Computer and Information Science
www.ccsenet.org/journal.html

Vol. 2, No. 2
May 2009

Design of the Nonlinear System Predictor Driven by the Bayesian-Gaussian Neural Network of Sliding Window Data

Yijian Liu
School of Electrical & Automation Engineering, Nanjing Normal University, Nanjing 210042, China
Department of Automation, Wuhan University, Wuhan 430072, China
Yanjun Fang
Department of Automation, Wuhan University, Wuhan 430072, China
Tel: 86-27-6877-6862 E-mail: liuyijian_2002@163.com

The research is financed by the National Natural Science Foundation of China (No.60704024 & No.60772107). (Sponsoring information)

Abstract

The model identification of the nonlinear system has been concerned by the industrial community all along. The relationship of the nonlinear dynamic system is contained in the data accumulated in the scene. To better utilize the data about the industrial objects, in this article, we put forward the nonlinear system predictor driven by the Bayesian-Gaussian neural network (NN) model, use the trained threshold matrix and sliding window data to realize the online output prediction for the nonlinear dynamic system. The simulation experiment indicates that the Bayesian-Gaussian NN based on the sliding window data can fulfill the demands of the online identification and prediction of the adaptive nonlinear system.

Keywords: Sliding window data, Bayesian-Gaussian neural network, Predictor, Nonlinear

1. Introduction

Most industrial control objects are nonlinear objects with time-varying, time-lag and saturation, so the input and output relationship model of the controlled system can not be exactly established. And general classic control method is designed based on the exact model of the system, so the model is difficult to be established and many antinomies exist in many control designs. On the other hand, in the dynamic running process, much input and output data will be produced, and these data are exterior representations of the nonlinear structure characteristics of the system, and these data can help us to establish the structure model of the system.

The confirmation of the input and output nonlinear structure model of the nonlinear dynamic system is the identification problem essentially, and it is composed by the identification model with proper parameters and the performance function which adjusts the parameters through optimizing the errors between the unknown system identification and the model output (Zhang, 2000, P.566-568). The NN model is a sort of effective function approximation tool, and it has been applied in the nonlinear system identification (Li, 2001, P.499-502, Zeng, 2009, P.2293-2300, Yan, 2007, P.232-236). In theory, any three-layer forward NN can approximate any nonlinear function, but the disadvantage of the NN is that the confirmation of the hidden layer mainly depends on the experiments and experiences, and if the network weight parameters are too much, the adjustment process of the weights will get in the local minimum. Otherwise, when the structure character of the nonlinear dynamic system changes, the trained NN model always can not fit the nonlinear system after structure change, so the NN must be retrained, and the pure forward NN doesn't adapt the time-varying identification and prediction of the dynamic system.

However, in the response process of the nonlinear dynamic system, large numbers of input and output data have described the structure characters of the nonlinear dynamic system from the exterior. As viewed from the probability theory, the structure character of the dynamic system should be included in the relationship of these data. Based on Bayesian inference and Gaussian hypothesis, in this article, we put forward a sort of Bayesian-Gaussian NN reasoning model based on sliding window data which can integrate sliding window data into the structure of the reasoning model. Only through confirming same threshold matrix parameters with the nonlinear system, we can use the historical data in the sliding window to realize the output prediction of the present system, and when the structure of the system changes, we can realize the online follow identification output of the system.

2. Description of nonlinear dynamic system

In the nonlinear dynamic system of the discrete time seen in Figure 1, suppose the system is stable, and the input and output nonlinear relationship of the system is

$$y(k) = f(y(k-1), y(k-2), \cdots, y(k-n); u(k), u(k-1), \cdots, u(k-m)) \tag{1}$$

Where, $y(k)$ denotes the output of the k'th step of the system, $y(k-i)$ (i=1, 2… n) denotes the system output of the former n steps, $u(k)$ denotes the input of the k'th step of the system, $u(k-i)$ (i=1, 2… m) denotes the system control inputs of the former m steps, f denotes the dynamic relationship between input and output of the dynamic system, and the nonlinear function relationship can be approximated by the identification method, and the target of the article is to use the Bayesian-Gaussian NN model based on sliding data window to approximate the structure of the nonlinear function f and the online identification and prediction of the dynamic system.

3. Bayesian-Gaussian NN based on sliding window data

Suppose the input vector of the nonlinear system identification model can be denotes as

$$X_k = [\hat{y}(k-1), \hat{y}(k-2), \cdots \hat{y}(k-n), u(k), u(k-1), \cdots, u(k-m)]^{\mathrm{T}} \tag{2}$$

X_k denotes the input of the system at the k'th sampling, and it is the column vector with n+m+1 lines. The output $Y_k = \hat{y}(k)$ is real number, and the input and output relationship of the system can denoted as

$$Y_k = f(X_k) \tag{3}$$

Based on historical input and output data, utilizing Bayesian inference and Gaussian hypothesis, the Bayesian-Gaussian model can realize the prediction $\hat{y}(k)$ of the system output $y(k)$, and the superscript "Λ" denotes the identification output of the model.

3.1 Deducing of Bayesian-Gaussian reasoning model

Suppose (X_i, y_i) (i=1, 2… N) is the sample set of the training, X_i is the sampling input of the i'th step, and it denotes the column vector of the m'th line, X_i=(X_{i1}, X_{i2},……,X_{im})$^{\mathrm{T}}$. y_i denotes the output of the system, and based on Bayesian inference and Gaussian hypothesis, the output y can be generated by the method of probability under the new input X.

3.1.1 When the single historical data (X_i, y_i) is known, what is the probability that X exports Y?

Under Gaussian hypothesis, Y possesses the probability density function p(Y) and fulfills the Gaussian normal distribution Y~N (y_0, σ_0^2), where y_0 is the mean value and σ_0^2 is the variance. Suppose Y is appointed, Y_i fulfils the normal distribution Y_i~N(Y, σ_i^2) and possesses the probability density function p(y_i|Y=y).

$$p(Y) = \frac{1}{\sqrt{2\pi}\sigma_0} e^{-\frac{1}{2}\frac{(Y-y_0)^2}{\sigma_0^2}} \tag{4}$$

$$p(y_i \mid Y = y) = \frac{1}{\sqrt{2\pi}\sigma_i} e^{-\frac{1}{2}\frac{(Y-y_i)^2}{\sigma_i^2}} \tag{5}$$

And the Bayesian theorem is

$$p(Y \mid y_i) = \frac{p(Y).p(y_i \mid Y)}{p(y_i)} \tag{6}$$

Substitute above (4) and (5) into (6), and simplify it and we can obtain

$$p(Y \mid y_i) = c_1 \frac{1}{\sqrt{2\pi}\sigma_{0,i}} e^{-\frac{1}{2}\frac{(Y-y_{0,i})^2}{\sigma_{0,i}^2}} \tag{7}$$

Where, c_1 is the normalization parameter, and the mean parameter $y_{0,i}$ and the variance parameter $\sigma_{0,i}^2$ can be expressed as

$$\sigma_{0,i}^{-2} = \sigma_0^{-2} + \sigma_i^{-2} \tag{8}$$

$$y_{0,i} = \sigma_{0,i}^2 (\sigma_0^{-2} y_0 + \sigma_i^{-2} y_i) \tag{9}$$

3.1.2 When the historical data sample (X_i, y_i) ($i = 1,2,\cdots,N$) is known, what is the probability that X exports Y?

Suppose the prior probability of y_i to Y is $p(Y \mid y_i)$, y_i and $y_j (i, j = 1,2,\cdots,N, i \neq j)$ are independent each other under the appointed condition Y, so the conditional probability that N data samples generate the output Y for the new input X is

$$p(Y \mid Y_1, Y_2, \cdots, Y_N) = k \frac{\prod_{i=1}^{N} p(Y \mid Y_i)}{p^{N-1}(Y)} \tag{10}$$

K is the normalization constant independent with Y and Ye Haiwen's ariticle (Haiwen Ye, 1999, P.21-36) gives the proof process in detail.

3.1.3 Bayesian-Gaussian reasoning model

Substitute (7) into (10), we can obtain

$$p(Y \mid Y_1, Y_2, \cdots, Y_N) = c_2 \frac{\prod_{i=1}^{N} \frac{1}{\sqrt{2\pi}\sigma_{0,i}} e^{-\frac{1}{2}\frac{(Y-y_{0,i})^2}{\sigma_{0,i}^2}}}{\left(\frac{1}{\sqrt{2\pi}\sigma_0}\right)^{N-1} e^{-\frac{N-1}{2}\frac{(Y-y_0)^2}{\sigma_0^2}}} \tag{11}$$

c_2 is a normalization constant independent of Y, because the distribution of the prior probability approximate as the constant, so the prior variance σ_0^2 is big, and (8) and (9) can be respectively approximated as $\sigma_{0,i}^{-2} = \sigma_i^{-2}$ and $y_{0,i} = y_i$.

Under the Gaussian hypothesis, (11) can be simplified as

$$\begin{aligned} p(Y \mid Y_1, Y_2, \cdots, Y_N) &= c_3 \frac{1}{\sqrt{2\pi}} \prod_{i=1}^{N} \frac{1}{s_i} e^{-\frac{1}{2}\frac{(Y-y_{ii})^2}{s_i^2}} \\ &= c_3 \frac{1}{\sqrt{2\pi}} \prod_{i=1}^{N} \frac{1}{\sigma_i} e^{-\frac{1}{2}\sum_{i=1}^{N}\frac{Y^2-2y_iY+y_i^2}{\sigma_i^2}} \\ &= c_4 \frac{1}{\sqrt{2\pi}\sigma(N)} \prod_{i=1}^{N} \frac{1}{\sigma_i} e^{-\frac{1}{2}\sum_{i=1}^{N}\frac{(Y-y'(N))^2}{\sigma(N)^2}} \end{aligned} \tag{12}$$

In the above formula, c_4 is the normalization constant independent of Y, and the estimated mean $y'(N)$ and the variance $\sigma(N)$ are respectively expressed as

$$Y'(N) = \sigma(N)^2 \sum_{i=1}^{N} \sigma_i^{-2} y_i \tag{13}$$

$$\sigma(N)^{-2} = \sum_{i=1}^{N} \sigma_i^{-2} \tag{14}$$

Suppose the variance fulfills (15)

$$\sigma_i^{\ 2} = \sigma_0^{\ 2} e^{(X-X_i)^T D(X-X_i)} \tag{15}$$

In (15), D is called as the threshold matrix.

$$D=\begin{bmatrix} d_{11}^{-2} & & \\ & d_{jj}^{-2} & \\ & & d_{mm}^{-2} \end{bmatrix} \tag{16}$$

Therefore, the formulas (13), (14) and (15) composes the Bayesian-Gaussian reasoning model, and the parameters of the whole model mainly include the threshold matrix D and the initial estimation variance σ_0^2, and the dimension of the threshold matrix is equal to the input amount of the nonlinear dynamic system, so the parameters which need to be confirmed from the network are few, and the operation time of the reasoning model can be largely saved.

3.2 Bayesian-Gaussian NN

Based on above Bayesian-Gaussian reasoning model, we can obtain the Bayesian-Gaussian NN seen in Figure 2, and it adopts the nerve cell nodes (seen in Figure 3) as same as general NN, and the network includes five layers.

The first layer: Store present system input, $X=[x_1,x_2,\cdots,x_m]$.

The second layer: Store N groups historical input data samples, and each group of sample includes *m* input variables. For the *j*'th node in the *i*'th group, its input and output relationship can be expressed as

$$s_{ij}^{[2]}=x_j,\quad f^{[2]}(s_{ij}^{[2]})=\frac{(s_{ij}^{[2]}-x_{ij})^2}{d_{jj}^2} \tag{17}$$

The superscript "$^{[2]}$" denotes the second layer of the Bayesian-Gaussian NN, and the corresponding third layer and the fourth layer are denoted as "$^{[3]}$" and "$^{[4]}$". The threshold matrix parameter of the second layer has been included in the encouragement function. From the experiment process, the N groups of historical input data samples are very important to the prediction of the system, and to reduce the operation of the Bayesian-Gaussian NN and follow the dynamic responses on line, we adopt the sliding window data method to select the historical input data in N groups.

The third layer: In N nodes, the *i*'th node corresponds with the *i*'th input sample in the second layer, and the input and output relationship is denoted as

$$s_i^{[3]}=\sum_{j=1}^{m}\frac{(x_j-x_{ij})^2}{d_{jj}^2},\quad f^{[3]}(s_i^{[3]})=\sigma_0^{-2}e^{-s_i^{[3]}}=\sigma_i^{-2} \tag{18}$$

The fourth layer: Includes two nodes and the relationship of the first node and the second layer can be expressed as

$$s_1^{[4]}=\sum_{i=1}^{N}y_i\sigma_i^{-2},\quad f_1^{[4]}(s_1^{[4]})=s_1^{[4]} \tag{19}$$

$$s_2^{[4]}=\sum_{i=1}^{N}\sigma_i^{-2},\quad f_2^{[4]}(s_2^{[4]})=s_2^{[4]} \tag{20}$$

The fifth layer: Includes two nodes and the input output relationships are

$$s_2^{[5]}=\sum_{i=1}^{N}\sigma_i^{-2},\quad f_2^{[5]}(s_2^{[5]})=\frac{1}{s_2^{[5]}}=\sigma(N)^2 \tag{21}$$

$$s_1^{[5]}=\sum_{i=1}^{N}y_i\sigma_i^{-2},\quad f_1^{[5]}(s_1^{[5]})=\frac{s_1^{[5]}}{s_2^{[5]}}=y'(N) \tag{22}$$

3.3 Working procedure of Bayesian-Gaussian NN based on sliding window data

The work process can be divided into the network off-line training and the online prediction application, and the Bayesian-Gaussian NN training is mainly to confirm the threshold matrix parameter D, and the online prediction application is to predict the present system output by N groups of historical input sample, and N groups of prediction sample set adopts the sliding window method to confirm, and above two approaches can be respectively described as follows.

3.3.1 Off-line training of Bayesian-Gaussian NN

First, to the N_l training sample (X_i,y_i), $i=1,2,\cdots,N_1$, use the following performance evaluation function

$$V_N(D)=\frac{1}{2N_1}\sum_{i=1}^{N_1}(y_i-\hat{y}_i)^2 \tag{23}$$

Where, y_i denotes the actual system output, $\hat{y}_i$ denote other N_l-1 training samples except for X_i, use (13) and (15) to train the Bayesian-Gaussian NN and obtain the prediction value.

The target of the train is to find out proper threshold matrix D which can make the output of actual system and the prediction value better fit, and make (23) to be least or fulfill the application precision demand of the engineering.

Above process is the process to minimize the formula (23), and we can adopt the optimization algorithm based on the grads such as the least square method and the simplex method (Yin, 2003, P.135-137, 145), and we can also adopt the genetic algorithm, the ant colony algorithm, the particle swarm optimization and other random evolutionary optimization algorithms which have been deeply researched and applied in recent years (Guo, 2003, P.70-73, Aaron, 2005, P.175-191, Susuki, 2008, P.249-253). According to the foraging process of the colon bacillus (Liu, 2007, P.991-994), we put forward the improved foraging optimization algorithm (seen in (24) and (25)), and validate they can be used to optimize these parameters through the experiment.

$$X_i^{n+1} = X_i^n + C(i)\phi(n) \tag{24}$$

$$X_i^{n+1} = X_i^n + w_1 r_1^n (P_i^{pbest} - X_i^n) + w_2 r_2^n (G^{gbest} - X_i^n) \tag{25}$$

The concrete contents and symbol parameters of the improved foraging optimization algorithm are in Liu's article (Liu, 2007, P.991-994), and in this article, we use the improved foraging optimization algorithm to optimize the threshold matrix parameter in (23), and the concrete optimization includes following six approaches.

Approach 1: Initialize relative parameters, and the approach includes optimizing the field range of the parameter θ, the step number of the chemical trend N_c, the step number of the walking operator N_s, the step length $C(i)$, the number of the species group S, the initial position of *E.Coli* X_i ($i = 1,2,\cdots,S$), the weighted coefficient w_1, w_2 and the condition that the algorithm ends in advance.

Approach 2: To every *E.Coli* individual, update the position by (23), and evaluate the adaptive function $eval_i^{n+1}$.

Approach 3: If $eval_i^{n+1} < eval_i^n$, suppose the walking counter $Counter$ is 0, and keep the walking direction $\phi(n)$ unchangeable, implement the walking operator until the walking step number N_s achieves maximum or doesn't fulfill $eval_i^{n+1} < eval_i^n$. And update $Pbest_i$, P_i^{pbest}, $Gbest$ and G^{gbest}.

Approach 4: If $eval_i^{n+1} > eval_i^n$, update the position of *E.Coli* by the formula (25).

Approach 5: If fulfilling the end condition, quit from the computation, or else, go on.

Approach 6: Go to the next chemical trend step.

Through above optimization process, we can obtain the threshold matrix D, and realize the training and learning process of the nonlinear dynamic system.

3.3.2 Bayesian-Gaussian NN based on sliding window data

Through above network training by the threshold matrix D, to realize the online prediction in the dynamic response process of the system, Yinli (Yin, 2003, P.135-137, 145) adopted the self-adjusted Bayesian-Gaussian NN model to sustain the constant of the number of N. Suppose there are N historical input data samples (X_i, y_i), *i*=1,2,…,*N*, when add one sample (X_μ, y_u), predict the *i*'th sample by other N samples, and compute the mean square prediction error (MSPE) of the *i*'th sample.

$$MSPE_i = E(y_i - Y_i(N))^2 + \sigma_i(N)^2 \tag{26}$$

If one sample can be predicted by other data samples, so its MSPE computed by (26) should be small, i.e. the sample can be obtained by the prediction from other samples. So we can eliminate the same from N+1 samples and keep the number of the input data samples of the online prediction unchangeable.

Above self-adjusted process of the Bayesian-Gaussian NN can bring extra computation time, especially when the input data sample number N is numerous. So the self-adjusted method has deficiencies for the online prediction application of the nonlinear dynamic system.

In this article, we use the sliding data window to confirm the input samples of Bayesian-Gaussian NN in the online prediction application. The data near the present time contribute most to the output of the present system, i.e. the data sample near the present time can predict the present output with higher precision.

The aim adopting the sliding data window is to sustain the prediction data sample scale N unchangeable for the Bayesian-Gaussian NN when predicting the output y, and the concrete method is seen in Figure 4.

Figure 4 shows three windows, and the data sample quantity of every sliding window, i.e. the width of the sliding window is N, and in the change from window 1 to window 2, only eliminate the data which is farthest to the present time of the window 1, and compose the window 2 with the data sample which is nearest to the present time, and the change from window 2 to window 3 is similar with above process, and in this way, the sliding window data form.

4. Simulation experiments

To validate the online identification effect of the Bayesian-Gauss NN based on sliding data window, we adopt following two nonlinear dynamic systems to test.

4.1 Single input and single output nonlinear system

$$y(t) = \sin(x) + e, \quad x \in (0, 2\pi) \tag{27}$$

Where, e is the random white noise, and it is the zero mean in the experiment and its variance is 0.2. Suppose the structure of the system changes from the 320th sampling time, and it becomes into $y(t) = 2\sin(2x) + e$, and to the 450th sampling time, it becomes into the original system.

The system samples 600 numbers and the input and output curve is seen in the real line of Figure 5.

First, we use the former 300 training samples in Figure 5, and the initial parameters of the foraging optimization algorithm include S=10, N_c=200, w_1=0.3, w_2=0.2, and the initial walking step length is 0.02, and because the system is only relative wit h the input x, so there is one parameter d_{11} in the threshold matrix of the Bayesian-Gaussian NN model to be confirmed, so the parameter dimension in the foraging optimization algorithm p is 1. The threshold matrix improved by the foraging optimization algorithm D is [43.3729], and the performance index $V_N(D) = 0.0128$.

Then we utilize the improved threshold matrix parameters to dynamically predict the output of the nonlinear system, and the adopted window width N is 10, and we can obtain the following curve of broken line in Figure 5. From Figure 5, the Bayesian-Gaussian NN based on sliding window can realize the prediction and the structure change following to the dynamic nonlinear system.

4.2 Multiple inputs and single output nonlinear system

For the nonlinear dynamic system,

$$\begin{aligned} y(t) = {} & (0.8 - 0.5e^{(-y^2(t-1))})y(t-1) - (0.3 + 0.9e^{(-y^2(t-1))})y(t-2) \\ & + u(t-1) + 0.2u(t-2) + 0.1u(t-1)u(t-2) + e(t) \end{aligned} \tag{28}$$

Where, $y(t)$ denotes the system output of the present time, $y(t-i), i = 1,2$ denotes the system output of the former two steps, $u(t-i), i = 1,2$ denotes the system control input of the former steps, so the input of the Bayesian-Gaussian NN $X = [y(t-1), y(t-2), u(t-1), u(t-2)]^T$, so the corresponding threshold diagonal matrix D = diag $(d_{11}^{-1}, d_{22}^{-1}, d_{33}^{-1}, d_{44}^{-1})$. The control variable u is the random sequence which mean is 2 and its variance is 1, and $e(t)$ is the Gaussian white noise which mean is 0 and its variance is 0.04.

From the formula (28), 100 sampling numbers are taken as the training samples, and the initial parameters of the improved foraging optimization algorithm include S=20, N_c=80, w_1=0.2, w_2=0.2, and the initial walking step length is 0.01, the X of the Bayesian-Gaussian NN model include four inputs, so there are four parameters to need being confirmed, and the parameter dimension in the foraging optimization algorithm p is 4. The threshold matrix improved by the foraging optimization algorithm D=diag[28.6618, 45.5806, 36.7235, 27.7467], and the performance index $V_N(D) = 0.0247$.

From the 101st time, the system structure changes, and the input output formula is changed to the formula (29).

$$\begin{aligned} y(t) = {} & (0.8 + 0.5e^{(-y^2(t-1))})y(t-1) - (0.3 - 0.9e^{(-y^2(t-1))})y(t-2) \\ & + u(t-1) + 0.2u(t-2) + 0.1u(t-1)u(t-2) + e(t) \end{aligned} \tag{29}$$

We utilize the threshold matrix D obtained by the improved foraging optimization algorithm and select 20 as the width of the sliding data window N to implement online prediction of the system, and we can obtain the broken line output in Figure 6.

5. Analysis and conclusions

From above simulation experiment of the nonlinear dynamic system, we can see that the Bayesian-Gaussian NN based on the sliding data window can fully utilize the window data to realize the online prediction of the system and acquire better effect of the online prediction and follow. From the experiment, we can also see that the Bayesian-Gaussian NN can better adapt the structure change of the nonlinear dynamic system, because the Bayesian-Gaussian NN can integrate window data into the structure, sustainably update the structure of the network through the continual sliding of the window, and quickly capture the change of the nonlinear system structure, and this character is attractive for the dynamic system which characters change often, and it can adapt the online prediction application for the nonlinear dynamic system.

References

Aaron C. Zecchin, Angus R. Simpson and etc. (2005). Parametric study for an ant algorithm applied to water distribution system optimization. *IEEE Transactions on Evolutionary Computation.* No.9(2). P.175-191.

Guo C.Y., He G. & Chen G.J. (2003). Application of genetic algorithm in identifying nonlinear regressive model. *Journal of Naval university of Engineering.* No.15(2). P.70-73.

Haiwen Ye & Weidou Ni. (1999). Nonlinear system identification using a Bayesian-Gaussian neural network for predictive control. *Neurocomputing.* No.28. P.21-36.

Li, Lirong, Shen, Chunlin & Hanmei. (2001). Identification Method of Thermal Process Based on BP Neural Network. *Journal of Nanjing University of Aeronautics & Astronautics.* No.33(5). P.499-502.

Liu, Yijian & Fang, Yanjiu. (2007). Simplified E.Coli foraging optimization algorithm and its application to parameter identification of nonlinear system model. *Control Theory & Applications.* No.24(6). P.991-994.

Susuki, Ryohei, Kawai Fukiko and etc. (2008). Parameter optimization of model predictive control using PSO. *SCIE Annual Conference.* No.1. P.249-253.

Yan Cai & Chao Gao. (2007). Nonlinear Modeling of Switched Reluctance Motor Based on BP Neural Network. *Third International Conference on Natural Computing.* No.1. P.232-236.

Yinli & Wang, Huiwen et al. (2003). Two Typical Applications of the Related Partial Least Square Algorithms in the Field of System Modeling. *Journal of System Simulation.* No.15(1). P.135-137, 145.

Zeng Yuhong & Huai Wenxin. (2009). Application of artificial neural network to predict the friction factor of open channel flow. *Communications in Nonlinear Science and Numerical Simulation.* No.14(5). P.2293-2300.

Zhang, Zhihua, Zheng, Nanning & Zhao, Zhenxuan. (2000). The Nonlinear Dynamic System Identifier Based on Fuzzy Bilateral NN. *Control Theory & Applications.* No.17(4). P.566-568.

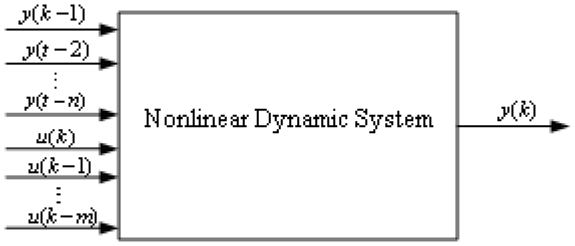

Figure 1. Nonlinear Dynamic System

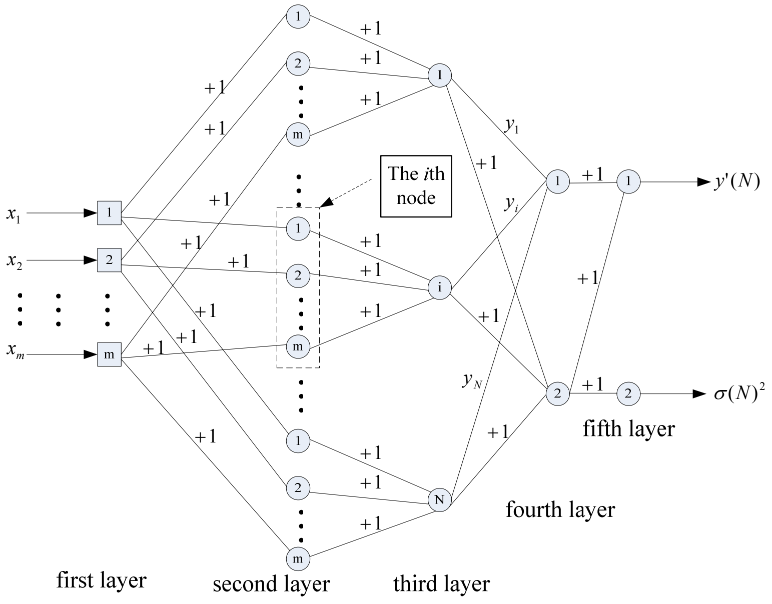

Figure 2. BGNN Network Model

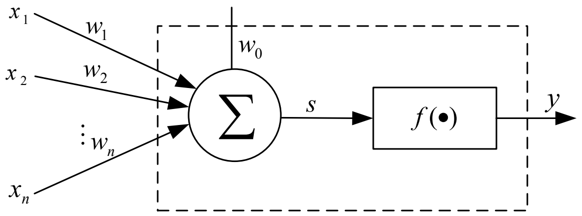

Figure 3. Nerve Cell Node Structure

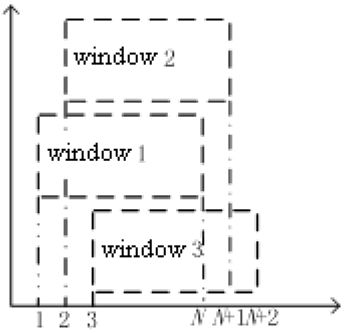

Figure 4. Work Process of Sliding Window

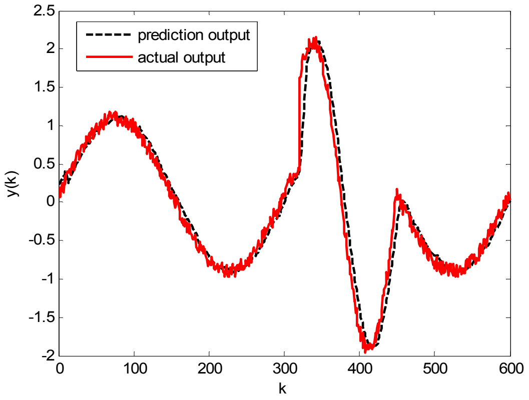

Figure 5. Online Prediction of SISO Nonlinear System

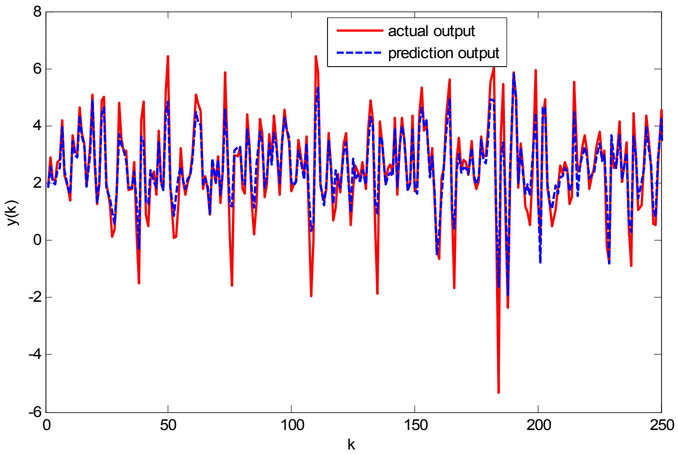

Figure 6. Online Prediction of MISO Nonlinear System