COMPARISON OF SOCIAL NETWORK STRUCTURE FOR PARTICLE SWARM OPTIMIZATION

CHEY KOK HUAT

A thesis submitted in fulfilment of the
requirements for the award of the degree of
Master of Science (Computer Science)

Faculty of Computer Science and Information System
Universiti Teknologi Malaysia

JUNE 2008

ABSTRACT

Swarm Intelligence (SI) originated from the study of colonies, or swarms of social organisms. Studies of the social behavior of organisms in swarms prompted the design of very efficient optimization and clustering algorithms. One of the major techniques in SI is Particle Swarm Optimization (PSO) while it is a technique where several particles (solutions) interacting between each other to find the best solutions. PSO shares many similarities with evolutionary computation techniques such as Genetic Algorithms (GA). GA evolution operators such as crossover and mutation, that chromosomes share information with each other, so the whole population moves like a one group towards an optimal area. Therefore, the various optimization techniques of PSO have been implemented in learning to increase the performance and validate the effectiveness of the social network structure. PSO is a functional procedure by initializing a population of random solutions and searches its member, called particle are initialized by assigning random positions and velocities. The potential particle solutions are then flown through the hyperspace to get the optimum solutions. However, to investigate the efficiency of PSO in optimization problem, a classifier must be incorporated particularly for classification problem. The most common classifier that is normally integrated with PSO is Artificial Neural Network. In this study, PSO is chosen and applied in feedforward neural network to enhance the learning process in terms of convergence rate and classification accuracy.

ABSTRAK

Kepintaran Berkumpulan (KB) berasal daripada proses pembelajaran tanah jajahan, ataupun perkumpulan organisma sosial. Tujuan pengajian kelakuan organisma sosial dalam berkumpulan adalah supaya mendapati pengoptimaan yang cekap dan menghasilkan algoritma yang berkelompok dengan pantas. Salah satu teknik yang penting daripada KB adalah Pengoptima Partikal Berkumpulan (PPB) di mana ia berinteraksi antara beberapa partikal untuk mendapatkan proses penyelesaian yang terbaik. PPB mempunyai bahagian persamaan dengan teknik evolusi perkiraan seperti Algoritma Genetik (AG). Untuk mendapatkan penyelesaiaan yang terbaik, kaedah pengoptimuman diperlukan seperti PPB untuk meningkatkan pembelajaran RN dari aspek prestasi rangkaian dan ketepatan pengelasan. PPB berfungsi dengan mengisytiharkan satu populasi penyelesaian secara rawak dan carian bagi nilai optimum dengan memperbaharui penghasilan generasi. Populasi berkenaan dan ahlinya dikenali sebagai partikel diberi nilai awalan dengan mengumpukkan kedudukan partikel secara rawak dan nilai pecutan bagi partikel yang berkaitan. Dapatan penyelesaian partikel yang berpotensi akan dikenalpasti dan ditaburkan pada ruang bertahap tinggi bagi mendapat satu penyelesaian yang optimum. Walau bagaimanapun, untuk menyelidik kecekapan bagi PPB dalam pengoptimaan, fungsi ketepatan pengkelasan amat diperlukan. Secara umumnya, penyelidikan kecekapan yang bersepadu dengan PPB adalah Rangkaian Neural. Dalam kajian ini, algoritma pengoptima yang terkini iaitu Pengoptima Partikal Berkumpulan telah dipilih dan digunakan dalam Rangkaian Neural untuk meningkatkan keupayaan proses pembelajaran dari segi masa penumpuan dan ketepatan pengkelasan.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 Introduction

Swarm intelligence (SI) originated from the study of colonies, or swarms of social organisms. Studies of the social behavior of organisms (individuals) in swarms prompted the design of very efficient optimization and clustering algorithms. With the latest research in soft computing, Swarm Intelligence (SI) technique was introduced in 1995 by James Kennedy, a social psychologist and Russell C. Eberhart, Associate Dean for Research, Purdue School of Engineering and Technology. SI is a bio-inspired technique and the latest Artificial Intelligence technique based on the study of collective behavior in decentralized and self organized systems. SI is defined as any attempt to design algorithms or distributed problem-solving devices inspired by the collective behavior of the social insect colonies and other animal societies [1]. SI systems are typically made up from a population of agents interacting locally with one another and with their environment and local interactions between such nodes often lead to the emergence of global behavior. There are two major techniques in SI: Ant Colony Optimization (ACO) and Particle Swarm Optimization (PSO). The ACO algorithm is a probabilistic technique for solving computational problems which can be reduced to search for good paths through graphs. They are inspired by the behavior of ants in finding paths from the colony to food. While PSO is a technique where several particles (solutions) interacting between each other to find the best solutions.

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). PSO algorithm is an optimization tool based on population, and the system is initialized with a population of random solutions and can search for optima by the updating of generations. However, unlike GA, PSO has no evolution operators such as crossover and mutation [2]. In the PSO algorithm, the potential solutions, called as particles, are obtained by ‘‘flowing’’ through the problem space by following the current optimum particles [3]. Generally speaking, the PSO algorithm has a strong ability to find the most optimistic result, but it has a disadvantage of easily getting into a local optimum [4, 5, 6, 7]. PSO has been successfully applied in many areas such as function optimization, artificial neural network training, fuzzy system control, and other areas [8].

Particle Swarm Optimization (PSO) is an optimization technique in the area of Swarm Intelligence that presents several advantages:

(i) It is easy to describe

(ii) It is simple to implement

(iii) There are few parameters to adjust

(iv) It uses a relatively small population

(v) It needs a relatively small number of function evaluations to converge

(vi) It is fast

The importance components in PSO are the Social Network Structure. The social structure for PSO is determined by the formation of overlapping neighborhoods, where particle within a neighborhood influence one another. Within the PSO, particle in the same neighborhood communicate with one another by exchanging information about the success of each particle in that neighborhood. The performance of the PSO depends strongly on the structure of the social network. The flow of information through a social network depends on (1) the degree of connectivity among nodes of the network, (2) the amount of clustering and (3) the average shortest distance from one node to another [9]. Researchers have investigated how deferent topologies for such a network affect performance [10, 11, 12, 13, 14, 15, 16]. For example, 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 [7]. With these studies, it has become clear that, in the standard PSO, there are three features that bias the particle to look in a better place (the particle remembers its best position, it identifies its best neighbour, and it knows that neighbour's best position so far).

The popularity of PSO in the field of numeric optimization has been increasing since it was created in 1995. It can be applied in the areas of system design, classification, pattern recognition, system modeling, scheduling, planning, signal processing, robotic applications, decision making, simulation and identification. However, 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).

Neural Network (NN), or artificial neural network (ANN) to be more precise, 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. Many applications have been developed using NN algorithm and most of the applications are on predicting future events based on historical data. Processing power in ANN allows the network to learn and adapt, in addition to making it particularly well suited to tasks such as classification, pattern recognition, memory recall, prediction, optimization, and noise filtering [17].

An artificial neural network (NN) is a layered network of artificial neuron (AN). An NN may consist of an input layer, hidden layers and an output layer. ANs in one layer are connected, fully or partially, to the ANs in the next layer. Feedback connections to previous layers are also possible.

Several different NN types have been developed, for example (the reader should note that the list below is by no means complete):

- Single-layer NNs, such as the Hopfield network;
- Multilayer feedforward NNs, including, for example, standard backpropagation, functional link and product unit networks;
- Temporal NNs, such as the Elman and Jordan simple recurrent networks as well as time-delay neural networks;
- Self-organizing NNs, such as the Kohonen self-organizing feature maps and the learning vector quantizer;
- Combined supervised and unsupervised NNs, e.g. some radial basis function networks.

These NN types have been used for a wide range of applications, including diagnosis of diseases, speech recognition, data mining, composing music, image processing, forecasting, robot control, credit approval, classification, pattern recognition, planning game strategies, compression, and many others. Besides this, the primary significance for a NN is the ability of the network to learn from its environment and to improve its performance through learning [18]. Learning is a process of modifying the weights and biases to the neurons and continued until a preset condition is met such as defined error function. Learning process is usually referred as training process in NN. The objective of training process is to classify certain input data patterns to certain outputs before testing with another group of related data.

Hence, the main contribution of this thesis is an investigation of social interaction between the individuals in the particle swarm algorithm. It presents a detailed study of the social structure between the individuals and how it influences the behavior of the algorithm. PSO with difference social network structures is integrated in ANN classifier to validate the effectiveness of incorporating different social network structure.

1.2 Problem Background

In this section, two terms related to the issue are explained: First, Particle Swarm Optimization and Genetic algorithm, and Second is a social network structure in Particle Swarm Optimization learning.

Genetic algorithms (GA) have been popular because of the parallel nature of their search and essentially because of their ability to effectively solve non-linear, multi-modal problems. They can handle both discrete and continuous variables without requiring gradient information. In comparison, PSO is well-known for its easy implementation, its computational inexpensiveness and its fast convergence to optimal areas of the solution space. Although it yields its best performance on continuous-valued problems, it can also handle discrete variables after slight modifications.

The convenience of realization and promising optimization ability in various problems, PSO algorithm has been paid more and more attention to by researchers [34]. PSO and GA for excess return evaluation in stock market [35]. Based on their experiment, it is proven that PSO algorithm is better compared to GA. PSO can reach the global optimum value with less iteration, keep equilibrium versus GA and shows the possibility to solve the complicated problem using only basic equation without crossover, mutation and other manipulation as in GA. The application for stock trading using PSO done and it shows good profit accumulation results [36].

Besides that, many researchers have compared both optimization techniques over the last years. [16] had compared these algorithms by applying to the atomic cluster optimization problem. The task consists of minimizing a highly multi-modal energy function of clusters of atoms. The results illustrated that PSO was noticeably superior to both a generic GA and a purpose-built problem-specific GA. [17] had found both techniques by training ANN to control virtual racecars. Due to the continuity of the neural weights being optimized, PSO turned out to be superior to GA for all accomplished tests - yielding a higher and much faster growing mean fitness and has disclosed that both techniques with a set of eight well-known optimization benchmark test problems were equally effective and more efficient in general [18]. Nevertheless, the superiority of PSO turned out to be problem-dependent. The difference in computational efficiency was found to be greater when the search strategies were used to solve unconstrained problems with continuous variables and less when they were applied to constrain continuous or discrete variables. Furthermore,

[19] implemented both approaches by identifying two mathematical model parameters. Although he noted that both approaches were equally effective, but he concluded that GA outperformed PSO with regard to efficiency. However, attention should be paid to Jones' PSO variant, as it used the same random variables for all dimensions during one velocity update - which turned out to perform worse than using different variables for each dimension as proposed originally. This could be the reason for the poor results for PSO presented by Jones. [20] has analyzed the abilities of PSO and GA to optimize dual-band planar antennas for mobile communication applications. They found that PSO was able to obtain slightly better results than GA, but that PSO took more CPU-time.

Table 1 briefly described from several researchers of the higher efficiency generally ascribed to PSO comparison with GA.

Table 1: higher efficiency generally ascribed to PSO comparison with GA.

Title/ Researcher /Year	Brief Description
Solving constrained onlinear optimization problem with Particle Swarm Optimization. [24]	This paper shows the investigated Particle Swarm Optimization (PSO) algorithm for constrained nonlinear optimization problem. PSO is started with a group of feasible solutions and a feasibility function is used to check if the new explored solutions satisfy all the constraints. Eleven test cases were tested and showed that PSO is an efficient and general solution to solve optimization problems.
A comparison of	Particle Swarm Optimization (PSO) and the GA move from

Particle Swarm Optimization and the Genetic Algorithm. [25]	a set of points (population) to another set of points in a single iteration with likely improvement using a combination of deterministic and probabilistic rules. The drawback of the GA is its expensive computational cost. It appears that PSO outperforms the GA with a larger differential in computational efficiency when used to solve unconstrained nonlinear problems with continuous design variables and less efficiency differential when applied to constrained nonlinear problems with continuous or discrete design variables.
Improvement of Genetic Algorithm Using PSO and Euclidean Data Distance. [28]	This paper introduces improvement of Genetic Algorithm using PSO and Euclidean Data Distance. When obtain an optimal solution using GA (Genetic Algorithm), operation such as crossover, reproduction, and mutation procedures is using to generate for the next generations. In this case, it is possible to obtain local solution because chromosomes or individuals which have only a close affinity can convergent. The result show applies PSO (Particle Swarm Optimization) to have a faster convergence.
Applying Particle Swarm Optimization to Software Testing. [26]	A research is conducted by Andreas Windisch of evolutionary structural testing is an approach to automatically generating test cases that achieve high structural code coverage. It typically uses genetic algorithms (GAs) to search for relevant test cases. In recent investigations particle swarm optimization (PSO), an alternative search technique, often outperformed GAs when applied to various problems. The results show that PSO outperforms GAs for most code elements to be covered in

	terms of effectiveness and efficiency.
Dynamic Model Updating Using Particle Swarm Optimization Method. [27]	This paper proposes the use of particle swarm optimization method (PSO) for finite element (FE) model updating. The PSO method is compared to the existing methods that use simulated annealing (SA) or genetic algorithms (GA) for FE model for model updating. The proposed method is tested on an unsymmetrical H-shaped structure. As the result, PSO achieves that accuracy at a computational speed that is faster than that by the GA and a full FE model which is faster than the SA and a full FE model.

In addition, [29] had shown that the structure of the social network can assist the behavior of particle swarms. Those created and test a general model of communication and consensus which puts the details of the dynamics and the optimum seeking behavior of PSOs into the background. The model includes the forms of communication currently implemented in PSOs, but it is significantly more general. [30] had investigated the effects of various population topologies on the particle swarm algorithm systematically. Random graphs were generated to specifications, and their performance on several criteria was compared. The particle swarm algorithm can be described generally as a population of vectors whose trajectories oscillate around a region which is defined by each individual's previous best success and the success of some other particle. [31] proposed the canonical particle swarm algorithm is a new approach to optimization, drawing inspiration from group behavior and the establishment of social norms. It is gaining popularity, especially because of the speed of convergence and the fact that it is easy to use. As well, advocated Particle Swarm Optimization as a novel algorithm where a population of candidate problem solution vectors evolves "social" norm by influencing their topological neighbors [32]. An individual was influenced by its best performance acquired in the past and the best experience observed in its neighborhood. The experimental results show that the topologies influenced in these two variants are important. [33] Modified the mechanism of PSO individual interacts with its neighbors. The performance of an individual depends on population topology as well as algorithm version. It appears that a fully informed particle swarm is more susceptible to alterations in the topology, but with a good topology, it can outperform the canonical version. In the next section, a brief comparison of GA and PSO is given to further discerning their differences.

1.2.1 Comparisons between Genetic Algorithm and PSO

Most of evolutionary techniques have the following procedure:

1. Random generation of an initial population
2. Reckoning of a fitness value for each subject.
3. Reproduction of the population based on fitness values.
4. If requirements are met, then stop. Otherwise go back to 2.

From the procedure, we can learn that PSO shares many common points with GA. Both algorithms start with a group of randomly generated populations, both have fitness values to evaluate the populations. Both update the population and search for the optimum with random techniques. Both systems do not guarantee success. However, PSO does not use genetic operators like crossover and mutation. Particles update themselves with the internal velocity. They also have memory, which is important to the algorithm.

Compared with GAs, the information sharing mechanism in PSO is significantly different. In GAs, chromosomes share information with each other.

So the whole population moves like a single group towards an optimal area. In PSO, only gBest (or lBest) gives out the information to others. It is a one-way information sharing mechanism. The evolution looks only for the best solution. Compared with the GA, all the particles tend to converge to the best solution quickly even in the local version in most cases.

According to the previous works, PSO is relatively recent optimization technique that can yield higher effectiveness.

1.3 Problems Statement

Swarm Intelligence technique called Particle Swarm Optimization is employed to probe the convergence rate and the classification accuracy of social network structure. However, in this study, the investigation of PSO Social Network Structures integrated with Artificial Neural Network will be conducted to attest its effectiveness in classification problem. Hence, the hypothesis of this study can be stated as:

PSO algorithm can enhance the classifications rate by optimizing different Social Network Structure

1.4 Purpose of Study

The purpose of the project is to explore the effectiveness of the integration of PSO with Artificial Neural Network using different social network structures for comparison convergence rate and classification problems. A multilayer ANN architecture with learning parameters will be used for training. Furthermore, the performance is also compared with PSO star structure (*gbest*) and PSO ring structure (*lbest*) for learning parameter on ANN with MSE will also be investigated, i.e., star and ring topologies. The three real classification problems are used to examine the performance of learning parameters for various size of data set.

1.5 Objectives

The purpose of the study is to explore the effectiveness of PSO social network structure in enhancing ANN learning. To achieve this objective, the following tasks must be conducted:-

i. To develop PSO network structure for Artificial Neural Network.
ii. To analyze the implicit association of these social network structure topologies with the ANN performance.
iii. To evaluate the efficiency of these social network structures.
iv. To compare and validate the effectiveness of each PSO social network structure.

1.6 Project Scope

The scope of this project is limited to the following areas:-

i. Three dataset are used: Iris, Breast Cancer Wisconsin and Herberman's Survival.
ii. The investigate comparison program is developed using Microsoft Visual C++ 6.0.
iii. Social network structures of star and ring topologies are used to compare the effectiveness for ANN learning.
iv. Statistical *t-test* is used to validate the significant of the social network structures in assisting the convergence rate of ANN.

1.7 Significance of Project

The study investigates the performance of PSO based ANN learning, in terms of accuracy and convergence rate, for classification problems. The improved-PSONN that utilizes the improved error function with learning parameters of particle *gbest* and *lbest* are developed and its performance is examined. The performance is compared among different PSO social network structure approaches to see which approach can give better and faster convergence for the training and classification accuracy. Furthermore, the result of this study is contributed to identify the effectiveness of these PSO social network structure as alternative optimization procedure for tuning of learning parameters on ANN training for different set of classification data. The

REFERENCES

1 Bonabeau, E., Dorigo, M and Theraulaz, G. (1999). Swarm Intelligence: From Natural to Artificial System. Oxford University Press. New York.

2 EBERHART R.C., SHI Y., “Comparison between genetic algorithms and particle swarm optimization”, *Evolutionary Programming VII, San Diego, 1998.*

3 Y. C. Foo, S. F. Chien, Andy L. Y. Low, and C. F. Teo (2005). New strategy for optimizing wavelength converter placement. *Optical Society of America 2005*. Vol. 13, No. 2, 545-551

4 Eberhart, R. and Hu. X., “Human tremor analysis using particle swarm optimization.” Proceedings of the Congress on Evolutionary Computation, 1999, Washington, DC, pp. 1927-1930.

5 Eberhart, R. and Shi, Y., “Evolving artificial neural networks.” Proceedings of the International Conference on Neural Networks and Brain, 1998, Beijing, P.R.C. PL5-PL13.

6 Kennedy, J., “Small worlds and mega-minds: effects of neighborhood topology on particle swarm performance.” Proceedings of the Congress on Evolutionary Computation, 1999, pp. 1931-1938.

7 Kennedy, J. and Eberhart, R., Swarm Intelligence, Morgan Kaufmann Academic Press, 2001.

8 E. Alfassio Grimaldi, F. Grimaccia, M. Mussetta, R. E. Zich (2004).Comparison Of Two Neural Network Optimization Approaches. *l0th Int, Conf. on Mathematical Methods in Electromagnetic Theory,* 461-463

9 D.J. Watts and S.H.Strogatz. Collective Dynamics of 'Small-World' Network. Nature, 393(6684):440-442, 1998.

10 Kennedy, J., Mendes, R.: Topological structure and particle swarm performance. In Fogel, D.B., Yao, X., Greenwood, G., Iba, H., Marrow, P., Shackleton, M., eds.: Proceedings of the Fourth Congress on Evolutionary Computation (CEC-2002), Honolulu, Hawaii, IEEE Computer Society (2002)

11 Peer, E., van den Bergh, F., Engelbrecht, A.: Using neighborhoods with guaranteed convergence pso. In: IEEE Swarm Intelligence Symposium, Indianapolis (2003) 235–242

12 Mendes, R., Neves, J.: What makes a successful society? experiments with population topologies in particle swarms. In Bazzan, A.L., Labidi, S., eds.: Advances in Artificial Intelligence. XVII Brazilian Symposium on Artificial Intelligence - SBIA'04, So Lus, Maranho – Brazil, Springer (2004) 346–355

13 Mendes, R.: Population Topologies and Their Influence in Particle Swarm Performance. PhD thesis, Escola de Engenharia, Universidade do Minho (2004) Phd Thesis.

14 Mendes, R., Kennedy, J., Neves, J.: The fully informed particle swarm: Simpler, maybe better. IEEE Transactions of Evolutionary Computation 8(3) (2004) 204–210

15 Mendes, R., Kennedy, J., Neves, J.: Avoiding the pitfalls of local optima: How topologies can save the day. In: Proceedings of the 12th Conference Intelligent Systems Application to Power Systems (ISAP2003), Lemnos, Greece, IEEE Computer Society (2003)

16 Kennedy, J., Mendes, R.: Neighborhood topologies in fully-informed and best-of-neighborhood particle swarms. In: 2003 IEEE SMC Workshop on Soft Computing in Industrial Applications (SMCia03), Binghamton, NY, IEEE Computer Society (2003)

17 Luger, George F., (2002). Artificial Intelligence: Structures and Strategies for Complex Problem Solving. 4th ed.. Addison-Wesley/Pearson Education Limited, arlow, England. 417-473.

18 Haykin, S. (1999). Neural Network. A Comprehensive Foundation. 2nd Ed: rentice Hall.

19 R. J. W. Hodgson. Partical swarm optimization applied to the atomic cluster optimization problem. In *GECCO*, pages 68–73, 2002.

20 B. Clow and T. White. An evolutionary race: A comparison of genetic algorithms and particle swarm optimization for training neural networks. In *Proceedings of the International Conference on Artificial Intelligence, IC-AI '04, Volume 2*, pages 582–588. CSREA Press, 2004.

21 R. Hassan, B. Cohanim, and O. de Weck. A comparison of particle swarm optimization and the genetic algorithm. In *Proceedings of the 46th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, 2005.

22 K. O. Jones. Comparison of genetic algorithm and particle swarm optimization. In *Proceedings of the International Conference on Computer Systems and Technologies*, 2005.

23 L. Oliva J. Hor´ak, P. Chmela and Z. Raida. Global optimization of the dual-band planar antenna: Pso versus ga. In *Radioelektronika*, 2006.

24 Xiaohui Hu and Russell Eberhart. Solving constrained Nonlinear Optimization Problems with Particle Swarm Optimization. In *Department of Biomedical Engineering Purdue University, West Lafayette, IN, USA*, 2003

25 Rania Hassan, Babak Cohanim and Olivier de Weck. A comparison of Particle Swarm Optimization and the Genetic Algorithm. In *Massachusetts Institute of Technology, Cambridge, MA, 02139,* 2005

26 Andreas Windisch. Applying Particle Swarm Optimization to Software Testing. *In DaimlerChrysler AG Research and Technology Alt-Moabit 96a, D-10559 Berlin, Germany*, 2007

27 Tshilidzi Marwala, Dynamic Model Updating Using Particle Swarm Optimization Method. In *School of Electrical and Information Engineering University of the Witwatersrand P/Bag 3, Wits, 2050, South Africa,* 2007

28 Dong Hwa Kim, Improvement of Genetic Algorithm Using PSO and Euclidean Data Distance. *Dept. of Instrumentation and Control Eng., Hanbat National University*, 2006

29 Nick Holden, Communication, Leadership, Publicity and Group Formation in Particle Swarms. *Computing Laboratory University of Kent, UK Technical Report CSM-453 Department of Computer Science University of Essex*, May 2006

30 James Kennedy and Rui Mendes, Population Structure and Particle Swarm Performance. Bureau of Labor Statistics Washington, DC. 2004

31 Rui Mendes, Member, IEEE, James Kennedy, and José Neves, Population Structure and Particle Swarm Performance. 2004

32 Rui Mendes, James Kennedy and Jose Neves, Watch Thy Neighbor Or How the Swarm Can Learn From Its Environment. *Departmento de Information Universidala do Minho PORTUGAL*. 2003

33 James Kennedy and Rui Mendes, Neighborhood Topologies in Fully Informed and Best-of-Neighborhood Particle Swarms. *Department of Informatics, University ofMinho, 4704-553 Braga, Portugal.* 2006

34 Song, M. P. and Gu, G.C. (2004). Research on Particle Swarm Optimization: A Review. IEEE: 2236-2241.

35 Lee, J. S., Lee, S., Chang, S. and Ahn, B. H. (2005). A Comparison of GA and PSO for Excess Return Evaluation in Stock Markets. Springer-Verlag: 221-230.

36 Nenortaite, J. and Simutis, R. (2004). Stock Trading System Based on the Particle Swarm Optimization Algorithm. Springer-Verlag: 843-850.

37 Madan, M.G., Liang J. and Noriyasu H. (2003). Static and Dynamic Neural Networks: From Fundamentals to Advanced Theory. Canada: John Wiley & Sons.

38 Van den Bergh, F. (1999). Particle Swarm Weight Initialization In Multi-Layer Perceptron Artificial Neural Networks. Accepted for ICAI. Durban, South Africa.

39 Alba, E. and Chicana, J. F. (2004). Training Neural Networks With GA Hybrid Algorithms. Proc. of GECCO'04. Seattle, Washington. LNCS 3102. 2004. 852-863.

40 Zhang, C., Shao, H. and Li, Y. (2000). Particle Swarm Optimization for Evolving Artificial Neural Network. IEEE: 2487-2490.

41 Ben Kröse, Patrick van der Smagt (1996). An Introduction to Neural Networks. The University of Amsterdam.

42 Bonabeau, E., Dorigo, M and Theraulaz, G. (1999). Swarm Intelligence: From Natural to Artificial System. Oxford University Press. New York.

43 Bonabeau, E., Dorigo, M., and Theraulaz, G. (1999). Swarm Intelligence : From Natural to Arti cial Systems. Oxford University Press,

44 Shi, Y. (2004). Particle Swarm Optimization. IEEE Neural Network Society: 8-1 Eberhart, R. and Shi, Y. (2001). Particle Swarm Optimization: Developments, Application and Resources. IEEE: 81-86.

45 Eberhart, R. and Shi, Y. (2001). Particle Swarm Optimization: Developments, Application and Resources. IEEE: 81-86.

46 Jones M.T (2005). AI Application Programming. 2nd Ed. Hingham, Massachusetts: Charles River Media Inc.

47 Al -kazemi, B. and Mohan, C.K. (2002). Training Feedforward Neural Network Using Multi -phase Particle Swarm Optimization. Proceeding of the 9th International Conference on Neural Information Processing. New York.

48 D.J. Watts and S.H. Strogatz. Collective Dynamics of 'Small-World' Networks. Nature, 393(6684):440–442, 1998.

49 R.C. Eberhart, P.K. Simpson, and R.W. Dobbins. Computational Intelligence PC Tools. Academic Press Professional, first edition, 1996.

50 J. Kennedy. Small Worlds and Mega-Minds: Effects of Neighborhood Topology on Particle Swarm Performance. In Proceedings of the IEEE Congress on Evolutionary Computation, volume 3, pages 1931–1938, 1999.

51 J. Kennedy and R. Mendes. Population Structure and Particle Performance. In Proceedings of the IEEE Congress on Evolutionary Computation, pages 1671–1676, 2002.

52 R. Mendes, P. Cortez, M. Rocha, and J. Neves. Particle Swarms for Feedforward Neural Network Training. In Proceedings of the International Joint Conference on Neural Networks, pages 1895–1899, 2002.

53 E.S. Peer, F. van den Bergh, and A.P. Engelbrecht. Using Neighborhoods with the Guaranteed Convergence PSO. In Proceedings of the IEEE Swarm Intelligence Symposium, pages 235–242, 2003.

54 P.N. Suganthan. Particle Swarm Optimiser with Neighborhood Operator. In Proceedings of the IEEE Congress on Evolutionary Computation, pages 1958–1962, 1999.

55 T. Krink, J.S. Vesterstrøm, and J. Riget. Particle Swarm Optimisation with Spatial Particle Extension. In Proceedings of the Fourth Congress on Evolutionary Computation, volume 2, pages 1474–1479, 2002.

56 Kennedy, J. (1999). Small worlds and mega-minds: Effects of neighborhood topology on particle swarm performance. In Proceedings of the 1999 Conference on Evolutionary Computation, pages 1931$^{\perp}$ 1938.

57 Mohankrishnan, N., Lee, W.S. and Paulik, M.J.(1996). Multi - layer neural network classification of on- line signatures. IEEE 39th Midwest symposium on Volume 2, 18-21 August 1996: 831 - 834.

58 Wu C.H., Epidemiology A. and Chang T.C. (1991). Protein classification using a neural network database system. Proceedings of the conference on Analysis of neural network applications. Fairfax, Virginia, United States: 29 - 41

59 Saeed Khan, M.K., Al -Khatib, W.G. and Moinuddin, M. (2004). Automatic classification of speech and music using neural networks. Proceedings of the 2nd ACM international workshop on Multimedia databases. Washington DC, USA: 94 - 99.

60 D.J. Watts and S.H. Strogatz. Collective Dynamics of ‘Small-World’ Networks.Nature, 393(6684):440–442, 1998.

61 Song, M. P. and Gu, G.C. (2004). Research on Particle Swarm Optimization: A Review. IEEE: 2236-2241.

62 Kim, G-H., Yoon, J-E., An, S-H., Cho, H-H. and Kang, K-I. (2004). Neural Network Model Incorporating A Genetic Algorithm in Estimating Construction Cost. Building and Environment. 39(11): 1333-1340.

63 Siti Mariyam Hj. Shamsuddin (2004). Lecture Note Advanced Artificial Intelligence: Number of Hidden Neurons. Universiti Teknologi Malaysia. Unpublished.

64 Charytoniuk, W. and Chen, M.S (2000). Neural Network Design for Short-term Load Forecasting. International Conference on Electric Utility Deregulation and Restructuring and Power Technologies 2000. 4-7 April 2000. City University, London, 554-561.

65 Van den Bergh, F. and Engelbrecht, A. P. (2000). Cooperative Learning in Neural Networks using Particle Swarm Optimizers. South African Computer Journal. (26):84-90.

66 Shi, Y. (2004). Particle Swarm Optimization. IEEE Neural Network Society: 8-13.

67 Haza Nuzly Bin Abdull Hamed (2006). *Particle Swarm Optimization for Neural Network Learning Enhancement,* Universiti Teknologi Malaysia. Unpublished.

68. Timothy Masters. *Practical Neural Network Recipes in C++*. Morgan Kaufmann is an imprint of Academic Press (81-85) (1993).

69 J. Kennedy and R.C. Eberhart. Particle Swarm Optimization. In Proceedings of the IEEE International Joint Conference on Neural Networks, pages 1942–1948, 1995.

70 M. Clerc and J. Kennedy. The Particle Swarm-Explosion, Stability, and Convergence in a Multidimensional Complex Space. IEEE Transactions on Evolutionary Computation, 6(1):58–73, 2002.

71 I.C. Trelea. The Particle Swarm Optimization Algorithm: Convergence Analysis and Parameter Selection. Information Processing Letters, 85(6):317–325, 2003.

72 F. van den Bergh. An Analysis of Particle Swarm Optimizers. PhDthesis, Department of Computer Science, University of Pretoria, Pretoria, South Africa, 2002.

73 F. van den Bergh and A.P. Engelbrecht. A Study of Particle Swarm Optimization Particle Trajectories. Information Sciences, 176(8):937–971, 2006.

74 Paulo F. Ribeiro, W.Kyle Schlansker, *Member, IEEE*. A Hybrid Particle Swarm and Neural Network Approach for Reactive Power Control, 2006

75 Mohamad Firdaus bin Ab Aziz (2006), Enhancement of Elman Recurrent Network Learning with Particle Swarm Optimization, Universiti Teknologi Malaysia. Unpublished.

76 Andries P.Engelbrecht, Fundamentals of Computational Swarm Intelligence, 2005.