

# Identification of Nonlinear System Based on Additive Legendre Neural Network

Bin Yang<sup>1 +</sup>

<sup>1</sup> School of Information Science and Engineering, Zaozhuang University, Zaozhuang, China 277160

**Abstract.** In this paper, an efficient identification method based on additive Legendre neural network (ALNN) model and hybrid evolutionary method is proposed to identify nonlinear systems. In order to improve efficiency of Legendre neural network (LNN), additive Legendre neural network is proposed. For finding the optimal structure and parameters of ALNN model, a new hybrid evolutionary method based on binary particle swarm optimization (BPSO) algorithm and firefly algorithm is employed. Two nonlinear system identification experiments are used to test ALNN model. The results reveal that ALNN model performs better than LNN and other classic neural networks.

**Keywords:** additive, Legendre neural network, binary particle swarm optimization, firefly algorithm.

## 1. Introduction

In order to understand and model the internal mechanism of real nonlinear systems, we usually face the problem of nonlinear system identification. System identification is modeling procedure where the mathematical representation of the output, or past input or both, can be obtained with the observed data.

Artificial neural networks (ANNs) are powerful mathematical methods that can be used to learn complex non-linear continuous functions, and have been successfully applied to identify nonlinear system in the past decades. Su et al. proposed a double-hidden layer neural network to identify a nonlinear dynamic system [1]. Juang et al. proposed a TSK-type recurrent fuzzy network (TRFN) structure for dynamic system identification [2].

Due to that traditional neural network has some disadvantages such as low efficiency, long learning time and easy to fall into the local minimum solution, Legendre neural network (LNN) was proposed. LNN model has no hidden layer and could add dimensionality of the input layer with a set of nonlinear functions. Thus LNN model has been applied for time series prediction and system identification [3-4].

The structure of Legendre neural network is very simple and learning speed is fast. But the number of input Legendre polynomials is large due to the fact that each input variable has  $n$  order Legendre polynomials. The structure of LNN is fixed and only task is to optimize the parameters of LNN. To reduce the optimization complexity and improve efficiency, this paper proposes a novel Legendre neural network model, namely additive Legendre neural network (ALNN). Binary particle swarm optimization (BPSO) algorithm is proposed to select proper input Legendre polynomials in order to construct proper structure. Firefly algorithm is used to optimize the parameters of ALNN. Two nonlinear systems are used to evaluate the performance of ALNN.

## 2. Method

### 2.1. Structure of ALNN

---

<sup>+</sup> Corresponding author. Tel.: +18369296801.  
E-mail address: batsi@126.com.

Legendre neural network (LNN) was first proposed by Yang and Tseng for function approximation in 1996 [3]. LNN has less parameter and does not have hidden layer, which uses Legendre orthogonal polynomials as the activation functions of hidden layer neurons. Due to the absence of hidden layer, LNN provides computational advantage over the MLP.

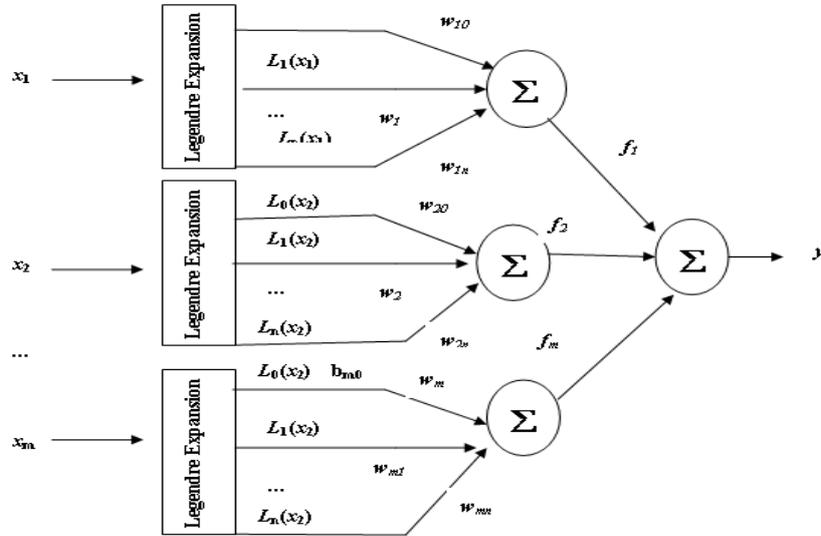


Fig.1: The structure of ALNN.

In order to select proper Legendre polynomials for specific problems, additive Legendre neural network (ALNN) is proposed. The structure of an ALNN is shown in Fig. 1 [5]. Suppose that input vector  $[x_1, x_2, \dots, x_m]$  and  $n$  order Legendre polynomials.

$$\begin{cases} L_0(x) = 1 \\ L_1(x) = x \\ L_2(x) = \frac{1}{2}(3x^2 - 1) \\ \vdots \\ L_n(x) = \frac{1}{n}[(2n-1)xL_{n-1}(x) - nL_{n-2}(x)] \end{cases} \quad (1)$$

The output of  $f_i$  is defined as followed.

$$f_i = \tanh\left(\sum_{k=0}^n L_k(x_k) \times b_{ik} \times w_{ik}\right) \quad (2)$$

Where  $\tanh()$  is Hyperbolic tangent,  $b_{ik}$  is Boolean value (0 or 1). When  $b_{ik}$  is equal to 1, Legendre polynomial  $L_0(x_k)$  is selected as input data and weight  $w_{ik}$  is assigned.

The final output  $y$  is defined as followed.

$$y = \sum_{k=1}^m f_k \quad (3)$$

## 2.2. Structure Optimization of ALNN

Additive Legendre neural network could not allow all Legendre polynomials as input data. It uses evolutionary method to select proper Legendre polynomials. In this paper, binary particle swarm optimization (BPSO) is used to gain the optimal vector  $[b_{10}, b_{11}, \dots, b_{1n}, \dots, b_{m0}, b_{m1}, \dots, b_{mn}]$ .

In BPSO, the moving trajectory and velocity of each particle is defined in term of probability. The moving trajectory represents changes of probabilities of a certain value. The moving velocity is defined as probability of a state or another state. Thus each bit  $x_i(t)$  of one particle is restricted to 0 or 1. Suppose that ALNN has  $m$  input variables and  $n$  order Legendre polynomials. The length of each particle is  $m \times (n+1)$ .

Each  $v_i(t)$  represents the probability of bit  $x_i(t)$  taking the value 1. A new velocity  $v_i(t)$  for particle  $i$  is updated as same as PSO, which is defined as followed.

$$v_i(t+1) = w * v_i(t) + c_1 r_1 (Pbest_i - x_i(t)) + c_2 r_2 (Gbest - x_i(t)). \quad (4)$$

Where  $w$  is the inertia weight,  $c_1$  and  $c_2$  are positive constants and  $r_1$  and  $r_2$  are uniformly distributed random number.  $Pbest_i$  is the best fitness of particle  $i$  and  $Gbest$  is the best position among all particles.

$x_i(t)$  is calculated as followed [6].

$$x_i(t) = \begin{cases} 1, & r < Sig(v_i(t)) \\ 0, & other. \end{cases} \quad (5)$$

Where  $r$  is created randomly from range  $[0,1]$ , and the function  $Sig$  is defined as followed.

$$Sig(v_i(t)) = \frac{1}{1 + e^{-v_i(t)}}. \quad (6)$$

### 2.3. Parameters Optimization of ALNN

According to the optimal structure of ALNN, tally the number ( $p$ ) of 1 in the optimal particle  $[x_1, x_2, \dots, x_{m*(n+1)}]$  of BPSO.  $p$  parameters  $[w_1, w_2, \dots, w_p]$  need be optimized. Firefly algorithm (FA) is an efficient optimization algorithm [7]. It is very simple, has few parameters and easy to apply and implement, so this paper uses firefly algorithm to optimize the parameters of Legendre neural network.

Firefly algorithm is the random optimization method of simulating luminescence behavior of firefly in the nature. The firefly could search the partners and move to the position of better firefly according to brightness property. A firefly represents a potential solution. In order to solve optimization problem, initialize a firefly vector  $[x_1, x_2, \dots, x_n]$  ( $n$  is the number of fireflies). As attractiveness is directly proportional to the brightness property of the fireflies, so always the less bright firefly will be attracted by the brightest firefly.

The brightness of firefly  $i$  is computed as

$$B_i = B_{i0} * e^{-\gamma r_{ij}} \quad (7)$$

Where  $B_{i0}$  represents maximum brightness of firefly  $i$  by the fitness function as  $B_{i0} = f(x_i)$ .  $\gamma$  is coefficient of light absorption, and  $r_{ij}$  is the distance factor between the two corresponding fireflies  $i$  and  $j$ .

The movement of the less bright firefly toward the brighter firefly is computed by

$$x_i(t+1) = x_i(t) + \beta_i (x_j(t) - x_i(t)) + \alpha \varepsilon_i \quad (8)$$

Where  $\alpha$  is step size randomly created in the range  $[0, 1]$ , and  $\varepsilon_i$  is gaussian distribution random number.

## 3. Experiments

### 3.1. The First Nonlinear System

The nonlinear system to be identified is described by

$$y(k) = 0.72y(k-1) + 0.025y(k-2)u(k-2) + 0.01u^2(k-3) + 0.2u(k-4). \quad (1)$$

Where  $y(k)$  is the output of the system at the  $k$ -th time point and  $u(k)$  is the plant input.

In order to make the comparison fairly, the input signals used for training ALNN model are same as in literatures [2], which is an iid uniform sequence over  $[-2,2]$  for about half of the 900 time points and the remaining data is given by  $1.05 \sin(\pi \frac{k}{45})$ . To test the performance of identification model, the following input is used for test.

$$u(k) = \begin{cases} \sin(\pi \frac{k}{25}) & k < 250 \\ 1.0 & 250 \leq k < 500 \\ -1.0 & 500 \leq k < 750 \\ 0.3 \sin(\pi \frac{k}{25}) + 0.1 \sin(\pi \frac{k}{32}) + 0.6 \sin(\pi \frac{k}{10}) & 750 \leq k < 1000 \end{cases} \quad (2)$$

Fig.2 gives the identification performance among actual data, LNN and ALNN models. From the Fig.2, it can be seen that identification systems are nearly as same as actual one, but the error by ALNN is smaller than the one by LNN. The RMSE values of identification system using some different methods for testing data are listed in Table 1. From Table 1, it is clear that ALNN model performs better than other classic neural network models.

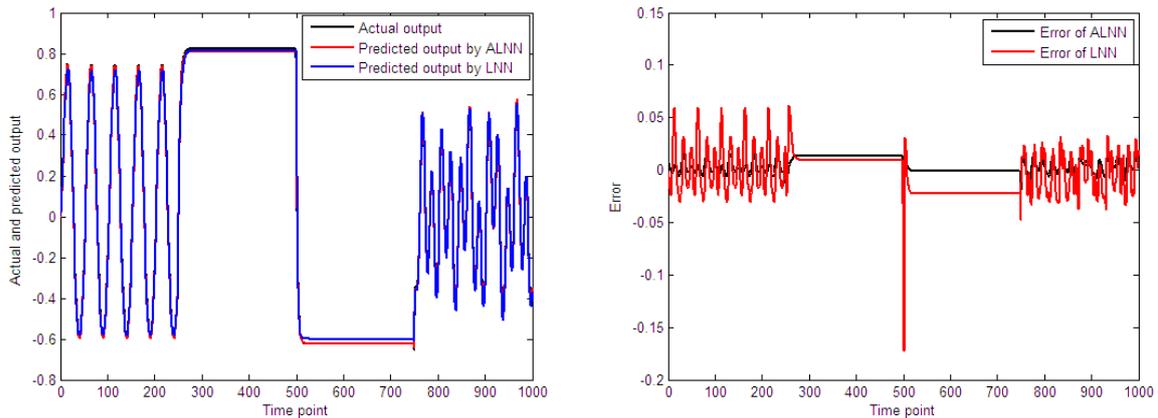


Fig. 2: Predicted data (left) and predicted error (right) by LNN and ALNN for Example 1.

Table 1: Comparison of different models for Example 1.

| Model           | ERNN [8] | RSONFIN [9] | TRFN-S [2] | FWNN [8] | FWNN-inline-PSO [8] | FWNN [8] | FWNN-inline-PSO [8] | LNN     | ALNN            |
|-----------------|----------|-------------|------------|----------|---------------------|----------|---------------------|---------|-----------------|
| RMSE of testing | 0.078    | 0.06        | 0.0313     | 0.0201   | 0.0313              | 0.0201   | 0.02015             | 0.02015 | <b>0.008283</b> |

### 3.2. The Second Nonlinear System

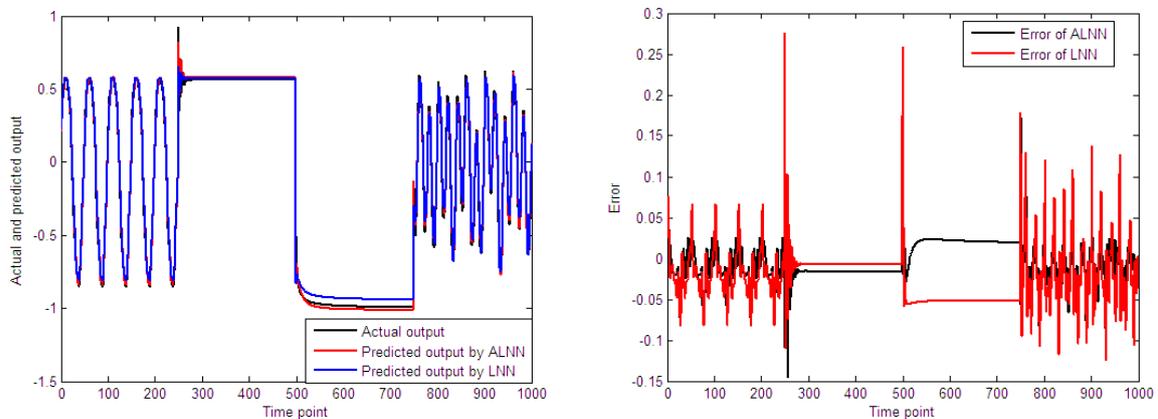


Fig. 3: Predicted data (left) and predicted error (right) by LNN and ALNN for Example 2.

In this section, the plant to be identified is described by the following equation:

$$y(k+1) = f(y(k), y(k-1), y(k-2), u(k), u(k-1)), \quad (3)$$

where

$$f(x_1, x_2, x_3, x_4, x_5) = \frac{x_1 x_2 x_3 x_5 (x_3 - 1) + x_4}{1 + x_3^2 + x_2^2}. \quad (4)$$

In order to make the comparison fairly, the training and testing signals are same as in literatures [2]. Fig.3 gives the predicted performance among actual data, LNN and ALNN models. We also can see that predicted output by LNN and ALNN are nearly as same as actual one. From Fig.3, it can be seen that the predicted error by ALNN is smaller than the one by LNN. The RMSE values of identification system using six different methods for testing data are listed in Table 2. From Table 2, it is clear that ALNN model performs better than other proposed neural network models.

Table 2: Comparison of different models for Example 2.

| Model           | ERNN [8] | RSONFIN [9] | TRFN-S [2] | FWNN [8] | LNN    | ALNN          |
|-----------------|----------|-------------|------------|----------|--------|---------------|
| RMSE of testing | 0.0575   | 0.0780      | 0.0346     | 0.0301   | 0.0430 | <b>0.0284</b> |

## 4. Conclusions

In order to improve efficiency of Legendre neural network, this paper propose a novel Legendre neural network model, namely additive Legendre neural network. ALNN utilizes binary particle swarm optimization algorithm to select proper input Legendre polynomials in order to determine the struture of ALNN. Firefly algorithm is used to optimize the parameters of ALNN model. Two nonlinear system identification experiments are used to test ALNN model. The results reveal that ALNN model performs better than LNN and other classic neural networks.

## 5. Acknowledgments

This study was funded by the PhD research startup foundation of Zaozhuang University (No.2014BS13).

## 6. References

- [1] Y. Su, Y. SHEEN. Neural network for system identification. *International Journal of Systems Science*. 1992, **23**(12): 2171-2186.
- [2] C. F. Juang. A TSK-Type Recurrent Fuzzy Network for Dynamic Systems Processing by Neural Network and Genetic Algorithms. *IEEE Transactions on Fuzzy Systems*. 2002, **10**(2): 155-170.
- [3] A. Q. Pei, J. Wang, W. Fang. Predicting agent-based financial time series model on lattice fractal with random Legendre neural network. *Soft Computing*. 2015, 1-16.
- [4] J. C. Patra, C. Bornand. Nonlinear dynamic system identification using Legendre neural network. *International Joint Conference on Neural Networks. IJCNN 2010*, 2010, 18-23.
- [5] J. C. Patra, P.K. Meher, G. Chakraborty. Nonlinear channel equalization for wireless communication systems using Legendre neural networks. *Signal Processing*. 2009, **89**(11): 2251-2262.
- [6] H. Nezamabadi-pour, M. Rostami. Binary Particle Swarm Optimization: challenges and New Solutions. *The Journal of Computer Society of Iran (CSI) On Computer Science and Engineering (JCSE)*. 2008, **6**(1-A): 21-32.
- [7] X. S. Yang. Firefly algorithms for multimodal optimization, *Stochastic Algorithms: Foundations and Applications. Lecture Notes in Computer Sciences*. 2009, **5792**: 169-178.
- [8] R. Cheng, Y. P. Bai. A novel approach to fuzzy wavelet neural network modeling and optimization. *International Journal of Electrical Power & Energy Systems*. 2015, **64**: 671-678.
- [9] C. F. Juang, C. T. Lin. A recurrent self-organizing neural fuzzy inference network. *IEEE Trans Neural Networks*. 1999, **10**(4): 828-45.