ACSIJ Advances in Computer Science: an International Journal, Vol. 3, Issue 1, No.7 , January 2014
ISSN : 2322-5157
www.ACSIJ.org

Island Model based Differential Evolution Algorithm for Neural Network Training

Htet Thazin Tike Thein

University of Computer Studies, Yangon
Yangon, Myanmar
htztthein@gmail.com

Abstract

There exist many approaches to training neural network. In this system, training for feed forward neural network is introduced by using island model based differential evolution. Differential Evolution (DE) has been used to determine optimal value for ANN parameters such as learning rate and momentum rate and also for weight optimization. Island model used multiple subpopulations and exchanges the individual to boost the overall performance of the algorithm. In this paper, four programs have developed; Island Differential Evolution Neural Network (IDENN), Differential Evolution Neural Network (DENN), Genetic Algorithm Neural Network (GANN) and Particle Swarm Optimization with Neural Network (PSONN) to probe the impact of these methods on ANN learning using various datasets. The results have revealed that IDENN has given quite promising results in terms of convergence rate smaller errors compared to DENN, PSONN and GANN.

Keywords: *Artificial neural network, Island Model, Differential Evolution, Particle Swarm Optimization, Genetic Algorithm.*

1. Introduction

A neural network is a computing system made up of a number of simple, interconnected processing neurons or elements, which process information by its dynamic state response to external inputs [1]. The development and application of neural networks are unlimited as it spans a wide variety of fields. This could be attributed to the fact that these networks are attempts to model the capabilities of of human. It had successfully implemented in the real world world application which are accounting and finance [2,3], [2,3], health and medicine [4,5], engineering and manufacturing [6,7], marketing [8,9] and general applications [10,11,12]. Most papers concerning the use of of neural networks have applied a multilayered, feed-forward, fully connected network of perceptions [13, 14]. Reasons for the use of simple neural networks are done by the simplicity of the theory, ease of programming, good results and because this type of NN represents an universal function in the sense that if the topology of the network is allowed to vary freely it can take take the shape of any broken curve [15]. Several types of learning algorithm have been used for neural network in the the literature.

The DE algorithm is a heuristic algorithm for global optimization. It was introduced several years ago (in 1997) and has been developed intensively in recent years [16]. Its advantages are as follows: the possibility of finding the global minimum of a multimodal function regardless of the initial values of its parameters, quick convergence, and the small number of parameters that needs to be set up at the start of the algorithm's operation [17].

Differential evolution is a relatively new global search and optimization algorithm that is suitable for the real variable optimization. It used the vector difference and elite selection for the selection process and have a relatively few parameter compared to other evolutionary algorithm. Neural network weight can be trained or optimized using differential evolution. Island based model works by running multiple algorithms and shares the results at regular interval promoting the overall performance of the algorithm. This system will propose the island based differential evolution algorithm for training feed forward neural network.

2. Literature Review

The most widely used method of training for feed forward ANNs is back propagation (BP) algorithm [18]. Feed forward ANNs are commonly used for function approximation and pattern classifications. Back propagation algorithm and its variations such as Quick Prop [19] and RProp [20] are likely to reach local minima especially in case that the error surface is rugged. In addition, the efficiency of BP methods depends on the selection of appropriate learning parameters. The other training methods for feed forward ANNs include those that are based on evolutionary computation and heuristic principles such as Differential Evolution (DE), Genetic Algorithm (GA), and Particle Swarm Optimization (PSO).

2.1 Artificial Neural Network (ANN)

An Artificial Neural Network, often just called a neural network, is a mathematical model inspired by biological neural networks. A neural network consists of an interconnected group of artificial neurons, and it processes information using a connectionist approaches to computation. In most cases a neural network is an adaptive system that changes its structure during a learning phase. Neural networks are used to model complex relationships between inputs and outputs or to find patterns in data.

2.2 Differential Evolution (DE)

Differential evolution (DE) algorithm is a simple evolutionary algorithm that creates new candidate solutions by combining the parent individual and several other individuals of the same population. A candidate replaces the parent only if it has better fitness. This is rather greedy selection scheme that often outperforms the traditional evolutionary algorithm. In addition, DE is a simple yet powerful population based, direct search algorithm with the generation and test feature for globally optimizing functions using real valued parameters. Among DE's advantages are its simple structure, ease of use, speed and robustness. Due to these advantages, it has many real-world applications. DE starts with random population as like other evolutionary algorithm. Solutions are encoded using chromosomes, for neural network training, weight of neural network are encodes in the chromosome. For each iteration of DE, fitness of each chromosome is evaluated. Fitness determines the quality of solution or chromosomes. For training neural network, fitness function is generally MSE (mean square error) of neural network. Each chromosome undergoes mutation and crossover operation to produce trial vector. Each fitness of trial vector is compared with the parent vector and the one with greater fitness survived and the next generations begin.

2.3 Particle swarm optimization (PSO)

Particle swarm optimization (PSO) [21] [22] is a stochastically global optimization method that belongs to the family of Swarm Intelligence and Artificial Life. Similar Similar to artificial neural network (ANN) and Genetic Algorithms (GA) [23][24] which is the simplified models models of the neural system & the natural selections of the the evolutionary theory, PSO is based on the principles that that flock of birds, school of fish, or swarm of bee's searches for food sources where at the beginning the perfect perfect location is not known. However, they eventually eventually they reach the best location of food source by means of communicating with each other.

2.4 Genetic Algorithm

Genetic algorithms are stochastic search techniques that guide a population of solutions towards an optimum using the principles of evolution and natural genetics. In recent years, genetic algorithms have become a popular optimization tool for many areas of research, including the field of system control, control design, science and engineering. Significant research exists concerning genetic algorithms for control design and off-line controller analysis.

Genetic algorithms are inspired by the evolution of populations. In a particular environment, individuals which better fit the environment will be able to survive and hand down their chromosomes to their descendants, while less fit individuals will become extinct. The aim of genetic algorithms is to use simple representations to encode complex structures and simple operations to improve these structures. Genetic algorithms therefore are characterized by their representation and operators. In the original genetic algorithm an individual chromosome is represented by a binary string. The bits of each string are called genes and their varying values alleles. A group of individual chromosomes are called a population. Basic genetic operators include reproduction, crossover and mutation [28]. Genetic algorithms are especially capable of handling problems in which the objective function is discontinuous or non differentiable, non convex, multimodal or noisy. Since the algorithms operate on a population instead of a single point in the search space, they climb many peaks in parallel and therefore reduce the probability of finding local minima.

2.5 Island Model (IM)

An island model (IM) is an approach to distribute EA. It divides individuals into subpopulations and allows for occasional exchange of individuals (migrations). The simplest island mode assumes the same global parameters for islands and the same global parameters for migrations. Populations are characterized by their number, size and the evolutionary algorithm type. Migrations are described by the topology.

3. Island Model based Differential Evolution Algorithm (IDE)

The DE algorithm was proposed by Price and Storn [25]. The DE algorithm has the following advantages over the traditional genetic algorithm: it is easy to use and it has efficient memory utilization, lower computational complexity (it scales better when handling large problems), and lower computational effort (faster convergence) [26]. DE is quite effective in nonlinear constraint optimization and is also useful for optimizing multimodal problems [27].
Its pseudocode form is as follows:

a) Create an initial population consisting of *PopSize* individuals
b) While (termination criterion is not satisfied) Do Begin
c) For each *i*th individual in the population Begin
d) Randomly generate three integer numbers: $r_1,r_2,r_3 \in [1;PopSize]$, where $r_1 \neq r_2 \neq r_3 \neq i$
e) For each *j*th gene in *i*th individual ($j \in [1;n]$) Begin
   $v_{i,j} = x_{r1,j} + F \bullet (x_{r2,j} - x_{r3,j})$
f) Randomly generate one real number randj∈[0; 1)
g) If $rand_j < CR$ then $u_{i,j} := v_{i,j}$
   Else $u_{i,j} := x_{i,j}$
   End;
h) If individual u_i is better than individual x_i then replace individual x_i by child u_i individual
   End;
   End;

3.1. Migration Topology

There are four types of migration topology. They are ring, torus, random and fully connected topology. This system investigates the ring topology.

3.2. Migration Strategy

A migration strategy consists of two parts. The first part is the selection of individuals, which shall be migrated to another island. The second part is to choose which individuals are replaced by the newly obtained individuals. Four migration strategies are common:

- Select the best individuals replace the worst individuals.
- Select random individuals, replace the worst individuals.
- Select the best individuals replace random individuals.
- Select random individuals, replace random individuals.

This system experiments the best individuals replace the worst individuals.

3.3. Migration Interval

In order to distribute information about good individuals among the islands, migration has to take place. This can either be done in synchronous way every nth generation or in an asynchronous way, meaning migration takes place at non-periodical times. It is commonly accepted that a more frequent migration leads to a higher selection pressure and therefore a faster convergence. But as always with a higher selection pressure comes the susceptibility to get stuck in local optima. In this system, various migration intervals will be experimented to find the best solution for the neural network training.

3.4. Migration Size

A further important factor is the number of individuals which are exchanged. According to these studies the migration size has to be adapted to the size of a subpopulation of an island. When one migrates only a very small percentage, the influence of the exchange is negligible but if too much individuals are migrated, these new individuals take over the existing population, leading to a decrease of the global diversity. In this system, migration size will also be investigated which can yield the best performance.

4. System Design

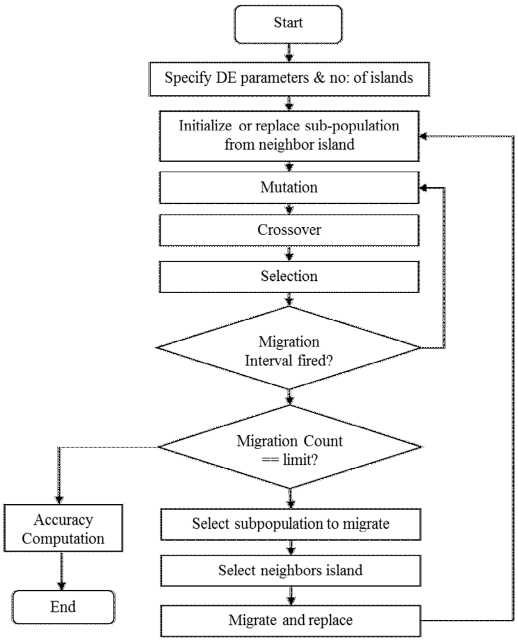

Fig. 1 System Flow Diagram

Island model used different subpopulation with each own island. Each island operates its own execution as like in DE algorithm. Each island initializes the population at the start of the algorithm or replace the subpopulation migrates from other neighbor. Mutation, crossover and selection are performed on the individual chromosome. If the migration interval is not fired, the next iteration begins within island, otherwise, a portion of its own population and neighbor is selected for migration. If the migration occurs, island sends sub-population to neighbor island. Neighbor island replaces the sub-population send by its neighbor and replace with its portion of population and algorithm continue.

5. EXPERIMENTAL RESULT

Currently the system experiment the island model with simple ring topology, migration strategy select the best individuals replace the worst individuals. The island model used the iteration as the migration interval and one-third of the old population is used to migrate and replace. Learning rate of this system is set to 0.01. Four programs have been developed: Island Differential Evolution Feed Forward Neural Network (IDENN), Differential Evolution Feed Forward Neural Network (DENN), Particle Swarm Optimization Feed Forward Neural Network (PSONN) and Genetic Algorithm Feed Forward Neural Network (GANN) using four dataset: XOR, Cancer, heart and Iris. The results for each dataset are compared and analyzed based on the convergence rate and classification performance. All algorithm are run for different numbers of iteration, among of them, MSE (Mean Square Error) of Island DE is much lower than other algorithms.

5.1 Results on XOR Dataset

Table 1: Result of IDENN, DENN, PSONN and GANN on XOR Dataset

	IDENN	DENN	PSONN	GANN
Learning Iteration	20	41	51	61
Error Convergence	0.003	0.0048865	0.00473763	0.04125
Convergence Time	4 sec	7 sec	12 sec	37 sec
Classification (%)	99.2	98.97	95.17	85.66

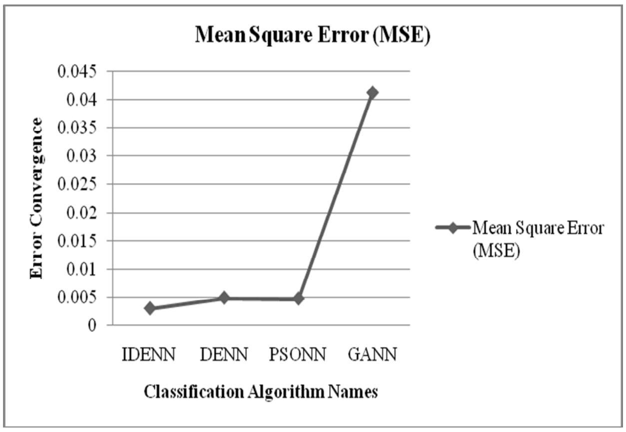

Fig. 2 MSE on XOR dataset

5.2 Results on Cancer Dataset

Table 2: Result of IDENN, DENN, PSONN and GANN on Cancer Dataset

	IDENN	DENN	PSONN	GANN
Learning Iteration	200	443	219	10000
Error Convergence	0.00201	0.00499	0.004870	0.50049

Convergence Time	103 sec	195 sec	110 sec	273 sec
Classification (%)	99.01	98.40	98.65	97.73

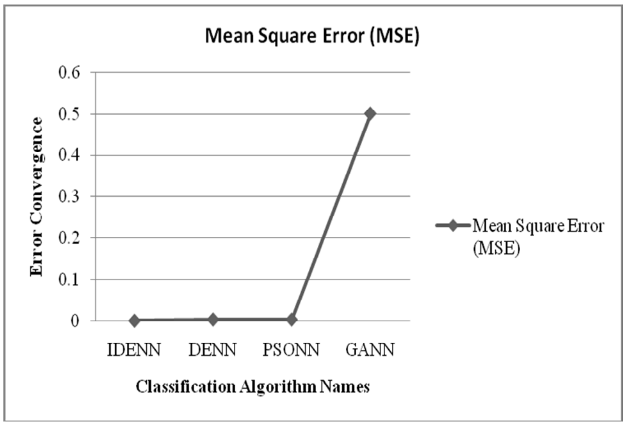

Fig. 3 MSE on Cancer dataset

5.3 Results on Iris Dataset

Table 3: Result of IDENN, DENN, PSONN and GANN on Iris Dataset

	IDENN	DENN	PSONN	GANN
Learning Iteration	28	61	818	10000
Error Convergence	0.0205	0.049803	0.049994	1.88831
Convergence Time	5 sec	16 sec	170 sec	256sec
Classification (%)	96.39	95.014972	93.86	97.72

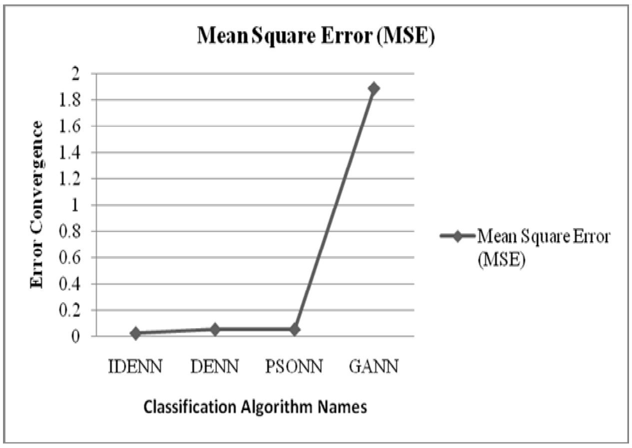

Fig. 4 MSE on Iris dataset

5.4 Results on Heart Dataset

Table 4: Result of IDENN, DENN, PSONN and GANN on Heart Dataset

	IDENN	DENN	PSONN	GANN
Learning Iteration	40	58	10000	9000
Error Convergence	0.039	0.048925	1.46392	3.00
Convergence Time	7 sec	16 sec	170 sec	110 sec
Classification (%)	88.93	85.50	89.56	92.83

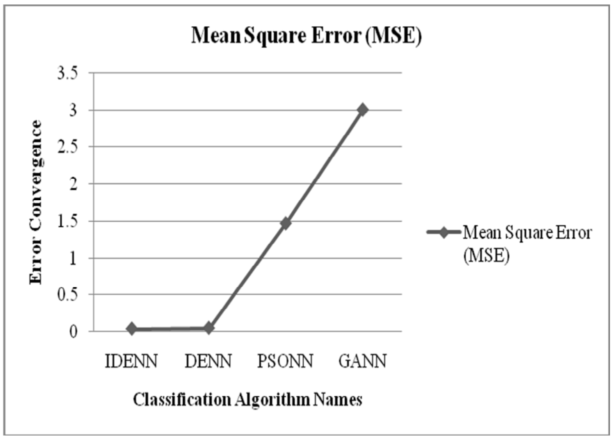

Fig. 5 MSE on Heart dataset

6. Comparison IDENN, DENN, PSONN and GANN

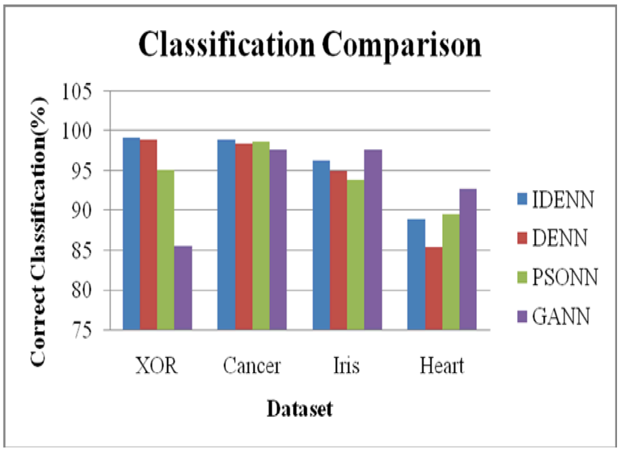

Fig. 6 Comparison of correct classification Percentage IDENN, DENN, PSONN and GANN

For XOR dataset, the results show that IDENN has better results on convergence time and correct classification percentage. IDENN convergence in a short time with high correct classification percentage. For Cancer dataset, IDENN classification results are better than DENN, PSONN and GANN. For Iris dataset, GANN classification results are better than IDENN, DENN, and PSONN. For Heart dataset, GANN classification results are better than IDENN, DENN, and PSONN. For overall performance, the experiments show that IDENN significantly reduces the error with minimum iterations. IDENN produces feasible results in terms of convergence time and classification percentage.

7. Conclusion

This system presents the neural network training algorithm using island model based differential algorithm. By exploiting the global search power of differential evolution algorithm in conjunction with island model will boost the training performance of the algorithm. The system will converge quickly to the lower mean square error. Island model encourage the diversity among the individual among islands which increase search capability and by migration island model can share the best experiences of each other. By using island model rather than single DE, it can get advantages from parallel problem solving and information sharing which lead to faster global search.

References

[1] A. Majdi, M. Beiki: Evolving Neural Network Using Genetic Algorithm for Predicting the Deformation Modulus of Rock Masses. International Journal of Rock Mechanics and Mining Science, vol. 47(2), pp. 246-253, 2010.
[2] Lee, K., Booth, D., and Alam, P.: A Comparison of Supervised and Unsupervised Neural Networks in Predicting Bankruptcy of Korean Firms. Expert Systems with Applications, vol. 29(1), pp.1–16, 2005.
[3] Landajo, M., Andres, J. D., and Lorca, P.: Robust Neural Modeling for the Cross-Sectional Analysis of Accounting Information. European Journal of Operational Research, vol. 177(2), pp. 1232–1252, 2007.
[4] Razi, M. A., and Athappily, K.: A Comparative Predictive Analysis of Neural Networks (NNs), Nonlinear Regression and Classification and Regression Tree (CART) Models. Expert Systems with Applications, vol. 2(1), pp. 65–74, 2005.
[5] Behrman, M., Linder, R., Assadi, A. H., Stacey, B. R., and Backonja, M. M.: Classification of Patients with Pain Based on Neuropathic Pain Symptoms: Comparison of an Artificial Neural Network against an Established Scoring System. European Journal of Pain, vol. 11(4), pp. 370–376, 2007.
[6] Yesilnacar, E., and Topal, T.: Landslide Susceptibility Mapping: A Comparison of Logistic Regression and Neural Networks Methods in a Medium Scale Study, Hendek region (Turkey). Engineering Geology, vol. 79 (3–4), pp. 251–266, 2005.
[7] Dvir, D., Ben-Davidb, A., Sadehb, A., and Shenhar, A. J.: Critical Managerial Factors Affecting Defense Projects Success: A Comparison between Neural Network and Regression Analysis. Engineering Applications of Artificial Intelligence, vol. 19, pp. 535–543, 2006.
[8] Gan, C., Limsombunchai, V., Clemes, M., and Weng, A.: Consumer Choice Prediction: Artificial Neural Networks Versus Logistic Models. Journal of Social Sciences, vol. 1(4), pp. 211–219, 2005.
[9] Chiang, W. K., Zhang, D., and Zhou, L.: Predicting and Explaining Patronage Behavior toward Web and Traditional Stores Using Neural Networks: A Comparative Analysis with Logistic Regression. Decision Support Systems, vol. 41, pp. 514–531, 2006.
[10] Chang, L. Y.: Analysis of Freeway Accident Frequencies: Negative Binomial Regression versus Artificial Neural Network. Safety Science, vol. 43, pp. 541–557, 2005.
[11] Sharda, R., and Delen, D.: Predicting box-office success of motion pictures with neural networks. Expert Systems with Applications, vol. 30, pp. 243–254 , 2006.
[12] Nikolopoulos, K., Goodwin, P., Patelis, A., and ssimakopoulos, V.: Forecasting with cue information: A comparison of multiple regressions with alternative forecasting approaches. European Journal of Operational Research, vol. 180(1), pp. 354–368, 2007.
[13] S. Curteanu and M. Cazacu.: Neural Networks and Genetic Algorithms Used For Modeling and Optimization of the Siloxane - Siloxane Copolymers Synthesis. Journal of Macromolecular Science, Part A, vol. 45, pp. 123–136, 2007.
[14] C. Lisa and S. Curteanu: Neural Network Based Predictions for the Liquid Crystal Properties of Organic Compounds.

Computer-Aided Chemical Engineering, vol. 24, pp. 39–45, 2007.

[15] Fernandes and L.M.F. Lona: Neural Network Applications in Polymerization Processes, Brazilian Journal Chemical Engineering, vol. 22, pp. 323–330, 2005.

[16] K. V. Price, R. M. Storn, and J. A. Lampinen, Differential Evolution: APractical Approach to Global Optimization. Berlin, Germany: Springer Verlag, 2005.

[17] R. L. Becerra and C. A. Coello Coello, “Cultured differential evolution for constrained optimization,” Comput. Methods Appl. Mech. Eng., vol. 195,no. 33–36, pp. 4303–4322, Jul. 1, 2006.

[18] Hecht Nelso R., back propagation neural network, 1989.

[19] Scott E. Fahlman: An Empirical Study of Learning Speed in Back-Propagation Networks, September 1988.

[20] Rprop – Description and Implementation Details Martin Riedmiller, Technical report, 1994.

[21] Kennedy, J.; Eberhart, R. "Particle Swarm Optimization". Proceedings of IEEE International Conference on Neural Networks, 1995.

[22] Kennedy, J.; Eberhart, R.C. Swarm Intelligence. Morgan Kaufmann, 2001.

[23] Zhang, G.P., Neural networks for classification: a survey. IEEE Transactions on Systems Man and Cybernetics, 2000.

[24] Rudolph, G., Local convergence rates of simple evolutionary algorithms with cauchy mutations, 1997.

[25] R. Storn and K. Price, “Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces,”J. Glob. Optim. vol. 11, no. 4, pp. 341–359, Dec. 1997.

[26] K. V. Price, R. M. Storn, and J. A.Lampinen, Differential Evolution—A Practical Approach to Global Optimization. Berlin, Germany: SpringerVerlag, 2005.

[27] D. Karaboga and S. Okdem, “A simple and global optimization algorithm for engineering problems: Differential evolution algorithm,” Turkish J. Elect. Eng. Comput. Sci., vol. 12, no. 1, pp. 53–60, 2004.

[28] D. Whitley, "Applying Genetic Algorithms to Neural Network Problems," International Neural Network Society p. 230, 1988.

Htet Thazin Tike Thein received her B.C.Tech. degree in Computer Technology from University of Computer Studies in Yangon Myanmar and M.C.Tech. degree in Computer Technology from University of Computer Studies in Yangon Myanmar. She is currently a student of Ph.D. degree at University of Computer Studies in Yangon Myanmar. She is currently been working toward her Ph.D. degree in Neural Network and classification.