Proceedings of the ASME 2013 Conference on Information Storage and Processing Systems
ISPS2013
June 24-25, 2013, Santa Clara, California, USA

ISPS2013-2948

Neural-Network Based Automatic PID Gain Tuning in the Presence of Time-Varying Disturbances in Hard Disk Drives

Liting Sun[1,2]*, Xu Chen[2], Masayoshi Tomizuka[2]

[1]University of Science and Technology of China, Hefei, Anhui, 230027, China
litingsun@me.berkeley.edu

[2]University of California, Berkeley
Department of Mechanical Engineering, University of California, Berkeley, CA, 94720, USA
Email: {maxchen, tomizuka}@me.berkeley.edu

Introduction

In hard disk drive (HDD) systems, disturbances commonly contain different frequency components that are time-varying in nature. Different HDD systems may subject to different excitation disturbances. In this case, it is difficult for fixed-gain PID controllers to maintain a good overall performance. When the characteristics of the disturbances change, or when servos are designed for different drive products, the PID gains have to be re-tuned. This paper presents automatic online gain tuning of PID controllers based on neural networks. The proposed control scheme can automatically adjust the PID parameters online in the presence of time-varying disturbances, or for different disturbances among different HDD products, and find the optimal sets of PID gains through the self-learning ability of neural networks.

Neural network has been widely applied in control systems due to its self-learning ability, adaptability, and nonlinear mapping between its inputs and outputs. Gao et. al [1] and Zhang et. al [2] both adopted an adaptive PID controller based on neural network in an electro-hydraulic position system and a crank angular speed control system where coupling effect between the motor and mechanism was considered. High adaptability and strong robustness of the controller were observed. Seung [3] used a primary PID controller and an auxiliary controller based on neural network for learning and compensating for the inherent nonlinearities in a pneumatic servo system, which improved the tracking performance and enhanced the robustness of the controller. Cheng et. al [4] applied the neural-network-based-PID controller in a nanopositioning system driven by ultrasonic motor and achieved a positioning accuracy of 10nm. Yang et. al [5] effectively improved the positioning performance of a micro-positioning stage using an adaptive neural-fuzzy PID controller, which improved the positioning accuracy from 92nm to 23nm at the X axis and from 102nm to 28nm at the Y axis.

In this paper, the self-learning ability of the neural network is employed to improve the adaptability and robustness of the PID controller with time-varying or product-dependent disturbances. The PID gains will be automatically tuned and optimized online for different HDD products with different disturbance characteristics.

The Structure of Neural-Network based PID Controller

Fig.1 shows the structure of neural-network based PID controller and the structure of a two-layer back-propagation neural network (BPNN) [6]. In Fig.1(b), $G_p(z^{-1})$ represents the dynamics of the HDD system

*Corresponding author: Liting Sun

and $d(k)$ is the disturbance signal. Our goal is to automatically tune the PID gains for a better regulation performance to match the current disturbance characteristics. In the neural-network based PID controller, PID gains are the outputs of the back-propagation neural network, as shown in Fig.1(a).

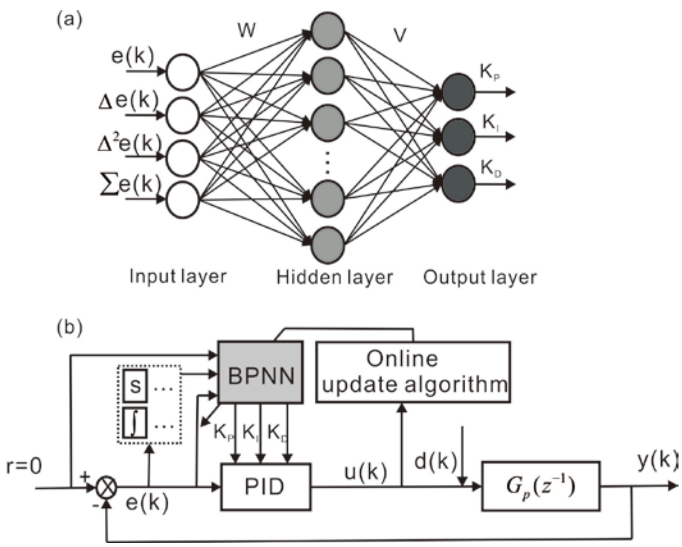

Fig.1 Structure of a two-layer back-propagation neural network (a) and the neural-network based PID controller (b)

The central idea for the BPNN based PID controller is to find a set of PID parameters that minimizes the cost function

$$E = e^2(k)/2$$

through the online update process of the neural network parameters, denoted as W and V in Fig. 1(a). $e(k)$ in the feedback loop depends on PID gains $[K_p(k),\ K_I(k),\ K_d(k)]$, which are functions of the weighting parameters W and V. Hence $e(k)$ depends on W and V. It thus makes sense to minimize the cost function with respect to W and V.

As shown in Fig.1(a), the structure of the BPNN adopted in this paper is 4-7-3. The input nodes include the position error signal (PES), the derivative of PES, the second differential of PES and the sum of PES. The order of the hidden layer is selected based on the empirical formula [7, 8]. In selecting this value, many factors such as the training algorithm, the training dataset, the input and the output neuron numbers and the complexity of the activation functions have to be considered. There is no guide about how to compute the optimal order of the hidden layer and it is obtained by trimming the network size without degrading its performance. It should be noted that the order of the hidden layer has nothing to do with the order of the plant to be controlled. This is one benefit of the neural network as it requires no information on the plant model.

The working principle of the BPNN can be expressed as follows.

Input layer: for each node i, the input and the output are

$$y_{1,i} = x_{1,i},\ i = 1,2,3,4; \tag{1}$$

where $x_{1,i}$ are $e(k)$, $\Delta e(k)$, $\Delta^2 e(k)$, $\sum e(k)$.

Hidden layer: for each node j, the input and the output relation is

$$\begin{cases} y_{2,j} = f_1(x_{2,j}) = f_1(\sum_{i=1}^{4} y_{1,i} w_{ij}) \\ f_1(x) = \dfrac{e^x - e^{-x}}{e^x + e^{-x}} \end{cases} \tag{2}$$

where w_{ij} is the updated weight coefficients that connect the nodes in the input and the hidden layers, and $f_1(x)$ is called the activation function of the hidden layer.

Output layer: for each node k, the input and the output relation is

$$\begin{cases} y_{3,k} = f_2(x_{3,k}) = f_2(\sum_{j=1}^{7} y_{2,j} v_{jk}) \\ f_2(x) = \dfrac{1}{1+e^{-x}} \end{cases} \tag{3}$$

where the weight coefficient v_{jk} connects the nodes in the hidden and the output layers, and $f_2(x)$ is the activation function of the hidden layer.

Online Update Algorithm. In order to automatically tune the PID gains in the presence of time-varying disturbances or dramatic disturbance changes, the weight coefficients w_{ij} and v_{jk} are updated online to minimize the PES. By using the idea that is similar in adaptive inverse control, the mean square error is first approximated by the instantaneous error square $E = e^2(k)/2$, and the update law is derived by using the steepest descent method and the backward error propagation concept:

$$
\begin{aligned}
\Delta v_{jk} &= -\eta \frac{\partial E}{\partial v_{jk}} = -\eta \frac{\partial E}{\partial e}\frac{\partial e}{\partial u}\cdots\frac{\partial y_{3,k}}{\partial x_{3,k}}\frac{\partial x_{3,k}}{\partial v_{jk}} = \eta e\cdot sgn(\frac{\partial e}{\partial u})\, f_2{}' y_{2,j} \\
\Delta w_{ij} &= -\eta \frac{\partial E}{\partial w_{ij}} = -\eta \frac{\partial E}{\partial y_{2,j}}\frac{\partial y_{2,j}}{\partial x_{2,j}}\frac{\partial x_{2,j}}{\partial w_{ij}} = -\eta\, f_1{}' y_{1,i}\frac{\partial E}{\partial y_{2,j}} \\
&= \eta\, f_1{}' y_{1,i}\sum_k(-e\cdot sgn(\frac{\partial e}{\partial u})\frac{\partial y_{3,k}}{\partial x_{3,k}})\left(\frac{\partial(\sum_j v_{jk} y_{2,j})}{\partial y_{2,j}}\right) \\
&= \eta e\cdot sgn(\frac{\partial e}{\partial u})\, f_1{}' y_{1,i}\sum_k y_{2,j}\, f_2{}' v_{jk}
\end{aligned}
\tag{4}
$$

$$
\begin{aligned}
w_{ij}(k+1) &= w_{ij}(k) + \eta\Delta w_{ij} + m_c(w_{ij}(k) - w_{ij}(k-1)) \\
v_{jk}(k+1) &= v_{jk}(k) + \eta\Delta v_{jk} + m_c(v_{jk}(k) - v_{jk}(k-1))
\end{aligned}
\tag{5}
$$

In Eq. (4), $sgn(x)$ is the sign function: i.e. $sgn(x)=1$ if $x>0$ and $sgn(x)=-1$ if $x<0$. η is the learning rate of the weight coefficients and m_c is the momentum adopted to avoid local minima.

Application to Time-varying Disturbance Rejection

To illustrate the effectiveness of the BPNN based automatic PID gains tuning process, the HDD benchmark [9] is used to study disturbance rejection with time-varying frequencies. We take a piecewise sinusoidal signal as the disturbance signal, as shown in Fig.2. Each segment has a different frequency, ranging from 20Hz to 200Hz.

Fig.3 shows a comparison of the results using a traditional fixed-gain PID controller and a BPNN based PID controller with automatic gain tuning. It can be seen that the BPNN based PID controller exhibits a better overall rejection performance with time-varying disturbances. This comes at the cost of a larger computational amount because the neural network weights have to be updated online to tune the PID[1].

Fig.4 also shows the tuning process of the PID parameters: K_p, K_i and K_d. As the frequency of the disturbance varies, the PID gains are tuned automatically and converge to different values for different disturbance characteristics.

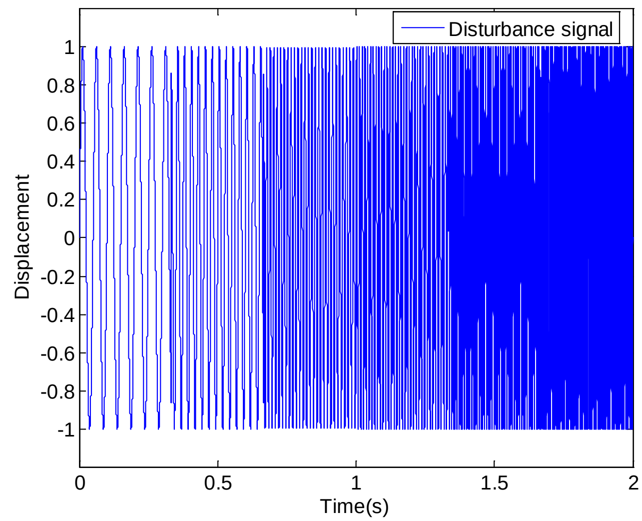

Fig.2 Disturbance signal with time-varying frequencies

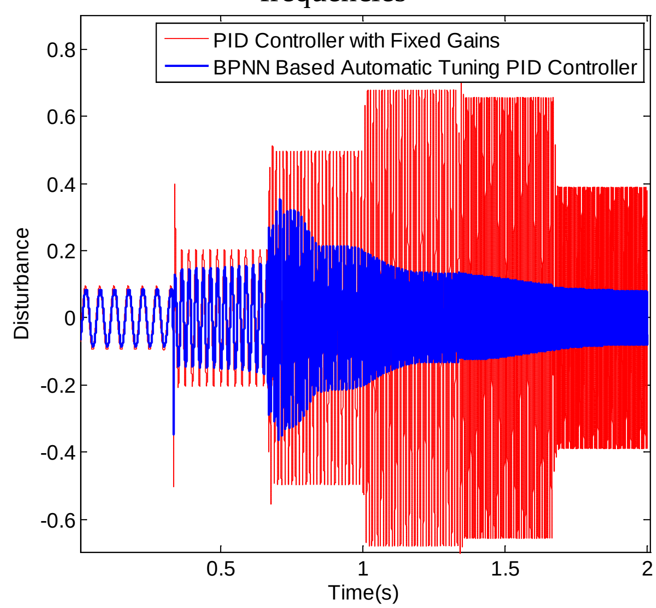

Fig. 3 Disturbance rejection results

[1] With this simulated BPNN structure, it requires 49 (=4*7+7*3) multiplications and 39 (=3*7+6*3) additions in the feedforward process and 147 (=3*49) multiplications and 117 (=3*39) additions during the error back-propagation process. So with simple BPNN structure, the computational amount still remains reasonable.

Conclusion

In this paper, automatic gain tuning of PID controllers based on neural-network is developed for suppressing time-varying disturbance. Using the self-learning ability of the neural network, this controller can adjust PID gains online for different disturbances to get an improved overall disturbance rejection performance. The design concepts are supported by simulation results on a HDD benchmark problem.

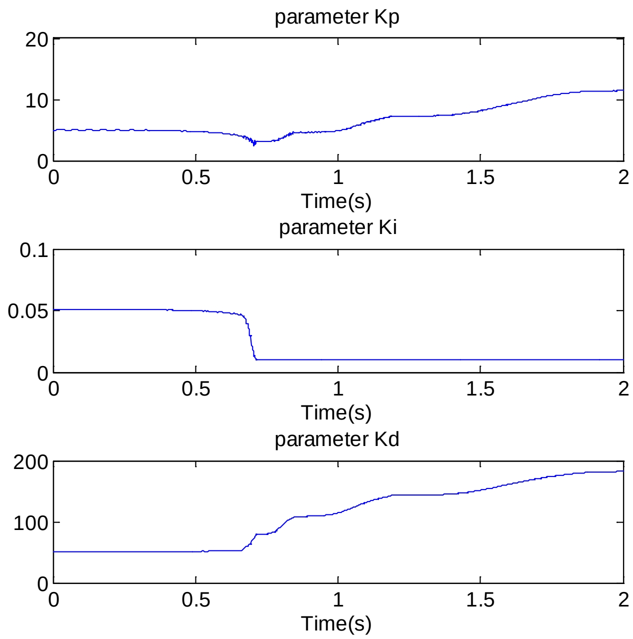

Fig. 4 Evolution of the PID parameters Kp, Ki and Kd

References

[1] Beitao Guo, Hongyi Liu, Zhong Luo, Fei Wang, 2009, "Adaptive PID Controller Based on BP Neural Network", *2009 International Joint Conference on Artificial Intelligence*.

[2] Yi Zhang, Chun Feng, and Bailin Li, 2006, "PID Control of Nonlinear Motor-Mechanism Coupling System Using Artificial Neural Network", *Advances in Neural Networks*, 2: 1096-1103.

[3] Seung Ho Cho, 2009, "Trajectory tracking control of a pneumatic X-Y table using neural network based PID control", *International Journal of Precision Engineering*, 10(5): 37-44.

[4] Fang Cheng, Kuang-Chao Fan, Jinwei Miao, Bai-Kun Li, Hung-Yu Wang, 2012, "A BPNN-PID based long-stroke nanopositioning control scheme driven by ultrasonic motor", *Precision Engineering*, 36:485-493.

[5] Yang Chuan, Zhao Qiang, Wang Hairong, Zhang Zhi, 2010, "Study on intelligent control system of two-dimensional platform based on ultra-precision positioning and large range", *Precision Engineering*, 34:627-633.

[6] Stuart Russell and Peter Norvig, 1995, *Artificial Intelligence: A Modern Approach*. Prentice Hall.

[7] Berry, M. J. A., and Linoff, G., 1997, *Data Mining Techniques*, John Wiley & Sons, NY, USA.

[8] Kevin L. Priddy and Paul E. Keller, 2005, *Artificial Neural Networks: An Introduction*, The Society of Photo-Optical Instrumentation Engineers, WA, USA.

[9] IEEJ, Technical Commitee for Novel Nanoscale Servo Control, "NSS benchmark problem of hard disk drive systems," http://mizugaki.iis.u-tokyo.ac.jp/nss/, 2007.