

The Impact of Social Network Structure in Particle Swarm Optimization for Classification Problems

Razana Alwee, Siti Mariyam Shamsuddin, Firdaus Ab. Aziz
K.H. Chey and Haza Nuzly Abdull Hameed
Soft Computing Research Group, University of Technology Malaysia,
81310 Skudai Johor, Malaysia

Abstract: Particle Swarm Optimization (PSO) is a mechanism that involves several particles (solutions) interacting among each other to find the best solutions. It is a functional procedure by initializing a population of random solutions and searches its member by assigning random positions and velocities. The potential particle solutions are then flown through the hyperspace to get the optimum solutions. In this study, social network structure of PSO is incorporated into Artificial Neural Network (ANN) to investigate its learning efficiency. The results yield that the classification and convergence rates of ANN with ring structure (lbest) is better compared to the star social structure (gbest). These results are further validated by executing statistical significant test for better justification.

Key words: ANN, particle swarm optimization, MLP, social network structure, global best, local best

INTRODUCTION

Particle Swarm Optimization (PSO) has been an increasingly interesting topic in the area of computational intelligence. It is an optimization tool that based on population space. The algorithm is initialized with a population of random solutions that can search for optimality by updating particles accordingly.

The importance component in PSO is its Social Network Structure (SNS). The SNS for PSO is determined by the formation of overlapping neighborhoods, in which each particle within a neighborhood influences one another. In PSO, particle in the same neighborhood communicates with one another by exchanging information about the success of other particle within the same neighborhood. It has been reported that with unimodal problems, a highly connected network (like the one available in a gbest-type of PSO) provides better performance, while the lbest PSO topology performs well on multimodal functions (Kennedy and Eberhart, 2001). To investigate the efficiency of PSO in optimization problems, a classifier must be incorporated particularly for classification problems. The most common classifier that is normally integrated with PSO is Artificial Neural Network (ANN).

ANN is an information processing paradigm that is inspired by the way biological nervous systems process information, such as the brain. The computation is highly complex, nonlinear and parallel. Processing power of ANN

allows the network to learn and adapt. In addition, ANN is well suited to tasks such as classification, pattern recognition, memory recall, prediction, optimization and noise filtering (Luger, 2002).

The primary significance of ANN is the ability of the network to learn from its environment and to improve its performance through learning (Haykin, 1999). Learning is a process of modifying the weights and biases to the neurons and continued until a preset condition is met. The objective of the training process is to classify some input data patterns to certain outputs prior to a testing with new data.

This study presents, the effectiveness of integrating difference social network structures of PSO in ANN classifier. The results are tested on various standard datasets available from UCI machine learning database.

OVERVIEW OF PSO

Each particle in the swarm represents a candidate solution to the optimization problem (Eberhart and Shi, 2001). Each particle keeps track of its best fitness position in hyperspace that has achieved so far, thus we called it personal best or pbest. The overall best value obtained by any particle in the population is called global best or gbest. During each epoch (or iteration), every particle is accelerated towards its own personal best in the direction of the global best position. This is achieved by calculating a new velocity term for each particle based on

the distance from its personal best and also the distance from the global best position. These two components (personal and global velocities) are then randomly weighted to produce the new velocity value for the particle. This value will affect the new position of the particle during the next epoch (Van den Bergh, 1999).

There are two equations involved in PSO. The movement equation (Eq. 1) and velocity update equation (Eq. 2) (James and Rui, 2003). Equation 1 provides the actual movement of the particles using their specific vector velocity, while Eq. 2 gives velocity vector adjustment given the two competing forces (gbest and pbest).

$$x_n = x_n + v_n \times \Delta t \quad (1)$$

$$v_n = v_n + C_1 \times \text{rand}() \times (g_{\text{best},n} - x_n) + C_2 \times \text{rand}() \times (p_{\text{best},n} - x_n) \quad (2)$$

Equation 1 is performed for each element of the position (x) and velocity (v) vector. The Δt parameter (which is usually set to 1.0) defines the discrete time interval over which, the particle will move. The result is a new position for the particle. In Eq. 2, the dimensional element is subtracted from the dimension of the best vector. Subsequently, it multiplies with a random number (between 0.0 and 1.0) and an acceleration constant (c_1 and c_2). The sum of these products is added to the given velocity. This process is performed for each element of the velocity vector. The random numbers provide randomness in the path to help the particle move around the solution space. The parameters c_1 and c_2 provide some control to decide, which particles should be given more emphasis; gbest or pbest.

PSO for neural network learning: Implementation of ANN learning with PSO is based on the position for each particle represents a set of weight. The dimensionality of each particle is the number of weights associated with the network (Al-kazemi and Mohan, 2002; Lee and Siti, 2008). The particle moves within the weight space attempting to minimize learning error. Changing the particle position gives the signal of updating the network weight accordingly as well as to reduce the error of the current epoch. In each epoch, all the particles update their position by calculating the new velocity. The new position is a set of new weights used to obtain the new error. For PSO, the new weights are adapted even though, no improvement is observed. This process is repeated for all the particles. The particle with the lowest error is considered as the global best particle. The training

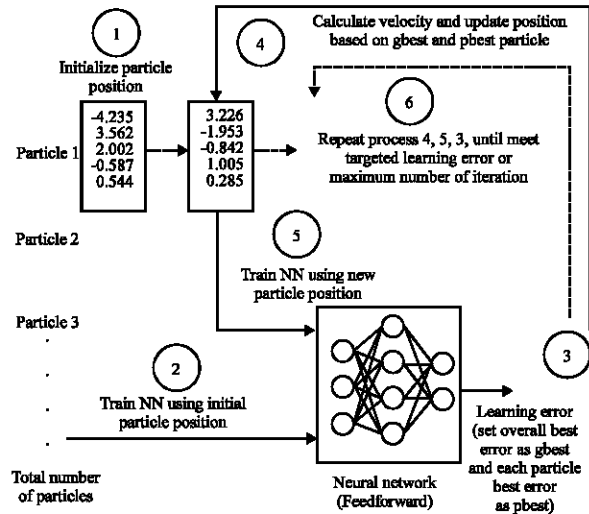


Fig. 1: PSO in ANN learning process

process continues until satisfactory error is achieved by the best particle or the maximum threshold is met. The weights are calculated for the classification error upon completion of the training process. The same set of weights is used to test the network with new dataset.

Feed forward ANN yields the learning error (particle fitness) based on set of weight and bias (PSO positions). The pbest value and gbest value are applied to Eq. 2 to produce a value for positions adjustment to the best solution or targeted learning error. The new sets of positions (ANN weight and bias) are produced by adding the calculated velocity value from Eq. 2 to the current position value using Eq. 1. These new set of positions are used for producing new learning error (particle fitness) in feed forward ANN. This process is repeated until the stopping conditions are met (minimum learning error or maximum number of iteration). Figure 1 shows the process of PSO in ANN learning.

PSO network structure: The social structure for PSO is determined by the formation of overlapping neighborhoods, where particles within a neighborhood influence one another. This is an analogy with observations of animal behavior, where an organism is most likely to be influenced by others in its neighborhood. Organisms that are more successful will have a greater influence on members of the neighborhood than the less one.

Global best PSO: For the global best PSO, or gbest PSO, the neighborhood for each particle is in its entire swarm. The social network structure of gbest PSO is star topology. For this topology, the social component of the

```

Create and initialize an  $n_s$ -dimensional swarm, S; repeat

    for each particle  $i=1, \dots, S.n_s$  do
        //set the personal best position
        if  $f(S.x_i) < f(S.y_i)$  then
             $S.y_i = S.x_i$ ;
        end
        //set the global best position if  $f(S.y_i) < f(S.\hat{y})$  then
             $S.y = S.y_i$ ;
        end
    end
end
for each particle  $i=1, \dots, S.n_s$  do
    update the velocity using Eq. (2c);
    update the position using Eq. (2a);
end
until stopping condition is true;
    
```

Fig. 2: gbest algorithm

update particle velocity reflects the obtained information from all the particles in the swarm. In this case, the social information is the best position found by the swarm and this is denoted as $\hat{y}(t)$.

For gbest PSO, the velocity of particle i is calculated as:

$$v_{ij}(t+1) = v_{ij}(t) + C_1 r_{1j}(t) [y_{ij}(t) - x_{ij}(t)] + C_2 r_{2j}(t) [\hat{y}_j(t) - x_{ij}(t)] \quad (3)$$

Where:

- $v_{ij}(t)$ = The velocity of particle i in dimension $j = 1, \dots, n$ at time step t
- $x_{ij}(t)$ = The position of particle i in dimension j at time step t
- C_1 and C_2 = Positive acceleration constants used to scale the contribution of the cognitive and social components, respectively and $r_{ij}(t)$, $r_{2,j}(t) \sim U(0, 1)$ are random values in the range of $(0, 1)$

These random values cause a stochastic element to PSO algorithm. The personal best position, y_i , associated with particle i is the best position the particle has been visited. The gbest PSO is shown in Fig. 2.

Local best PSO: The local best PSO, or lbest PSO, uses a ring social network topology, in which smaller neighborhoods are defined for each particle. The social component reflects the exchanged information within the neighborhood of the particle. With reference to Eq. 2, the social contribution to particle velocity is proportional to the distance between a particle and the best position found by the neighborhood of particles. The velocity is calculated as:

```

Create and initialize an  $n_s$ -dimensional swarm, S;
repeat
    for each particle  $i=1, \dots, S.n_s$  do
        //set the personal best position
        if  $f(S.x_i) < f(S.y_i)$  then
             $S.y_i = S.x_i$ ;
        end
        //set the neighborhood best position
        if  $f(y_j) < f(y_i)$  then
             $S.y = S.y_j$ ;
        end
    end
    for each particle  $i=1, \dots, S.n_s$  do
        update the velocity using equation (2d);
        update the position using equation (2a);
    end
end
until stopping condition is true;
    
```

Fig. 3: lbest algorithm

$$v_{ij}(t+1) = v_{ij}(t) + C_1 r_{1j}(t) [y_{ij}(t) - x_{ij}(t)] + C_2 r_{2j}(t) [\hat{y}_j(t) - x_{ij}(t)] \quad (4)$$

where, \hat{y}_{ij} is the best position, found by the neighborhood of particle i in dimension j . The lbest algorithm shown in Fig. 3.

PSO parameters: There are four basic parameters in PSO (Lee and Siti, 2008):

- The acceleration constants for gbest (C_1)
- The acceleration constants for pbest (C_2)
- The time interval (Δt)
- The number of particle

The acceleration constants are used to define how particles swarm in the simulation (Luger, 2002). The acceleration constant and C_2 represent the stochastic acceleration that pulls each particle toward pbest and gbest position. The C_1 constant affects the influence of the global best solution over the particle, whereas, the C_2 constant affects how much influence of personal best solution has over the particle. When C_1 is higher than C_2 , the swarm tends to move around the global best solution, whereas the inverse causes greater flow around the individual personal best solution.

The dt (or Δt) parameters defines the time interval over, which movement takes place in the solution space. Decreasing the Δt values will provide higher granularity movement within the solution space and higher Δt value performs lower granularity movement. For the number of particles in the swarm, the more particles that are presented the greater the amount of space that is covered in the problem, thus optimization becomes slower.

Table 1: PSO parameters

Parameters	Values
C_1	2.0
C_2	2.0
Δt	0.1
Number of particles	20
Problem dimension	Based on dataset NN architecture
Range of particle	Not specified. Free to move anywhere
Stop condition	NN minimum error or maximum number of iteration

Depending on the solution space, this parameter can be adjusted to achieve better optimization. There are also, some other parameters that depend on the problems such as particle dimension, number of particles and stopping condition. In present study, the number of dimension is referring to the number of weight and bias that is based on the dataset and ANN architecture. Equation 5 shows how dimension is calculated in this study. The number of particles in the swarm affects the run-time significantly, thus a balance between variety (more particles) and speed (fewer particles) must be sought (Van den Bergh and Engelbrecht, 2000). PSO with well-selected parameter set can have good performance (Shi, 2004).

$$\text{Dimension} = (\text{input} \times \text{hidden}) + (\text{hidden} \times \text{output}) + \text{hidden}_{\text{bias}} + \text{output}_{\text{bias}} \quad (5)$$

Parameters that have been used in this study are shown in Table 1, while particle position (weight and bias) values are initialized randomly with initial position velocity value is set to 0. The C_1 and C_2 constant are set to 2 as suggested (Eberhart and Shi, 2001).

RESULTS AND DISCUSSION

In order to investigate the impacts of different social network structure of PSO in ANN learning, three dataset have been used in this study and these include Iris, breast cancer Wisconsin and Herberman's survival dataset. The results are evaluated and analyzed based on the convergence rate and classification performance.

An impact of gbest and lbest on iris dataset: The network architecture used for Iris dataset consists of 4 input nodes, 9 hidden nodes and 3 output nodes. Forty data patterns used to train the network. For PSO parameters, 20 particles are used, C_1 and $C_2 = 2$, $\Delta t = 0.1$, problem dimension is set to 75 and the stopping conditions are set to minimum error of 0.005 or maximum iteration of 1000. The experimental results are shown in Table 2 and Fig. 4.

Table 2 depicts that gbest has converged at minimum error of 0.04638, while minimum error for lbest is 0.00293. The classification is better for lbest than gbest with 99.99% accuracy compared to 97.16%. Hence, lbest converges faster at only 9 iterations compared to gbest with 10 iterations. Thus, the errors are reduced significantly (Fig. 4).

Table 2: Performance of Iris dataset

Parameters	Global best	Local best
Learning iteration	10	9
Error convergence	0.04638	0.00293
Classification (%)	97.16	99.99

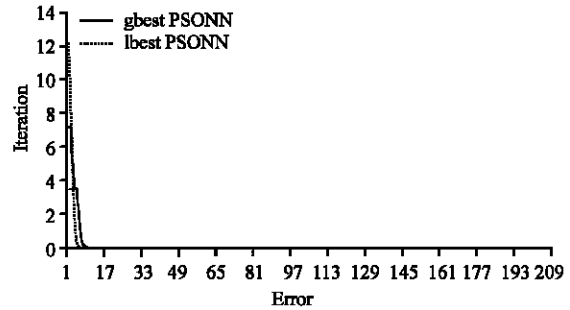


Fig. 4: Convergence of Iris dataset

An impact of gbest and lbest on breast cancer dataset: For breast cancer Wisconsin problems, 50 data patterns have been used with the network size consists of 9 nodes in the input layer, 19 nodes in the hidden layer and 1 node in the output layer. For PSO parameters, 20 particles are used, C_1 and $C_2 = 2$, $\Delta t = 0.1$ and problem dimension equals 210. The stopping conditions are set to a minimum error of 0.005 or maximum iteration of 1000. The experimental results for gbest and lbest are shown in Table 3 and Fig. 5.

In breast cancer Wisconsin learning process, both algorithms converge using the maximum number of pre-specified iteration. In this experiment, lbest manages to converge with 8 iterations, while gbest converges at 208 iterations. For classification accuracy, it shows that lbest is better than gbest with 99.67% compared to 99.35%. Figure 5 depicts that lbest significantly reduces the error at small number of iteration compared to gbest.

An impact of gbest and lbest on herberman's survival dataset: The network size that has been used to train the Herberman's survival problem consists of 3 input layer nodes, 7 hidden layer nodes and 2 output layer nodes with 100 data patterns. For PSO particles are set to 20, C_1 and $C_2 = 2$, $\Delta t = 0.1$ and problem dimension is set to 44. The stopping conditions are set to minimum error of 0.005 or maximum iteration of 1000. The experimental results for gbest and lbest are shown in Table 4 and Fig. 6.

In Herberman's survival learning process, both algorithms converge using the maximum number of pre-specified iteration. In this experiment, lbest and gbest managed to converge with minimum iterations. Figure 6 depicts that lbest is better than gbest 99.98% compared to 98.97%.

This analysis is carried out to compare the result between ANN learning with star structure PSO (gbest) and ring structure PSO (lbest). The learning patterns for

Table 3: Performance breast cancer Wisconsin dataset

Parameters	Global best	Local best
Learning iteration	208	8
Error convergence	0.00495	0.00333
Classification (%)	99.35	99.67

Table 4: Performance of Herberman's survival dataset

Parameters	Global best	Local best
Learning iteration	11	11
Error convergence	0.00369	0.00031
Classification (%)	98.97	99.98

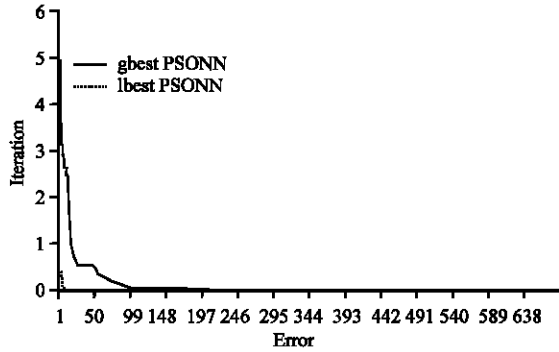


Fig. 5: Convergence of breast cancer Wisconsin dataset

both algorithms are compared using the same datasets as above. Figure 7 shows that ANN with ring structure PSO (lbest) gives better convergence rate and classification accuracy with minimum iterations.

Statistical test: The efficiency of the results are validated by using statistically simulations, t-test to investigate the significant of 2 groups. This test is conducted by calculating the mean of the collected data from two random samples of independent observations, each from an underlying normal distribution. t-test is also a method of validating stochastic models of population viability. It is based on assessing the mean and variance of the predicted population size. When, carrying out t-test, two hypotheses will be conducted:

H_0 : Population means are the same

$$\mu_1 = \mu_2$$

H_1 : Population means are not the same

$$\mu_1 \neq \mu_2$$

The mean of two sample data needs to be determined. Mean is the arithmetic average of a set of values or distribution, Eq. 6:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \quad (6)$$

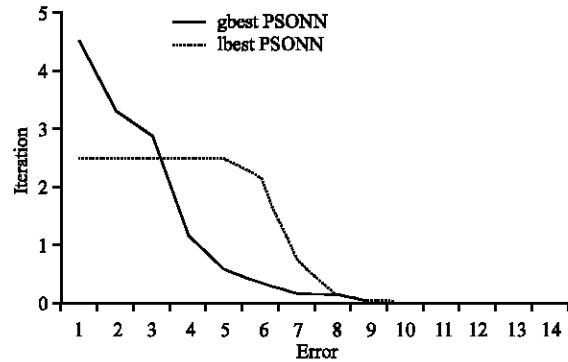


Fig. 6: Convergence of herberman's survival dataset

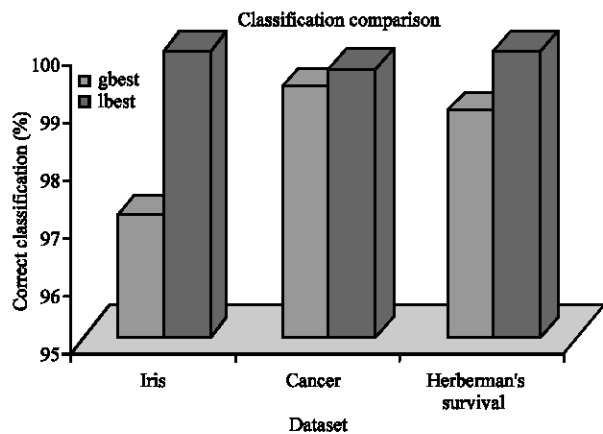


Fig. 7: Comparative between gbest and lbest

Standard deviation is a measure of the dispersion of a set of values. It can be applied to a probability distribution, a random variable or a population. The standard deviation is usually denoted with the letter of σ . It is defined as the square root of the variance as shown Eq. 7. The standard deviation of the mean is given as Eq. 7:

$$\sigma = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2} \quad (7)$$

Where:

- σ = The standard deviation of the variable and
- n = The number of observations

Degree of Freedom (DoF) is the number of parameters, which may be independently varied and may in a problem, population and distributions. For two data samples, there are $n_1 + n_2$ observations. There are two means to be estimated. That leaves $n_1 + n_2 - 2$ DoF for estimating variability.

In this study, the t-value for Iris dataset is given as 0.83428 and the critical value of t is 1.9845. Therefore, two-tailed probability values of a t-test are 0.406141. The t-test for Iris dataset is shown in Table 5.

Table 5: t-test for Iris dataset

Parameters	Local best	Global best
Number of iteration	50	50
MSE	3.40038	2.74219
SD	2.88000	4.77761

Table 6: t-test for cancer dataset

Parameters	Local best	Global best
Number of iteration	50	50
MSE	0.34248	0.21028
SD	0.76497	0.19032

Table 7: t-Test for Herberman's dataset

Parameters	Local best	Global best
Number of iteration	50	50
MSE	1.41579	1.19193
SD	1.18817	1.60826

The t-value was 0.83428 and the critical value of t is 1.9845. Here, the value of 1.9845 is >0.83428 , hence, the null hypothesis H_0 is rejected. We can conclude that there is a significant difference between population means.

Table 6 shows the significant test for cancer dataset. The estimate difference mean is 0.09517 and the t-value is 1.1858. The critical value of t is 1.9845, hence, two-tailed probability values of a t-test are 0.23857.

The value of 1.9845 is >1.18585 hence, the null hypothesis H_0 is rejected. We can conclude that there is a significant difference between population means for cancer dataset.

Table 7 depicts the significant test for Herberman's survival dataset. The estimate difference mean is 0.22386 with significance level of 0.05 ($\alpha = 0.05$). The Confidence Interval (CI) is equal to 95%. The t-value is 0.79164 and the critical value of t is 1.9845. Therefore, two-tailed probability values of a t-test are 0.43051.

CONCLUSION

In this study, network architecture and selection of network parameters for the dataset have manipulated the convergence and the performance of network learning. By reducing the hidden nodes, it minimizes the number of weight (position) and also reduces the problem dimension. In this study, PSO social network structure, lbest and gbest are implemented in feed forward ANN to improve the learning.

The comparison between gbest and lbest are carried out to validate the effectiveness of each social network structure in terms of convergence rate and accuracy.

The results are further validated with statistical test. Based on the analysis, it shows that the classification accuracy and convergence rates of ANN with ring structure PSO (lbest) is better compared to ANN with star structure PSO (gbest).

ACKNOWLEDGEMENTS

This research was supported by Ministry of Higher Education (MOHE) under Fundamental Research Grant Scheme (FRGS VOT 78243). Authors would like to thank RMC, Universiti Teknologi Malaysia for research activities, Soft Computing Research Group (SCRG) for the support and anonymous reviewers for their incisive comments to improve this research .

REFERENCES

- Al-kazemi, B. and C.K. Mohan, 2002. Training Feedforward Neural Network Using Multi-phase Particle Swarm Optimization. Proceeding of the 9th International Conference on Neural Information Processing. New York.
- Eberhart, R. and Y. Shi, 2001. Particle swarm optimization: developments, application and resources. IEEE Proceeding of the Congress on Evolutionary Comput., 1: 81-85. DOI: 10.1109/CEC.2001.934374.
- Haykin, S., 1999. Neural Network: A Comprehensive Foundation. 2nd Edn. Prentice Hall. ISBN: 10-0132733501.
- James, K. and M. Rui, 2003. Neighborhood topologies in fully informed and best-of-neighborhood particle swarms. Proceed. IEEE. Workshop on Soft Comput. Industrial Applications Issue, pp: 45-50. DOI: 10.1109/SMCIA.2003.1231342.
- Kennedy, J. and R. Eberhart, 2001. Swarm Intelligence, Morgan Kaufmann Academic Press. ISBN: 1-55860-595-9, San Francisco, CA.
- Lee, Y.S. and M.S. Siti, 2008. Bounded PSO v_{max} function in neural network learning. Proc. IEEE Comput. Soc. Press on the 8th Int. Conf. Intelligent Syst. Designs and Application (ISDA), Taiwan, pp: 474-479. ISBN: 978-0-7695-3382-7.
- Luger, G.F., 2002. Artificial Intelligence: Structures and Strategies for Complex Problem Solving. 4th Edn. Pearson Education Limited, England, ISBN: 3827370027.
- Shi, Y., 2004. Particle Swarm Optimization. IEEE Neural Network Society. Electronic Data Syst. Inc, pp: 8-13.
- Van den Bergh, F., 1999. Particle Swarm Weight Initialization in Multi-layer Perceptron Artificial Neural Networks. Development and Practice of Artificial Intelligence Techniques, Durban, South Africa, pp: 41-45.
- Van den Bergh, F. and A.P. Engelbrecht, 2000. Cooperative learning in neural networks using particle swarm optimizers. South Afr. Comput. J., (26): 84-90. DOI: 10.1.1.27.4265.