Modeling of Chaotic Behavior of Benchmark Datasets using Hybrid Heuristic Optimization

Urvashi Dhawale
Department of Electronics and Telecommunication Engineering,
S. B. Jain Institute of Technology, Management & Research,
Nagpur (M.S.) India.
dhawaleurvashi31@gmail.com,

Pravin R. Kshirsagar
Department of Electronics Engineering,
Rajiv Gandhi College of Engineering, Research and Technology,
Gondwana University, Chandrapur (M.S),India.
pravinrk88@yahoo.com,

Sudhir G.Akojwar
Associate Professor,
Department of Instrumentation Engineering,
Government College of Engineering,
Gondwana University, Chandrapur (M.S), India.
sudhirakojwar@gmail.com,

Pratik R. Hajare
Department of Electronics and Telecommunication Engineering,
S. B. Jain Institute of Technology, Management & Research, Nagpur (M.S.) India.
pratik_hajare74@rediffmail.com,

Abstract— Optimization is required for producing the best results. Heuristic algorithm is one of the techniques which can be used for finding best results. By making use of artificial neural network and particle swarm optimization values can be predicted and chaotic signals can be modeled which forms the base of this project. The chaotic signals here use are Mackey series and Box Jenkins Gas Furnace data series. The results of this work shows the comparative study of predicted number of neurons in the second hidden layer also it gives the value of mean square error while making the prediction.

Keywords—*ANN, BP, FFNN, PSO, GA, MSE.*

*****

I. Introduction

Classification of chaotic signals and their predictions forms the base of many research technologies[9]. Such technologies are helpful for mankind in various ways. They find application in technical as well as biomedical engineering streams. Certain unknown phenomena like natural calamities with unpredicted behavior are harmful to nature and such chaotic behavior can be previously predicted by using various modeling techniques. One of the techniques is hybrid heuristic optimization which incorporated modeling of chaotic signals. Last one value of the chaotic series can be predicted by making use of particle swarm optimization (PSO) which is trained using feed forward neural network. By observing that no previous research work is available for performing modeling of chaotic signals this work becomes very beneficial.

A heuristic algorithm can solve a problem in a faster and more efficient way as compared to the traditional methods by sacrificing optimality, accuracy, precision, or completeness for speed. Heuristic algorithms are often times used to solve NP-complete problems which are a class of decision problems. In case of NP-problems there is no such method known for finding out the solution but the solution can be predicted correctly if some information related to the problem is provided. Heuristics can be combined with other optimization algorithms to find good solutions for specific problems or they can be use individually also. Heuristic algorithms are mostly used when approximate solutions are adequate and expensive set up is required for finding out meticulous solutions.

The need of prediction can be justified by the explanation given below. Today, the most used tool for classification and prediction of data is neural network. When the solution represents the network topological information but not the weight values, a network with a full set of weights must be used to calculate the training error for the cost function. This is often done by performing a random initialization of weights and by training the network using one of the most commonly used learning algorithms, such as back propagation. This strategy may lead to noisy fitness evaluation, since different weights and biases as starting point initializations and training parameters can produce different results for the same topology (Eberhart and Shi, 2000; Clerc and Kennedy, 2002).

In order to implement an efficient and appropriate optimization technique for acquiring correct results, the neural network is so trained. Effective training algorithm and better-understood system behavior are the advantages of this type of neural network. Selection of network input parameters and performance of classifier are important in epileptic seizure detection. The efficiency of this technique can be explained by using the result of experiments. This project clearly

ISSN: 2454-4248
189 – 193

demonstrates that this method is applicable for detecting epileptic seizure. The qualities of the method are, it is simple to apply, and it does not require high computation power.

II. GENETIC ALGORITHM

The genetic algorithm is Search and optimization techniques that generate solutions to optimization problems using techniques inspired by natural evolution[8]. Optimization is essential to any problem involving whether it is engineering or economics. [1] Genetic Algorithms were first introduced by John Holland in 1970. Genetic algorithms also implement the optimization strategies by simulating evolution of species through natural selections. Genetic algorithm is generally composed of two processes. First process is selection of individual for the production of next generation and second process is manipulation of the selected individual to form the next generation by crossover and mutation techniques [2]. The selection mechanism determines which individual are chosen for reproduction and how many offspring each selected individual produce. The main principle of selection strategy is the better is an individual; the higher is its chance of being parent. Generally, crossover and mutation explore the search space, whereas selection reduces the search area within the population by discarding poor solutions. However, worst individuals should not be discarded and they have some chances to be selected because it may lead to useful genetic material. A good search technique must find a good trade-off between exploration and exploitation in order to find a global optimum. Hence, it is important to find a balance between exploration (i.e. poor solutions must have chance to go to the next generation) and exploitation (i.e. good solutions go to the next generation more frequently than poor solutions) within the mechanism of the selection.

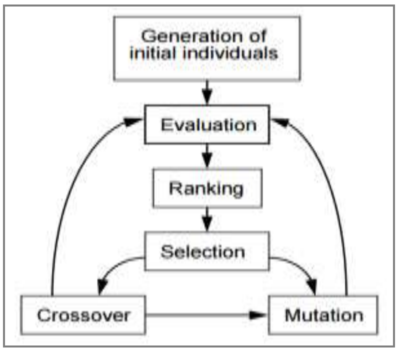

Figure 1.Principle Structure of A Genetic Algorithm

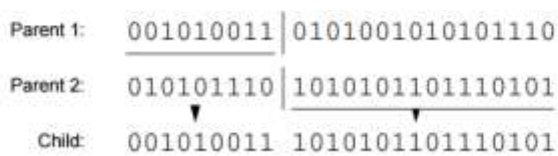

Figure 2 One-Point Crossover

Two-point crossover (Figure 3) differs from the previous version merely in the point that two random cuts are made, so three pieces have to be put together in order to produce an offspring.

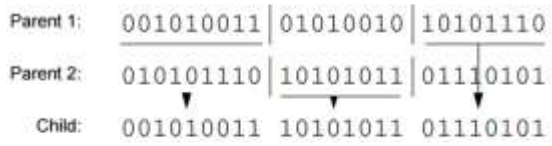

Figure 3 Two-Point Crossover

These two are the two original crossover operations. The third one, uniform crossover is suggested in [Syswerda, 1989]. It is illustrated in figure 2.4. Here, for each bit, it is randomly decided, if it is copied from parent one or two. During a generation a fixed number of crossovers and mutations are performed.

III. PARTICLE SWARM OPTIMIZATION

Eberhart and Kennedy in 1995 proposed the concept of particle swarm. The method first developed was based on individual particle to perform optimization but later on the algorithm was simplified and it was realized that the individuals here typically known as particles were actually performing optimization [3]. It was originally meant for simulating the social behavior of a bird flock. Particle Swarm Optimization is based on the social behavior of a colony of swarm of insects, such as ants, birds, fishes, termites, bees, and wasps. And PSO mimics their behavior. Each one in the swarm behaves in the distributed way using its own personal intelligence and the group intelligence collectively. Therefore, when one particle is able to find good path to the food, the other members of the team follow that particle irrespective of their distances from the good one [4]. Optimizations based on these facts of swarm intelligence are called behaviorally inspired algorithms and algorithms such as genetic algorithms are called Evolutionary algorithms, where the next generation is formed by mutating the older generations.

In PSO, the particles are initially scattered at random positions in the search-space, moving in random directions with different velocities. The direction of a particle is then gradually changed depending on the best previous positions of itself and its best neighbor, searching in their vicinity and wishing that discovering even better positions. The inertia weight controls the amount of recurrence in the particle's velocity so that no two particles moving in the search space are at the same position at any instant [5]. Also enforcing search-space boundaries after a particle's position is update, it is also required to impose limitations on the distance update, it is also required to impose limitations on the distance the particle may move in only one step [6]. This is done by limiting a particle's velocity to the full dynamic range of the search-space, so the particle may at most moves from 1 search-space boundary to the other in single step.

IV. BACKPROPAGATION ALGORITHM

GDBP algorithms have been widely used, well investigated and one of the most popular learning algorithm class for FNNs. Their effectiveness has been well documented. For example, an online GDBP learning algorithm for time-varying inputs [Zhao 1996], an adaptive learning algorithm with reduced complexity [Zhou and Si 1998], a fast learning algorithm based on the gradient descent of neuron space [Parisi et al 1995], the LMBP algorithm [Hagan and Menhaj 1994], and a general BP learning algorithm for FNN [Yu, Efe and Kaynak 2002]. [7]

A common dynamic characteristic in this class of algorithm is that because of the fixed learning rate, an asymptotic convergence is incurred, the result being that the closer to the desired weights, the slower the convergence speed - making it difficult for fast convergence with higher precision tolerance.

V. BENCHMARK DATASETS

Two classification problems are taken into account for testing of designed algorithm. Basically this work shows the performance of PSO+GA when it is blended into PNN to find accurate spread factor for PNN. The mean square error obtained for classifications shows the effectiveness of the research work also the designed algorithm is implemented for the network in such a way that when dataset varies, most of the PSO does not require change in values that is they are generalized from research studies and experiments.

1. Mackey Series

500 samples were trained for Mackey series and another 500 samples were predicted. Samples were taken as x(t-18), x(t-12), x(t-6) and x(t) and the predicted sample was x(t+6). Samples started from 101 to 600 and test samples started from 601 to 1100. 500 samples started from 101 and 601 respectively for train and test data in four columns.[8]

For Mackey Series data, the neurons in the hidden layer were selected to be 4 and in the output layer to be 1 with transfer functions log sigmoid for both. The network was trained for 500 samples and then test for another 500 samples. The result shows the predicted and the actual output for all train and test samples. Another figure plot shows the error in prediction by normal FFNN with random weights and biases and PSO gained weights and biases to FFNN.

2. Box Jenkins Gas Furnace Data Series

200 samples of Box Jenkins time series data were used for training and 90 samples were used for testing. The instantaneous values of output y(t) depends on ten variables mainly the previous past values of y(t) for past four sampling times and u(t) for past six sampling times i.e. y(t − 1), y(t − 2), y(t − 3), y(t − 4), u(t − 1), u(t − 2), u(t − 3), u(t − 4) ,u(t − 5), u(t −6) . The train array was 10x200 and the test array was 10x90. The result shows the predicted and the actual output for all train and test samples.

VI. PROPOSED SYSTEM

The proposed hybrid optimization techniques use Feed Forward Neural Network for the prediction of chaotic signal. The hybrid approach uses Feed Forward Neural Network and Particle Swarm optimization. Two hidden layers are incorporated for prediction in which the neurons in the first layer are priory initialized and neurons in the second layer are optimized using PSO. Both layers of DNN use Back Propagation for training the network and log-sigmoidal function for both the hidden layers. The output layer uses a linear function. The figure below shows the block diagram for selecting number of neurons in hidden layer of feed forward neural network using hybrid PSO and GA optimization.

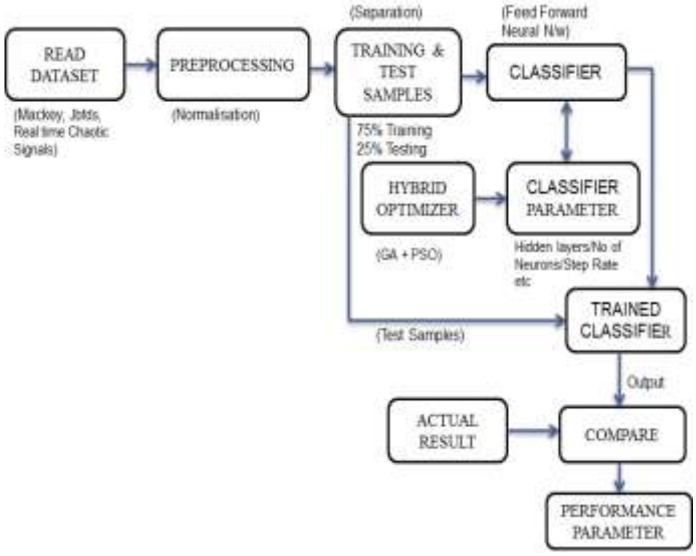

Figure 4 Block Diagram of proposed system

PSO Parameters:

The PSO parameters that are used in the project are specified as follows. These parameters are initialized at the time of initialization.

Number of particles – 4	C2f – 2
Wmax – 0.9	C2i – 1
Wmin – 0.5	POPmax- 20
C1f – 1	POPmin – 2
C1i – 2	Vmax -10
	Vmin - 1

$$w = \frac{w_{max} - (w_{max} - w_{min}) * iteration\ number}{maximum\ iteration}$$

$$C_1 = \frac{(c_{1f} - c_{1i}) * iteration\ number}{maximum\ iteration} + c_{1i}$$

$$C_2 = \frac{(c_{2f} - c_{2i}) * iteration\ number}{maximum\ iteration} + c_{2i}$$

Feed-forward Network Parameters:

1. Mackey Series:

Number of hidden layers	2
Number of neurons in first hidden layer	6
Number of neurons in second hidden layer	To be found out by PSO
Transfer Function	Logsig, logsig, purelin

ISSN: 2454-4248
189 – 193

Epochs (for PSO optimization)	200
Epochs (for finding minimum MSE)	1000
Learning rate	0.1
MSE	10^{-6}
Learning Algorithm	Leverbeg , Marquardt

2. Box Jenkins Gas Furnace Data:

Number of hidden layers	2
Number of neurons in first hidden layer	6
Number of neurons in second hidden layer	To be found out by PSO
Transfer Function	Logsig, logsig, purelin
Epochs (for PSO optimization)	500
Epochs (for finding minimum MSE)	1000
Learning rate	0.1
MSE	10^{-6}
Learning Algorithm	Leverbeg , Marquardt

Algorithm Steps:

For Box Jenkins Data, training samples are 200 and testing samples are 90. For Mackey Series 500 samples are used for both training and testing. The results are included in the next chapter.

1. Load Dataset – training and testing samples.
2. Initialize PSO parameters.
3. Initialize Deep Neural Network parameters.
4. Evaluate DNN for all particles once.
5. Find the best particle - G_{BEST}.
6. For number of iterations, iterate.
7. For number of particles, evaluate.
8. Calculate w.
9. Calculate C_1 and C_2.
10. Update velocities of particles.
11. Restore velocities in V_{min} and V_{max}.
12. Update particle position.
13. Restore positions in POP_{min} and POP_{max}.
14. Train the DNN with NN_{POS}= particle position.
15. Calculate fitness and MSE.
16. Find G_{BEST}.
17. Go to step (8), if all particles are not evaluated.
18. Go to step (7), if iterations are not completed.
19. Store G_{BEST}.
20. Evaluate the DNN with this G_{BEST} value for 5 iterations, since weights and biases corresponding to this G_{BEST} value are not retained.
21. Find the best performance of DNN.
22. Show results.

VII. RESULT AND CONCLUSION

In this work, the hybrid optimization of chaotic signals is performed using prediction technique. The data sets that are used for the work are taken from Mackey Series and Box Jenkins Gas Furnace Data Series. The Neural Network that is used for the process is Feed Forward Neural Network. PSO has been utilized to predict the value in the series for both Mackey as well as Box Jenkins. The results obtained individually are as follows.

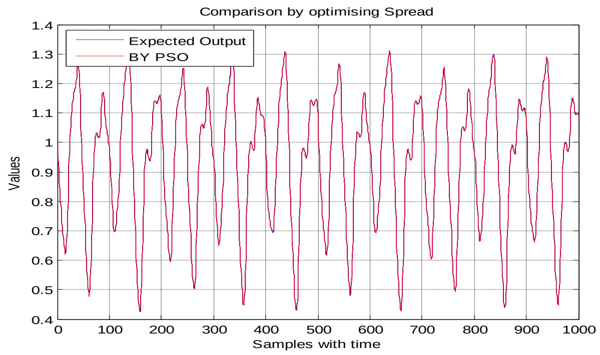

Figure 5 Comparisons for Mackey Series

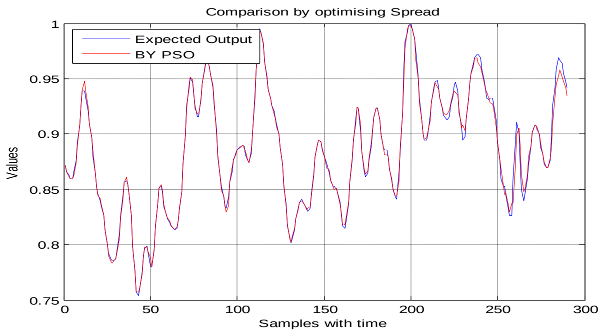

Figure 6 Comparisons for Box Jenkins Data

Sr. No	Series Type	Mean Square Error with PSO	Result Obtained
1	Mackey Series	$1.6591x\ 10^{6}$	6 | 17 | 1
2	Box Jenkins Series	$4.8019x\ 10^{5}$	6 | 2 | 1

Table 1 Comparison of MSE

ISSN: 2454-4248
189 – 193

The result obtained in the above mentioned table can be explained as follows.

- First value of the table denotes the number of neurons in the first hidden layer.
- Second value denotes the number of neurons in the second hidden layer which are predicted by the PSO.
- Third value denotes the number of predicted values which will be always one.

CONCLUSION

Most often when a normal PNN with radial basis function neural network is trained for an input data, the optimal output cannot be expected due to number of radial basis functions or hidden units, center of hidden units and the spread factor at the first execution.. For classification problems, it is clear that the classification is more accurate when spread is priory obtained by hybrid heuristic optimization and then given to PNN than what is achieved by random spread with normal PNN. Thus combination of PSO+GA with PNN is an optimization tool for neural networks. The same can be with Back propagation for weights and biases.

The above work is based on comparative study of hybrid optimization of chaotic signals. The feed forward neural network is trained with the mentioned parameters and the help of PSO the number of neurons in the second layer are predicted. The types of series used as chaotic signals are Mackey series and BJGFD Series.

REFERENCES

[1] Richa Garg, GGITC, Ambala Saurabh mittal GGGI, Ambala India "Optimization by Genetic Algorithm" International Journal of Advanced Research in Computer Science and Software Engineering Volume 4, Issue 4, April 2014

[2] Noraini Mohd Razali, John Geraghty,"A genetic algorithm performance with different selection strategies", Proceedings of the World Congress on Engineering Vol II, 2011.

[3] Thomas Stutzle and Holger H. Hoos, "Min-Max Ant System", Future generation computer systems, Elsevier, pp. 889-914, 2000.

[4] Marco Dorigo and Thomas Stutzle, "Ant colony optimization", A Bradford Book, The MIT Press, Cambridge, Massachusetts, London, England, 2003.

[5] B. S. Jung1, B. W. Karney and M. F. Lambert," Benchmark Tests of Evolutionary Algorithms: Mathematic Evaluation and Application to Water Distribution Systems", Journal of Environmental Informatics vol. 7, issue 1, pp. 24-35, 2006.

[6] Rahib H. Abiyev and Mustafa Tunay," Optimization of High-Dimensional Functions through Hypercube Evaluation", Computational Intelligence and Neuroscience, Hindavi, volume 2015, pp. 1-11.

[7] R.A.Caruana and J.D. Schaffer: "Representation and Hidden Bias: Gray vs. Binary Coding for Genetic Algorithms", in: Proceedings of the Fifth International Conference on Machine Learning, p. 153-161, Morgan Kaufmann

[8] Pravin Kshirsagar and Sudhir Akojwar,"Hybrid Heuristic Optimization for Benchmark Datasets", International Journal of Computer Application (0975-8887), Vol.146-No.7, July 2016.

[9] Pravin Kshirsagar and Sudhir G. Akojwar,"Prediction of Neurological Disorders using Optimized Neural Network" International conference on Signal Processing, Communication, Power and Embedded System (SCOPES),Oct. 2016 .

[10] Pravin Kshirsagar and Sudhir Akojwar "Classification of Human Emotions using EEG Signals" ,International Journal of Computer Applications (0975 – 8887) Volume 146 – No.7, July 2016