, pp. 1942–1948, 1995.

[3] R. C. Eberhart and J. Kennedy. "A new optimizer using particle swarm theory. InProceedings of the Sixth International Symposium on Micro Machine and HumanScience, Nagoya, Japan, pp. 39–43, 1995.

[4] A. Bandura.In: Social Foundations of Thought and Action: A Social Cognitive Theory, Prentice-Hall, Englewood Cliffs, NJ, 1986.

[5] M. Ethan."Swarm Behavior of Animals." International Journal Swarm EvolComput 10:e205,2021.

[6] M. Imran, R. Hashim and N. E. Khalid."An Overview of Particle Swarm Optimization Variants", Malaysian Technical Universities Conference on Engineering & Technology 2012, MUCET 2012 Part 4 Information and Communication Technology, Procedia Engineering 53, 2013, pp. 491 – 496.

[7] Y. Liu, B. Ying, C. Sui, Y. Luo, Z. Zhang and R. Liu, Research Article: Simulation Study of Swarm Intelligence Based on Life Evolution Behavior. In:Hindawi Publishing Corporation Discrete Dynamics in Nature andSociety Volume 2015, Article ID 291298, 2015.

[8] A. Beatriz and A. Roberto. Designing ArtificialNeural Networks Using Particle Swarm Optimization Algorithms. In:Hindawi PublishingCorporation Computational Intelligence and Neuroscience Vol. 2015, Article ID 369298, 2 June 2015.

[9] Y. Zhang, P. Agarwal, V. Bhatnagar, S.Balochian, and J. Yan, (eds) In: Swarm Intelligence and Its Applications, Hindawi Publishing Corporation. TheScientific World Journal Volume 2013, Article ID 528069, 2013.

[10] Y. Shi."Feature article on particle swarm optimization, IEEE Neural Network Society",Feature Article, 2014, pp. 8–13.

[11] A. Abido."Optimal power flow using particle swarm optimization, InternationalJournal of Electrical Power Energy System", Vol. 24, No. 7, 2002, pp. 563–571.

[12] Y. Valle, G. Venayagamoorthy, S. Mohagheghi, J.C. Hernandez, R. G. Harley. Particleswarm optimization: basic concepts, variants and applications in power systems. In: IEEETrans. on Evolutionary Computation, Vol. 12, No. 2, 2008, pp. 171–195.

[13] S.Slama, A.Errachdi and M.Benrejeb, In: Chapter: Tuning Artificial Neural NetworkController Using Particle Swarm Optimization Technique for NonlinearSystem, 2021.

[14] In: Tuning Artificial Neural Network Controller Using Particle Swarm OptimizationTechnique…doi: http://dx.doi.org/10.5772/intechopen.96424.

[15] H. R. Ahmed and J. I. Glasgow.In: Swarm Intelligence: Concepts, Modelsand Applications, Technical Report 2012-585, Conference Paper, 2012.