Online tuning of a model-based controller by perturbation of its poles

Keiichiro FURUYA*, Shinichi ISHIZUKA* and Itsuro KAJIWARA*
* Division of Human Mechanical Systems and Design, Hokkaido University
Kita 13 Nishi 8, Kita-ku, Sapporo, Hokkaido, 060-8628, Japan
E-mail: k.furuya@frontier.hokudai.ac.jp

Received 23 January 2015

Abstract
This study proposes an online tuning method using a model-based controller with adaptive parameters in the controller to effectively maintain the control performance and stability due to characteristic variations in the structure. Although model-based control generally provides a highly controllable performance, its performance depends on the modeling accuracy of the controlled object. Typically modeling errors, characteristics that change over time, etc. cause the performance to deteriorate. Hence, tuning of the model-based controller’s characteristics is proposed as a method to adapt to the errors between a real object and its model. The main idea of the tuning method proposed in this study is that tuning the poles of the controller greatly affects control performance and stability. The tuning algorithm in the proposed method employs the simultaneous perturbation stochastic approximation (SPSA), which is well suited for optimization problems with multiple design variables. To evaluate the effectiveness of the proposed tuning method, it is applied to vibration control simulations in which the model of the controlled object is perturbed to change its physical characteristics, and then the controller is tuned to adapt to these changes. Since SPSA is a stochastic optimization method, Monte Carlo simulations are also conducted to demonstrate the effectiveness of the proposed tuning method.

Key words: Online tuning, Simultaneous Perturbation Stochastic Approximation (SPSA), Smart structure, Adaptive control, Self-tuning, Model-based control

1. Introduction

As advanced mechanical systems become more compact and precise, control performance must also be improved. Hence, various active control methods have been proposed. For example, PID control is widely employed due to its simple constitution and reliability from accumulated experience. In contrast, model-based control effectively manages performance, but its applications are limited due to complex control theory and model construction. In particular, model-based control performance depends on the accuracy of the model, but it is impossible to construct an exact model free from modeling errors due to calculation costs, product dispersion, time-dependent characteristics, etc. These errors cause the control property to deteriorate and destabilize, which interfere with the utilization of model-based control.

To cope with such problems, control system tuning theories have been studied. For example, a self-tuning regulator (STR) can identify the unknown parameters of a controlled object automatically and tune the control parameters. Astrom et al. have developed a STR using the minimum variance criterion (Astrom et al., 1977), while Wellstead et al. suggested the STR to assign the closed-loop poles (Wellstead et al., 1979). In other studies, model reference adaptive control (MRAC) tunes the control parameters via tracking errors with respect to the reference model. Xu and Jia proposed MRAC in which the unmodeled hysteresis effect is estimated (Xu and Jia, 2014). However, the optimization problem in these studies is based on parameterization of a closed-loop system or errors with respect to the reference model, resulting in a complex tuning system. Although some tuning methods do not require a mathematical model or complex theory, they are mainly discussed about simple control theory like PID control, and similar method for the model-based controller is rare.
In 1987, Spall proposed a simultaneous perturbation stochastic approximation (SPSA) algorithm, which is occasionally used in various adaptive control methods. Because this algorithm only requires two measurements of loss function regardless of the number of parameters, it is uniquely suited to complex optimization problems. Applications of SPSA to several cases, including parameter estimation in system identification (Spall, 1998a and Hirokami, et al., 2004), self-tuning of the PID control parameters (Xu, et al., 2012), and a neural network (Song, et al., 2008), have confirmed its effectiveness.

Herein an intuitive tuning method for a model-based controller is proposed. This study aims to maintain the performance and stability by tuning the poles of a model-based controller. Although a model of the controlled object is used to design the controller, which is defined as the base controller, it is unnecessary when the controller is tuned. In the present method, perturbations are given to the poles of the controller, and the control frequency and gain are tuned to adapt to the modeling errors. The adaptive parameters applied to the poles are updated using SPSA in real-time. To evaluate the effectiveness of the proposed tuning method, vibration control simulations using finite element method (FEM) are carried out in which the time-dependent characteristics in the model are considered by providing the controlled object with the characteristic variation. The principal purpose of this study is to propose a new adaptive control scheme and to evaluate the fundamental features of the proposed method, and so we consider a case in which the controlled object is a time-varying system whose characteristics change gradually and continuously. This paper verifies the effectiveness of the proposed tuning method by Monte Carlo simulations.

2. Problem setting

2.1. Modeling of the controlled object

To evaluate the efficacy of the proposed tuning method, vibration control simulations of a flat plate with one fixed end are conducted. In the model of the flat plate (Fig.1), a piezoelectric actuator (40×5 mm) is used for the actuator of the control system and an accelerometer is attached to measure the acceleration of the flat plate. The state-space equation of this finite element model is derived, and the equation of motion for the structural system is described as

\[ M\ddot{x} + C\dot{x} + Kx = B_1w + B_2u \]  

(1)

where \( M \) is the mass matrix, \( K \) is the stiffness matrix, and \( C \) is the assumed proportional viscous damping matrix. \( x, w, \) and \( u \) are the displacement, disturbance, and control input vector, respectively. Because the degree-of-freedom in Eq.(1) is generally too large, the control system should not be designed directly with this equation. Thus, the coordinate transformation into the modal space is appropriate for the model reduction, and Eq.(1) is transformed into the modal space.

\[ \ddot{\xi} + \Phi^T C\Phi \dot{\xi} + \Lambda \xi = \Phi^T B_1w + \Phi^T B_2u \]  

(2)

where

\[ \Lambda = \Phi^T K \Phi \]

\( \Lambda \) is the diagonal eigenvalue matrix and the modal matrix \( \Phi \) is normalized with the mass matrix. From Eq.(2), the reduced-order state-space equation is described as

\[ \dot{q} = Aq + B_1w + B_2u \]  

(3)

where

\[ q = \{ \xi \} , \quad A = \begin{bmatrix} 0 & I \\ -\Lambda & -\Phi^T C\Phi \end{bmatrix} , \quad B_1 = \begin{bmatrix} 0 \\ \Phi^T B_{1s} \end{bmatrix} , \quad B_2 = \begin{bmatrix} 0 \\ \Phi^T B_{2s} \end{bmatrix} \]

(4)

In this study, an accelerometer detects the acceleration as a feedback signal, which is used for observed output \( y_2 \) and controlled output \( z_1 \). Then the detected acceleration \( y_2 \) is derived by Eq.(2) and Eq.(4) as

\[ y_2 = C_\epsilon \dot{\xi} \]

(5)

\[ = C_\epsilon (\Phi^T C_\epsilon \Phi \xi - \Lambda \xi + \Phi^T B_{1s}w + \Phi^T B_{2s}u) \]
From these equations, the reduced-order state-space equation can be derived. This study considers the first to the sixth modes. Figure 2 shows the open-loop frequency response function (FRF) of the displacement of the reduced-order controlled object, which is considered as a nominal plant whose characteristics are changed in the tuning simulation.

2.2. Control system

Figure 3 shows a block diagram of the vibration suppression. In Fig.3(a), \( G(s) \) is the generalized plant of the controlled object \( P(s) \) shown in Fig.1 and \( K(s) \) is the controller. Figure 1 also depicts disturbance \( w \), control input \( u \), observed output \( y_2 \), and controlled output \( z_1 \). Figure 3(b) shows the details of generalized plant \( G(s) \). The controlled variable is described as

\[
y_1 = \begin{bmatrix} Qz_1 \\ Ku \end{bmatrix}
\]

where \( Q \) and \( R \) are the weight matrices with respect to the control performance and the control input cost, respectively. \( H_\infty \) control theory is used to design the controller. The control problem in \( H_\infty \) control theory is described as

\[
\min \| T_{y_1 w} \|_\infty
\]

where \( T_{y_1 w} \) is the transfer function matrix between disturbance \( w \) and controlled variable \( y_1 \).

In this study, the controller is designed with \( Q = 1.0 \) and \( R = 32 \) to suppress all adopted peaks of the nominal plant. Figure 4 shows the FRF between the disturbance and the displacement controlled with the designed controller, confirming that the designed controller achieves a high control performance. We call this controller the nominal controller.
their imaginary parts dominate the control frequency. Generally, a model-based controller is expressed by the state-space equation, which is described as

\[
\begin{align*}
\dot{x}_c &= A_c x_c + B_c y_2 \\
u &= C_c x_c + D_c y_2
\end{align*}
\]
(8)

The eigenvalues of system matrix \(A_c\) correspond to the poles of the transfer function of the system. To tune the poles, perturbations are given to the elements of matrix \(A_c\), which have \(n\times n\) elements in an \(n\)th-order system. To reduce the calculate costs for online tuning, the controller state equation is transformed into the diagonal canonical form in which the diagonal elements of system matrix \(A_c\) are the poles, which are considered as tuning variables.

\[
\tilde{A}_c = T^{-1}A_c T
\]
\[
= \begin{bmatrix}
\lambda_1 & 0 \\
0 & \ddots \\
0 & 0 & \lambda_n
\end{bmatrix}
\]
(9)

where \(T\) is the transformation matrix, which consists of eigenvectors \(v_i\) \((n \times 1)\) of matrix \(A_c\).

\[
T = \begin{bmatrix}
v_1 \\
v_2 \\
\vdots \\
v_n
\end{bmatrix}
\]
\((i = 1, 2, \cdots, n)\)
(10)

From this transformation, the number of tuning variables becomes \(n\) because matrix \(A_c\) is transformed to a diagonal matrix.
with eigenvalues $\lambda_i$. Then the state-space equation is described as
\begin{align}
\dot{\tilde{x}}_c &= \tilde{A}_c \tilde{x}_c + \tilde{B}_c y_2 \\
u &= \tilde{C}_c \tilde{x}_c + \tilde{D}_c y_2
\end{align}
(11)
where the other system matrices in Eq. (8) are also transformed as
\begin{align}
\tilde{B}_c &= T^{-1} B_c \\
\tilde{C}_c &= C_c T \\
\tilde{D}_c &= D_c
\end{align}
(12)
In addition, it is easily shown that this transformation does not alter transfer function $G(s)$.
\begin{align}
G(s) &= \tilde{C}_c(sI - \tilde{A}_c)^{-1} \tilde{B}_c + \tilde{D}_c \\
&= C_c T(sI - T^{-1} A_c T)^{-1} T^{-1} B_c + D_c \\
&= C_c (sI - A_c)^{-1} B_c + D_c
\end{align}
(13)
In this study, the poles of the controller are perturbed to adapt to the modeling error. These perturbations change the controller gain and control frequency, which are expressed by the real and imaginary parts of the poles, respectively. Then the $j$th pole is described as $\lambda_j = \alpha_j + i\beta_j$, and perturbed pole $\lambda'_j$ is described as
\begin{equation}
\lambda'_j = k_{re}\alpha_j + ik_{im}\beta_j
\end{equation}
(14)
where $k_{re}$ and $k_{im}$ are the positive perturbations given to each part of the pole. Perturbed controller is always stable because the nominal controller is stable and given perturbations are positive. Here, the complex conjugate relationship is maintained by giving the same perturbation to each pair of complex conjugate poles. Matrix $\tilde{A}_c$, which includes the perturbations, is expressed by matrix $\tilde{A}'_c$ (Eq.(15)). Because the poles are generally conjugate complex numbers or real numbers, the diagonal elements of matrix $\tilde{A}_c$ include complex numbers, but the system matrix becomes a real matrix even after the inverse transformation of Eq. (9) by this method.
\begin{equation}
T\tilde{A}'_c T^{-1} = \begin{bmatrix} v_1 & v_2 & \cdots & v_n \end{bmatrix} \begin{bmatrix} \lambda'_1 & 0 & \cdots & 0 \\
0 & \ddots & \vdots & \vdots \\
0 & \cdots & \lambda'_n \end{bmatrix} \begin{bmatrix} w_1 \\
w_2 \\
\vdots \\
w_n \end{bmatrix} = \sum v_i \lambda'_i w_i = \lambda'_c
\end{equation}
(15)
where $T^{-1}$ is the inverse matrix of $T$, and $w_i$ is the $i$th row vector $(1 \times n)$ of $T^{-1}$.
\begin{equation}
T^{-1} = \begin{bmatrix} w_1 \\
w_2 \\
\vdots \\
w_n \end{bmatrix}
\end{equation}
(16)
If $\lambda'_j = \lambda_k$ ($\lambda'_j$ is the complex conjugate number of $\lambda_j$), then the corresponding elements of $T$ and $T^{-1}$ become $v^*_j = v_k$ and $w^*_j = w_k$. Thus, the $j$th and $k$th factors of Eq.(15) can be described as Eq.(17), and the complex numbers cancel.
\begin{equation}
v_j \lambda'_j w_j + v_k \lambda'_k w_k = v_j \lambda'_j w_j + v_j \lambda'_j w^*_j \\
= v_j \lambda'_j w_j + (v_j \lambda'_j w_j)^* \in \mathbb{R}^{n \times n}
\end{equation}
(17)
By this method, the elements of system matrix $\tilde{A}'_c$ are real number.

3.2. SPSA algorithm
The controller poles are effectively tuned by the SPSA algorithm. SPSA is a gradient method using the stochastic approximation algorithm proposed by Spall in 1987 (Spall, 1987). The SPSA algorithm is very effective for large-scale optimization problems.
The gradient method minimizes loss function \( L(\theta) \) using its gradient \( g(\theta) = \partial L(\theta)/\partial \theta \) where \( \theta \) is the design variable vector of a \( p \)-dimension multivariate system. Generally, it is difficult to determine gradient \( g(\theta) \) directly. Thus the stochastic gradient method uses estimated gradient \( \hat{g}_i \) and uploads design variable \( \theta_i \), which is described as

\[
\theta_{i+1} = \theta_i - a_i \hat{g}_i(\theta_i)
\]  

where \( i \) is the iteration number, \( a_i \) represents the scalar gain coefficient, and \( \hat{g}_i(\theta_i) \) represents stochastic approximation of an unknown gradient of the loss function. A major stochastic method, which is called finite difference stochastic approximation (FDSA), gives a small positive and negative perturbation to each parameter, and estimates the gradient by the differences of both sides of the loss function described as

\[
\hat{g}_{ti}(\theta_t) = \frac{L(\theta_t + c_i e_i) - L(\theta_t - c_i e_i)}{2c_i}
\]

where \( e_i \) is the index vector and \( c_i \) is a coefficient that determines a degree of given perturbation. Since FDSA requires \( 2p \) times the calculation of loss function to update all parameters, the calculation cost increases in proportion to the number of parameters.

In contrast, SPSA is based on a highly efficient gradient approximation, which requires that the loss function is measured twice regardless of the number of parameters. In a study by Spall (1987, 1992), \( \hat{g}_i(\theta_t) \) is represented by a two-measurement approximation, which is described as

\[
\hat{g}_{ti}(\theta_t) = \frac{L(\theta_t + c_i \Delta_i) - L(\theta_t - c_i \Delta_i)}{2c_i \Delta_{ij}}
\]

where \( \Delta_i \in R^p \) is the vector of \( p \) mutually independent mean-zero random variables, such as the symmetric Bernoulli \( \pm 1 \) distribution. The SPSA algorithm gives perturbations to all design variables simultaneously. Thus, the design variables are updated by calculating the loss function twice. Because the calculation is independent of the number of parameters, the calculation cost of SPSA becomes \( 1/p \) compared to FDSA. In general, two coefficients \( a_i \) and \( c_i \) become smaller as iteration number \( t \) increases (Spall, 1998b). \( a_i \) and \( c_i \) are described as

\[
a_i = \frac{a}{(A + t + 1)\alpha}
\]

\[
c_i = \frac{c}{(t + 1)^\gamma}
\]

where \( a, c, A, \alpha, \) and \( \gamma \) are non-negative coefficients that decrease exponentially as the iterations progress and converge to zero to terminate the adaptation process. However, to adapt to successive characteristic changes such as time-dependent characteristics, the controller must be tuned continuously. Consequently, \( a_i \) and \( c_i \) are defined as constants in this study that are determined by the degree of the characteristic change, noise level of control system, etc.

### 3.3. Tuning control system

To carry out the online tuning of the controller, loss function \( L(\theta_t) \) is evaluated in real-time. \( L(\theta_t) \) is defined by the sum of the mean-square of the observed output, which is described as

\[
L(\theta_t) = \frac{1}{f_t} \sum_{k=(t-1)m+1}^{m} y^2_k(t)
\]
where \( m \) is the number of evaluation points within the evaluating period, \( f_s \) is the sampling frequency, and \( y_2(k) \) is the observed output at a sampling point \( k \). Since SPSA requires that the loss function is calculated twice to simultaneously update all design variables, updating period \( T_e \) is described as

\[
T_e = \frac{2m}{f_s}.
\]  

(25)

The perturbations given to the design variables are kept as \( \theta_t \leftarrow \theta_t + c_t \Delta_t \) during first \( T_e = 2 \) s, and they are kept as \( \theta_t \leftarrow \theta_t - c_t \Delta_t \) during another period. The loss functions corresponding to each perturbation are calculated in this way, and the gradient is estimated. Figure 5 depicts the block diagram of the tuning system. To evaluate the effectiveness of the proposed tuning method, the controller during the simulation is tuned to adapt continuously to time-dependent characteristics.

![Block diagram of the tuning system](image.png)

**Fig. 5 Block diagram of the tuning system**

4. Tuning simulation

4.1. Simulation conditions

The dynamic characteristics of the controlled object are changed in real-time tuning simulation. In this study, we consider a case in which the controlled object is a time-varying system whose characteristics change gradually and continuously. Aging of the mechanical systems and environmental condition changes such as a temperature change influence mechanical properties of materials, such as rigidity and elasticity, and cause a gradual and continuous characteristic variation of the system. It is important that the controller keeps the control performance and stability against this kind of characteristic variation. The fundamental features of the proposed method are revealed through the simulation.

As the dynamic characteristics change, the natural frequency of the controlled object increases linearly from 0 to +20\%. The rate of a given perturbation is determined by simulation period \( T_s \), which is described as \( \delta_f = 20/T_s \% \) per second. Figure 6 shows the FRFs of the nominal and perturbed plants where the perturbation is +20\% for all natural frequencies. The control performance of the closed-loop system with the perturbed plant (+20\%) and the nominal controller becomes worse over 300Hz compared to that with the nominal plant. Figure 7 shows the Nyquist diagram of the open-loop system with the perturbed plant and the nominal controller. It is observed from Fig.7 that the closed-loop system with this plant and controller becomes unstable. These results confirm that the nominal controller cannot maintain the control performance and stability for dynamic characteristics changes. Hence, the proposed tuning method is employed to adapt to the change in the dynamic characteristics and to assess whether the proposed tuning method can maintain the control performance and stability.

Because the order of the nominal controller is 12, there are 12 controller poles that consist of 6 pairs of complex conjugate numbers in the simulation. To keep the complex conjugate relationship when the poles are perturbed, half of the poles are considered to be design variables. In addition, the perturbations are given to both the real and imaginary parts of the poles, increasing the number of design variables to 12. Perturbation range of \( k_{re} \) and \( k_{im} \) is limited to 0.5 – 1.5 in the simulation. Because simulation period \( T_s = 200s \), the change in the dynamic characteristics of the controlled object are given \( \delta_f = +0.1\% \) per second. Disturbance \( w \), which is applied to the controlled object, is considered to be white noise. Table 1 details the simulation conditions.

4.2. Simulation Result

Figure 8 shows the results for one of the tuning simulations for the time history response of the loss function. The
blue dashed line indicates the results for the nominal controller. The closed-loop system with the nominal controller becomes unstable after 30s when the perturbation to the controlled object exceeds +3%, demonstrating that the nominal controller easily makes the closed-loop system unstable. In contrast, the tuned controller, which is denoted by the red line, maintains the control performance and stability until 200s. Although the performance occasionally declines, its stability is maintained. Figure 9 shows the Nyquist diagrams of the open-loop system with the perturbed plant and the nominal controller, and the perturbed plant and the tuned controller at 200s. As mentioned before, the nominal controller cannot maintain the stability of the closed-loop system during the variation, but the tuned controller can. Figure 10 shows the FRFs of the closed-loop system with the nominal plant and the nominal controller, the perturbed plant and the nominal controller, and the perturbed plant and the tuned controller at 200s. Although the performance of the tuned controller is worse than that of the nominal controller when controlling the nominal plant, it is superior when controlling perturbed plant. Especially, the performance of the tuned controller around 1.2kHz is superior to that of the nominal controller. It is confirmed from these results that the proposed tuning method is very effective against the characteristic variation of the controlled object.
Because SPSA is a stochastic optimization method and the disturbance is white noise, the tuning simulation results always differ for each trial. To stochastically evaluate the effectiveness of the proposed tuning method, Monte Carlo simulations are carried out. Figure 11 shows the frequency distribution of the time history response of the loss function expressed in a color map after 1000 Monte Carlo simulations. Although the closed-loop system with the nominal controller becomes unstable around 30s, the proposed tuning method is more likely to maintain its control performance and stability, especially until 120s. After 120s, the dispersion of the loss function becomes larger and this means that the control performance tends to worsen. This result implies that the proposed tuning method has a limitation on characteristic variation range of the controlled object. In this application, the tuned controller can mostly maintain the control performance and stability during the variation until over 10% perturbations of the plant natural frequencies.

However, the proposed tuning system sometimes causes the closed-loop system to become unstable. Figure 12 shows another result from the same tuning simulation. As discussed above, the stability of the closed-loop system isn’t always guaranteed during variation because the perturbations to the poles are given by random numbers. Although the control performance and stability tend to be restored quickly, system destabilization similar to Fig.12 must be prevented for practical applications of this method. An approach for the improvement of such problems is to increase the update frequency. Although it is difficult to prevent the destabilization perfectly because of uncertainty of the SPSA algorithm, the control performance and stability can be improved if the design variable is updated quickly before the deterioration is enlarged. Ishizuka proposed the overlap evaluation of loss function, which can evaluate continuously while running (Ishizuka and Kajiwara, 2015). It is shown that the overlap evaluation increases the update frequency and improves the follow-up performances. This effect enables to extend the structural variation range and the applications with other types of the variation. The features of the proposed method to other types of the characteristic variation should be examined in the future.

5. Conclusion

This study proposed an online tuning method of the model-based controller in which the controller poles perturbed. To evaluate the performance of the proposed tuning method, online tuning simulations in which the characteristic perturbations were applied to the controlled object were carried out. The tuned controller showed a higher control performance and stability than the nominal controller. Additionally, Monte Carlo simulations showed that the proposed method could stochastically maintain the control performance and stability. Although Closed-loop systems sometimes became unstable using the proposed tuning method, the stability could be quickly restored.
In the future, we intend to refine the tuning system so that the stability of the closed-loop system is enhanced. The overlap evaluation of the loss function will be employed to increase the update frequency of the design variable. It is expected that this approach improves the stability of the closed-loop system. In addition, a tuning experiment using the proposed method will be conducted to evaluate the performance of the refined tuning method. Furthermore, the achievable control performance will be compared to that with other adaptive control methods to evaluate the effectiveness of the proposed method.

References

Astrom, K. J., Borisson, U., Ljung, L. and Wittenmirk, B., Theory and applications of self-tuning regulators, Automatica, Vol.12, No.5 (1977), pp.457-476

Hirokami, T., Maeda, Y. and Tsukada, H., Parameter Estimation using Simultaneous Perturbation Stochastic Approximation, IEEJ Transactions on Electronics, Information and System, Vol.124, No.11 (2004), pp.2241-2248

Ishizuka, S. and Kajiwara, I., Online adaptive PID control for MIMO systems using simultaneous perturbation stochastic approximation, Journal of Advanced Mechanical Design, Systems, and Manufacturing, Vol.9, No.2 (2015), pp.1-16

Song, Q., Spall, J. C., Soh, Y. C. and Ni, J., Robust Neural Network Tracking Controller Using Simultaneous Perturbation Stochastic Approximation, IEEE Transactions on Neural Networks, Vol.19, No.5 (2008), pp.817-835

Spall, J. C., A stochastic approximation technique for generating maximum likelihood parameter estimates, Proceedings of Control Conference (1987), pp.1161-1167

Spall, J. C., Multivariate stochastic approximation using a simultaneous perturbation gradient approximation, IEEE Transactions on Automatic Control, Vol.37, No.3 (1992), pp.332-341

Spall, J. C., An Overview of the Simultaneous Perturbation Method for Efficient Optimization, JOHNS HOPKINS APL TECHNICAL DIGEST, Vol.19, No.4 (1998a), pp.482-492

Spall, J. C., Implementation of the Simultaneous Perturbation Algorithm for Stochastic Optimization, Aerospace and Electronic Systems, IEEE Transactions on, Vol.34, No.3 (1998b), pp.817-823

Wellstead, P. E., Prager, D. and Zanker, P., Pole assignment self-tuning regulator, Proceeding of the Institution of Electrical Engineers, Vol.126, No.8 (1979), pp.781-787

Xu, Q. and Jia, M., Model Reference Adaptive Control With Perturbation Estimation for a Micropositioning System, IEEE Transactions on Control Systems Technology, Vol.22, No.1 (2014), pp.352-359

Xu, P., Li, G., Wang, K., Self Tuning of PID Controller Based on Simultaneous Perturbation Stochastic Approximation, Advances in EECM Vol.1, LNEE 139 (2012), pp.647-652