Training ANFIS Using Genetic Algorithm for Dynamic Systems Identification

Article · December 2016
DOI: 10.18201/ijisae.266053

Bulent Haznedar
Hasan Kalyoncu University

International Journal of
Intelligent Systems and
Applications in Engineering
IJISAE
ISSN:2147-6799 www.atscience.org/IJISAE
Advanced Technology and Science
Original Research Paper

Training ANFIS Using Genetic Algorithm for Dynamic Systems Identification

Bulent Haznedar*[1], Adem Kalinli[2]

Accepted 3rd September 2016

Abstract: In this study, the premise and consequent parameters of ANFIS are optimized using Genetic Algorithm (GA) based on a population algorithm. The proposed approach is applied to the nonlinear dynamic system identification problem. The simulation results of the method are compared with the Backpropagation (BP) algorithm and the results of other methods that are available in the literature. With this study it was observed that the optimisation of ANFIS parameters using GA is more successful than the other methods.

Keywords: *Neuro-Fuzzy, ANFIS, Genetic Algorithm, System Identification.*

1. Introduction

System identification is a model generating process that is developed to predict the behaviour of a system between its input and output signals. The approaches based on the fuzzy neural networks and artificial neural networks are among the methods that are used commonly for the dynamic systems identification. However, due to some of its superior features, studies towards using ANFIS for the purpose of dynamic system identification, have started to increase gradually in the recent years [1-5].

Training the ANFIS model is basically the determination process of the optimal values for the model's premise and consequent parameters. Derivative-based algorithms are commonly used for training of ANFIS. But there is a problem of getting stuck in a local minimum in derivative-based algorithms. In this context, various methods have been proposed in recent years in order to train ANFIS parameters. Some of these methods are artificial intelligence optimization algorithms and heuristic algorithms such as Genetic algorithm, PSO and Differential Evolution Algorithm that are not derivative-based [6].

Ghomsheh et al. [3] proposed Particle Swarm Optimization (PSO) algorithm that is modified for the optimization of ANFIS parameters. Carrano et al. [7] used GA for the training of ANFIS in the solution of multi-objective optimization problems. Cus et al. [8] used Ant Colony Optimization (ACO) algorithm for training of ANFIS parameters. They tried the performance analysis of the developed method on the non-linear systems.

In this study, an approach is presented towards the training ANFIS by using GA based on a population algorithm. The performance of the presented approach is tested on non-linear system and the results are compared with different methods. In the following section, the basic structure of ANFIS is introduced, in the third section, information about genetic algorithm is given, in the fourth section, information about the presented approach is given and the obtained results are given in the section 5.

[1] *Hasan Kalyoncu University, Department of Computer Engineering, Gaziantep, Turkey.*
[2] *Erciyes University, Department of Computer Engineering, Kayseri, Turkey.*
* *Corresponding Author: Email: bulent.haznedar@hku.edu.tr*
Note: This paper has been presented at the 3rd International Conference on Advanced Technology & Sciences (ICAT'16) held in Konya (Turkey), September 01-03, 2016.

2. Adaptive Neuro-Fuzzy Inference System (ANFIS)

Adaptive Network-based Fuzzy Inference System (ANFIS) is a network model which Sugeno-type fuzzy system is combined with neural learning ability. The main aim of ANFIS is to optimize the parameters of the equivalent fuzzy logic system by using input-output data sets via a learning algorithm. Parameter optimization is carried out in such a way that the error value between the actual output and the target output to be minimum.

ANFIS contains two parameters as antecedent and consequent parameters. Those two parameter types connect the fuzzy rules to each other and training of the model is provided with the optimization of these parameters. ANFIS basically consists of five layers. A basic ANFIS structure consisting of two inputs and one output is given in (Figure.1).

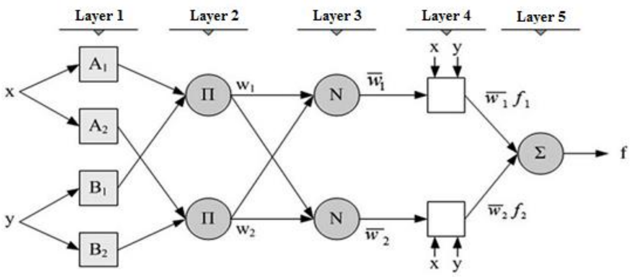

Figure 1. A basic structure of ANFIS [9]

A. Layer 1

This layer is named as the fuzzification layer. The signal that is obtained from each node is transfered to the other layer. In this layer, for the outputs of the cells, (O_{1i}), Equation (1) and Equation (2) are given [10].

$$O_{1i} = \mu A_i(x) \qquad i = 1,2 \qquad (1)$$

$$O_{1i} = \mu B_{i-2}(x) \qquad i = 3,4 \qquad (2)$$

In here, A_i and B_i are any membership functions belong to the inputs, μA_i and μB_i are membership degrees that are calculated for this function. For the bell-shaped membership function μA_i is calculated with the equation below.

$$\mu A_i = e^{-\frac{1}{2}(\frac{x-c}{a})^2} \qquad i = 1,2 \qquad (3)$$

In here, a_i and c_i are sigma and central parameters of the membership function respectively.

B. Layer 2

This layer is named as the rule layer. Each rule's firing strength is calculated with the membership degrees that are coming from the previous layer.

$$O_{2i} = w_i = \mu A_i(x).\mu B_i(y) \qquad i = 1,2 \qquad (4)$$

C. Layer 3

In this layer which is named as the normalisation layer, each rule's normalised firing strength is calculated.

$$O_{3i} = \overline{w_i} = \frac{w_i}{w_1 + w_2} \qquad i = 1,2 \qquad (5)$$

D. Layer 4

This layer is the defuzzification layer. Output value for each rule is calculated using the value of firing strength from previous layer.

$$O_{4i} = \overline{w_i}.f_i = \overline{w_i}.(p_i x + q_i y + r_i) \qquad i = 1,2 \qquad (6)$$

E. Layer 5

This is the sum layer. ANFIS's output is obtained by collecting the output values belong to the each rule that are obtained in defuzzification layer.

$$O_{5i} = f = \sum \overline{w_i}.f_i = \frac{\sum w_i.f_i}{\sum w_i} \qquad i = 1,2 \qquad (7)$$

3. Genetic Algorithm (GA)

Genetic Algorithm (GA) which its fundamental principles are set forth by John Holland in 1970's, is implemented with success on many problem types [11]. GA is a heuristic algorithm which is used for being able to find exact or approximate results in optimization or search problem. This algorithm is developed by being inspired from the techniques in the evolutionary biology such as inheritance, mutation, selection and crossover. GA can be implemented quite effortlessly even on the multidimensional problems with large-size search space and also with too many number of variables.

GA is a population based optimization algorithm. Equivalent of the candidate solutions that generate the population is chromosomes. Due to the various evolution processes, these chromosomes transform into the solution candidates that represent better results. This process is maintained until reaching an acceptable compliance value or until meeting the criteria such as pre-determined processing time or generation number. Basic steps of the Genetic Algorithm are given in (Figure.2).

1. Generate random population of *n* chromosomes
2. Evaluate the fitness of each chromosome
3. Create a new population by repeating following steps until the new population is complete
   (i) Select two parent chromosomes from a population according to their fitness
   (ii) With a crossover probability cross over the parents to form new offspring
   (iii) With a mutation probability mutate new offspring at each locus
   (iv) Place new offspring in the new population
4. Use new generated population for a further run of the algorithm
5. If the end condition is satisfied, **stop**, and return the best solution in current population
6. Go to step **2**

Figure 2. Main steps of the basic GA algorithm [12]

4. Training ANFIS Using GA

In this section, GA usage for updating ANFIS parameters is explained. ANFIS has two parameter types that have to be updated. These are premise parameters and consequent parameters. Premise parameters belong to the gauss membership function that is given as $\{a_i, c_i\}$ in Equation (3). The total number of the premise parameters is equal to the sum of the parameters in all membership functions. Consequent parameters are the ones that are used in defuzzification layer, shown in Equation (6) as $\{p_i, q_i, r_i\}$.

The method that is suggested in this study is applied to the dynamic system identification problem. For the error value between the output obtained from ANFIS and the output obtained from non-linear dynamic system to be minimum, ANFIS parameters are optimized with GA. Block diagram showing this structure is given in (Figure.3). *RMSE* error function which is obtained by using Equation (8), is used for determining the error value of the solution. $\overline{y_i}$ is used in Equation (8), is showing the output obtained by ANFIS at the time *i* and y_i is showing the actual output of the system. *N* is showing the instance number used in the application.

$$RMSE = \sqrt{\frac{[\sum_{i=1}^{N}(y_i - \overline{y_i})]}{N}} \qquad (8)$$

The selection of the input array which will be used in training is important on the success of training. The elements of the input array are usually chosen from sinusoidal and random input values [13]. In the carried out simulation studies within this study, *u(k)* input signal array which has proper distribution, randomly generated in the range of $[-2,2]$ is used.

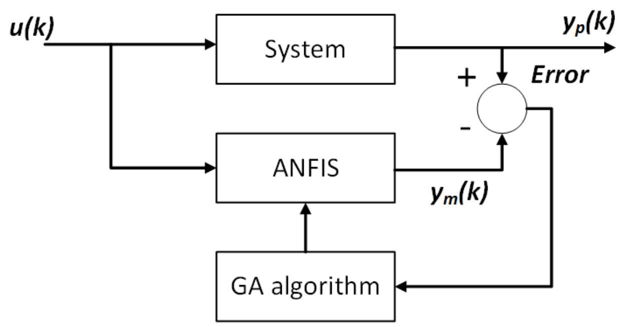

Figure 3. Block diagram showed the proposed method

5. Simulation Results

In this section, Applications have been performed by using non-linear dynamic system for the training ANFIS with GA. In order to measure the performance of the system 250 input and output data sets are generated. 200 of these data sets are used for training, the remaining 50 are used for the test.

For the dynamic system, that is going to be modelled, ANFIS structure that consists of 3 inputs and 1 output is created respectively. For each of the input of created ANFIS structure, gauss membership function with 2 parameters is used. Therefore there are 2^n rules in the used ANFIS model [14]. In this way, ANFIS structure consisting 12 premise parameters and 32 consequent parameters with 8 rules is used for the system.

In the study, many attempts have been made in order to decide upon the various parameter values that are required for GA. As a consequence of these attempts, population size is chosen as 200, crossover rate is chosen as 0.8 and mutation rate is chosen as 0.01.

Besides, the performance of the suggested approach is also compared with the results taken from the literature, which belong to the Elman feedback artificial neural network that is commonly used in the identification of the dynamic systems [15,16]. In the considered studies, Kalinli has suggested an approach based on training Elman network by using Tabu Search Algorithm (TS), SA algorithm and Parallel Tabu Search Algorithm (PTS) for the identification of dynamic systems and achieved more successful results than the traditional BP algorithm [15,16].

Sample non-linear system which is used for simulation belongs to a simple pendulum, swinging through small angle that is taken from [15] and defined with Equation (9). The system's discrete time definition is as follows:

$$y(k) = A_1 y(k-1) + A_2 y(k-2) + A_3 y^3(k-2) + B_1 u(k-2) \quad (9)$$

Where, $A_1 = 1.040000,\ A_2 = -0.824000,$

$A_3 = 0.130667,\ B_1 = -0.160000$

RMSE error values that are obtained following the simulation studies, and the error values belong to different methods and that are taken from the literature are given in Table 1. From the results that are given on Table 1, it is clearly seen that the performance of the approach suggested in this study has quite high success comparing to the other methods. Standard deviation for the training and testing processes of the error values that are obtained in 15 different attempts towards training ANFIS with genetic algorithm is found as 0.0032 and 0.0022 respectively. The low standard deviations show that *RMSE* values are close to each other, so it means that the results are reliable. The system that belongs one of the attempts towards training ANFIS with GA and the answers of ANFIS model are given in (Figure.4).

Table 1. Comparison of the results

Model	*Train ($RMSE_{AVG}$)*	*Test ($RMSE_{AVG}$)*
Elman – BP [15]	-	2.6182E-01
Elman – TS [15]	-	1.1990E-01
Elman – SA [15]	-	2.9430E-02
Elman – PTS[16]	-	2.83189E-02
ANFIS - BP	7.99112E-03	9.60747E-03
ANFIS – GA	**8.69121E-03**	**8.32214E-03**

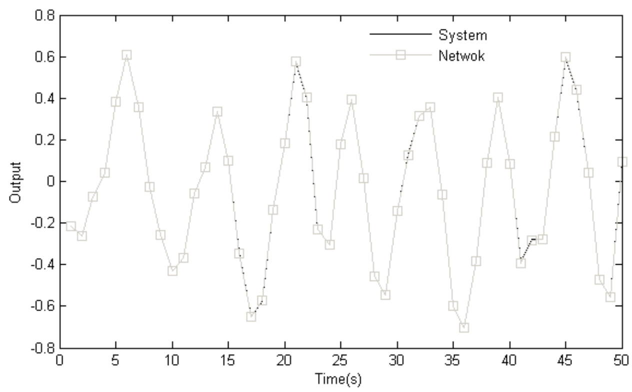

Figure 4. Response of the system (*RMSE* error= 5.33145E-03)

6. Conclusions

It has been seen that the best training and test error values obtained are found with the suggested method. Following the studies, finding the admissible *RMSE* error value shows that the training of ANFIS using GA for identification of non-linear systems is quite effective. Besides, the results with the low deviation value has been obtained within the suggested method. Having low deviation value shows that *RMSE* error values obtained from trainings started with different initial solutions are close to each other which means the results are reliable. Also, by reason of the fact that Genetic algorithm does not have restrictions as the derivative-based algorithms have and it is easily applicable on the problems, it is assessed that ANFIS model can be used on its applications for different problems.

Acknowledgements

This work was supported by the Research Fund of Erciyes University of Turkey, grant number: FDK-2016-6371.

References

[1] D. Karaboga and E. Kaya, "Training ANFIS using artificial bee colony algorithm for nonlinear dynamic system identification," in: IEEE 22nd Signal Processing and Communications Applications Conference (SIU), 2014, pp. 493-496.

[2] P. Liu, W. Leng and W. Fang, "Training ANFIS model with an improved quantum-behaved particle swarm optimization algorithm," Mathematical Prob. in Eng. vol. 2013, 2013.

[3] V.S. Ghomsheh, M.A. Shoorehdeli and M. Teshnehlab, "Training ANFIS structure with modified PSO algorithm," In Control and Automation, Med'07. Mediterranean Conference on IEEE, 2007, pp. 1-6.

[4] M.A. Shoorehdeli, M. Teshnehlab, A.K. Sedigh and M.A. Khanesar, "Identification using ANFIS with intelligent hybrid stable learning algorithm approaches and stability analysis of training methods," Applied Soft Comput. vol. 9, no. 2, pp. 833–850, 2009.

[5] M.A. Shoorehdeli, M. Teshnehlab and A.K. Sedigh, "Training ANFIS as an identifier with intelligent hybrid stable learning algorithm based on particle swarm optimization and extended Kalman filter," Fuzzy Sets and Systems. vol. 160, pp. 922–948, 2009.

[6] D. Simon, "Training fuzzy systems with the extended Kalman Filter," Fuzzy Sets Syst. vol. 132, pp. 189–199, 2002.

[7] E.G. Carrano, R.H.C. Takahashi, W.M. Caminhas and O.M. Neto, "A genetic algorithm for multiobjective training of ANFIS fuzzy networks," IEEE Congress on Evolutionary Computation, pp. 3259–3265, 2008.

[8] F. Cus, J. Balic and U. Zuperl, "Hybrid ANFIS-ants system based optimisation of turning parameters," Journal of Achievements in Materials. vol. 36, no. 1, pp. 79-86, 2009.

[9] S. Uzundurukan, "Determination and modeling of basic parameters that affect the swelling properties of soils," Ph.D. Thesis, Suleyman Demirel University, Turkey, 2006.

[10] J.S.R. Jang and C.T. Sun, "Neuro-Fuzzy modeling and control," The proceedings of the IEEE. vol. 83, no. 3, 378-406, 1995.

[11] M. Mitchell, "An Introduction to Genetic Algorithms," MIT Press, 1998.

[12] S.Lek, M.Scardi, P.F.M. Verdonschot, J.-P. Descy and Y.-S. Park, *Modelling Community Structure in Freshwater Ecosystems*. Berlin, Germany: Springer-Verlag, 2005.

[13] D.E. Rumelhart and J.L. McClelland, "Explorations in the micro-structure of cognition," parallel distrubuted processing 1. Cambridge, MA: MIT Press, 1986.

[14] C.F. Juang, "A TSK-type recurrent fuzzy network for dynamic systems processing by neural network and genetic algorithms," IEEE Transactions on Fuzzy Systems. vol. 10, no. 2, pp. 155–170, 2002.

[15] A. Kalinli, "Training Elman network using simulated annealing algorithm," Journal of the Institute of Science and Technology of Erciyes University. vol. 19, pp. 28-37, 2003.

[16] A. Kalinli and D. Karaboga, "Training recurrent neural networks by using parallel tabu search algorithm based on crossover operation," Engineering Applications of Artificial Intell. vol. 17, pp. 529-542, 2004.