Tikrit Journal of Pure Science 24 (1) 2019

ISSN: 1813 – 1662 (Print)
E-ISSN: 2415 – 1726 (On Line)

TJPS

Tikrit Journal of Pure Science

Journal Homepage: http://tjps.tu.edu.iq/index.php/j

Gray Wolf Optimization and Least Square Estimatation As A New Learning Algorithm For Interval Type-II ANFIS

Blqees K. Faraj , Nazar K. Hussein

Department of Mathematics, College of Computer Science and Mathematics, University of Tikrit, Iraq.

https://doi.org/10.25130/tjps.v24i1.339

ARTICLE INFO.

Article history:

-Received: 16 / 9 / 2018

-Accepted: 18 / 1 / 2019

-Available online: / / 2019

Keywords: of interval type-2, ANFIS, Gray Wolfe Optimization, Meckey-Glass time series, meta-heuristic.

Corresponding Author:

Name: Blqees K. Faraj

E-mail:

math_bkamzwt93@st.tu.edu.iq

Tel:

ABSTRACT

Gray Wolfe Optimization (GWO) is one of the meta-heuristic method and it is a popular technique in Many engineering and economic applications. GWO and Least Square Estimatation (LSE) are used to optimize the antecedents and consequents parameters of interval type-2 ANFIS respectively. We are checking the new learning algorithm by using the interval type-2 ANFIS in prediction of Mackey-Glass time series and the results were very encouraging compared to other algorithms.

I- Introduction

Zadeh in his paper [1] was present the idea of fuzzy sets, where it was the begin new covenant of knowledge, because the concept of fuzzy logic gave the computer ability to make a decision differently than normal logic. The dealing with relation becomes flexible where terms “tall”, “big” ,”hot”,…etc. In 1975 [2], Zadeh expanded his concept of fuzzy logic to which gave a wider area of resolution than the previous type as it gives several opinions instead of one opinion the type-II fuzzy logic

The Artificial Neural Networks (ANN) are systems has adjusted the processing of information [3, 4] , ANNs with FLSs are good hybrid systems which are used the features of two techniques ANNs and FLSs, one of this kinds is Adaptive Neuro Fuzzy Inference System (ANFIS) which was suggested by Jang 1993 [5].

II- Architectural of ANFIS

The architectural of ANFIS is almost similar to the structure of FLS because of both are Receive the crisp values as inputs and then converts to fuzzy value Then a series of operations are carried out on the fuzzy values, in order to finally remove the fuzziness from them for crisp values. Since this system is composed between two intelligence systems then we will achieve a system in quality or we can say that we have a fuzzy thinking and Network computation .

As in the FLS then ANFIS in general consists of two parts, the antecedents is the first part and the consequents part is the second part. The structure of ANFIS of five layers which is shown in the fig.(1), where the fuzzification procedure is in the first layer, in the second layer the implementation of fuzzy and the rule by the fuzzy process, the normalized of the membership function is executed in the third layer, in the fourth layer the process of finding values of consequents part and the fifth layer is used to find the finally output of ANFIS [5].

$$O_i^1 = \mu_{A_i}(x) \quad \text{For } i = 1,2 \qquad (1)$$

$$O_i^2 = w_i = \mu_i(x) \times \mu_i(x), \quad i = 1,2 \qquad (2)$$

Where A_i is the fuzzy set of inputs, $\mu_{A_i}(x)$ is the membership function of x in A_i, $\overline{w}_i$ is the output of layer 3, and $\{p_i, q_i, r_i\}$ is the consequent parameters of this node i.

$$\overline{w}_i = \frac{w_1+w_2}{w_i}, \quad i = 1,2 \qquad (3)$$

$$O_i^4 = \overline{w}_i f_i = \overline{w}_i(p_i x_1 + q_i x_2 + r_i) \qquad (4)$$

$$O_i^5 = overalloutput = \sum_i \overline{w}_i f_i = \frac{\sum_i w_i f_i}{w_i} \qquad (5)$$

Fin ally, the values of the consequent parameters can we find by use a least square algorithm, as follow:

$O_1^5 = y = (w_1x_1)p_1 + (w_1x_2)q_1 + w_1r_1 + (w_2x_1)p_2 + (w_2x_2)q_2 + w_2r_2$

$$y = [w_1x_1 \quad w_1x_2 \quad w_3 \quad w_2x_1 \quad w_2x_2 \quad w_2]\begin{bmatrix} p_1 \\ q_1 \\ r_1 \\ p_2 \\ q_2 \\ r_2 \end{bmatrix} = \mathbf{XW}$$

….(6)

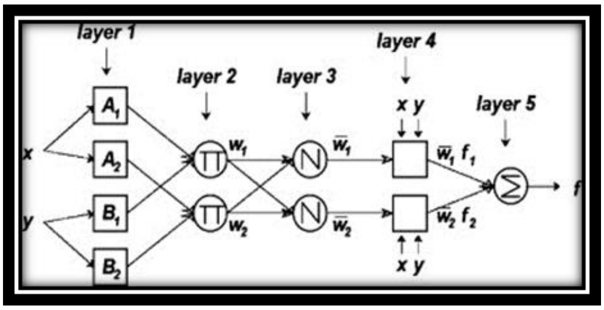

Figure (1)ANFIS architecture for a two-input, two-rule of first-order Sugeno model

III- The hybrid learning algorithm

Since ANFIS is a Neural Network then the parameters of ANFIS will adjusted with a finite number of iterations whenever the error is converge to a specified value. The hybrid learning algorithm is used to adjusted these parameters, in [5] was present procedure to update the parameters of ANFIS by hybrid learning algorithm. The basic learning method of ANFIS is a hybrid method which combined between gradient descent (GD) and least square estimate (LSE), since each iteration (epoch) in ANFIS consists of forward pass and backword passs; The set of the training data are entered into ANFIS, and then these inputs are processed to be neuron outputs. This is done layer by layer, after which rule consequent parameter values are determined using LSE [6]. The antecedent parameter are update by using GD.

Since the fuzzy inference in Takagi-Sugeno (TS) an output y is a linear function, then if we have K of linear equations in terms of the consequent parameters

$$\begin{cases} y(1) = \bar{w}_1(1)f_1(1) + \bar{w}_2(1)f_2(1) + \cdots + \bar{w}_n(1)f_n(1) \\ y(2) = \bar{w}_1(2)f_1(2) + \bar{w}_2(2)f_2(2) + \cdots + \bar{w}_n(2)f_n(2) \\ \vdots \\ y(k) = \bar{w}_1(k)f_1(k) + \bar{w}_2(k)f_2(k) + \cdots + \bar{w}_n(k)f_n(k) \end{cases} \quad (7)$$

where $f_i(j) = p_{i0} + p_{i1}x_1(j) + p_{i2}x_2(j) + \cdots + p_{im}x_m(j)$

s.t. $i, j = 1,2, \dots, k$

and the number of input variables is m and the number of neurons in the rule layer is n.wher

$$y = \begin{bmatrix} y(1) \\ y(2) \\ \vdots \\ y(k) \end{bmatrix}$$

and $y = AS$ (8)

s.t. A is $q \times n(1+m)$ matrix

$$A = \begin{bmatrix} \bar{w}_1(1) & \bar{w}_1(1)x_1(1) \dots \bar{w}_1(1)x_m(1) \dots \bar{w}_n(1) & \bar{w}_n(1)x_1(1) \dots \bar{w}_n(1)x_m(1) \\ \bar{w}_1(2) & \bar{w}_1(2)x_1(2) \dots \bar{w}_1(2)x_m(2) \dots \bar{w}_n(2) & \bar{w}_n(2)x_1(2) \dots \bar{w}_n(2)x_m(2) \\ & \vdots & \\ \bar{w}_1(k) & \bar{w}_1(k)x_1(k) \dots \bar{w}_1(k)x_m(k) \dots \bar{w}_n(k) & \bar{w}_n(k)x_1(k) \dots \bar{w}_n(k)x_m(k) \end{bmatrix}$$

and S is $n(1+m) \times 1$ vector

$S = [p_{10} \ p_{11} \ \dots \ p_{1m} \ p_{20} \ p_{21} \ \dots \ p_{2m} \dots p_{n0} \ p_{n1} \ \dots \ p_{nm}]$

The number of input and output patterns is usually q, which is used by training greater than the number of consequent parameters $n(1+m)$. This means that finding the solution of (eq.8) accurately may not be possible, so we use guessing using the method a least-square estimate of $S; S^*$, that minimises the squared error $\|AS - y\|^2$. It is done by using the pseudoinverse technique [6].

$S^* = (A^T A)^{-1}A^T y$ (9)

where A^T is the transpose of A, and $(A^T A)^{-1}A^T$ is the pseudoinverse of A if $(A^T A)$ is non-singular. In this case, the parameters of the rule consequent are estimated. Then, the forward pass of this epoch is complete and we can calculate the error vector as follow

$e = y_d - y$ (10)

In the backward pass, the error signals are propagated back and the rule antecedent parameters are updated by use gradient descent method: Calculate partial derivatives per chain rule

Since $E = \frac{1}{2}e^2 = \frac{1}{2}(y - y_d)^2$ (11)

and since we want to update the parameters of membership functions, may be triangular or may be trapezoidal or other, then if we suppose that the type of membership is trapezoidal, therefore we have four parameters $a, b, c, and\ d$

$$\Delta a = -\alpha\frac{\partial E}{\partial a} = -\alpha\frac{\partial E}{\partial e} \times \frac{\partial e}{\partial y} \times \frac{\partial y}{\partial(\bar{w}_i f_i)} \times \frac{\partial(\bar{w}_i f_i)}{\partial(\bar{w}_i)} \times \frac{\partial(\bar{w}_i)}{\partial(w_i)} \times \frac{\partial(w_i)}{\partial\mu_{A_i}(x)} \times \frac{\partial\mu_{A_i}(x)}{\partial a} \quad (12)$$

then,

$a = a + \Delta a$ (13)

as similar, we can update the other parameters $b, c, and\ d$.

The proposed training is also a hybrid of the two methods, the first is the least-squares method or the forward pass has not changed is the same as in the original of the ANFIS training that means that values of linear equations will be updated in the original method, but the change is in the backward pass, where update to the values of membership functions variables will be updated based on the Gray Wolfe Optimization (GWO) method instead of gradient descent in the beginning give initial population and initial values of these variables and so on in accordance with GWO steps are updating those values that reach specific resolution or a specific error is allowed.

IV- INTERVAL TYPE-2 ADAPTIVE NEURO-FUZZY INFERENCE SYSTEMS (IT2ANFIS)

IT2ANFIS with each of upper and lower MFs to be as two separable ANFISs, i.e. there are two ANFISs where the overall outputs of each of them are y upper (denotes y_u) and y lower (denotes y_l) and the overall output of IT2 ANFIS is the average of these two outputs. The system used in this study has four inputs x_1, x_2, x_3 and x_4 each in two fuzzy sets and one output y.

As previously mentioned that an ANFIS is multi-layers, ANFIS architecture consists of five layers, Fig. 1, the nodes in the first and fourth layers are adaptive, while the nodes in the other layers are fixed [3]. Note that a symbol $O_{l,i}$ will be used to refer to the ith node output in layer l. These layers can be briefly described, as used in this study and for each ANFIS in this model, as follows [7]:

Layer 1: The adapted of each node in this layer being by the following function

$O_{1,i} = \mu_{\tilde{A}_i}(x1) \quad i = 1,2 \quad (14)$

or $O_{1,i} = \mu_{\tilde{B}_{i-2}}(x2) \quad i = 3,4 \quad (15)$

or $O_{1,i} = \mu_{\tilde{C}_{i-4}}(x3) \quad i = 5,6 \quad (16)$

or $O_{1,i} = \mu_{\tilde{D}_{i-6}}(x4) \quad i = 7,8 \quad (17)$

In other words, each $O_{1,i}$ is a membership grade of a fuzzy set $\tilde{A}$ where $\tilde{A} = \tilde{A}_1, \tilde{A}_2, \tilde{B}_1, \tilde{B}_2, \tilde{C}_1, \tilde{C}_2, \tilde{D}_1 \text{ or } \tilde{D}_2$. Here an IT2 Gaussian MFs with uncertain standard deviation and fixed mean parameters of the layer 1 named **premise**.

Layer 2: Each node in this layer is labeled $\prod$, whose output is obtained using T-norm operators (product or min), that perform AND on all incoming signals, i.e. it the firing strength of every rule.

$O_{2,i} = w_i =$

$The\ firing\ strength\ of\ each\ rule\ i \quad (18)$

for all $i = 1, 2, \cdots, 16$

Layer 3: Each ith node in this layer, is labeled N, whose output is calculated by the following form:

$O_{3,i} = \overline{w}_i = \frac{w_i}{\sum_{i=1}^{16} w_i} \quad (19)$

where w_i is the ith rule's firing strength. Therefore the outputs here called **normalized firing strengths**.

Layer 4: Every ith node in this layer, is an adaptive node, as previously mentioned, with a node function:

$O_{4,i} = \overline{w}_i f_i = \overline{w}_i(c_{1i}x_1 + c_{2i}x_2 + c_{3i}x_3 + c_{4i}x_4 + c_{0i})$ …. (20)

where $\overline{w}_i$ is the output of layer 3 (a normalized firing strength) and c_{1i}, c_{2i}, c_{3i}, c_{4i} and c_{0i} are the parameters whose called **consequent**.

Layer 5: In this layer there is a single node denoted by Σ, which output is the overall output y computes as follows:

$y = O_{5,i} = \sum_{i=1}^{16} \overline{w}_i f_i = \frac{\sum_{i=1}^{16} w_i f_i}{\sum_{i=1}^{16} w_i} \quad (21)$

Since we are working with two ANFISs, hence there are two overall outputs, y_u and y_l. Therefor the overall output of the IT2 ANFIS is computing as follows:

$y = \frac{y_u + y_l}{2} \quad (22)$

V- The hybrid learning algorithm

As previously mentioned, an ANFIS is supervised system with a hybrid learning algorithm. In this study, a hybrid learning of Least-Squared Estimator (LSE), [7] and Grey Wolf Optimizer (GWO) used to optimize the consequent and antecedent parameters, respectively. Briefly, a GWO method can be described in these following steps [8]:

Calculate the coefficient vectors $\vec{A}$ and $\vec{C}$ as follows:

$\vec{A} = 2\vec{a} \cdot \vec{r}_1 - \vec{a} \quad (23)$

$\vec{C} = 2 \cdot \vec{r}_2 \quad (24)$

where $\vec{a}$ is linearly decreased function from 2 to 0 can be taken as:

$\vec{a} = 2\left(1 - \frac{epoch}{\max number\ of\ iterations}\right) \quad (25)$

for all epoch=1, 2, … , max number of iteration and random vectors $\vec{r}_1$ and $\vec{r}_2$ are their values are in the inteval [0,1].

Then calculate:

$\vec{D}_\alpha = |\vec{C}_1 \cdot \vec{m}_\alpha - \vec{m}| \quad (26)$

$\vec{D}_\beta = |\vec{C}_2 \cdot \vec{m}_\beta - \vec{m}| \quad (27)$

$\vec{D}_\delta = |\vec{C}_3 \cdot \vec{m}_\delta - \vec{m}| \quad (28)$

$\vec{m}_1 = \vec{m}_\alpha - \vec{A}_1 \cdot \vec{D}_\alpha \quad (29)$

$\vec{m}_2 = \vec{m}_\beta - \vec{A}_2 \cdot \vec{D}_\beta \quad (30)$

$\vec{m}_3 = \vec{m}_\delta - \vec{A}_3 \cdot \vec{D}_\delta \quad (31)$

$\vec{m}(epoch + 1) = \frac{\vec{m}_1 + \vec{m}_2 + \vec{m}_3}{3} \quad (32)$

$\vec{m}$, $\vec{m}_\alpha$, $\vec{m}_\beta$ and $\vec{m}_\delta$ will described in the pseudo code of LSE-GWO

The hybrid LSE-GWO pseudo code of its algorithm is as follows:

Initialize population of (GWO) m_i (which refer to the means) $(i = 1, 2, \cdots, n)$
Initialize the rest of T2 Gaussian MF parameters (σ_1 and σ_2)
Initialize a, A, and C
Compute the IT2 ANFIS consequent parameters using LSE
Compute the efficiency, RMSE, of every search agent
m_α=the first optimum search agent
m_β=the second optimum search agent
m_δ=the third optimum search agent
***for** each epoch (epoch=1:maximum no. of epochs)*
***for** every search agent*
Update the position of the current search agent by eq.(34)
end for
Update a, A, and C
Compute the consequent of all search agents by LSE
Update m_α, m_β and m_δ
end for
return m_α

Pseudo code of the LSE-GWO algorithm

Next, the experimental analysis and adaption (or simulation) results discussion presented. we used the root mean square error (RMSE) to be the implement criterion for all experiments which calculates as (Eyoh et al., 2017; Ren et al., 2011)

$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y^a - y)^2} \quad (33)$

where N is the training (or checking) data points number, y is the model output and y^a is the desired output.

VI- EXPERIMENTS AND RESULTS

To evaluate the IT2 ANFIS performance, we implemented it for a MG time series that is a well-known dataset which defined by a non-linear differential delay equation as following [9]:

$\dot{x}(t) = \frac{dx(t)}{dt} = \frac{a\,x(t-\tau)}{1+x^n(t-\tau)} - b\,x(t) \quad (34)$

s.t a, b and n are constant, t refers to the current time and τ is a time delay parameter non-negative constant used to produce the chaotic behavior in the data [10], which it turns to the chaotic when $\tau > 17$ [11]

To allow comparison with other works, the initial condition used in the method was $x(0) = 1.2$, the time step was 0.1, $a = 0.2$, $b = 0.1$, $n = 10$ and $\tau = 17$. Similar to [7] we extracted 1000 input-output data pairs from the chaotic MG time series $x(t)$ of the following format:

$[x(t-18), x(t-12), x(t-6), x(t); x(t+6)]$

where $x(t+6)$ is the output obtained from the input vector $[x(t-18), x(t-12), x(t-6), x(t)]$, where $t = 118$ to 1117. The dataset divided into two sets, the first 500 pairs used to be the training dataset for the IT2 ANFIS and the remaining 500 pairs of the data used to be the checking dataset for validating the identified IT2 ANFIS.

After iterating these two models many times, we had $RMSE_{trn} = 0.001683$ and $RMSE_{chk} = 0.001465$ from model I. While the results obtained from model II were $RMSE_{trn} = 0.000719$ and $RMSE_{chk} = 0.000551$ as shown in Table I. The results obtained from applying different approaches to Mackey-Glass prediction are listed in Table II.

TABLE II: The results comparison for prediction MG time series

Models	RMSE
ANFIS Ensemble with IT2FLS [12]	0.04933
IT2FLS [13]	0.0335
SA-T2FLS [9]	0.0089
IT2IFLS-TSK [11]	0.0079
ANFIS [7]	0.0015
The proposed method	0.001465

VII- CONCLUSION

The new hybrid LSE-GWO learning algorithm is used to optimize the IT2 ANFIS by searching for the best parameters of antecedents and consequents by using two models. The MG time series has been predicted by these proposed methods. Both type-1 and interval type-2 ANFIS, in their ability, have been compared to handle uncertainty and a good results had been obtained comparing with the other hybrid algorithms. In future work, we will use a new algorithm to optimize the parameters of the Interval type-2 ANFIS.

References

1.Zadeh, L.A., *Information and control.* Fuzzy sets, 1965. **8**(3): p. 338-353.

2.Zadeh, L.A., *The concept of a linguistic variable and its application to approximate reasoning—II.* Information sciences, 1975. **8**(4): p. 301-357.

3.Suparta, W. and K.M. Alhasa, *Modeling of tropospheric delays using ANFIS*. 2016: Springer.

4.Mohamad, A.A., N.K. Hussein, and I.A. Morbat, *Prediction of Fuzzy Sunspot Time Series By Using RBFANN.* Tikrit Journal of Pure Science, 2017. **22**(11).

5.Jang, J.-S., *ANFIS: adaptive-network-based fuzzy inference system.* IEEE transactions on systems, man, and cybernetics, 1993. **23**(3): p. 665-685.

6.Negnevitsky, M., *Artificial intelligence: a guide to intelligent systems*. 2005: Pearson Education.

7.Jang, J.-S.R., C.-T. Sun, and E. Mizutani, *Neuro-fuzzy and soft computing-a computational approach to learning and machine intelligence [Book Review]*. IEEE Transactions on automatic control, 1997. **42**(10): p. 1482-1484.

8.Mirjalili, S., S.M. Mirjalili, and A. Lewis, *Grey wolf optimizer.* Advances in engineering software, 2014. **69**: p. 46-61.

9.Almaraashi, M. and R. John. *Tuning of type-2 fuzzy systems by simulated annealing to predict time series*. in *Proceedings of the world Congress on Engineering*. 2011.

10.Hassan, S., et al. *Hybrid model for the training of interval type-2 fuzzy logic system*. in *International Conference on Neural Information Processing*. 2015. Springer.

11.Eyoh, I., R. John, and G. De Maere, *Interval type-2 intuitionistic fuzzy logic system for non-linear system prediction.* 2016.

12.Soto, J., P. Melin, and O. Castillo. *A new approach for time series prediction using ensembles of ANFIS models with interval type-2 and type-1 fuzzy integrators*. in *Computational Intelligence for Financial Engineering & Economics (CIFEr), 2013 IEEE Conference on*. 2013. IEEE.

13.Castillo, O., et al. *Type-2 fuzzy logic: theory and applications*. in *Granular Computing, 2007. GRC 2007. IEEE International Conference on*. 2007. IEEE.

خوارزمية الذئب الرمادي وطريقة تقدير المربعات الصغرى كخوارزمية مهجنة جديدة لتدريب النظام (IT-II-ANFIS)

نزار خلف حسين ، بلقيس خالد فرج

قسم الرياضيات ، كلية علوم الحاسوب والرياضيات ، جامعة تكريت ، تكريت ، العراق

الملخص

خوارزمية الذئب الرمادي هي واحدة من خوارزميات الفوق حدسية وهي من الخوارزميات التي تستخدم بشكل واسع في التطبيقات الهندسية والاقتصادية. خوارزمية الذئب الرمادي مع طريقة تقدير المربعات الصغرى استخدمت كخوارزمية تدريب مهجنة لتدريب واختيار المعلمات المثلى في نظام (IT-II_ANFIS). تم اختبار خوارزمية التدريب الجديدة باستخدام النظام (IT-II_ANFIS) في التنبأ بالقيم لمتسلسلة ميكي– كلاص الزمنية وقد كانت النتائج مشجعة مقارنة بالخوارزميات الأخرى.