Fibers and Polymers 2006, Vol.7, No.1, 62-65

Real-time Identification of the Draft System Using Neural Network

Soon Yong Chun[1], Han Jo Bae, Seon Mi Kim, Moon W. Suh[2], P. Grady[2], Won Seok Lyoo, Won Sik Yoon, and Sung Soo Han*

School of Textiles, Yeungnam University, Gyeongsan 712-749, Korea
[1]School of IT Electronic Engineering, Dongyang University, Yeongju 750-711, Korea
[2]College of Textiles, North Carolina State University, Raleigh, NC 27695, USA
(Received August 30, 2005; Revised December 23, 2005; Accepted February 27, 2006)

Abstract: Making a good model is one of the most important aspects in the field of a control system. If one makes a good model, one is now ready to make a good controller for the system. The focus of this thesis lies on system modeling, the draft system in specific. In modeling for a draft system, one of the most common methods is the "least-square method"; however, this method can only be applied to linear systems. For this reason, the draft system, which is non-linear and a time-varying system, needs a new method. This thesis proposes a new method (the MLS method) and demonstrates a possible way of modeling even though a system has input noise and system noise. This thesis proved the adaptability and convergence of the MLS method.

Keywords: Draft system, Sliver, Control, Neural network, Modeling

Introduction

In designing the controller of draft systems, the accurate system modeling has a significant effect on the performance of the controller. In particular, the issue of identifying the dynamic behavior of draft system has long been researched as an important subject particularly by many researchers in the area of textile machine control [1-3].

Theory

Until now most methods used in draft system modeling were based on the algorithm of the least-square method. When parameters are estimated using the least-square method, estimation is impossible in time-varying non-linear systems. However, because a draft system has very high non-linearity, a new algorithm is required in modeling the system. The present study suggested a modified least square method (MLS method) with improved adaptability and convergence, assuming that a draft system is a probabilistic system that has uncertainty in the behavior of slivers according to the irregularity of input slivers as well as temperature and humidity. Figure 1 is a block diagram that estimates parameters using the existing least-square method.

Figure 2 is a block diagram of MLS method proposed in this study. Here, x is the thickness of input slivers and y is the thickness of output slivers. The input and output of the system are used to learn the neural network method by the estimator having the algorithm of MLS method, and the result of learning is taken as the initial value of the least-square method and, as a result, it becomes possible to estimate the parameter. Recently the high performance of computer has reduced the learning time of the neural network method considerably, which makes possible the real-time estimation of the initial value of the least-square method. The MLS method based on the neural network method perceives the value of input data and estimates parameters fit for the value and, accordingly, it is possible to estimate parameters in real-time and the system is highly adaptable to environmental changes in irregularity, temperature, humidity, etc. [4].

*Corresponding author: sshan@yu.ac.kr

Experiments

Estimator Using Least Square Method

The least-square method algorithm was proposed by Johann K. Gauss in order to estimate the trajectory of planets.

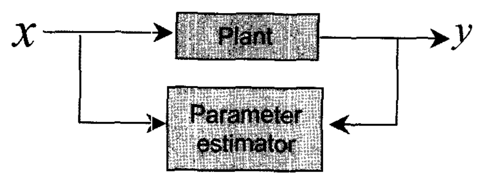

Figure 1. Block diagram of least-square method.

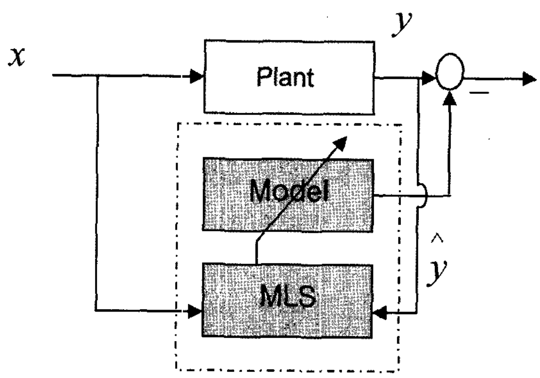

Figure 2. Block diagram of modified least-square method (MLS method).

This is widely used as a tool for defining the parameters of feedback approximation [5,6].

Estimation function $\hat{\theta}(N)$ can be expressed as equation (1) by the least-square method.

$$\hat{\theta}(N) = (X_N^T X_N)^{-1} X_N^T y_N \quad (1)$$

$\hat{\theta}(N)$ can be calculated using input data and output data, namely, $x(N)$ and $y(N)$, which are actual measurement data. In addition, we can calculate $\hat{\theta}(N+1)$ using $x(N+1)$ and $y(N+1)$. $\hat{\theta}(N)$ estimates parameters through iterative mathematical development.

The numerical expression is as follows;

$$\hat{\theta}(N+1) = \hat{\theta}(N) + \alpha(N+1)P(N) \\ x(N+1)[y(N+1) - x^T(N+1)\hat{\theta}(N)] \quad (2)$$

where, $P(N)$ is the initial covariance and $\alpha(N+1)$ is the weight. When estimating value $\hat{\theta}(N+1)$ using $P(N)$ and $\alpha(N+1)$, estimation is possible to some degree in case the irregularity of input slivers and the variation of temperature and humidity are ignored, but the estimated value cannot be used under the influence of these factors. Figure 3 shows the result of parameter estimation according to the irregularity of input slivers. This is a graph of estimating the value of a parameter, the virtual target of which is 0.277. The input used Gaussian noise having different covariances. Input covariances were 0, 0.003, 0.006 and 0.009 respectively.

As the result of simulation shows, although only the irregularity of input slivers was considered without considering system noise, when covariance was 0.009 parameter estimation was impossible.

Figure 4 shows the result when system noise was 0.002 without considering the irregularity of input slivers (that is, covariance is 0). In the figure, the estimated value diverges. Thus, even if the irregularity of input slivers is not considered, if system noise is considered, it is impossible to estimate parameters with the least-square method. Accordingly, we need an estimation method applicable even when the irregularity of input slivers and system noise are considered and this is possible through the real-time initial value estimation of the least-square method.

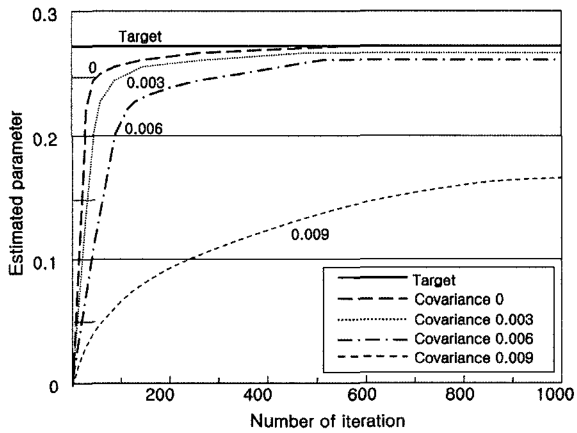

Figure 3. The graph of parameter estimation, using Gaussian noise that has four different error covariances.

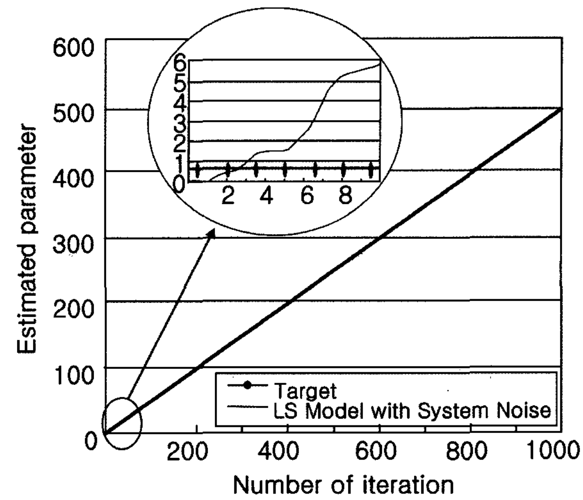

Figure 4. The result of parameter estimation, which has the model with system noise.

Neural Network Method

The neural network method is a numerical expression of calculation that humans do [7-11].

First, the human brain is basically composed of neurons and the neurons are firmly connected with one another through axons. Between the neurons, namely, between axons and dendrites is a space called synapse, through which signals can be intensified or attenuated. The synapse is called weight (the intensity of connection) in the neural network method.

Figure 5 shows how to express biological neurons in

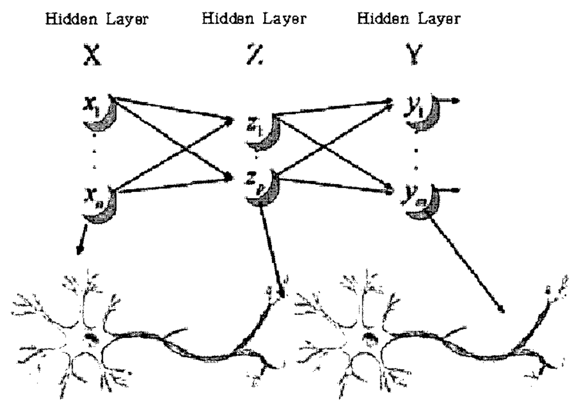

Figure 5. Structure of multi-layered artificial neural network.

mathematical structure.

Figure 5 used a hidden layer, n input neurons and m output neurons. Such a multi-layered neural network method uses various learning methods and one of common methods is back propagation learning.

In the multi-layered neural network method, back propagation learning can be expressed as follows.

$$NET_Z = XV^T \quad (3)$$

$$NET_y = ZW^T \quad (4)$$

NET_Z is the sum of weights in the hidden layer, and the output of the hidden layer is Z. NET_y is the sum of the output layer and the final output is y. Here X ($X = [x_1 \ldots x_n]$) is input vector, Z ($Z = [z_1 \ldots z_p]$) hidden layer vector, and y ($y = [y_1 \ldots y_m]$) output layer vector. V is the weight between the input layer and the hidden layer, and W weight between the hidden layer and the output layer.

When the neural network method is learned, first learning pattern pairs are selected and weights V and W are initialized with small random values. Then the learning rate is set at an appropriate value larger than 0. Next, the weights are changed by inputting the learning pattern pairs in order.

Equation (5) shows error signals δ_y and δ_z.

$$\delta_y = (d-y)y(1-y)$$

$$\delta_z = f(NET_z)\sum_{i=1}^{m}\delta_y W \quad (5)$$

The variation of weight between the hidden layer and the output layer at k^{th} learning step is estimated as follows. ΔW^k shows the variation of weight between the hidden layer and the output layer, and ΔV^k the variation of weight between the input layer and the hidden layer. γ is a weight.

$$\Delta W^k = \gamma\delta_y Z$$

$$\Delta V^k = \gamma\delta_z X \quad (6)$$

Equation (7) estimates the weight in the k+1th learning step. The weight between the hidden layer and the output layer is W^{k+1} and the weight between the input layer and the hidden layer is V^{k+1}.

$$W^{k+1} = W^k + \Delta W^k$$

$$V^{k+1} = V^k + \Delta V^k \quad (7)$$

The weights are changed iteratively, and when error E ($E = 1/2(d-y)^2$) converges within the limit of allowable error learning process ends.

In this research, back propagation was used in learning, and in the simulation the allowable error limit was 0.05, initial learning value 0.16, the number of iteration 5,000, and the number of neurons in the hidden layer 3 in considering of learning speed and convergence speed.

Modified Least-square Method

The output of a draft system is significantly affected by the irregularity of input slivers. Thus, in this study, training on the irregularity of input slivers was made through learning based on the neural network method in order for the system to recognize each sliver.

In the entire flow of MLS method, the structure of the neural network method set the initial weight, initial value, input value and target value and, using these values, estimates the output of the hidden layer and the output layer. The error is calculated using the estimated output value, and if the error is larger than the initially set target error, the weight is changed. In this research, the allowable error limit was 0.05 and weight was changed until the error was smaller than the error limit.

After learning ends, $P(N)$ in the least-square method is reset with the initial value obtained through the neural network method. The least-square method estimates the initial parameter using an expression containing the initial input and output values and noise. The value is reset with the initial value obtained through the neural network method. The result of neural network learning is used in resetting the initial value of the least-square method, and the output of the model having the estimated parameter is compared with the real output, and if the output is below the set value the learning stops. If the system changes by any elemental factor, learning is resumed so that the estimated output of the model approaches to the real output.

In this research, a draft system was modeled using ARMA model as in equation (8).

$$\begin{aligned} y_k = {} & 0.277y_{k-1} + 0.15548y_{k-2} + 0.135y_{k-3} \\ & + 0.3y_{k-4} - 0.0247x_{k-1} + 0.00945x_{k-2} \\ & + 0.3475x_{k-3} - 0.0399x_{k-4} + w(t) \end{aligned} \quad (8)$$

y_k is k^{th} output and $y_{k-1}, y_{k-2}, y_{k-3}, y_{k-4}$ are k-1th, k-2th, k-3th and k-4th output respectively. $x_{k-1}, x_{k-2}, x_{k-3}, x_{k-4}$ are k-1th, k-2th, k-3th and k-4th input respectively. $W(t)$ is system noise.

MLS method was used for biquadratic ARMA model to estimate the parameter. The figure shows parameter estimation in a draft system containing uncertainty.

Here, axis x is the number of iteration and axis y is the parameter value.

Figure 6 shows the result of estimating the parameter with target value of 0.277 using MLS method for a model containing system noise. The solid line is parameter estimation in a

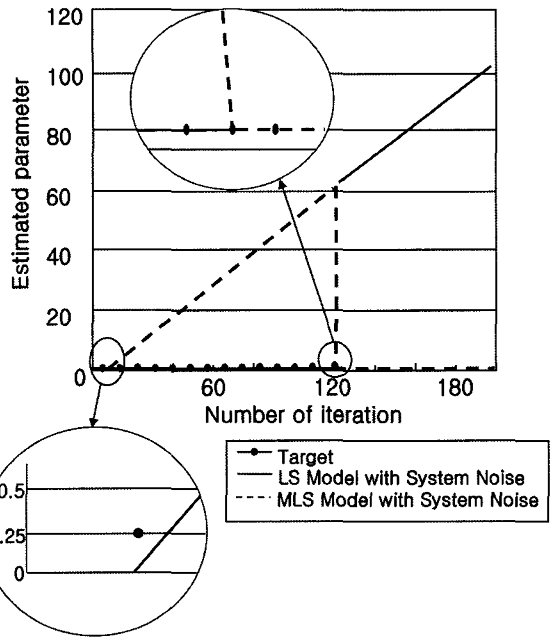

Figure 6. Identification of system parameter (The first parameter value is 0.277).

draft system considering non-linearity using the least-square method. Here, the system noise was 0.002, input noise used Gaussian noise and error covariance was 0.003.

As shown in the figure, the result of parameter estimation in the least-square method marked with a solid line does not converge when the irregularity of input slivers and system noise are considered. Therefore, as suggested earlier, it is impossible to estimate the parameter using the ordinary least-square method. On the other hand, the dotted line in the figure follows the least-square method when the learning of the neural network method has not been completed but it converges accurately as soon as the learning is completed. When it is examined closely in the magnified figure, the estimated parameter value converges exactly to 0.277.

As the figure shows, convergence was improved when the parameter was estimated using MLS method. We confirmed that accuracy and convergence were superior when MLS algorithm was used in estimating eight system parameters for slivers with high irregularity compared to when the least-square method was used.

Conclusions

The present study proposed MLS algorithm as a modeling technique suitable for draft systems.

In modeling non-linear draft systems, draft systems using the neural network method showed higher adaptability and convergence in modeling than those using the least-square method.

This study showed through computer simulation that real-time perception is possible in draft systems using MLS algorithm, and the algorithm can be applied to actual draft systems.

As MLS algorithm was used, it became possible to perceive plants containing probabilistic noise in draft systems, and we could derive a more practically usable system model than modeling methods proposed in previous researches. In addition, as the modeling of draft systems is possible through fast convergence, we can control draft systems to obtain high-quality slivers.

Acknowledgement

This work was supported by grant No. RTI04-01-04 from the Regional Technology Innovation Program of the Ministry of Commerce, Industry, and Energy (MOCIE).

References

1. S. S. Han, *J. Korean Fiber Soc.*, **33**, 1067 (1996).
2. Y. Huh, W. S. Kwon, and S. H. Jang, *J. Korean Fiber Soc.*, **37**, 171 (2000).
3. B. Choi and Y. Huh, *J. Korean Fiber Soc.*, **32**, 300 (1995).
4. R. Johansson, "System Modeling Identification", pp.74-132, Prentice Hall Inc., 1993.
5. A. A. Giordano and F. M. Hsu, "Least Square Estimation with Applications to Digital Signal Processing", John Wiley, 1985.
6. D. C. Park, Identification of Stationary/Non-stationary System Using Artificial Neural Networks, M.S., 1990.
7. R. Rojas, "Neural Networks", Springer, 1996.
8. F. H. She, L. X. Kong, S. Nahauandi, and A. Z. Kouzani, *Text. Res. J.*, **72**, 594 (2002).
9. C. C. Huang and K. T. Chang, *Text. Res. J.*, **71**, 987 (2001).
10. D. W. Patterson, "Introduction to Artificial Intelligence and Expert Systems", Prentice Hall, 1990.
11. N. B. Karayiannis and A. N. Venetsanopoulos, "Artificial Neural Networks : Learning Algorithms, Performance Evaluation, and Applications", Kluwer Academic, 1993.