ANN Model identification: Two Soft Computing Based Approaches

Kalra Ashima[1], Kumar Shakti[2], Walia Sukhbir Singh[3]
[1]Research Scholar, Punjab Technical University, Jalandhar, Punjab.
[2]Baddi University of Emerging Sciences & Technology, Baddi(HP) India.
[3]IKG Punjab Technical University, Kapurthla, Punjab, India.

Received June 10, 2017 Accepted June 20, 2017

ABSTRACT *This paper proposes an optimized ANN model identification approach. The approach is a two step approach. The first step decides the architecture and in the second step the ANN is trained with the given training data. It is an iterative process that begins with only one neuron. If the output error is not within acceptable limits; the additional neurons, followed by additional hidden layers are added until the output error is within acceptable limits. The proposed approach used Particle Swarm Optimization (PSO) and Artificial Bee Colony (ABC) based approaches to train the ANN model. To validate our proposed approach, we implemented the approach in the MATLAB and compared its performance with other 3 well known classical training approaches namely LM, EBP and Rprop. We tested our approach on two well known problems from the literature. We observe that LM is fastest converging algorithm with fair amount of accuracy. As the accuracy requirement is increased it was observed that the proposed approach gave best performance in terms of acceptable error (MSE) when the model was trained with PSO based algorithm. Out of the 5 approaches used for training PSO outperformed all other approaches followed by ABC algorithm based training approach.*

Key words: model identification, ANN (artificial neural network), optimization.

1 Introduction

An artificial neural network (ANN) is a massively parallel distributed network of artificial neurons having ability to store experiential knowledge and later making it available for further use [1]. In this paper we propose two soft computing based ANN model identification approaches for evaluation of the performance of Institutions of Higher Learning from the given training data set. The modeling of an ANN system consists of two steps : first we select the ANN architecture in which number of hidden layers and the number of neurons in each hidden layer is decided. In the second step we train this ANN system to the given training data. In this paper we propose to formulate the ANN model identification problem as a search and minimization problem. The optimization algorithms are applied in a way to automatically adjust the number of hidden layers, neurons in each of the hidden layers and identified values of synaptic weights in such a way so as to minimize the objective function which we define to be the MSE.

$$MSE = \frac{1}{N}\sum_{k=1}^{N}[OA - OC]^2$$

where OA is the actual output or desired output, OC is the computed output, N is number of training examples used for model identification.

The paper we demonstrates the application of Particle Swarm Optimization (PSO) and Artificial bee colony (ABC) based global optimization approaches for ANN model identification. Further, we compare the performance of proposed approaches to standard error back propagation (EBP), Resilent propagation (RProp) and Levenberg-Marquardt (LM) based classical training algorithms. This paper consists of 6 sections. Section 2 gives a brief of related work. Section 3 introduces the two different problems considered for this paper for ANN system identification. Section 4 discusses the modeling process used for implementation. Section 5 of this paper presents simulation results, discussions and compares the performance of the proposed two soft computing approaches with 3 classical approaches. Section 6 concludes the paper.

2. Related Work

An excellent survey [2-3] can be found in literature on the various learning approaches including classical as well as soft computing based optimization approaches for NN training. Error back propagation EBP proposed by Werbos[4], is the widely used method for ANN training [5]. Rumelhart [6] further elaborated it as the learning representations of NN by back propagating errors. This algorithm includes different versions like standard or incremental back propagation (IBP) by Freeman and Skappura [7] in which the network weights are updated after presenting each pattern from the learning data set, rather than once per iteration and batch back propagation (BBP), Hagan et.al [8] in which the network weights update takes place once per iteration, while all learning data pattern are processed through the network. To improve the convergence speed of EBP, Rumelhart extended his work by introducing the momentum term [9-10]. Resilent error back propagation (Rprop) proposed by Riedmiller and Braun[11] is also one of the methods to improve convergence speed that performs a direct adaptation of the weight step based on local gradient information. Among second order approaches, Levenberg-Marquardt (LM) algorithm is proposed by Bogdan M. Wilamowski [12] and is used as a successful algorithm for ANN training with MLP structure for increasing the convergence speed [13-14]. The above mentioned techniques are conventional techniques but in the recent past the researchers have move on to the soft computing techniques based training approaches. Particle Swarm Optimization (PSO) [15] is a soft computing based global optimization proposed by Eberhart and Kenedy. This is the simulation of social behaviour of bird flocking or fish schooling. Another soft computing based approach is Artificial Bee Colony (ABC) [16] simulating the intelligent foraging behavior of a honey bee swarms, proposed by Karaboga and Basturk. Both PSO and ABC are swarm based optimization approaches and well used for ANN training as per literature [17-18]. A good survey has been found on the extraction of rules from the trained neural networks in papers [19-23].

3. ANN model identification

In this section we discuss application of our proposed ANN modeling approach on two different problems namely Rapid Battery Charger (RBC) and Performance Evaluation of Institutions of Higher Learning from the given training data set. The complete modeling of an ANN system consists of two processes: first the selection of ANN architecture in which number of hidden layers and the number of neurons in each hidden layer is to be decided. Second is the training of this ANN system by the given training data. The problem here is formulated as search and minimization problem. The optimization algorithms are applied in a way to automatically adjust

the number of hidden layers, neurons in each of the hidden layers and identified values of synaptic weights in such a way so as to minimize the objective function i.e. MSE.

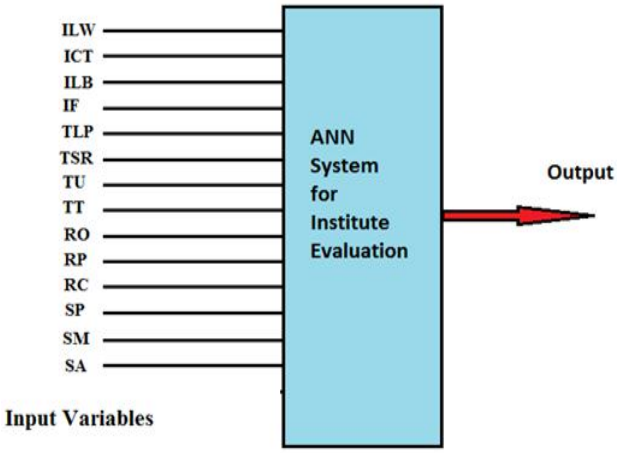

Fig. 1 Block Diagram of ANN System for IRS

In the first example we discussed the evaluation system for the institutes of higher learning [24]. Figure 1 represents the block diagram of the desired model. This is a multi-input single output system. The names of input variables and other details of the system are mentioned below in table 1.

Table 1
Input variables for institute rating system (IRS)

1	Laboratories And Workshops (ILW)	8	A/V Aids Used / Teaching Techniques (TT)
2	Class Rooms And Tutorials (ICT)	9	Research Orientation (RO)
3	Library (Book, Journals) (ILB)	10	Research Publications (RP)
4	Academic Facilities(IF)	11	Research Projects/ Conferences (RC)
5	Teaching-Learning Process (TLP)	12	Student Placements (SP)
6	Student/Teacher Ratio (TSR)	13	Students Merit (Pass Percentage) (SM)
7	Teacher Training/ Updation (TU)	14	Admission Preference (SA)

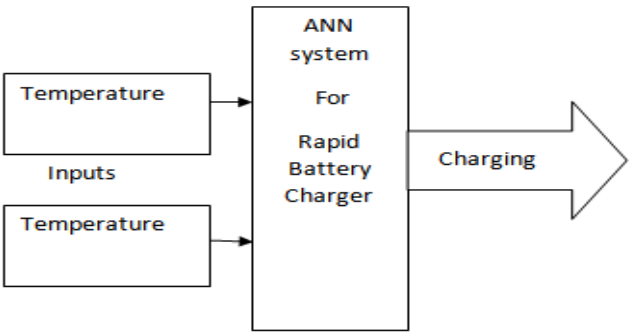

Fig. 2 Block Diagram of the Required ANN System for RBC

The second example of ANN model identification that we present is of Rapid Battery Charger (RBC) modeling problem [25-26]. This is a two input and single output system and belongs to the control systems category. The two inputs are (1) temperature (2) temperature gradient (temp_grad.) and charging current is the output for the system. Figure 2 above shows the block diagram of the system to be identified.

4. **Methodology**

Modeling an ANN System is a complex process involving number of steps. This complexity further increases with the increase in the number of input parameters and number of hidden layers. The main steps to be followed for designing a complete ANN System are given as below:

1. Begin with number of hidden layers NH = 0
2. Fix the number of neurons in each hidden layer.(values of k1, k2 … and so on) (we have taken 5 in first hidden and 3 neurons in second hidden)(maximum layers hidden is 2)
3. Randomly initialize the weights of ANN.
4. For each training pattern, evaluate output and error between the computed and desired output.
5. Compute mean square error for the model (MSE).
6. Minimize the objective function (MSE) by adjusting the weights using proposed approach by embedding following modules for training:

   Module1: EBP

   Module 2: Resilent error back propagation

   Module 3: LM algorithm

   Module 4: PSO

   Module 5: ABC
7. If MSE is acceptable (termination criterion is met) then go to step 9, else if number of hidden layers are non zero then increase the number of neurons in the hidden layers. After an upper limit of the number of neurons in the hidden layers has reached and if the performance is still not acceptable we increase the number of hidden layers.
8. Go to step 4
9. Stop

5. Simulation and results and discussion

In order to validate our two proposed approaches we implemented these algorithms in MATLAB on a DEL Laptop with Intel core i3 processor working at 2.40 GHz with 4 GB RAM, running on Windows 7 platform. We used rapid battery charger (RBC) data and identification for institute rating system (IRS) data to evolve the architecture as well as for training purpose. We used 20% data for training purpose. For each of the implemented approach we took 20 trials with 500, 1000, 2000, 5000 and 10,000 iteration and recorded the MSE of each evolved model. For the parameters given in table 2, the results showing mean value for MSE and time of system

identification for Rapid Battery Charger (RBC) and institute rating system (IRS) using various algorithms are given in Table 3 and Table 4 respectively.

Tab**le 2**. Parameters of RBC and IRS data set

Parameters	**Values for RBC**	**Values for IRS**
Size of population	10	10
Number of hidden layers	2	2
Number of neuron in first hidden layer	5	5
Number of neuron in first hidden layer	3	3
Number of iterations	500, 1000, 2000, 5000, 10000	500,1000, 2000, 5000, 10000
Number of input variables	2	14

Table 3. Results(mean value of MSE and Time) for ANN system identification using Five algorithms for Rapid Battery Charger (RBC) data

Iteration= 500					
Algorithm	**PSO**	**EBP**	**RPROP**	**LM**	**ABC**
MSE	0.01519	0.11933	0.11447	0.181001	0.111565
Execution time(sec)	1.92455	5.64463	8.980565	0.02649	18.6888
Iteration= 1000					
Algorithm	**PSO**	**EBP**	**RPROP**	**LM**	**ABC**
MSE	0.01873	0.11914	0.1148	0.166205	0.111705
Execution time(sec)	3.43082	10.6403	15.1261	0.02498	57.72772
Iteration= 2000					
Algorithm	**PSO**	**EBP**	**RPROP**	**LM**	**ABC**
MSE	0.01666	0.11938	0.11563	0.198865	0.111755
Execution time(sec)	10.4296	20.2409	30.88321	0.025435	69.25645
Iteration= 5000					
Algorithm	**PSO**	**EBP**	**RPROP**	**LM**	**ABC**
MSE	0.01372	0.11932	0.115855	0.133305	0.11163
Execution time(sec)	35.2559	50.1013	71.54164	0.025165	234.3459
Iteration= 10,000					
Algorithm	**PSO**	**EBP**	**RPROP**	**LM**	**ABC**
MSE	0.01018	0.119	0.115887	0.164307	0.111233
Execution time(sec)	133.855	105.8128	121.144	0.02916	496.6234

Table 4. Results (mean value of MSE and Time) for ANN system identification using Five algorithms for IRS (institute rating system) data

Iteration= 500					
Algorithm	**PSO**	**EBP**	**RPROP**	**LM**	**ABC**
MSE	0.02563	0.02817	0.02802	0.160475	0.023225
Execution time(sec)	1.27252	1.10002	1.3095	0.02613	13.91547
Iteration= 1000					
Algorithm	**PSO**	**EBP**	**RPROP**	**LM**	**ABC**
MSE	0.02325	0.027595	0.027335	0.301085	0.025695
Execution time(sec)	2.92384	2.17379	2.59752	0.042045	30.33184
Iteration= 2000					
Algorithm	**PSO**	**EBP**	**RPROP**	**LM**	**ABC**
MSE	0.02256	0.02726	0.02745	0.089187	0.024515
Execution time(sec)	7.97452	5.61909	5.93603	0.028465	52.59275
Iteration= 5000					
Algorithm	**PSO**	**EBP**	**RPROP**	**LM**	**ABC**
MSE	0.02232	0.02784	0.0271	0.198725	0.026735
Execution time(sec)	43.5106	13.68402	11.12177	0.02902	130.4593
Iteration= 10,000					
Algorithm	**PSO**	**EBP**	**RPROP**	**LM**	**ABC**
MSE	0.02068	0.027095	0.02749	0.176665	0.024565
Execution time(sec)	189.537	24.65806	24.98658	0.03718	316.4512

A look at table 3 and 4 makes it amply clear that LM is the quickest to converge algorithm with fair amount of accuracy. However, if adequate training time is available then the out of the five algorithms that we have considered, PSO was found to offers best performance with minimum

MSE of all the approaches. We observe this for both of the modeling examples that were considered for our study. Further we observed that ABC based approach ranks at number two after the PSO based approach. Both of these soft computing based approaches outperform classical three approaches namely LM, EBP and RPROP which were considered for comparison purposes.

6. Conclusions

In this paper we have proposed an ANN Model identification approach. The proposed approach decides the appropriate feed forward architecture for the ANN model and then trains the model using PSO based and ABC optimization algorithms based approaches. Both the approaches were found to generate the optimized ANN models with minimum MSE. In order to validate our approach, we implemented the proposed approaches in MATLAB. We used two soft computing based approaches namely PSO and ABC for training the ANN model in our approach. We tested our proposed approach on two problems available in the literature. The first problem was two input, single output ANN based rapid battery charger and the second one was the institute rating evaluation system (IRS). In case of both of the problems it was observed that the two approaches proposed by authors outperformed the other 3 classical approaches of ANN model training. We observed that out of the 5 approaches used PSO based approach produced the minimum MSE model followed by ABC and LM based approaches respectively. We further observed that LM based approach is quickest to converge with fair amount of accuracy. However, if the accuracy requirement is higher then PSO and ABC based approaches perform best in that order but with a requirement of higher training times. This became apparent from the observations that as the number of iterations was increased to 10000 for training, PSO gave the best performance followed by ABC, LM, Rprop and EBP approaches.

References

1. G. Zhang(2000), “Neural networks for classification: a survey,” IEEE Transactions on Systems, Man, and Cybernetics-Part C: Applications and Reviews, vol. 30, no. 4, pp. 451–462.
2. S. Kumar, S.S Walia, A. Kalra.(2015) “ANN Training: A Review of Soft Computing Approaches”, International Journal of Electrical & Electronics Engineering, Vol. 2, Spl. Issue 2, pp. 193-205.
3. A. Kalra, S. Kumar, S.S Walia.(2016) “ANN Training: A Survey of classical and Soft Computing Approaches”, International Journal of Control Theory and Applications, Vol. 9, pp. 715-736.
4. P. J. Werbos(1994), “The Roots of Backpropagation", John Wiley k Sons, Inc., New York.
5. P Sehgal et.al(2012), “Minimization of Error in Training a Neural Network Using Gradient Descent Method”, International Journal of Technical Research(IJTR),Vol 1,Issue 1, pp 10-12.
6. D. E. Rumelhart, G E Hinton and R J Williams(1986) “Learning internal representations by error propagation, MIT Press.
7. Freeman, J.A., Skappura, D.M. (1991), “Neural Networks Algorithms, Applications and Programming Techniques”. Addison-Wesley, Houston, pp. 12– 105.
8. Hagan, M.T., Demuth, H.B., Beale, M.H.(1996). Neural Network Design. PWS, Boston
9. M Moreira and E Fiesler(1995), “Neural Networks with Adaptive Learning Rate and Momentum Terms” IDIAP, technical report.

10. R.A. Jacobs(1988), "Increased rates of convergence through learning rate adaptation", Neural Networks, 1,pp 295-307.

11. Martin Riedmiller Heinrich Braun(1993) "A Direct Adaptive Method for Faster Backpropagation Learning: The RPROP Algorithm", IEEE, ,pp 586-591.Francisco, CA, 1993, pp. 586-591.

12. Hao Yu, Bogdan M. Wilamowski, Member IEEE "Levenberg–Marquardt Training".

13. M. T. Hagan and M. B. Menhaj(1994), "Training feed forward networks with the Marquardt algorithm," IEEE Trans. Neural Netw., vol. 5, no. 6, pp. 989–993.

14. Amir Abolfazl Suratgar, Mohammad Bagher Tavakoli, and Abbas Hoseinabadi(2007) "Modified Levenberg-Marquardt Method for Neural Networks Training",World Academy of Science, Engineering and Technology International Journal of Computer, Information, Systems and Control Engineering Vol:1 No:6,pp 1719-1721.

15. Kennedy, J.; Eberhart, R.(1995), "Particle Swarm Optimization". Proceedings of IEEE International Conference on Neural Networks. pp. 1942–1948.

16. D Karaboga ,B Basturk,(2007)"A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm", Journal of Global Optimization , Volume 39 Issue 3,pp 459 – 471.

17. Naseer mohammadi, Seyed javad Merabedini(2014), "Comparison of Particle Swarm optimization and back Propagation algorithms for training Feed Forward neural networks", Journal of mathematics and computer science 12 , 113 – 123.

18. Yana Mazwin Mohmad Hassim and Rozaida Ghazali, "Training a Functional Link Neural Network Using an Artificial Bee Colony for Solving a Classification Problems".

19. Duch, W., R. Adamczak, and K. Grabczewski, (1998),"Extraction of logical rules from neural networks," Neural Process. Lett., vol. 7, pp. 211–219.

20. Fu, L., Aug. (1994), "Rule generation from neural networks," IEEE Transactions on Systems, Man and Cybernetics, Vol. 24, No. 8, pp. 1114-1124.

21. Ishibuchi, H. et al., (1993), "Neural Networks that learn from Fuzzy if then rules," IEEE Trans. On Fuzzy Systems, Vol.1, pp.85-97.

22. Ishikawa, M., (1996), "Structural learning with forgetting," Neural Networks, vol. 9, pp. 509–521. Taha,I. A. and Ghosh,J. 1999. Symbolic interpretation of artificial neural networks. IEEE Trans. Knowl. Data Eng.11 : 448 – 463.

23. Towell, G.G., J. W. Shavlik, (1993), "Extracting refined rules from knowledge-based neural networks,"Machine learning, vol. 13, no.1, pp. 71-101.

24. Shakti Kumar , S S Walia, S S Bhatti(2013) "Performance Evaluation of Institutes of Higher Learning: A hierarchical Fuzzy System Approach" IRACST – Engineering Science and Technology: An International Journal (ESTIJ), ISSN: 2250-3498,Vol.3, No.4.

25. Shakti Kumar, Nitika Ohri, Savita Wadhvan (2004) a,"ANN based design of rapid battery charger", Trends Of Computational Techniques In Engineering Oct 15-16, , SLIET, Longowal Punjab pp 129-132.

26. Khosla, A., Kumar, S. and Aggarwal, K. K. (2003) Identification of fuzzy controller for rapid nickel cadmium batteries charger through fuzzy c–means clustering algorithm. Proceedings of 22nd International Conference of the North American Fuzzy Information Processing Society, Chicago, Illinois, USA, July 24–26, pp. 536–539.