Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

Iterative Feedback Tuning of Cascade Control for Position and Velocity of Two-Mass System

Hanul Jung * **Kiho Jeon** * **Sehoon Oh** *

* *Department of Robotics Engineering, DGIST, Daegu, Korea (e-mail: jungsky14@dgist.ac.kr, kiho3375@dgist.ac.kr and sehoon@dgist.ac.kr).*

Abstract: The inaccuracy of position and velocity controllers due to the vibration in the two-mass system in industrial robots cause devastating problems in both safety and productivity. To solve this problem, a method for tuning a cascade controller applied to two-mass systems based on Iterative Feedback Tuning (IFT) is proposed. The proposed new iterative feedback tuning method utilized a modified cost function to optimize the gains in a cascade control system and to address the complexity of the two-mass system due to higher-order dynamics. The performance of the proposed iterative feedback tuning method is verified through several simulations.

Keywords: Two-inertia system, iterative feedback tuning, cascade control

1. INTRODUCTION

Most industrial robots have elastic elements, such as bearings and gears, and the elasticity from these elements can cause vibrations during the operation of the robot. This problem makes it difficult for the robot to achieve its desired position and velocity (Hillsley and Yurkovich (1993)). The inaccurate position and velocity of industrial robots have devastating impacts on productivity, including product defects and production cycles. Hence, high precision on both position and velocity controllers is critical to industrial robots.

The cascade controller has two feedback loops to control the position and velocity of a robotic system. The outer control loop of the cascade structure receives the position data and then adjusts the set-point to be controlled by the inner control loop, while the inner control loop controls the set-point by receiving the velocity data. Therefore, the controllers of the inner and outer control loops need to be tuned separately, and it takes a lot of time and effort to optimize each of the two controllers.

Research on controller tuning has been classified into two categories in various literature. The first classification is the model-based method. The model-based controller utilizes the system model to generate control inputs that achieve the desired performance. The controller design is simplified because the model of the system is used for the controller design(Huang et al. (2005)). Although the model-based controller has these characteristics, various papers have described specific situations in which it is difficult to obtain the system model, and point out that additional time and procedures are required to obtain the system model (Pintelon et al. (1994)). In this respect, the data-driven method is proposed as an alternative method.

The data-based method is a model-free technique for tuning controllers by processing data from several experiments, such as virtual reference feedback tuning, correlation-based tuning, and iterative feedback tuning (IFT). Among these, IFT has been applied to motion controllers in various industrial applications such as semiconductor equipment and precision stages due to its flexibility to be used to various controllers. IFT updates the control parameter set to reduce performance-relevant cost function. By processing the data from a few experiments to obtain an unbiased gradient estimate, the gradient estimate is used to update a new set of control parameters every iteration. Since the gradient estimate has a direction of control performance change, changing the control parameter in the direction of the gradient estimate can update the control parameter set to minimize the cost function. This IFT method can also be applied to sophisticated controllers according to a specified set of control parameters.

The application of IFT to cascade controllers is presented in (Kissling et al. (2009)), however, this is not time efficient because the inner loop and outer loop controllers were tuned sequentially. To solve this problem, the method of tuning the controllers of the inner and outer control loops were studied at the same time, but the tuning freedom was reduced by simplifying the performance-relevant cost function (Tesch et al. (2016)).

In this paper, a novel IFT method of the cascade controller is proposed. Also, the proposed technique is applied to a two-mass system, shown in Fig. 1. Because these type of systems have higher-order dynamics, both the design of both position and velocity controllers becomes a challenging task Hu et al. (2014). However, the proposed tuning

⋆ This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIP) (NRF-2019R1A2C2011444).

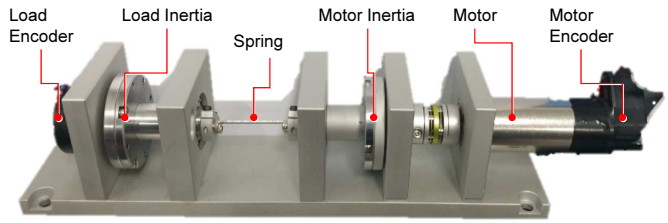

Fig. 1. Two-mass system.

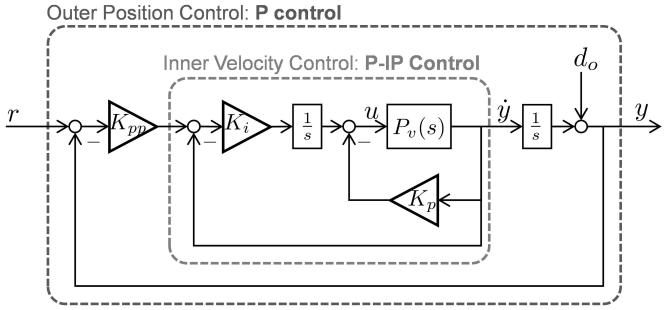

Fig. 2. Cascade control for the two-mass system.

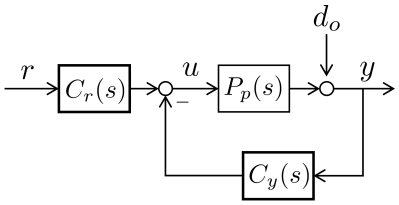

Fig. 3. Two-degree-of-freedom cascade control.

technique uses both position and velocity data from inner and outer control loops and also considers the influence of control effort.

2. POSITION AND VELOCITY OF TWO-MASS SYSTEM CASCADE CONTROLLER

The cascade controller of the two-mass system is shown in Fig. 2. Where P_v is the velocity output model of the system, r, y, $\dot{y}$ are the desired position reference, measured position and measured velocity, respectively, and u and d_o are respectively the control effort and outer disturbance. The control objective of the cascade control is to minimize $e = r - y$ and $e_v = \dot{r} - \dot{y}$. Where $\dot{r}$ is the derivative of the position command. Therefore, the outer control loop uses a proportional controller K_{pp} to minimize e and the inner control loop uses proportional K_p and integral K_i controllers to minimize e_v. Therefore, the closed-loop transfer functions of the cascade control are

$$\frac{y}{r} = \frac{K_{pp}K_iP_v(s)}{s^2 + (K_ps^2 + K_is + K_{pp}K_i)P_v(s)} \tag{1}$$

$$\frac{y}{d_o} = \frac{s}{s^2 + (K_ps^2 + K_is + K_{pp}K_i)P_v(s)} \tag{2}$$

This cascade controller can be replaced with the equivalent controller two-degree-of-freedom (TDOF) cascade controller separately from controller C_r for command and controller C_y for the output data.

$$u_c = C_r(\rho)r(t) - C_y(\rho)y(t) \tag{3}$$

where C_r and C_y are controllers of the TDOF cascade control and are composed of the control parameters $\rho = [K_p \;\; K_i \;\; K_{pos}]$. $K_{pos} = K_{pp}K_i$ is replaced to eliminate between K_i and K_{pp}. Therefore, the transfer function of TDOF cascade control C_r and C_y is as follows:

$$C_r(\rho) = \frac{K_{pos}}{s} \tag{4}$$

$$C_y(\rho) = \frac{K_ps^2 + K_is + K_{pos}}{s} \tag{5}$$

$$P_p(s) = \frac{P_v}{s} \tag{6}$$

where P_p is the system location output model. As a result, Fig. 2 can be represented by TDOF cascade control which is an equivalent controller, and the block diagram is shown in Fig. 3. TDOF cascade control is the control structure used in the general IFT tuning method (Hjalmarsson et al. (1998)).

3. ITERATIVE FEEDBACK TUNING FOR CASCADE CONTROLLER

3.1 Overview of Iterative Feedback Tuning Method

Iterative feedback tuning is used to solve the cost function $J(\rho)$ and H_2 optimization problem to obtain optimal control parameter ($\rho^\star$).

$$\rho^\star = \arg\min_{\rho} J(\rho) \tag{7}$$

where $J(\rho)$ is a performance-relevant cost function designed for the purpose of cascade controllers. The proposed cascade control considers the control effort (u) of the servo system while effectively minimizing e and e_v. Therefore, the performance-relevant cost function, which is proposed in this paper, is as follows:

$$J(\rho) = \frac{1}{2N}[e^Te + \lambda_v e_v^T e_v + \lambda_u u^T u] \tag{8}$$

where $e = [e(1) \dots e(N)]^T$, $e_v = [e_v(1) \dots e_v(N)]^T$, $u = [u(1) \dots u(N)]^T$ is respectively the position error, velocity error, and control effort. the data sampled signal as number of samples (N), and λ_v and λ_u are weights of velocity error and control effort, respectively.

$$\lambda_v = \frac{\|y - r\|}{\|\dot{y} - r_v\|} \tag{9}$$

$$\lambda_u = \lambda \tag{10}$$

where the λ is a positive scalar. To solve (7), an appropriate ρ must be found for (8) that satisfies:

$$0 = \frac{\partial J}{\partial \rho}(\rho) = \frac{1}{N}\left[\frac{\partial e^T}{\partial \rho}(\rho)e(\rho) + \lambda_v \frac{\partial e_v^T}{\partial \rho}(\rho)e_v + \lambda_u \frac{\partial u}{\partial \rho}(\rho)u\right] \tag{11}$$

If the gradient of the cost function, $\frac{\partial J}{\partial \rho}(\rho)$, is obtained, the optimal control parameters can be found using the following iterative algorithm.

$$\rho^{i+1} = \rho^i - \gamma^i R^{i-1}\frac{\partial J}{\partial \rho}(\rho) \tag{12}$$

where γ^i is a positive real scalar that determines the step size, and R^i is an approximation of the Hessian.

$$\gamma^i = \frac{\mu}{i} \tag{13}$$

$$R^i = \frac{1}{N}\frac{\partial J^T}{\partial \rho}(\rho)\frac{\partial J}{\partial \rho}(\rho) \tag{14}$$

where μ is the initial step size. To obtain the optimal set of control parameters using (12), the gradient of the cost functions of (11) and (14) is calculated. In the next subsection, the gradient of the cost function can be introduced experimentally (Robbins and Monro (1951)).

3.2 Gradient of Cost Function

The control objective of the proposed cascade control is divided in three-fold: 1) position tracking, 2) velocity tracking, and 3) efficiency of control effort.

1) Position Tracking — $e(\rho)$ is the difference between the expected output $y_d(\rho)$ from the command input r and the output $y(\rho)$ in the closed loop.

$$e(\rho) = T(\rho)r + S(\rho)d_o - y_d \tag{15}$$

where $T(\rho)$ and $S(\rho)$ are the complementary sensitivity and sensitivity functions of the closed-loop, respectively, then

$$T(\rho) = \frac{C_r(\rho)P_p(s)}{1 + C_y(\rho)P_p(s)} \tag{16}$$

$$S(\rho) = \frac{1}{1 + C_y(\rho)P_p(s)} \tag{17}$$

$$y_d = T_d(\rho)r \tag{18}$$

where $T_d(\rho)$ is the reference model. In order to obtain (11), $\frac{\partial e}{\partial \rho}(\rho)$ must be solved as follows:

$$\frac{\partial e}{\partial \rho}(\rho) = \frac{\partial T}{\partial \rho}(\rho)r + \frac{\partial S}{\partial \rho}(\rho)d_o \tag{19}$$

where the derivatives of $T(\rho)$ and $S(\rho)$ are:

$$\begin{aligned}\frac{\partial T}{\partial \rho}(\rho) &= \frac{\frac{\partial C_r}{\partial \rho}(\rho)P_p}{1 + C_y(\rho)P_p} - \frac{\frac{\partial C_y}{\partial \rho}(\rho)C_r(\rho){P_p}^2}{(1 + C_y(\rho)P_p)^2} \\ &= \frac{1}{C_r(\rho)}\frac{\partial C_r}{\partial \rho}(\rho)T(\rho) - \frac{1}{C_r(\rho)}\frac{\partial C_y}{\partial \rho}(\rho)T^2(\rho)\end{aligned} \tag{20}$$

$$\frac{\partial S}{\partial \rho}(\rho) = -\frac{P_p\frac{\partial C_y}{\partial \rho}(\rho)}{(1 + C_y(\rho)P_p)^2} = \frac{1}{C_r(\rho)}\frac{\partial C_y}{\partial \rho}(\rho)T(\rho)S(\rho) \tag{21}$$

Here, (19) is rearranged using (20) and (21) as follows.

$$\frac{\partial e}{\partial \rho}(\rho) = \frac{1}{C_r(\rho)}\left[(\frac{\partial C_r}{\partial \rho} - \frac{\partial C_y}{\partial \rho})T(\rho)r + \frac{\partial C_y}{\partial \rho}T(\rho)(r - y)\right] \tag{22}$$

where C_r, $\frac{\partial C_r}{\partial \rho}$, $\frac{\partial C_y}{\partial \rho}$ can be calculated mathematically using (4) and (5). Therefore, only $T(\rho)r$ and $T(\rho)(r - y)$ are the unknown terms. These unknown terms can be obtained using the following sets of data from three different experiments.

$$\begin{aligned} {r_1}^i &= r, \\ {y_1}^i(\rho^i) &= T(\rho^i)r + S(\rho^i){d_{o1}}^i, \end{aligned} \tag{23}$$

$$\begin{aligned} {r_2}^i &= r - {y_1}^i(\rho^i) = e, \\ {y_2}^i(\rho^i) &= T(\rho^i)e + S(\rho^i){d_{o2}}^i \end{aligned} \tag{24}$$

$$\begin{aligned} {r_3}^i &= r, \\ {y_3}^i(\rho^i) &= T(\rho^i)r + S(\rho^i){d_{o3}}^i, \end{aligned} \tag{25}$$

where the subscript is the number of experiments and superscript is the number of iteration. If d_0 is zero mean, the unknown terms can be obtained by y_2 and y_3. So (22) can be obtained as:

$$\left[\frac{\partial e}{\partial \rho}(\rho)\right] = \frac{1}{C_r(\rho)}\left[(\frac{\partial C_r}{\partial \rho} - \frac{\partial C_y}{\partial \rho})y_3(\rho) + \frac{\partial C_y}{\partial \rho}y_2(\rho)\right] \tag{26}$$

2) Velocity Tracking — $e_v(\rho)$ is the error of the inner control loop and can be expressed as follows:

$$e_v(\rho) = T(\rho)r_v + S(\rho)d_{ov} - y_{dv} \tag{27}$$

$\frac{\partial e_v}{\partial \rho}$ can be obtained in the same way as described in 3.2.1.

$$\begin{aligned}\frac{\partial e_v}{\partial \rho}(\rho) &= \frac{1}{C_r(\rho)}\left[(\frac{\partial C_r}{\partial \rho} - \frac{\partial C_y}{\partial \rho})T(\rho)r_v + \frac{\partial C_y}{\partial \rho}T(\rho)(r_v - \dot{y})\right] \\ &= \frac{s}{C_r(\rho)}\left[(\frac{\partial C_r}{\partial \rho} - \frac{\partial C_y}{\partial \rho})T(\rho)r + \frac{\partial C_y}{\partial \rho}T(\rho)(r - y)\right]\end{aligned} \tag{28}$$

Since $r_v = r \cdot s$ and $r_v - \dot{y} = (r - y) \cdot s$, (22) is multiplied by s. Thus, data from (24) and (25) can be used without further experimentation. Therefore, (28) can be rewritten as:

$$\left[\frac{\partial e_v}{\partial \rho}(\rho)\right] = \frac{s}{C_r(\rho)}\left[(\frac{\partial C_r}{\partial \rho} - \frac{\partial C_y}{\partial \rho})y_3(\rho) + \frac{\partial C_y}{\partial \rho}y_2(\rho)\right] \tag{29}$$

3) Efficiency of Control Effort — The closed-loop signal of the control effort is as follows:

$$u(\rho) = S(\rho)[C_r(\rho)r - C_y(\rho)d_o] \tag{30}$$

$\frac{\partial u}{\partial \rho}$ in (11) is calculated as follows:

$$\begin{aligned}\frac{\partial u}{\partial \rho}(\rho) &= S(\rho)\left[(\frac{\partial C_r}{\partial \rho} - \frac{\partial C_y}{\partial \rho})r + \frac{\partial C_y}{\partial \rho}(r - y)\right] \\ &= \frac{1}{C_r(\rho)}\left[(\frac{\partial C_r}{\partial \rho} - \frac{\partial C_y}{\partial \rho})S(\rho)r + \frac{\partial C_y}{\partial \rho}S(\rho)(r - y)\right]\end{aligned} \tag{31}$$

Here, $u(t)$ is obtained by the command used in the experiment of (23)∼(25) is as follows:

$${u_1}^i = C_r(\rho^i)S(\rho^i)r - C_y(\rho^i)S(\rho^i){d_{o1}}^i \tag{32}$$

$$\begin{aligned}{u_2}^i &= C_r(\rho^i)S(\rho^i)r \\ &\quad - C_r(\rho^i)S(\rho^i){y_1}^i - C_y(\rho^i)S(\rho^i){d_{o2}}^i\end{aligned} \tag{33}$$

$${u_3}^i = C_r(\rho^i)S(\rho^i)r - C_y(\rho^i)S(\rho^i){d_{o3}}^i \tag{34}$$

(31) can be rewritten with u obtained from (32) (34).

$$\left[\frac{\partial u}{\partial \rho}(\rho)\right] = \frac{1}{C_r(\rho)}\left[(\frac{\partial C_r}{\partial \rho} - \frac{\partial C_y}{\partial \rho})u_3(\rho) + \frac{\partial C_y}{\partial \rho}u_2(\rho)\right] \tag{35}$$

4. EVALUATION OF THE PROPOSED METHOD WITH SIMULATION

4.1 Description of Dynamics of Two-mass System

In this paper, the iterative feedback tuning of the developed cascade control is applied to a two-mass system as shown in Fig 1. u and F_{ext} are control effort and external disturbance in Fig. 4, respectively, and $\dot{\theta}_m$, $\dot{\theta}_s$ and $\dot{\theta}_l$ are the angular velocity of the motor, spring and load,

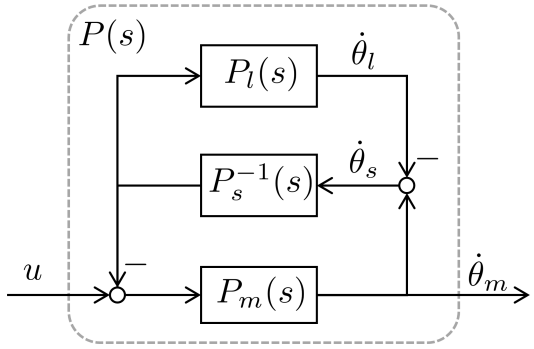

Fig. 4. Block diagram of two-mass system.

respectively. P_m, P_s and P_l are the dynamic models of the motor, spring, and load, respectively.

$$P_m(s) = \frac{1}{J_m s + B_m} = \frac{1}{2.50 \cdot 10^{-5} s + 2.84 \cdot 10^{-5}} \tag{36}$$

$$P_s(s) = \frac{s}{K_s} = \frac{s}{1.80} \tag{37}$$

$$P_l(s) = \frac{1}{J_l s + B_l} = \frac{1}{2.17 \cdot 10^{-5} s + 7.50 \cdot 10^{-5}} \tag{38}$$

As aforementioned, there is an accuracy problem in both position and velocity controllers because of the flexible link, which is described by $P_s(s)$. This flexibility of the link causes oscillations when the motor is transmitting force to the load. The effect of flexibility is also shown in the transfer function from the motor torque to the motor angular velocity:

$$P(s) = \frac{P_m(P_s + P_l)}{P_s + P_m + P_l} = \frac{s^2 + 5.53s + 76600}{2.5 \cdot 10^{-5} s^3 + 9.87 \cdot 10^{-4} s^2 + 3.56s + 6.40} \tag{39}$$

In this transfer function, the resonance is at 63 Hz, and the antiresonance is at 46Hz, which makes the tuning of the controller difficult. In the next subsection, the proposed iterative feedback tuning of the proposed cascade control is verified by simulation.

4.2 Simulation with Different Cases of Cost Functions

The purpose of the proposed cascade control is to minimize the positional and velocity errors of the system. For this purpose, the cost function of the developed iterative feedback tuning method is designed to include position error, velocity error and control effort.

The effect of the inclusion of the velocity error in the cost function is demonstrated. The initial control parameter ($\rho^1 = [0.01\ \ 0.1\ \ 40]$) and the weight of control effort ($\lambda_u^1 = 0.01$) are set to be equal and then compared according to the following case ($C1$ is case number 1, and $C2$ is case number 2).

- $C1$ —the cost function of the new iterative feedback tuning method is the same as (8) and the initial step size (μ_{C1}) is set to 17.
- $C2$ —the weight of velocity error (λ_v) is set to 0 to eliminate the velocity error in the cost function. Therefore, the cost function and cost function derivative under $C2$ are given as

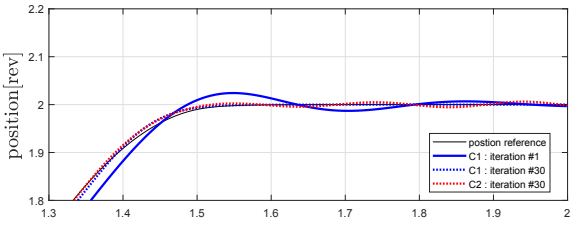

Fig. 5. Position control performance comparison according to $C1$ and $C2$.

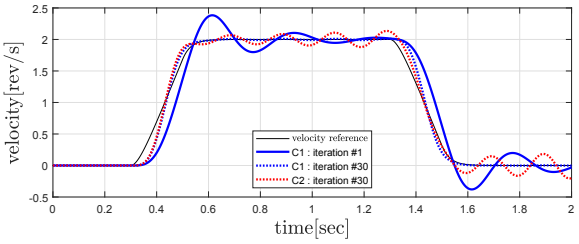

Fig. 6. Velocity control performance comparison according to $C1$ and $C2$.

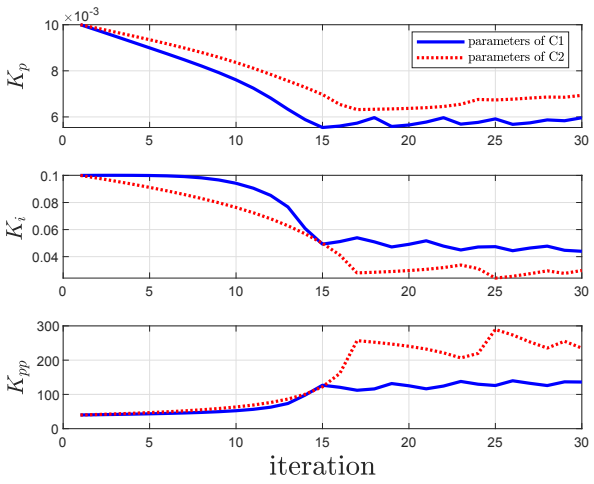

Fig. 7. Parameter convergence diagram with $C1$ and $C2$.

$$J(\rho) = \frac{1}{2N}[e^T e + \lambda_u u^T u] \tag{40}$$

$$\frac{\partial J}{\partial \rho}(\rho) = \frac{1}{N}\left[\frac{\partial e^T}{\partial \rho}(\rho) e(\rho) + \lambda_u \frac{\partial u}{\partial \rho}(\rho) u\right] \tag{41}$$

Note that μ_{C2} is set to 7.

The reason why μ is different in $C1$ and $C2$ is that the aspect of parameter convergence changes rapidly depending on the value of μ in $C2$. Although the μ of $C1$ is greater than that of $C2$, it can be seen that the control gain changes more rapidly in $C2$.

The results, which is shown in Fig. 5 and Fig. 6 show that the vibration remains in $C2$ after 30 times of iteration because the control gains in Fig. 7 were not updated in balance under $C2$. In particular, the K_{pp} increased sharply and did not settle even in the above 15 of iteration. However, the cost function of $C2$ is lower than $C1$ in Fig. 8 because it does not include velocity error.

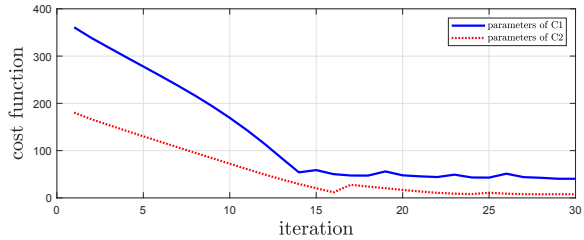

Fig. 8. Cost function reduction from iteration 1 to 30 with $C1$ and $C2$.

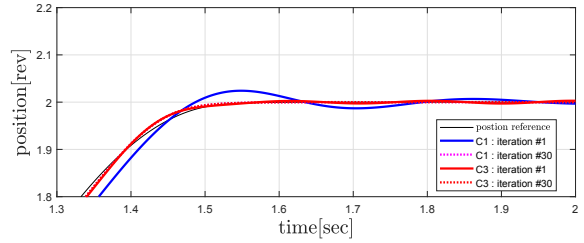

Fig. 9. Position control performance comparison according to $C1$ and $C3$.

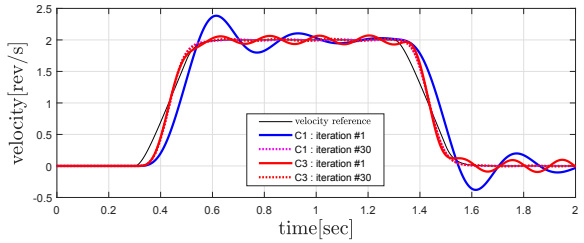

Fig. 10. Velocity control performance comparison according to $C1$ and $C3$.

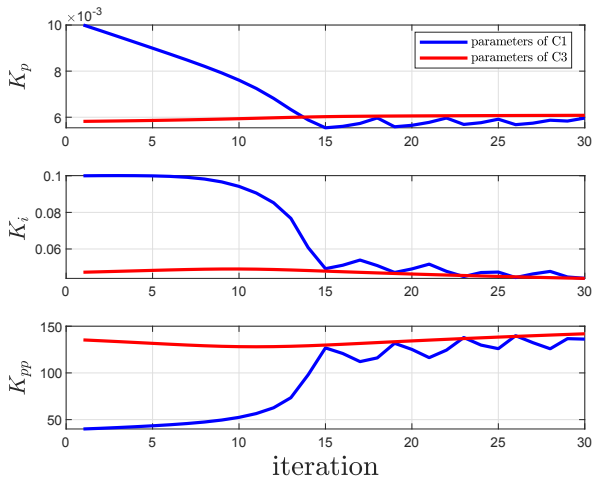

Fig. 11. Parameter convergence diagram with $C1$ and $C3$.

4.3 Simulation with Different Initial Values

Here, simulations are performed to verify that the proposed iterative feedback tuning algorithm drives the control parameter to converge to the optimal parameter. The initial control parameters in $C3$ are set to 95 % of the optimal control parameters obtained from $C1$ and then the

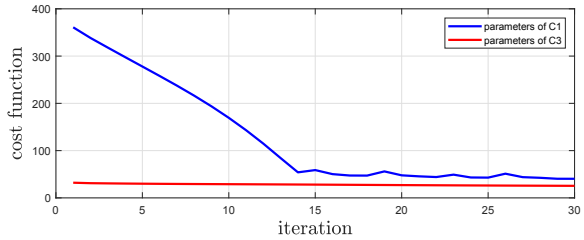

Fig. 12. Cost function reduction from iteration 1 to 50 with $C1$ and $C3$.

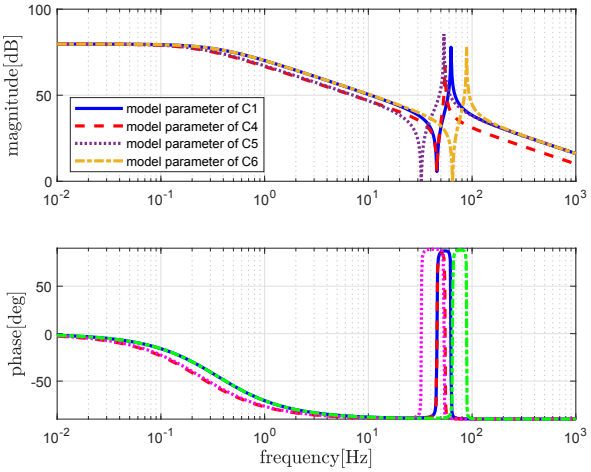

Fig. 13. Frequency characteristics of $C1$, $C4$, $C5$ and $C6$.

control parameters were seen to converge to the optimal control parameters. In addition, λ_v is activated and both case are set equal to $\lambda_u = 0.2$.

C3 —the value of 95 % of the optimal control parameter obtained from $C1$ is $\rho^1 = [0.00585\ 0.473\ 135.3]$ and step size to $\mu_{C3} = 40$.

Since the initial values of $C1$ and $C3$ are different, the results of Fig. 9 and Fig. 10 show that the initial data of $C1$ has errors and vibrations. But after 30 times in both $C1$ and $C3$, Fig. 9 and Fig. 10 show similar performance, because the control parameters converged to the optimal point in both $C1$ and $C3$ as shown in Fig. 11. Since the initial control parameter of $C1$ is set to a different value from the optimal parameter, these caused a drastic change in the cost function as shown in Fig. 12, However, control parameters in $C3$ did not change significantly, so the cost function of $C3$ was modest.

4.4 Simulation with Different Model Parameter

Here, the proposed IFT is verified to be applicable to various system models. Because iterative feedback tuning algorithm is a model-free method, it should be applicable regardless of any model change. For this, the two-mass system model was changed according to $C1$ and additional cases below. Fig. 13 is the frequency characteristic according to the model change of the two-mass system.

C4 —$J_m{}^* = 2 \cdot J_m$ in (39) reduces the resonance frequency and DC gain.

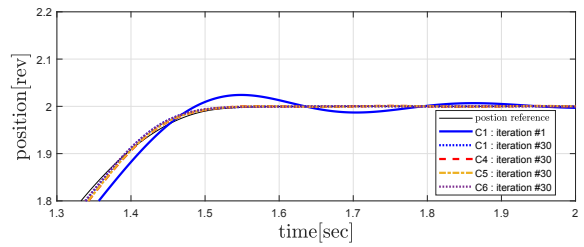

Fig. 14. Position control performance comparison according to $C1$, $C4$, $C5$ and $C6$.

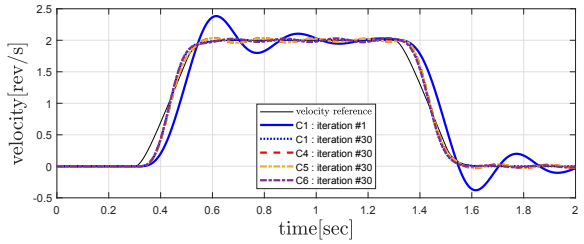

Fig. 15. Velocity control performance comparison according to $C1$, $C4$, $C5$ and $C6$.

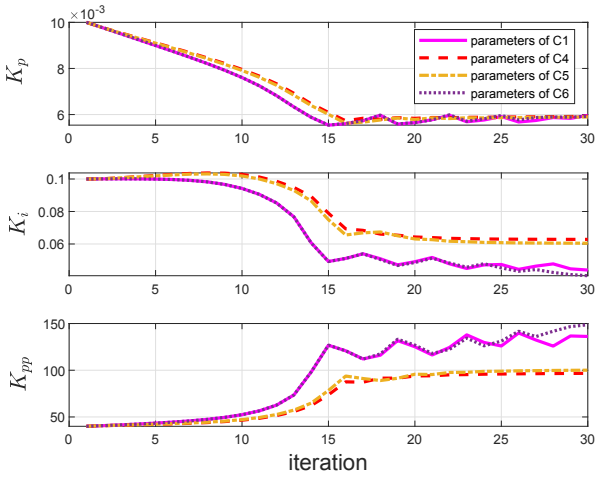

Fig. 16. Parameter convergence diagram with $C1$, $C4$, $C5$ and $C6$.

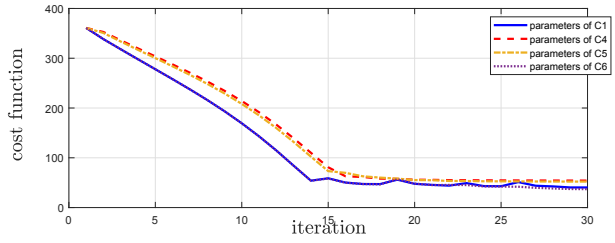

Fig. 17. Cost function reduction from iteration 1 to 50 with $C1$, $C4$, $C5$ and $C6$.

C5 —$J_l{}^* = 2 \cdot J_l$ in (39) reduces the resonance and antiresonant frequencies and also reduces DC gain.

C6 —$K_s{}^* = 2 \cdot K_s$ in (39) raises the resonance and antiresonance frequencies and reduces the DC gain.

The increase of the errors in $C4$ and $C5$ can be seen in Fig. 14 and Fig. 15. Therefore, it can be seen that there are more constraints on the controller design of the two-mass system with a low resonance point. On the other hand, $C6$ decreased DC gain and increased resonance antiresonance, but $C6$ seems to have more controller design margin than $C4$ and $C5$

Table 1. Optimal parameter after 30 iteration

Condition	K_p	K_i	K_{pp}
$C1$	0.005964	0.04396	136.2
$C2$	0.006936	0.02978	235.2
$C3$	0.006083	0.04393	141.9
$C4$	0.005921	0.06283	096.7
$C5$	0.005895	0.06047	100.1
$C6$	0.005924	0.04047	148.7

Model changes of J_m and J_l produce optimal parameter changes as shown in Fig. 16 and Fig. 17 . Notice that the cost function increases at a limited iteration.

5. CONCLUSION

This paper has proposed a novel iterative feedback tuning method of cascade control. Since the cascade control aims to control both position and velocity simultaneously, the cost function of the proposed iterative feedback tuning method includes the position and velocity errors. By analyzing the frequency characteristics of the system, the resonance and antiresonance frequencies has been considered in the IFT. In the verification using three different kinds of simulations with different conditions, the proposed method optimally tunes the position and velocity controllers of cascade controls applied to two-mass systems.

REFERENCES

Hillsley, K.L. and Yurkovich, S. (1993). Vibration control of a two-link flexible robot arm. *Dynamics and Control*, 3(3), 261–280.

Hjalmarsson, H., Gevers, M., Gunnarsson, S., and Lequin, O. (1998). Iterative feedback tuning: theory and applications. *IEEE control systems magazine*, 18(4), 26–41.

Hu, J.S., Hu, F.R., and Kang, C.H. (2014). On the two-inertia system: Analysis of the asymptotic behaviors to multiple feedback position control. *Asian Journal of Control*, 16(1), 175–187.

Huang, H.P., Jeng, J.C., and Luo, K.Y. (2005). Auto-tune system using single-run relay feedback test and model-based controller design. *Journal of Process Control*, 15(6), 713–727.

Kissling, S., Blanc, P., Myszkorowski, P., and Vaclavik, I. (2009). Application of iterative feedback tuning (ift) to speed and position control of a servo drive. *Control Engineering Practice*, 17(7), 834–840.

Pintelon, R., Guillaume, P., Rolain, Y., Schoukens, J., and Van Hamme, H. (1994). Parametric identification of transfer functions in the frequency domain-a survey. *IEEE transactions on automatic control*, 39(11), 2245–2260.

Robbins, H. and Monro, S. (1951). A stochastic approximation method. *The annals of mathematical statistics*, 400–407.

Tesch, D., Eckhard, D., and Bazanella, A.S. (2016). Iterative feedback tuning for cascade systems. In *2016 European Control Conference (ECC)*, 495–500. IEEE.