International Journal of Innovative Computing, Information and Control
Volume **18**, Number **3**, June **2022**
 ISSN 1349-4198
pp. **667–685**

AN UPDATE RULE OF PARAMETERS OF ONLINE-OFFLINE INTEGRATED LEARNING METHOD OF NEURAL NETWORK CONTROL

Masakazu Morita, Qingjiu Huang* and Mimpei Morishita

Electrical Engineering and Electronics Program
Graduate School of Engineering
Kogakuin University
1-24-2 Nishi-shinjuku, Shinjuku-ku, Tokyo 163-8677, Japan
cd18002@ns.kogakuin.ac.jp; morisita@cc.kogakuin.ac.jp
*Corresponding author: huang@cc.kogakuin.ac.jp

Received October 2021; revised January 2022

Abstract. *In order to deal with load fluctuations and disturbances, and to realize real-time control, we have already proposed an online-offline integrated learning method of neural network control for the position control of servo system in previous study. However, the relationship between the learning rate, the threshold for the number of iterations within the sampling time, and the responsiveness and the robustness of the servo system has not yet been clarified. In this study, the relationship was clarified and formulated from simulation and analysis data, and an update rule for the learning rate and the threshold for the number of iterations to enable self-tuning was proposed. Specifically, the relationship between the learning rate, the threshold for the number of iterations, and the responsiveness and the robustness of the servo system was clarified from simulation and analysis data. In addition, the relationship between the learning rate, the threshold for the number of iterations, a feedback error, and a settling time was formulated using the least squares method, and the update rule of the learning rate and the threshold for the number of iterations to enable self-tuning was proposed. Moreover, the index values of the feedback error and the settling time were set, and the learning rate and the threshold for the number of iterations were calculated based on the proposed update rule, and the effectiveness of this study was shown from simulation and analysis result before and after changing the load inertia.*
Keywords: Neural network control, Online-offline integrated learning, Update rule of learning rate and threshold for number of iterations, Least squares method, Self-tuning

1. **Introduction.** Neural network control, as one kind of adaptive control and nonlinear control, is expected to be used for high-precision position control of servo system such as robots and numerical control machines [1-6].

Until now, in previous study, in order to deal with load fluctuations and disturbances, and to realize real-time control, we proposed an online-offline integrated learning method of neural network control for the position control of servo system, in which the offline learning is performed if the feedback error does not exceed the threshold value, and the online learning with few repeating loops within the sampling time is performed if the feedback error exceeds the threshold value [7]. However, the relationship between the learning rate η, the threshold for the number of iterations N_ε within the sampling time, and the responsiveness and the robustness of the servo system has not yet been clarified, so the tuning of the learning rate η and the threshold for the number of iterations N_ε requires a lot of time, and it cannot say that the learning efficiency is good.

DOI: 10.24507/ijicic.18.03.667

In the studies on the update rule of the learning rate of neural network, Kanda et al. proposed a method to determine the learning rate using the Proportional-Integral-Differential (PID) control and the prediction [8,9]. And, Cater proposed the Heuristic Learning Algorithm (HLA) to increase the learning rate when the error evaluation function is large and its gradient is small [10]. Vogl et al. proposed a method to increase or decrease the learning rate depending on the change in the error evaluation function [11,12]. Moreover, there is the Kick Out method that suppresses the vibration of weights generated in the valley of the curved surface of the evaluation function based on the Jacobs method [13-15]. In addition, there is an update rule named as the Delta-Bar-Delta (DBD) for the weights with only the gradient term of the Jacobs method [16,17]. Although these update rules are such that the derivative of the learning rate increase or decrease is smoothed, which can suppress the oscillation of the weight increase or decrease, because the relationship between the efficiency of learning and the learning minimum value has not been formulated, it is difficult to connect the relationship between the learning rate increase or decrease with the responsiveness and the robustness of the servo system. And, it is necessary to formulate the relationship between the responsiveness and robustness of the servo system and the threshold for the number of iterations N_ε as the parameter proposed by the online-offline integrated learning method of the previous study.

For the above background, in this study, the relationship between the learning rate η, the threshold for the number of iterations N_ε, and the responsiveness and the robustness of the servo system is clarified from simulation and analysis data for the position control of the servo system. Moreover, the relationship between them is formulated using the least squares method, and the update rule of the learning rate η and the threshold for the number of iterations N_ε to enable self-tuning is proposed.

Specifically, first, for the position control of the servo system, an analysis environment using a dynamic model of the position control of the servo motor is built. Next, the online-offline integrated learning method is performed before and after changing the load inertia, and the relationship between the learning rate η, the threshold for the number of iterations N_ε, and the responsiveness and the robustness of the servo system are clarified from simulation and analysis data. In addition, the relationship between the learning rate η, the threshold for the number of iterations N_ε, the feedback error, and the settling time is formulated using the least squares method, and the update rule of the learning rate η and the threshold for the number of iterations N_ε to enable self-tuning is proposed. Finally, the index values of the feedback error and the settling time were set, and the learning rate η and the threshold for the number of iterations N_ε were calculated based on the proposed update rule, and the effectiveness of this study is shown from simulation and analysis result before and after changing the load inertia.

The structure of this paper is shown as follows. Section 2 describes the algorithm of the online-offline integrated learning method proposed by the previous study. Section 3 presents the analysis environment using the dynamic model of position control for the servo motor. Section 4 shows the steady-state characteristics and transient characteristics of the servo system related to the increase or decrease of the learning rate η and the threshold for the number of iterations N_ε. Section 5 presents the formulation of the relationship between the learning rate η, the threshold for the number of iterations N_ε, the feedback error, and the settling time, and shows the proposal of an update rule of the learning rate η and the threshold for the number of iterations N_ε to enable self-tuning. Section 6 summarizes the analysis results and considerations of the servo system based on the proposed update rule. Section 7 presents the conclusions of this study.

2. **Online-Offline Integrated Learning Method.** In this study, the servo system with neural network control was built based on the feedback error learning method [18]. Its structure is shown in Figure 1. And, the learning method of this neural network is the online-offline integrated learning method proposed by the previous study [7]. The flowchart is shown in Figure 2, and the algorithm of the learning method is shown in Equations (1)-(9).

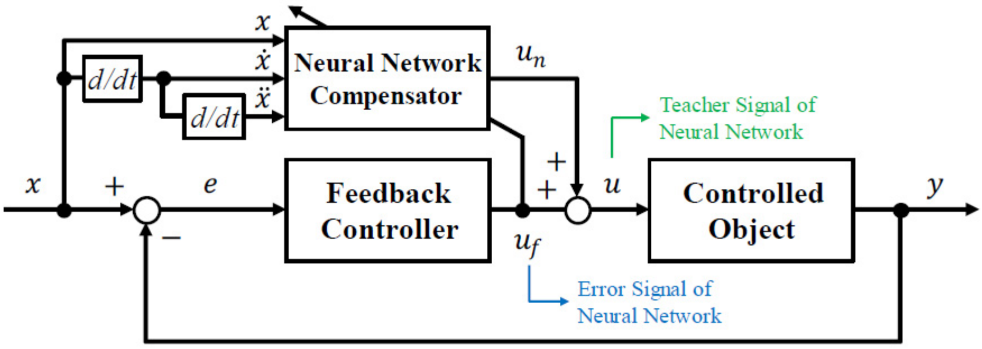


FIGURE 1. Structure of the servo system using neural network control

The explanation of the variables in Figure 1 is shown as follows:
x: Reference value, also the input to the feedback control system.
e: Tracking error, also the feedback error.
u_f: Output of the feedback controller, also the error signal of the neural network.
u_n: Output of the neural network.
u: Control input for the controlled object, also the teacher signal of the neural network.
y: Controlled variable, also the output to the feedback control system.

In this neural network, in order to obtain the inverse dynamic model of the controlled object with the learning, the input u for the controlled object is treated as a teacher signal. Therefore, the difference between the teacher signal u and the output signal u_n is the error of the neural network and is equal to u_f. Thus, the neural network works to match the output u_n with the teacher signal u through the learning, obtains the inverse dynamic model of the controlled object, and compensates the feedback controller as a feedforward controller. In Equations (4)-(9), N_s is the number of repeating loop.

$$e = x - y \tag{1}$$

$$y(j) = f_y(w(i,j), x(i)) = \frac{2}{1 + e^{-\sum_{i=1}^{c} w(i,j)x(i)}} - 1 \tag{2}$$

$$u_n(k) = z(k) = f_z(v(j,k), y(j)) = \frac{2}{1 + e^{-\sum_{j=1}^{b} v(j,k)y(j)}} - 1 \tag{3}$$

$$\Delta v(j,k)(N_s) = -\eta \frac{\partial E_n}{\partial v(j,k)} + \alpha \Delta v(j,k)(N_s - 1) \tag{4}$$

$$v(j,k)(N_s) = v(j,k)(N_s - 1) + \Delta v(j,k)(N_s) \tag{5}$$

$$\Delta w(i,j)(N_s) = -\eta \frac{\partial E_n}{\partial w(i,j)} + \alpha \Delta w(i,j)(N_s - 1) \tag{6}$$

$$w(i,j)(N_s) = w(i,j)(N_s - 1) + \Delta w(i,j)(N_s) \tag{7}$$

$$y(j) = f_y(w(i,j)(N_s), x(i)) = \frac{2}{1 + e^{-\sum_{i=1}^{c} (w(i,j)(N_s))x(i)}} - 1 \tag{8}$$

$$u_n(k) = z(k) = f_z(v(j,k)(N_s), y(j)) = \frac{2}{1 + e^{-\sum_{j=1}^{b}(v(j,k)(N_s))y(j)}} - 1 \quad (9)$$

Equation (1) shows the feedback error e. Equation (2) shows the function indicating neurons in a middle layer, and Equation (3) shows the function indicating neurons in an output layer. Equation (4) shows the update equation for the weights $v(j,k)$, and Equation (5) shows the current weights $v(j,k)$. Equation (6) shows the update equation for the weights $w(i,j)$, and Equation (7) shows the current weights $w(i,j)$. Equation (8) shows the $y(j)$, and Equation (9) shows the $z(k)$.

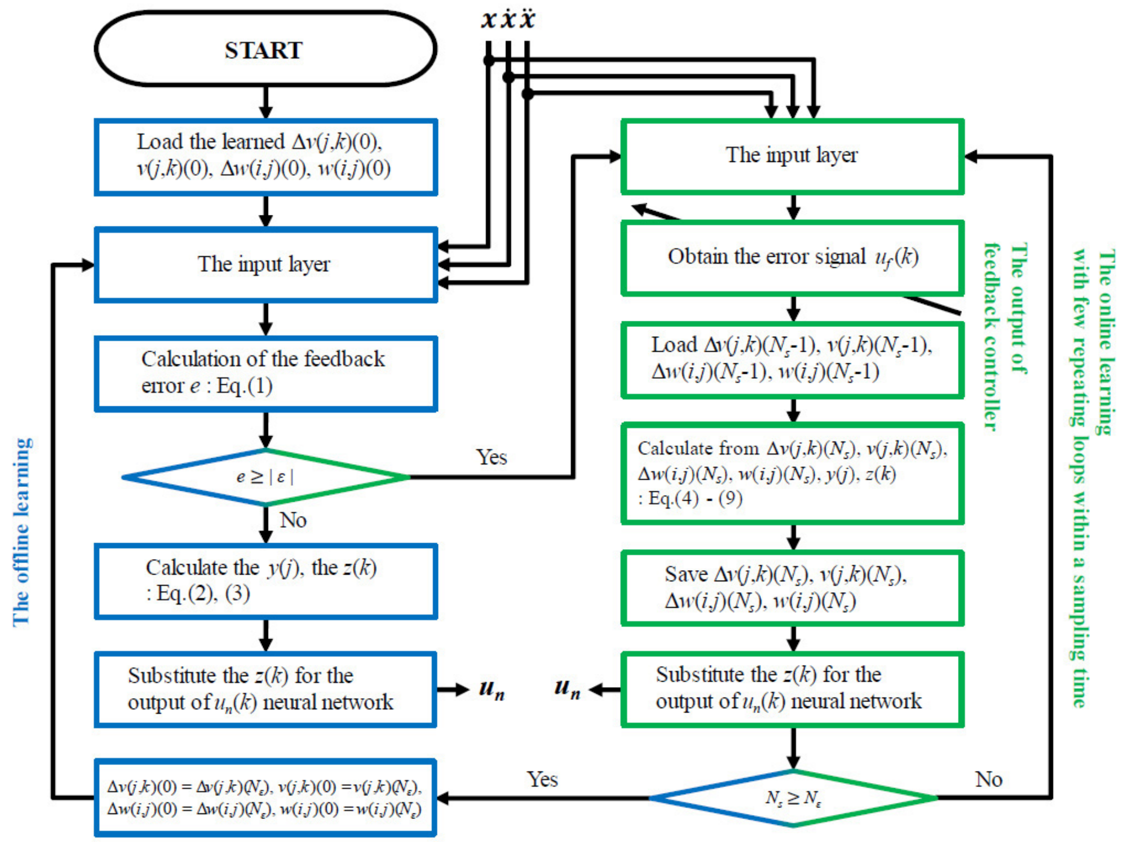


FIGURE 2. The online-offline integrated learning method proposed by the previous study

Regarding the flowchart in Figure 2, first, the weights obtained by enough online learning are loaded into the neural network and set as $\Delta v(j,k)(0)$, $v(j,k)(0)$, $\Delta w(i,j)(0)$, $w(i,j)(0)$. However, according to the results of the simulation for the position control of the servo motor in this paper, if the weights $v(j,k)$ is initialized with a value of zero, better results will be obtained. Therefore, the weights $v(j,k)$ is initialized with a value of zero in this study. Moreover, let the position x, the velocity $\dot{x}$, and the acceleration $\ddot{x}$ of the reference value be the inputs of neural network. In addition, the error signal E_n is obtained from the output $u_f(k)$ of the feedback controller. Next, the feedback error e is calculated by Equation (1). If the calculated error e does not exceed the threshold $|\varepsilon|$, the offline learning method is performed. And, $y(j)$ and $z(k)$ are calculated by Equations (2) and (3). And then, let $z(k)$ be the output $u_n(k)$ of the neural network. If the calculated error e exceeds the threshold $|\varepsilon|$, the online learning method is performed with few repeating loops within one sampling time. In each loop, the inputs of the neural network are given by the position x, the velocity $\dot{x}$, and the acceleration $\ddot{x}$ of reference value. In addition, the error signal E_n is obtained from the output $u_f(k)$ of the feedback controller. $\Delta v(j,k)(N_s-1)$, $v(j,k)(N_s-1)$, $\Delta w(i,j)(N_s-1)$, $w(i,j)(N_s-1)$ are loaded into the neural network, and $\Delta v(j,k)(N_s)$, $v(j,k)(N_s)$, $\Delta w(i,j)(N_s)$, $w(i,j)(N_s)$, $y(j)$, $z(k)$ are calculated by Equations (4)-(9). The value of $z(k)$ is given to $u_n(k)$. Next, if the number of repeating loops N_s within the sampling time does not reach the threshold N_ε, these calculations are repeated. If the number of repeating loops N_s within the sampling time reaches the threshold N_ε, the online learning method is terminated and the calculations return to the offline learning method. Then, let the obtained weights $\Delta v(j,k)(N_\varepsilon)$, $v(j,k)(N_\varepsilon)$, $\Delta w(i,j)(N_\varepsilon)$, $w(i,j)(N_\varepsilon)$ be the learned weights $\Delta v(j,k)(0)$, $v(j,k)(0)$, $\Delta w(i,j)(0)$, $w(i,j)(0)$ of the offline learning method.

3. **Analysis Environment.** In this section, it shows the dynamic model of the control system of the servo motor as the controlled object, and shows the analysis model of the control system built by MATLAB/Simulink in order to simulate the position control of the servo motor [7].

The structure of the servo control with neural network control is shown in Figure 3. In addition, the simulation model structured in MATLAB/Simulink is shown in Figure 4. In this model, the sampling time for analysis is 1 ms, and the analysis method for calculus is the 4th-order Runge-Kutta method. The block θ_r is the generator of the reference position, and generates the angular trajectory that is a sine wave with an amplitude of

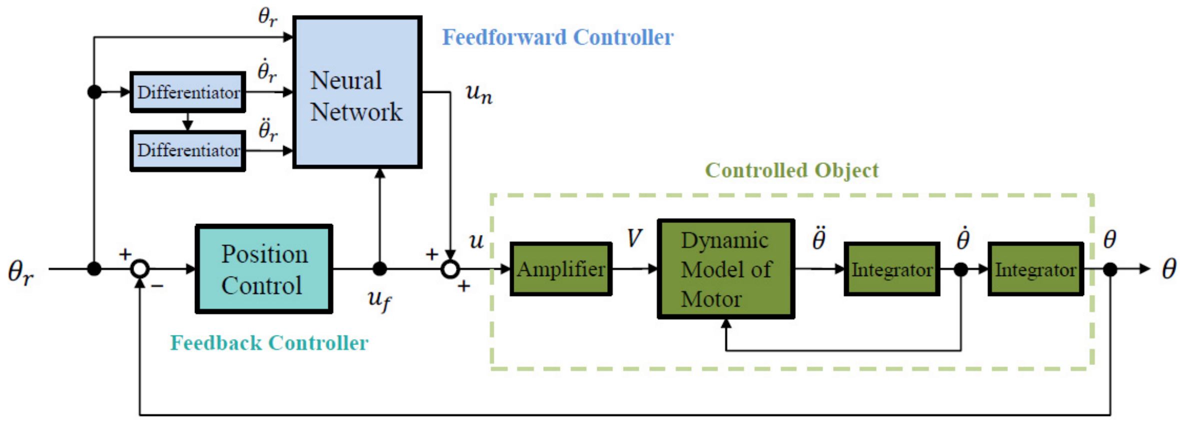


FIGURE 3. Structure of the servo system with neural network control

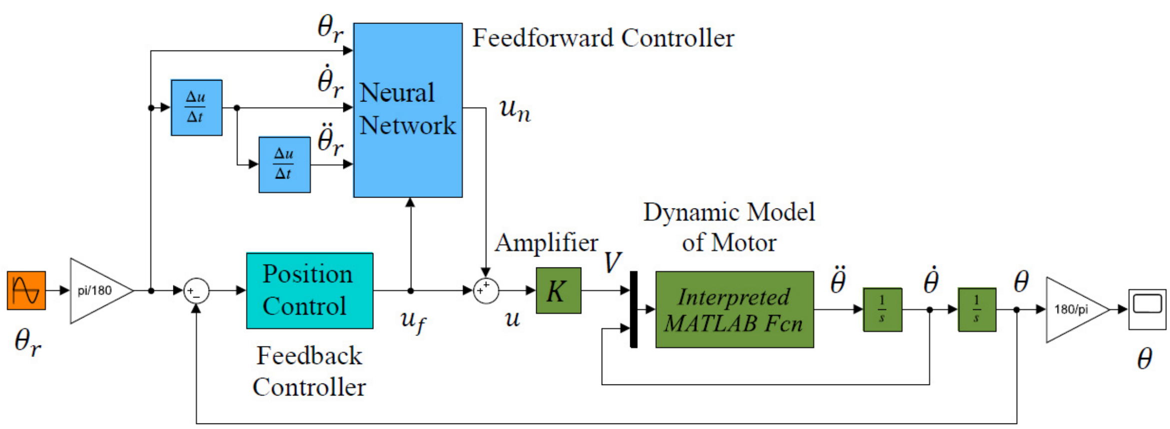


FIGURE 4. Simulation model of the servo system with neural network control

±90 degrees, a frequency of 1 Hz, and a phase of 0 degrees. The feedback controller uses the PID control. The gain of PID control is set with the value calculated by the pole placement method. The amplifier K is connected in front of the dynamic model of motor. And, θ_r, $\dot{\theta}_r$, $\ddot{\theta}_r$ are connected to the block of the neural network compensator. In addition, the angle θ and the angular velocity $\dot{\theta}$, which are the outputs of the motor, are obtained by integrating the angular acceleration $\ddot{\theta}$. In the simulation of this study, the DC servo motor deals with Maxon's EC-max 22, and its parameters are shown in Table 1 below. The setting of the load inertia J_L is shown in Section 4 [7].

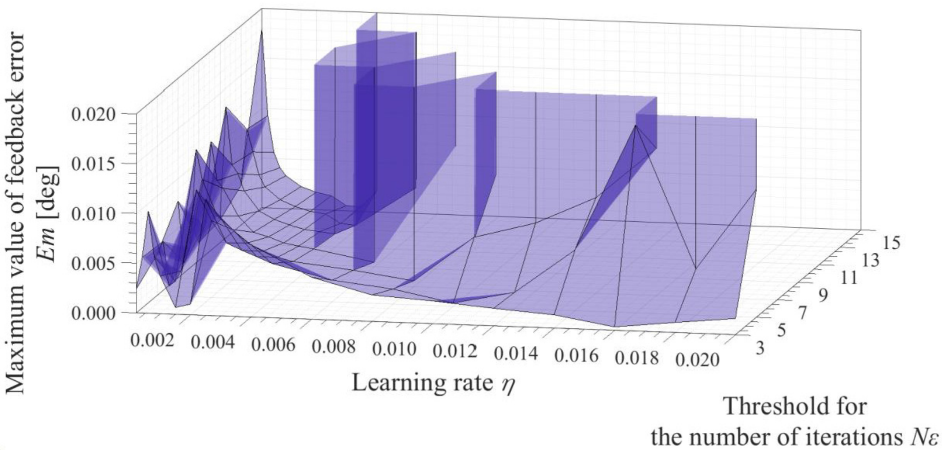
Table 1. Parameters of brushless DC motor (EC-max 22)

Symbol	Name	Value
V_n	Nominal voltage	24 V
I_n	Nominal current	0.657 A
N_n	Nominal speed	8250 rpm
T_n	Nominal torque	10.8 mNm
R	Terminal resistance phase to phase	12.4 Ω
K_t	Torque constant	18.1 mNm/A
J_M	Rotor inertia	2.25×10^{-7} kgm^2

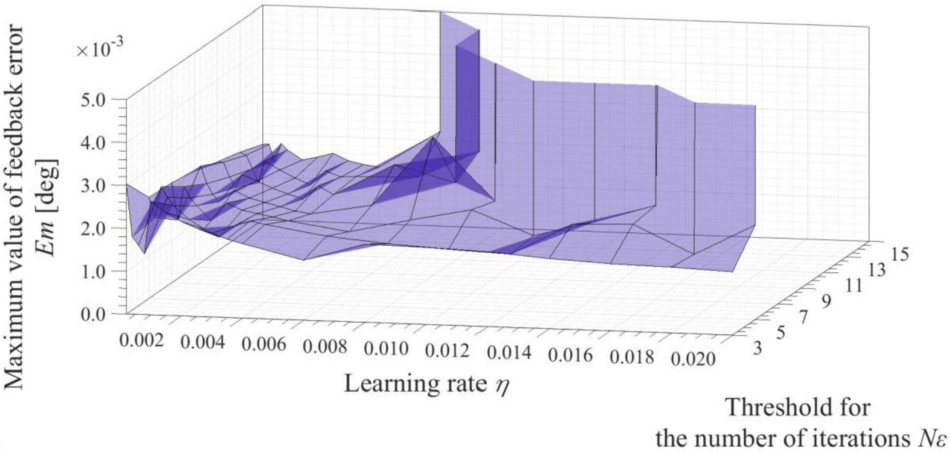
4. **Characteristics of Servo System Related to Increase or Decrease in Learning Rate.** In this section, the characteristics of the servo system on the increase or decrease in the learning rate using the online-offline integrated learning method are shown in Sections 4.1 and 4.2. In Section 4.1, the steady-state characteristics on three parameters, the maximum feedback error E_m, the learning rate η, and the threshold for the number of iterations N_ε before and after changing the load inertia are shown in Figure 5. In Section 4.2, the transient characteristics on the three parameters, the settling time T_s, the learning rate η, and the threshold for the number of iterations N_ε, before and after changing the load inertia are shown in Figure 6.

4.1. **Steady-state characteristics related to increase or decrease in learning rate.** Regarding the parameters of the online-offline integrated learning method, first, the learned weights were set as $\eta = 0.004$, $\alpha = 0.001$. Next, the learning rate was set from $\eta = 0.0002$ to $\eta = 0.02$, and the threshold for the number of iterations was set from $N_\varepsilon = 3$ to $N_\varepsilon = 15$. Here, the inertia coefficient was set to $\alpha = 0.001$. Finally, the load inertia value was set as $J_L = J_M = 2.25 \times 10^{-7}$ kgm^2. Figures 5(a) and 5(b) show the simulation results of the maximum feedback error, the learning rate, and the threshold for the number of iterations before and after changing the load inertia, respectively. These values are shown in Table 2 and Table 3. And, we define N as the number of data for the learning rate.

Before and after changing the load inertia, increasing the learning rate η tends to reduce the maximum value of the feedback error, and increasing the threshold for the number of iterations N_ε tends to reduce the maximum value of the feedback error up to a certain value. However, when the learning rate η and the threshold for the number of iterations N_ε exceed the certain value, the maximum value of the feedback error tends to increase. And, when the learning rate η is large and the threshold for the number of iterations N_ε is a lot, the maximum value of the feedback error tends to increasing.



(a) Before changing load inertia



(b) After changing load inertia

FIGURE 5. Maximum value of the feedback error, learning rate and threshold for the number of iterations with the online-offline integrated learning method

4.2. **Transient characteristics related to increase or decrease in learning rate.** Regarding the parameters of the online-offline integrated learning method, it was set in the same way as the previous section. Figures 6(a) and 6(b) show the simulation results of the settling time, the learning rate η, and the threshold for the number of iterations N_ε before and after changing the load inertia, respectively. These values are shown in Table 4 and Table 5. And we define N as the number of data for the learning rate.

Before and after changing the load inertia, increasing the learning rate η tends to shorten the settling time, and increasing the threshold for the number of iterations N_ε tends to shorten the settling time. However, when the learning rate η is large and the threshold for the number of iterations N_ε is large, the settling time tends to become longer.

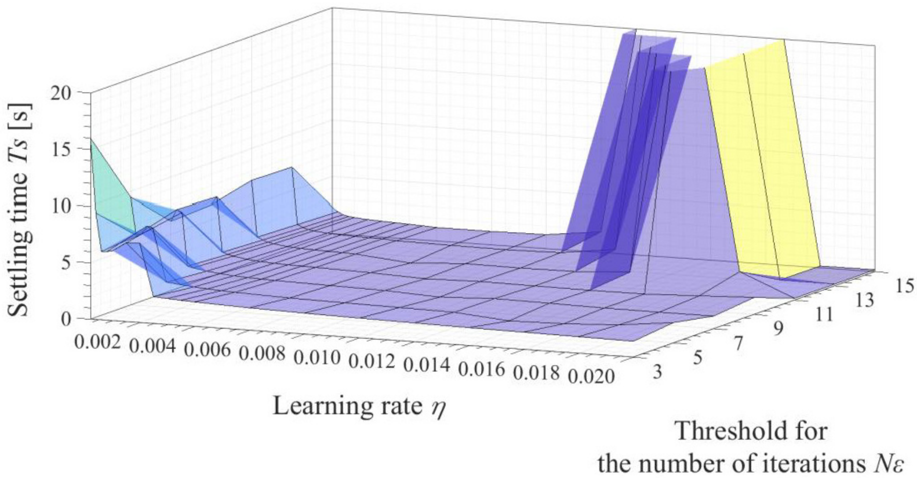
5. **Formulation of Learning Rate Based on Least Squares Method and Derivation of Update Rule of Learning Rate and Threshold for Number of Iterations.** In this section, the combination of parameters with the minimum error or the combination of parameters with the shortest convergence time is extracted from the characteristics of the servo system in Section 4. And, we apply the least squares method to the combinations of those parameters. For the above, we will describe the formulation of the

Table 2. Maximum value of the feedback error E_m (before changing the load inertia)

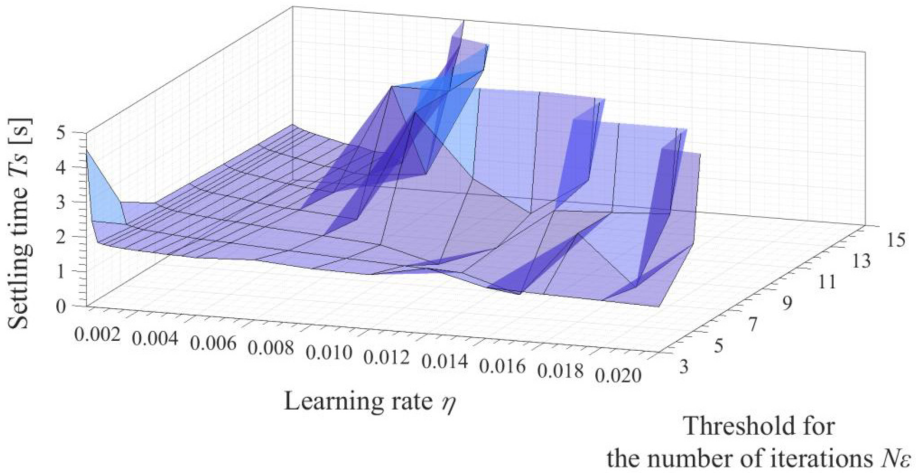
Number of data N	Learning rate η	Threshold for the number of iterations N_ε						
		3	5	7	9	11	13	15
1	0.0002	0.0023	0.0040	0.0077	0.0018	0.0022	0.0039	0.0179
2	0.0004	0.0055	0.0054	0.0015	0.0035	0.0137	0.0096	0.0093
3	0.0006	0.0102	0.0009	0.0031	0.0119	0.0078	0.0074	0.0056
4	0.0008	0.0057	0.0014	0.0129	0.0080	0.0070	0.0051	0.0041
5	0.0010	0.0037	0.0025	0.0099	0.0073	0.0053	0.0039	0.0033
6	0.0015	0.0006	0.0107	0.0063	0.0046	0.0033	0.0027	0.0022
7	0.0020	0.0010	0.0077	0.0050	0.0030	0.0025	0.0020	0.0017
8	0.0025	0.0116	0.0055	0.0037	0.0026	0.0020	0.0016	0.0014
9	0.0030	0.0098	0.0048	0.0029	0.0020	0.0016	0.0014	0.0005
10	0.0035	0.0084	0.0042	0.0024	0.0018	0.0014	0.0005	0.0004
11	0.0040	0.0072	0.0036	0.0020	0.0015	0.0013	0.0004	0.0021
12	0.0060	0.0042	0.0021	0.0014	0.0004	4.9431	0.0044	4.6882
13	0.0080	0.0026	0.0015	0.0005	5.5826	0.0106	4.9578	6.8082
14	0.0100	0.0023	0.0003	0.0051	0.0097	6.5867	6.6505	6.9485
15	0.0120	0.0017	0.0016	0.0068	6.4762	6.7786	6.9239	6.9364
16	0.0140	0.0013	0.0061	0.0096	6.3216	6.9894	7.0780	6.9886
17	0.0160	0.0003	0.0189	0.0147	6.9937	6.9735	7.1675	7.1116
18	0.0180	0.0010	0.0046	6.6503	7.0457	6.9121	7.0811	7.0688
19	0.0200	0.0016	0.0128	6.8744	6.9920	7.0130	6.9704	7.0120

Table 3. Maximum value of the feedback error E_m (after changing the load inertia)

Number of data N	Learning rate η	Threshold for the number of iterations N_ε						
		3	5	7	9	11	13	15
1	0.0002	0.0030	0.0023	0.0023	0.0022	0.0020	0.0019	0.0017
2	0.0004	0.0018	0.0018	0.0014	0.0023	0.0020	0.0018	0.0017
3	0.0006	0.0016	0.0026	0.0022	0.0019	0.0017	0.0015	0.0013
4	0.0008	0.0014	0.0023	0.0019	0.0017	0.0015	0.0012	0.0018
5	0.0010	0.0026	0.0021	0.0018	0.0015	0.0018	0.0017	0.0016
6	0.0015	0.0023	0.0018	0.0014	0.0018	0.0016	0.0015	0.0014
7	0.0020	0.0022	0.0016	0.0019	0.0016	0.0015	0.0013	0.0015
8	0.0025	0.0020	0.0014	0.0017	0.0015	0.0013	0.0012	0.0016
9	0.0030	0.0019	0.0017	0.0016	0.0014	0.0012	0.0015	0.0014
10	0.0035	0.0018	0.0018	0.0016	0.0013	0.0016	0.0014	0.0013
11	0.0040	0.0017	0.0018	0.0015	0.0013	0.0014	0.0014	0.0013
12	0.0060	0.0014	0.0015	0.0013	0.0015	0.0021	0.0015	0.0022
13	0.0080	0.0018	0.0014	0.0016	0.0032	0.0017	0.0021	3.3284
14	0.0100	0.0018	0.0013	0.0019	0.0017	0.6822	2.3544	2.2298
15	0.0120	0.0017	0.0013	0.0013	0.1770	3.6661	2.2298	2.2299
16	0.0140	0.0016	0.0020	0.0017	0.4300	2.2298	2.2298	2.2299
17	0.0160	0.0015	0.0021	0.0021	3.1378	2.2298	2.2299	2.2299
18	0.0180	0.0015	0.0014	0.0935	2.2298	2.2298	2.2299	2.2300
19	0.0200	0.0014	0.0022	3.4260	2.2297	2.2298	2.2300	2.2299



(a) Before changing load inertia



(b) After changing load inertia

FIGURE 6. Settling time, learning rate and threshold for the number of iterations with the online-offline integrated learning method

maximum value of the feedback error E_m, the learning rate η, and the threshold for the number of iterations N_ε in Section 5.1, and will describe the formulation of the settling time T_s, the learning rate η, and the threshold for the number of iterations N_ε in Section 5.2. Additionally, we will describe the derivation and the proposal of update rule of the learning rate and the threshold for the number of iterations in Section 5.3.

5.1. **Formulation of learning rates, threshold for number of iterations and error maximum value.** This section describes the formulation of the learning rate, the threshold for number of iterations and the error maximum value. For each learning rate η and the threshold for the number of iterations N_ε in Table 2 and Table 3, the maximum value of the feedback error obtained in the analysis is $E_m(i)$, and the estimated function of the maximum value of the feedback error related to the learning rate η and the threshold for the number of iterations N_ε is defined as $f_i(\eta, N_\varepsilon)$. The sum of squares of the difference between $E_m(i)$ and $f_i(\eta, N_\varepsilon)$ is defined as E_{em} and is shown in Equation (10).

$$E_{em} = \sum_{i=1}^{N} \left[E_m(i) - f_i(\eta, N_\varepsilon)\right]^2 = \sum_{i=1}^{N} \left[E_m(i) - \left(a\eta^2 + bN_\varepsilon^2 + c\eta + dN_\varepsilon + e\right)\right]^2 \quad (10)$$

TABLE 4. Settling time T_s (before changing the load inertia)

Number of data N	Learning rate η	Threshold for the number of iterations N_ε						
		3	5	7	9	11	13	15
1	0.0002	16.00	9.503	6.158	6.320	7.194	7.105	2.352
2	0.0004	9.453	6.199	7.214	7.039	2.254	2.109	1.975
3	0.0006	6.089	6.357	7.010	2.215	2.040	1.883	1.755
4	0.0008	6.100	7.108	2.279	2.065	1.883	1.738	1.640
5	0.0010	6.205	6.986	2.166	1.947	1.770	1.650	1.572
6	0.0015	7.000	2.241	1.959	1.745	1.614	1.534	1.479
7	0.0020	7.054	2.107	1.818	1.635	1.534	1.470	1.422
8	0.0025	2.398	1.992	1.717	1.567	1.482	1.423	1.373
9	0.0030	2.310	1.903	1.648	1.512	1.441	1.380	1.322
10	0.0035	2.238	1.830	1.597	1.482	1.405	1.377	1.267
11	0.0040	2.177	1.769	1.557	1.449	1.369	1.290	1.198
12	0.0060	1.996	1.612	1.445	1.326	0.914	1.226	1.196
13	0.0080	1.868	1.517	1.346	1.129	1.207	1.179	1.252
14	0.0100	1.767	1.439	1.107	1.229	1.157	1.255	1.836
15	0.0120	1.684	1.364	1.313	1.175	1.194	1.893	30.00
16	0.0140	1.611	0.940	1.247	1.127	1.477	30.00	30.00
17	0.0160	1.545	1.031	1.161	1.138	30.00	30.00	30.00
18	0.0180	1.482	1.269	1.129	1.322	2.198	0.194	0.172
19	0.0200	1.415	1.815	1.131	1.513	0.180	0.176	0.185

TABLE 5. Settling time T_s (after changing the load inertia)

Number of data N	Learning rate η	Threshold for the number of iterations N_ε						
		3	5	7	9	11	13	15
1	0.0002	4.540	2.400	1.770	1.720	1.670	1.630	1.600
2	0.0004	2.480	1.770	1.690	1.630	1.580	1.540	1.510
3	0.0006	1.870	1.730	1.640	1.570	1.530	1.480	1.450
4	0.0008	1.830	1.690	1.600	1.540	1.480	1.450	1.400
5	0.0010	1.800	1.660	1.570	1.500	1.450	1.410	1.360
6	0.0015	1.760	1.620	1.520	1.450	1.380	1.320	1.260
7	0.0020	1.730	1.580	1.480	1.400	1.310	1.230	1.160
8	0.0025	1.710	1.550	1.440	1.330	1.240	1.150	1.110
9	0.0030	1.690	1.520	1.410	1.280	1.180	1.120	1.000
10	0.0035	1.680	1.500	1.370	1.230	1.150	1.020	0.980
11	0.0040	1.660	1.480	1.330	1.210	1.100	1.000	0.740
12	0.0060	1.730	1.390	1.210	1.070	4.300	1.140	4.290
13	0.0080	1.610	1.300	1.080	4.290	4.290	4.300	15.00
14	0.0100	1.570	1.220	0.650	2.510	15.00	15.00	15.00
15	0.0120	1.890	1.150	1.260	1.639	15.00	15.00	15.00
16	0.0140	1.470	0.650	2.470	2.735	15.00	15.00	15.00
17	0.0160	1.410	2.557	2.545	15.00	15.00	15.00	15.00
18	0.0180	1.360	1.130	2.633	15.00	15.00	15.00	15.00
19	0.0200	1.310	2.530	15.00	15.00	15.00	15.00	15.00

The parameter values a, b, c, d, and e are calculated from Equations (11)-(15). And, the arranged equations from Equations (11)-(15) are shown in Equations (16)-(20).

$$\frac{\partial E_{em}}{\partial a}=\frac{\partial}{\partial a}\sum_{i=1}^{N}\left[E_m(i)-\left(a\eta^2+bN_\varepsilon^2+c\eta+dN_\varepsilon+e\right)\right]^2=0 \tag{11}$$

$$\frac{\partial E_{em}}{\partial b}=\frac{\partial}{\partial b}\sum_{i=1}^{N}\left[E_m(i)-\left(a\eta^2+bN_\varepsilon^2+c\eta+dN_\varepsilon+e\right)\right]^2=0 \tag{12}$$

$$\frac{\partial E_{em}}{\partial c}=\frac{\partial}{\partial c}\sum_{i=1}^{N}\left[E_m(i)-\left(a\eta^2+bN_\varepsilon^2+c\eta+dN_\varepsilon+e\right)\right]^2=0 \tag{13}$$

$$\frac{\partial E_{em}}{\partial d}=\frac{\partial}{\partial d}\sum_{i=1}^{N}\left[E_m(i)-\left(a\eta^2+bN_\varepsilon^2+c\eta+dN_\varepsilon+e\right)\right]^2=0 \tag{14}$$

$$\frac{\partial E_{em}}{\partial e}=\frac{\partial}{\partial e}\sum_{i=1}^{N}\left[E_m(i)-\left(a\eta^2+bN_\varepsilon^2+c\eta+dN_\varepsilon+e\right)\right]^2=0 \tag{15}$$

$$a\sum_{i=1}^{N}\eta^4+b\sum_{i=1}^{N}\eta^2N_\varepsilon^2+c\sum_{i=1}^{N}\eta^3+d\sum_{i=1}^{N}\eta^2N_\varepsilon+e\sum_{i=1}^{N}\eta^2=\sum_{i=1}^{N}\eta^2E_m(i) \tag{16}$$

$$a\sum_{i=1}^{N}\eta^2N_\varepsilon^2+b\sum_{i=1}^{N}N_\varepsilon^4+c\sum_{i=1}^{N}\eta N_\varepsilon^2+d\sum_{i=1}^{N}N_\varepsilon^3+e\sum_{i=1}^{N}N_\varepsilon^2=\sum_{i=1}^{N}N_\varepsilon^2E_m(i) \tag{17}$$

$$a\sum_{i=1}^{N}\eta^3+b\sum_{i=1}^{N}\eta N_\varepsilon^2+c\sum_{i=1}^{N}\eta^2+d\sum_{i=1}^{N}\eta N_\varepsilon+e\sum_{i=1}^{N}\eta=\sum_{i=1}^{N}\eta E_m(i) \tag{18}$$

$$a\sum_{i=1}^{N}\eta^2N_\varepsilon+b\sum_{i=1}^{N}N_\varepsilon^3+c\sum_{i=1}^{N}\eta N_\varepsilon+d\sum_{i=1}^{N}N_\varepsilon^2+e\sum_{i=1}^{N}N_\varepsilon=\sum_{i=1}^{N}N_\varepsilon E_m(i) \tag{19}$$

$$a\sum_{i=1}^{N}\eta^2+b\sum_{i=1}^{N}N_\varepsilon^2+c\sum_{i=1}^{N}\eta+d\sum_{i=1}^{N}N_\varepsilon+Ne=\sum_{i=1}^{N}E_m(i) \tag{20}$$

In addition, the calculated parameters a, b, c, d, and e are shown in Equation (21).

$$\begin{pmatrix} a\\ b\\ c\\ d\\ e \end{pmatrix}=\sum_{i=1}^{N}\begin{pmatrix} \eta^4 & \eta^2N_\varepsilon^2 & \eta^3 & \eta^2N_\varepsilon & \eta^2\\ \eta^2N_\varepsilon^2 & N_\varepsilon^4 & \eta N_\varepsilon^2 & N_\varepsilon^3 & N_\varepsilon^2\\ \eta^3 & \eta N_\varepsilon^2 & \eta^2 & \eta N_\varepsilon & \eta\\ \eta^2N_\varepsilon & N_\varepsilon^3 & \eta N_\varepsilon & N_\varepsilon^2 & N_\varepsilon\\ \eta^2 & N_\varepsilon^2 & \eta & N_\varepsilon & N \end{pmatrix}^{-1}\sum_{i=1}^{N}\begin{pmatrix} \eta^2E_m(i)\\ N_\varepsilon^2E_m(i)\\ \eta E_m(i)\\ N_\varepsilon E_m(i)\\ E_m(i) \end{pmatrix} \tag{21}$$

Apply the least squares method to the combination of the learning rate η and the threshold for the number of iterations N_ε that minimizes feedback error in Table 2 and Table 3, respectively. Thereby, the approximate curves $\tilde{E}_{mb}$ and $\tilde{E}_{ma}$ before and after changing the load inertia, that is Equations (22) and (23), are calculated. These curves are shown in Figure 7.

$$\tilde{E}_{mb}=10.2\eta^2-\left(8.8\times10^{-6}\right)N_\varepsilon^2-0.2\eta+\left(1.4\times10^{-4}\right)N_\varepsilon+\left(1.1\times10^{-3}\right) \tag{22}$$

$$\tilde{E}_{ma}=1.4\eta^2+\left(7.6\times10^{-6}\right)N_\varepsilon^2-\left(4.5\times10^{-2}\right)\eta-\left(1.7\times10^{-4}\right)N_\varepsilon+\left(2.4\times10^{-3}\right) \tag{23}$$

The approximate curves $\tilde{E}_{mb}$ before changing the load inertia and the approximate curves $\tilde{E}_{ma}$ after changing the load inertia tend to draw completely different curves when

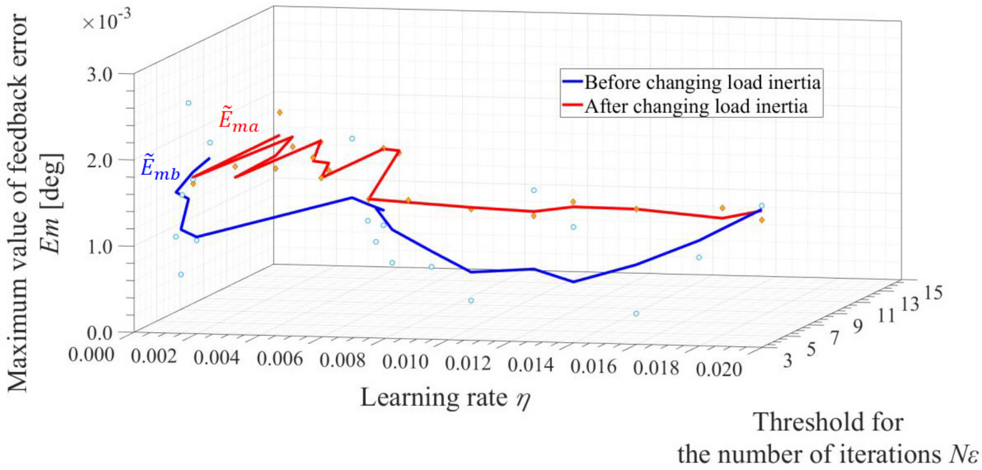


FIGURE 7. Approximation curve based on data of learning rate, threshold for the number of iterations and error maximum value

the learning rate η is increased. Also, $\tilde{E}_{mb}$ and $\tilde{E}_{ma}$ tend to draw completely different curves when the threshold for the number of iterations N_ε is increased.

5.2. **Formulation of learning rates, threshold for number of iterations and settling time.** This section describes the formulation of the learning rate, the threshold for number of iterations and the settling time. For each learning rate η and the threshold for the number of iterations N_ε in Table 4 and Table 5, the settling time obtained in the analysis is $T_s(j)$, and the estimated function of the settling time related to the learning rate η and the threshold for the number of iterations N_ε is defined as $f_j(\eta, N_\varepsilon)$. The sum of squares of the difference between $T_s(j)$ and $f_j(\eta, N_\varepsilon)$ is defined as E_{ts} and is shown in Equation (24).

$$E_{ts} = \sum_{j=1}^{N} [T_s(j) - f_j(\eta, N_\varepsilon)]^2 = \sum_{j=1}^{N} \left[T_s(j) - \left(a\eta^2 + bN_\varepsilon^2 + c\eta + dN_\varepsilon + e\right)\right]^2 \tag{24}$$

The parameter values a, b, c, d, and e are calculated from Equations (25)-(29). And, the arranged equations from Equations (25)-(29) are shown in Equations (30)-(34).

$$\frac{\partial E_{ts}}{\partial a} = \frac{\partial}{\partial a} \sum_{j=1}^{N} \left[T_s(j) - \left(a\eta^2 + bN_\varepsilon^2 + c\eta + dN_\varepsilon + e\right)\right]^2 = 0 \tag{25}$$

$$\frac{\partial E_{ts}}{\partial b} = \frac{\partial}{\partial b} \sum_{j=1}^{N} \left[T_s(j) - \left(a\eta^2 + bN_\varepsilon^2 + c\eta + dN_\varepsilon + e\right)\right]^2 = 0 \tag{26}$$

$$\frac{\partial E_{ts}}{\partial c} = \frac{\partial}{\partial c} \sum_{j=1}^{N} \left[T_s(j) - \left(a\eta^2 + bN_\varepsilon^2 + c\eta + dN_\varepsilon + e\right)\right]^2 = 0 \tag{27}$$

$$\frac{\partial E_{ts}}{\partial d} = \frac{\partial}{\partial d} \sum_{j=1}^{N} \left[T_s(j) - \left(a\eta^2 + bN_\varepsilon^2 + c\eta + dN_\varepsilon + e\right)\right]^2 = 0 \tag{28}$$

$$\frac{\partial E_{ts}}{\partial e} = \frac{\partial}{\partial e} \sum_{j=1}^{N} \left[T_s(j) - \left(a\eta^2 + bN_\varepsilon^2 + c\eta + dN_\varepsilon + e\right)\right]^2 = 0 \tag{29}$$

$$a\sum_{j=1}^{N}\eta^4+b\sum_{j=1}^{N}\eta^2N_\varepsilon^2+c\sum_{j=1}^{N}\eta^3+d\sum_{j=1}^{N}\eta^2N_\varepsilon+e\sum_{j=1}^{N}\eta^2=\sum_{j=1}^{N}\eta^2T_s(j) \quad (30)$$

$$a\sum_{j=1}^{N}\eta^2N_\varepsilon^2+b\sum_{j=1}^{N}N_\varepsilon^4+c\sum_{j=1}^{N}\eta N_\varepsilon^2+d\sum_{j=1}^{N}N_\varepsilon^3+e\sum_{j=1}^{N}N_\varepsilon^2=\sum_{j=1}^{N}N_\varepsilon^2T_s(j) \quad (31)$$

$$a\sum_{j=1}^{N}\eta^3+b\sum_{j=1}^{N}\eta N_\varepsilon^2+c\sum_{j=1}^{N}\eta^2+d\sum_{j=1}^{N}\eta N_\varepsilon+e\sum_{j=1}^{N}\eta=\sum_{j=1}^{N}\eta T_s(j) \quad (32)$$

$$a\sum_{j=1}^{N}\eta^2N_\varepsilon+b\sum_{j=1}^{N}N_\varepsilon^3+c\sum_{j=1}^{N}\eta N_\varepsilon+d\sum_{j=1}^{N}N_\varepsilon^2+e\sum_{j=1}^{N}N_\varepsilon=\sum_{j=1}^{N}N_\varepsilon T_s(j) \quad (33)$$

$$a\sum_{j=1}^{N}\eta^2+b\sum_{j=1}^{N}N_\varepsilon^2+c\sum_{j=1}^{N}\eta+d\sum_{j=1}^{N}N_\varepsilon+Ne=\sum_{j=1}^{N}T_s(j) \quad (34)$$

In addition, the calculated parameters a, b, c, d, and e are shown in Equation (35).

$$\begin{pmatrix} a \\ b \\ c \\ d \\ e \end{pmatrix}=\sum_{j=1}^{N}\begin{pmatrix} \eta^4 & \eta^2N_\varepsilon^2 & \eta^3 & \eta^2N_\varepsilon & \eta^2 \\ \eta^2N_\varepsilon^2 & N_\varepsilon^4 & \eta N_\varepsilon^2 & N_\varepsilon^3 & N_\varepsilon^2 \\ \eta^3 & \eta N_\varepsilon^2 & \eta^2 & \eta N_\varepsilon & \eta \\ \eta^2N_\varepsilon & N_\varepsilon^3 & \eta N_\varepsilon & N_\varepsilon^2 & N_\varepsilon \\ \eta^2 & N_\varepsilon^2 & \eta & N_\varepsilon & N \end{pmatrix}^{-1}\sum_{j=1}^{N}\begin{pmatrix} \eta^2T_s(j) \\ N_\varepsilon^2T_s(j) \\ \eta T_s(j) \\ N_\varepsilon T_s(j) \\ T_s(j) \end{pmatrix} \quad (35)$$

Apply the least squares method to the combination of the learning rate η and the threshold for the number of iterations N_ε, which has the shortest convergence time in Table 4 and Table 5, respectively. Thereby, the approximate curves $\tilde{T}_{sb}$ and $\tilde{T}_{sa}$ before and after changing the load inertia, that is Equations (36) and (37), are calculated. These curves are shown in Figure 8.

$$\tilde{T}_{sb}=\left(6.5\times10^3\right)\eta^2+\left(3.2\times10^{-4}\right)N_\varepsilon^2-220.8\eta-\left(9.9\times10^{-2}\right)N_\varepsilon+3.3 \quad (36)$$

$$\tilde{T}_{sa}=\left(7.3\times10^3\right)\eta^2+\left(6.9\times10^{-3}\right)N_\varepsilon^2-232.8\eta-0.3N_\varepsilon+3.8 \quad (37)$$

The approximate curve $\tilde{T}_{sb}$ before changing the load inertia and the approximate curve $\tilde{T}_{sa}$ after changing the load inertia tend to draw similar curves up to a certain value of

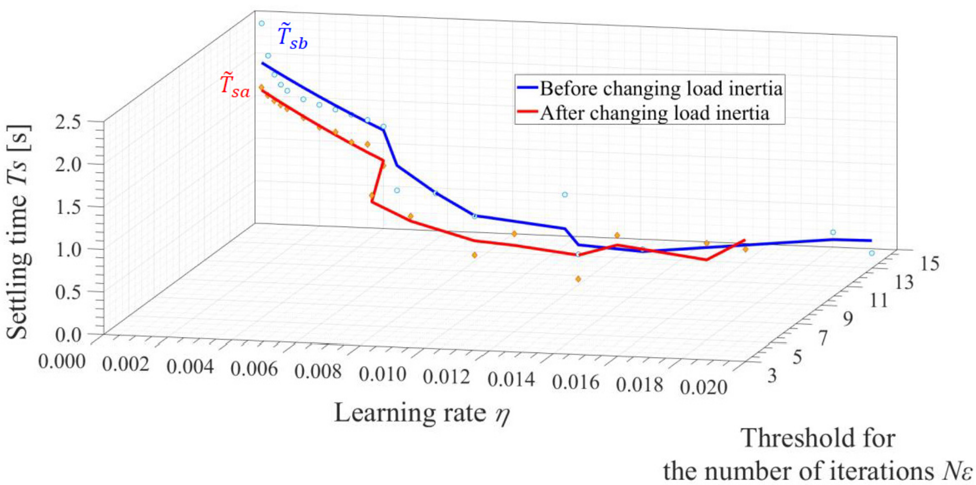


FIGURE 8. Approximation curve based on data of learning rate, threshold for the number of iterations and settling time

the learning rate when the learning rate η and the threshold for the number of iterations N_ε are increased. Also, $\tilde{T}_{sb}$ and $\tilde{T}_{sa}$ tend to draw different curves when the learning rate η and the threshold for the number of iterations N_ε exceed the certain value.

5.3. **Derivation of update rule of learning rate and threshold for number of iterations.** In order to derive the update rule, we formulate the learning rate. To do that, we set the index values of the maximum value of the feedback error E_{mr} and the settling time T_{sr}, and calculate the threshold for the number of iterations $\tilde{N}_\varepsilon$. In order to derive the threshold for the number of iterations $\tilde{N}_\varepsilon$, we calculate the simultaneous equations of the approximation curves $\tilde{E}_{mb}$ and $\tilde{T}_{sb}$ or the approximation curves $\tilde{E}_{ma}$ and $\tilde{T}_{sa}$. The parameters of $\tilde{E}_{ma}$ or $\tilde{E}_{mb}$ are defined as a_1, b_1, c_1, d_1, and e_1, respectively, $\tilde{T}_{sb}$ or $\tilde{T}_{sa}$ are defined as a_2, b_2, c_2, d_2, and e_2, respectively. In addition, the coefficients of the equation for the threshold for the number of iterations $\tilde{N}_\varepsilon$ are defined as A, B, and C, which are shown in Equations (38) and (39).

$$\tilde{N}_\varepsilon = A\tilde{\eta}^2 + B\tilde{\eta} + C \tag{38}$$

$$\begin{aligned} A &= -\frac{a_1b_2 - a_2b_1}{d_1b_2 - d_2b_1} \\ B &= -\frac{c_1b_2 - c_2b_1}{d_1b_2 - d_2b_1} \\ C &= \frac{b_2E_{mr} - b_1T_{sr} - (e_1b_2 - e_2b_1)}{d_1b_2 - d_2b_1} \end{aligned} \tag{39}$$

Substitute $\tilde{N}_\varepsilon$ into $\tilde{E}_{mb}$ or $\tilde{E}_{ma}$ and use Ferrari's solution for the expanded equation. The coefficients of the expanded equation are defined as D, E, F, G, and H, and are shown in Equations (40) and (41).

$$D\eta^4 + E\eta^3 + F\eta^2 + G\eta + H = 0 \tag{40}$$

$$\begin{aligned} D &= b_1A^2 \\ E &= 2b_1AB \\ F &= a_1 + b_1\left(B^2 + 2AC\right) + d_1A \\ G &= 2b_1BC + c_1 + d_1B \\ H &= b_1C^2 + d_1C + e_1 - \tilde{E}_{mb} \end{aligned} \tag{41}$$

Substituting the equation $\tilde{\eta} = z - E/4D$ with the coefficient z into Equation (40), and the coefficients of the expanded fourth power equation are defined as p, q, and r, and are shown in Equations (42) and (43).

$$z^4 + pz^2 + qz + r = 0 \tag{42}$$

$$\begin{aligned} p &= -\frac{3E^2}{8D^2} + \frac{F}{D} \\ q &= \frac{E^3}{8D^3} - \frac{EF}{2D^2} + \frac{G}{D} \\ r &= -\frac{3E^4}{256D^4} + \frac{E^2F}{16D^3} - \frac{EG}{4D^2} + \frac{H}{D} \end{aligned} \tag{43}$$

Take z to the fourth power in Equation (42) as the left side, and add the equation $tz^2 + t^2/4$ with the coefficient t to both sides. Then, completing the square of the equation is shown in Equation (44).

$$\left(z^2 + \frac{t}{2}\right)^2 = (t-p)\left\{z - \frac{q}{2(t-p)}\right\}^2 - \frac{q^2}{4(t-p)} + \frac{t^2}{4} - r \tag{44}$$

In order to make $-q^2/4(t-p)+t^2/4-r=0$ on the right side of Equation (44), the simplification of $-q^2/4(t-p)+t^2/4-r=0$ for t is shown in Equation (45).

$$t^3 - pt^2 - 4rt + 4pr - q^2 = 0 \tag{45}$$

Define one of the solutions of Equation (45) as t_0, Equation (44) can be transformed into Equation (46).

$$\left(z^2 + \frac{t_0}{2}\right)^2 = (t_0 - p)\left\{z - \frac{q}{2(t_0 - p)}\right\}^2 \tag{46}$$

Remove the square root of both sides of Equation (46) and find the solution for z with the quadratic formula. Then, substituting the solution of z into $\tilde{\eta} = z - E/4D$, the learning rate $\tilde{\eta}$ can be formulated with the following four solutions. However, if the four solutions include solutions that do not satisfy $0 < \tilde{\eta} < 1$ and complex numbers, they are not used.

$$\begin{aligned} \tilde{\eta}_{1,2} &= -\frac{E}{4D} + \frac{1}{2}\left(-\sqrt{t_0 - p} \pm \sqrt{-t_0 - p + \frac{2q}{\sqrt{t_0 - p}}}\right) \\ \tilde{\eta}_{3,4} &= -\frac{E}{4D} + \frac{1}{2}\left(\sqrt{t_0 - p} \pm \sqrt{-t_0 - p - \frac{2q}{\sqrt{t_0 - p}}}\right) \end{aligned} \tag{47}$$

Substitute the solution for the learning rate $\tilde{\eta}$ into Equation (38), the threshold for the number of iterations $\tilde{N}_\varepsilon$ can be formulated as Equation (48). However, if the solutions are the negative numbers, the complex numbers and the large numbers, they are not used.

$$\tilde{N}_{\varepsilon 1,\varepsilon 2} = A\tilde{\eta}_{1,2}^2 + B\tilde{\eta}_{1,2} + C, \quad \tilde{N}_{\varepsilon 3,\varepsilon 4} = A\tilde{\eta}_{3,4}^2 + B\tilde{\eta}_{3,4} + C \tag{48}$$

6. **Analysis Results of Servo System Based on Update Rule of Learning Rate and Threshold for Number of Iterations.** In this section, first, we substitute the calculated threshold for the number of iterations $\tilde{N}_\varepsilon$ and the learning rate $\tilde{\eta}$ based on the update rule of the learning rate η and the threshold for the number of iterations N_ε into the online-offline integrated learning method. Next, we simulate the position control of servo motor before and after changing the load inertia. The results of this simulation are shown in Section 6.1. Finally, the considerations of the match between the obtained value of the results from the simulation and the index values are summarized in Sections 6.2 and 6.3.

6.1. **Analysis results.** The index values are determined to be $E_{mr} = 0.002$ degrees and $T_{sr} = 1.5$ seconds. The learning rate and the threshold for the number of iterations before and after changing the load inertia calculated for those index values were $\tilde{\eta} = 0.022$, $\tilde{N}_\varepsilon = 2$ and $\tilde{\eta} = 0.00029$, $\tilde{N}_\varepsilon = 20$. And, Figure 9 shows the results of the feedback error e before and after changing load inertia. Figure 10 shows the results of the output u_n of the neural network compensator before and after changing load inertia. Figure 11 shows the output u_f of the feedback controller before and after changing load inertia.

6.2. **Considerations of steady-state characteristics.** First, consider the results before changing the load inertia. The maximum value of the steady-state error at the calculated learning rate $\tilde{\eta}$ and threshold for the number of iterations $\tilde{N}_\varepsilon$ is E_{m1} in Figure 9, which is 0.00046 degrees. The time at that time is 6.125 seconds. And then, the output of the neural network is u_{n1} in Figure 10, and the output of the feedback controller is u_{f1} in Figure 11. The maximum value of the steady-state error E_{m1} at the calculated learning rate $\tilde{\eta}$ and threshold for the number of iterations $\tilde{N}_\varepsilon$ is about 1/4.3 of the index value E_{mr} 0.002 degrees. Therefore, from the analysis results of the simulation, the maximum value of the steady-state error E_{m1} is within the range of the index value E_{mr}.

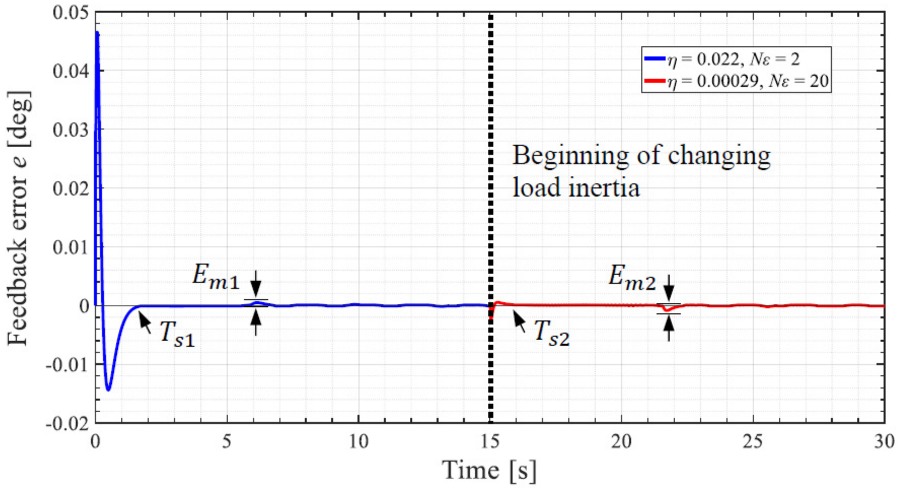


FIGURE 9. Feedback error with the online-offline integrated learning method (before and after changing load inertia)

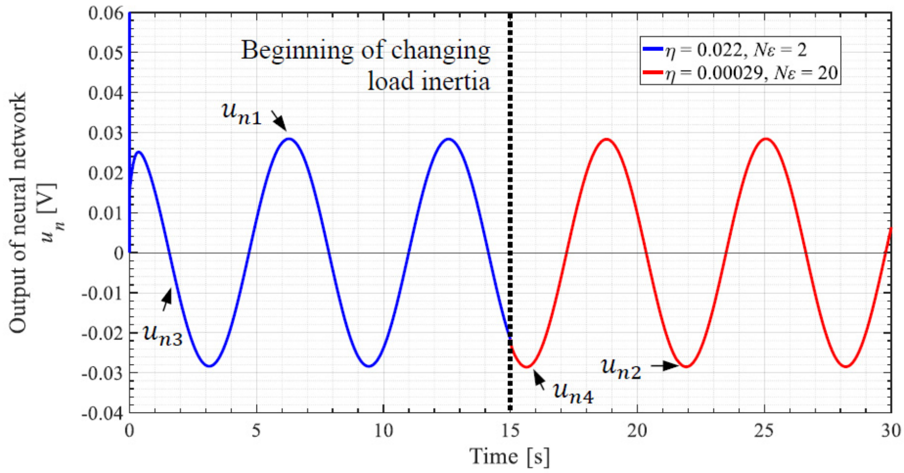


FIGURE 10. Output of neural network with the online-offline integrated learning method (before and after changing load inertia)

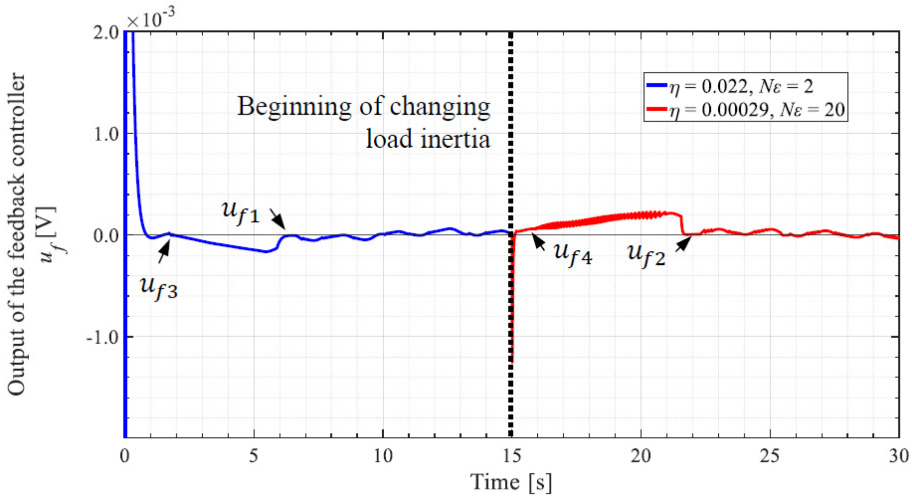


FIGURE 11. Output of the feedback controller with the online-offline integrated learning method (before and after changing load inertia)

Next, consider the results after changing the load inertia. The maximum value of the steady-state error of the calculated learning rate $\tilde{\eta}$ and threshold for the number of iterations $\tilde{N}_\varepsilon$ is E_{m2} in Figure 9, which is 0.00086 degrees. The time at that time is 21.73 seconds. And then, the output of the neural network is u_{n2} in Figure 10, and the output of the feedback controller at that time is u_{f2} in Figure 11. The maximum value of the steady-state error E_{m2} at the calculated learning rate $\tilde{\eta}$ and threshold for the number of iterations $\tilde{N}_\varepsilon$ is about 1/2.3 of the index value E_{mr} 0.002 degrees. Therefore, from the analysis results of the simulation, the maximum value of the steady-state error E_{m2} is within the range of the index value E_{mr}.

6.3. **Considerations of transient characteristics.** First, consider the results before changing the load inertia. The settling time of the calculated learning rate $\tilde{\eta}$ and threshold for the number of iterations $\tilde{N}_\varepsilon$ is T_{s1} in Figure 9, which is 1.692 seconds. The value of the feedback error at that time is within the threshold range. And then, the output of the neural network is u_{n3} in Figure 10, and the output of the feedback controller at that time is u_{f3} in Figure 11. The settling time T_{s1} at the calculated learning rate $\tilde{\eta}$ and threshold for the number of iterations $\tilde{N}_\varepsilon$ is about 1.1 times of the index value T_{sr} 0.002 degrees, but is close to the index value of settling time. Therefore, from the analysis results of the simulation, the maximum value of the steady-state error T_{s1} is within the range of the index value T_{sr}.

Next, consider the results after changing the load inertia. The settling time of the calculated the learning rate $\tilde{\eta}$ and the threshold for the number of iterations $\tilde{N}_\varepsilon$ is T_{s2} in Figure 9, which is 0.920 seconds. The value of the feedback error is within threshold range. Then, the output of the neural network is u_{n4} in Figure 10, which is a little larger than the output value before changing the load inertia, the output of the feedback controller at that time is u_{f4} in Figure 11. The settling time T_{s2} at the calculated learning rate $\tilde{\eta}$ and threshold for the number of iterations $\tilde{N}_\varepsilon$ is about 1/1.6 of the index value T_{sr} 0.002 degrees. Therefore, from the analysis results of the simulation, the maximum value of the steady-state error T_{s2} is within the range of the index value T_{sr}.

From the above results and considerations, the relationship between the learning rate η, the threshold for the number of iterations N_ε, and the responsiveness and the robustness of the servo system were clarified. And, the update rule for the learning rate η and the threshold for the number of iterations N_ε was derived from the formulation of the learning rate, the feedback error, and the settling time. Moreover, it can be determined that the proposed update rule for the learning rate η and the threshold for the number of iterations N_ε is suitable to enable self-tuning of the learning rate η and the threshold for the number of iterations N_ε.

7. **Conclusions.** Until now, in previous study, in order to deal with load fluctuations and disturbances, and to realize real-time control, we proposed an online-offline integrated learning method for the position control of servo system based on the neural network control [7]. However, the relationship between the learning rate η, the threshold for the number of iterations N_ε within the sampling time, and the responsiveness and the robustness of the servo system has not yet been clarified, so the tuning of the learning rate η and the threshold for the number of iterations N_ε requires a lot of time, and it cannot say that the learning efficiency is good.

Therefore, in this study, the relationship between the learning rate η, the threshold for the number of iterations N_ε, and the responsiveness and the robustness of the servo system for the position control was clarified from simulation and analysis data. Moreover, by using the least squares method, the relationship between them was formulated, and the update rule of the learning rate η and the threshold for the number of iterations N_ε to enable self-tuning was proposed.

First, for the position control of the servo system, the analysis environment using a dynamic model of the position control of the servo motor was built, and the algorithm of the online-offline integrated learning method proposed by the previous study was explained. Next, before and after changing the load inertia, the steady-state characteristics and transient characteristics of the servo system related to the increase or decrease of the learning rate η and the threshold for the number of iterations N_ε by simulating the online-offline integrated learning method were shown. And, the relationship between the learning rate η, the threshold for the number of iterations N_ε, the feedback error, and the settling time was formulated. With this formulation, the update rule of the learning rate η and the threshold for the number of iterations N_ε to enable self-tuning was proposed. Moreover, the index values of the feedback error and the settling time were set, and the learning rate η and the threshold for the number of iterations N_ε were calculated based on the proposed update rule, and the calculation the learning rate $\tilde{\eta}$ and the threshold for the number of iterations $\tilde{N}_\varepsilon$ was applied into the online-offline integrated learning method before and after changing the load inertia of the servo motor.

Based on the results and the considerations of simulation, the relationship between the learning rate η, the threshold for the number of iterations N_ε, and the responsiveness and the robustness of the servo system can be clarified by the formulation, and it can be concluded that the proposed update rule for the learning rate η and the threshold for the number of iterations N_ε is suitable to enable self-tuning of the learning rate η and the threshold for the number of iterations N_ε.

As for future research, we will study an update rule with better learning efficiency considering the minimum value of learning.

REFERENCES

[1] L. Chen and M. Tsutsumi, Compensation for positioning error of CNC machine tools using neural network. Compensation for multivariant positioning error, *JSME-Part C*, vol.61, no.583, pp.1224-1229, 1995.

[2] L. Chen, M. Tsutsumi, D. F. Ge and A. Moran, Stabilization for feed drive accuracy of CNC machine tools with autonomous control system. Application of neural network, *JSME-Part C*, vol.61, no.591, pp.4504-4509, 1995.

[3] Y. Shinzhi, R.-D. Pinzon-Morales and Y. Hirata, Adaptive robot control by the general-purpose cerebellar neural network model, *The Japanese Society for Artificial Intelligence*, SIG-AGI-002-01, 2016.

[4] Q. Zhu, Y. Lu, J. Zhou, X. Niu, O. Xie, Y. Shen, F. Niu and Y. Zhu, Two-degree-of-freedom compound control based on RBF neural network for air conditioning temperature control system, *Int. J. Innov. Comput. Inf. Control*, vol.16, no.2, pp.445-456, 2020.

[5] M. Zhuang and Q. Zhu, The internal mode fractional-order PID control based on neural network for the temperature of air-conditioned rooms, *Int. J. Innov. Comput. Inf. Control*, vol.17, no.3, pp.1019-1028, 2021.

[6] C. Lee and B.-D. Lee, Enhancement for automatic extraction of RoIs for bone age assessment based on deep neural networks, *ICIC Express Letters*, vol.14, no.2, pp.163-170, 2020.

[7] M. Morita, Q. Huang and M. Morishita, Online-offline integrated learning method of the neural network control for a servo system, *Int. J. Innov. Comput. Inf. Control*, vol.17, no.4, pp.1447-1464, 2021.

[8] A. Kanda, S. Fujita and T. Ae, Acceleration by prediction for error back propagation algorithm of neural network, *Transactions of the Institute of Electronics, Information and Communication Engineers*, vol.J76-D-II, no.1, pp.132-139, 1993.

[9] K. Watanabe, K. Serizawa and L. Hou, Speeding up of the neural network learning by adaptive tuning of the learning rate, *Transactions of the Society of Instrument and Control Engineers*, vol.30, no.9, pp.1093-1099, 1994.

[10] J. P. Cater, Successfully using peak learning rates of 10 (and greater) in back-propagation networks with the heuristic learning algorithm, *Proc. of the IEEE International Conference on Neural Networks*, pp.645-651, 1987.
[11] T. P. Vogl, J. K. Mangis, A. K. Rigler, W. T. Zink and D. L. Alkon, Accelerating the convergence of the back-propagation method, *Biological Cybernetics*, vol.59, pp.257-263, 1988.
[12] Y. Zhang, K. Mao, S. Leigh et al., A parametric study of 3D printed polymer gears, *The International Journal of Advanced Manufacturing Technology*, vol.107, nos.11-12, pp.4481-4492, 2020.
[13] K. Ochiai, N. Toda and S. Usui, New accelerated learning algorithm to reduce the oscillation of weights in multilayered neural networks, *IJCNN-92 (Baltimore)*, vol.1, pp.914-919, 1992.
[14] K. Ochiai, N. Toda and S. Usui, A new learning algorithm by reducing the oscillation of weights for feedforward neural network: Kick out algorithm, *IEEJ Transactions on Electronics, Information and Systems*, vol.113, no.12, pp.1154-1162, 1993.
[15] H. Fukuda, M. Odagaki and O. Hiwaki, Identification of stimulated sites using artificial neural networks based on transcranial magnetic stimulation-elicited motor evoked potentials and finger forces, *IEEJ Transactions on Electronics, Information and Systems*, vol.130, no.12, pp.2204-2209, 2010.
[16] R. A. Jacobs, Increased rates of convergence through learning rate adaptation, *Neural Networks*, vol.1, pp.295-307, 1988.
[17] Q. Shi, H.-K. Lam, B. Xiao and S.-H. Tsai, Adaptive PID controller based on Q-learning algorithm, *CAAI Transactions on Intelligence Technology*, vol.3, no.4, pp.235-244, 2018.
[18] H. Gomi and M. Kawato, Learning control of a closed loop system using feedback-error-learning, *Transactions of the Institute of Systems Control and Information Engineers*, vol.4, no.1, pp.37-47, 1991.

Author Biography

Masakazu Morita received the B.Sc. degree from Kogakuin University, Japan, and received the M.Sc. degree from Kogakuin University, Japan. He is currently a Ph.D. student at the Control System Laboratory and Electromagnetic Energy Application and System Laboratory, Graduate School of Engineering, Kogakuin University, Japan. His research interests include the control theory and the position control of Permanent Magnet Synchronous Motor.

Qingjiu Huang received the B.Sc. degree from Huazhong University of Science and Technology, China; the M.Sc. and Ph.D. degrees from Chiba University, Japan. He is currently an Associate Professor at the Control System Laboratory, Graduate School of Engineering, Kogakuin University, Japan. His research interests include the control theory, such as robust control, optimal control, adaptive control, and intelligent control; and the kinematics, dynamics, motion control of multi-joint robot arm, multi-legged robot, biped robot, and four-wheel car.

Mimpei Morishita received a Bachelor of Science degree and a Master's degree in Electrical Engineering from Waseda University in 1980 and 1982, respectively. After that, he joined Toshiba. He received his Ph.D. in electrical engineering from Waseda University in 2000. He entered Kogakuin University as a professor in the Department of Electrical and Electronic Engineering in 2010 and retired from Toshiba Corporation.

Prof. Morishita is currently a full-time professor at the Department of Electrical and Electronic Engineering, Kogakuin University. His current topics of interest are high-efficiency synchronous motors and generators, anti-torque ripple drive control, simplification of electromagnetic suspension systems, and the application of Halbach arrays of permanent magnets.