IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 29 May 2022, accepted 26 June 2022, date of publication 5 July 2022, date of current version 15 July 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3188713

RESEARCH ARTICLE

Non-Iterative Data-Driven Tuning of Model-Free Control Based on an Ultra-Local Model

SHUICHI YAHAGI[1] **AND ITSURO KAJIWARA**[2]
[1]6th Research Department, ISUZU Advanced Engineering Center Ltd., Fujisawa-shi, Kanagawa 252-0881, Japan
[2]Division of Mechanical and Aerospace Engineering, Hokkaido University, Sapporo, Hokkaido 060-8628, Japan

Corresponding author: Shuichi Yahagi (yahagi@iaec.isuzu.co.jp)

ABSTRACT In this paper, we present a data-driven tuning method for model-free control based on an ultra-local model (MFC-ULM), which is also called intelligent proportional-integral-derivative control. In industries, the control design must be easy, and it is important that the control law can be applied to nonlinear systems. The MFC-ULM has most of these features. However, in practice, trial-and-error tuning of MFC-ULM design parameters is necessary. To address this problem, we adopt a data-driven tuning approach. In the proposed method, the MFC-ULM design parameters can be tuned from single-experiment data without requiring system identification, and optimal parameters for the MFC-ULM are obtained using the least-squares method. Additionally, we adopt L_2-norm regularization to avoid overlearning. The effectiveness of this method was examined using simulations of two nonlinear systems. The results revealed that the MFC-ULM design parameters can be obtained directly without knowing the characteristics of the controlled object.

INDEX TERMS Data-driven control, model-free control, parameter tuning, PID control.

I. INTRODUCTION

In industrial systems, more than 90% of closed-loop control uses proportional-integral-derivative (PID) control [1] due to its simple structure, easy implementation, and good robustness [2]. Although the desired control performance can be obtained for highly linear controlled objects, it is difficult to achieve sufficient control performance with a fixed PID controller for nonlinear systems. Nonlinear control theory and model-based control can be applied to nonlinear systems, but the hurdles to nonlinear control applications are high due to limited controller performance, complexity of the theory, and large computational load. Additionally, industrial systems are complex, and it is often difficult to obtain accurate mathematical models, so model-based control may not be fully effective.

Recently, control system design methods that do not use system identification for a model of the system to be controlled have attracted attention [3]. There are many studies [4], including virtual reference feedback tuning (VRFT) [5]–[7], fictitious reference iterative tuning (FRIT) [8], [9], model-free adaptive control based on dynamic-linearization techniques (MFAC-DLT) [10], model-free control based on an ultra-local model (MFC-ULM) [11], [12], active disturbance rejection control (ADRC) [13], and adaptive fuzzy control (AFC) [14]. MFAC-DLT, MFC-ULM, and AFC can be applied to nonlinear systems and have been widely studied [10]–[20]. Additionally, control system design methods that do not use system models to be controlled are being applied to industrial systems, such as process systems and automotive systems [21]–[27].

The present paper focuses on the MFC-ULM proposed by Fliess and Join [11], [12], which is also called intelligent PID (iPID) control [11] and which is considered to be intuitively understandable. This is crucial for industries. Although the term ''model-free control'' generally means control techniques that do not explicitly use a model of the system to be controlled, in this paper, the ''model-free control based on an ultra-local model'' proposed by Fliess is referred to as ''MFC-ULM'' or simply ''MFC'' as in previous works [11], [28], [29].

As mentioned above, key features of MFC include its ease of understanding and its applicability to nonlinear systems. However, its design parameters, including α and PID gains, must be tuned in the field by trial and error. Reference [25] states that the tuning of an intelligent proportional (iP) controller is simpler than that of a conventional PID controller, and tuning guidelines have been proposed for an intelligent

The associate editor coordinating the review of this manuscript and approving it for publication was Azwirman Gusrialdi.

proportional (iP) controller. However, fine-tuning is necessary for the final stage, and there is no mention of the fine-tuning method or any tuning guidelines for iPID control. That is, trial-and-error tuning is occurring. To address these issues, a method has been proposed to tune the MFC design parameters directly using a data-driven control approach [28], [29]. Although the proposed method is useful, it does not treat all of the design parameters (α and PID gains). The tunable parameters of the MFC are only the PID gains, assuming that α, which is significantly related to the performance of the MFC, is known. Additionally, the cost function is not convex with respect to the tuning parameters, and nonlinear optimization is used to obtain the design parameters. The optimization of nonlinear cost functions requires a large amount of computation time, may lead to local solutions, and requires setting the hyperparameters of the optimization solver. For practical use, it is desirable to obtain design parameters easily and quickly. In reference [30], the cost function is made convex by setting α as a known parameter and adopting integral-proportional derivative (I-PD) control; however, for PID controllers, the cost function does not become convex. Thus, the same problem occurs as described above. Additionally, a part of the MFC is unused, and the full MFC is untargeted.

In this paper, we propose a method to tune all the MFC design parameters directly from the input/output data of the controlled object without a system model to be controlled. The proposed method is based on the VRFT approach, which uses a set of experimental data to find the optimal MFC design parameters. In other words, there is no need for repetitive hand tuning, such as repeated experiments. First, the cost function for VRFT-based MFC, which realizes direct tuning of the MFC design parameters, is derived as a convex function. This allows the optimal parameters to be obtained in a short time without the need for nonlinear optimization, unlike the previous literature [28], [29]. Additionally, L_2-norm regularization is introduced to prevent overlearning. L_2-norm regularization has a hyperparameter setting, which is determined by introducing cross-validation. In the proposed method, the hyperparameters do not need to be set, thereby allowing for simple automatic tuning. The contributions of this paper are summarized below:

- We propose a direct-tuning method for all MFC design parameters in the framework of data-driven control. Conventional methods treat only some of the MFC design parameters as tunable parameters.
- The proposed cost function is convex, and the optimal solution can be obtained quickly. In the conventional method, the cost function is not convex, and nonlinear optimization is required to obtain the optimal parameters.
- L_2-norm regularization is introduced to prevent overtraining, and hyperparameters are determined automatically. This has not been examined in previous studies.

The remainder of this paper is organized as follows. In Section 2, we provide an overview of the MFC and describe the problem setting. In Section 3, we propose a method for directly tuning the MFC design parameters based on a data-driven approach. In Section 4, we demonstrate the effectiveness of the proposed method using simulations. In Section 5, we provide a summary of this paper.

II. PRELIMINARY

First, we give an overview of MFC in discrete time. Then, we explain the problem setting.

A. MFC OVERVIEW

1) ULTRA-LOCAL MODEL

The controlled object is a nonlinear single-input, single-output system that can be expressed as

$$y(t) = f(y(t-1), \ldots, y(t-m), u(t-1), \ldots, u(t-l)), \quad (1)$$

where t represents discrete time, $f()$ is an unknown nonlinear function, $u \in R$ is the control input, $y \in R$ is the output, and l and m are the unknown orders of the input and output, respectively. The ultra-local model proposed in MFC [11], [12] is expressed as

$$p^n y(t) = \alpha u(t) + F(t), \quad (2)$$

where $n \geq 1$ is the order, $\alpha \in R$ is a design parameter which is not a physical parameter, and $F \in R$ represents unmodeled dynamics and disturbance. p is the differential operator which is defined as $p(q) = (1 - q^{-1})/T_s$. q^{-1} is the backward shift operator and T_s is the sampling period. Here, we assumed $n = 1$ based on prior literature [12], [25], [28], [29].

2) MFC CONTROL BASED ON AN ULTRA-LOCAL MODEL

Fig. 1 shows the MFC control law. y_r is the target value, ε is the deviation, and P is the controlled object. Based on (2), the MFC control law is expressed as

$$u(t) = \alpha^{-1}\left(-\hat{F}(t) + py_r(t)\right) + C(\varepsilon(t)) \quad (3)$$

with

$$\hat{F}(t) = py(t) - \alpha u(t-1), \quad (4)$$
$$\varepsilon(t) = y_r(t) - y(t), \quad (5)$$

where $\hat{F}(t)$ is the estimated value of F. Additionally, y_r is a differentiable signal, which is generated by passing the original target value through a low-pass filter or the like, and $C(\varepsilon(t))$ is the feedback controller. Here, the PID control law is used in the form

$$C(\varepsilon(t)) = K_p\varepsilon(t) + K_i T_s \sum \varepsilon(t) + \frac{K_d}{T_s}(\varepsilon(t) - \varepsilon(t-1)), \quad (6)$$

where K_p, K_i, and K_d are the proportional, integral, and derivative gains, respectively. The PID controller can be expressed as

$$C(q) = K^T \psi(q) \quad (7)$$

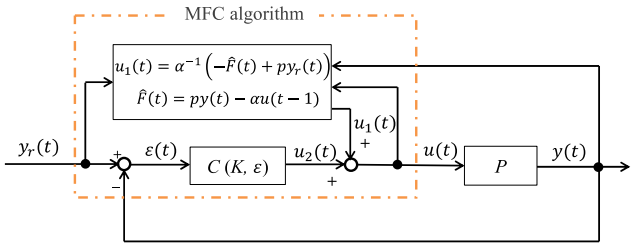

FIGURE 1. MFC algorithm based on an ultra-local model.

with

$$K = \begin{bmatrix} K_p & K_i & K_d \end{bmatrix}^T, \tag{8}$$

$$\psi(q) = \begin{bmatrix} 1 & \frac{1}{p(q)} & p(q) \end{bmatrix}^T. \tag{9}$$

When the feedback controller used in (3) is a PID controller, the MFC is also called iPID control, and similarly, when it is a P (PI) controller, it is also called iP (iPI) control. In the estimation of F, since noise removal is necessary for real systems, the filter W_F is added to (4) as follows

$$\hat{F}(t) = W_F(q)(py(t) - \alpha u(t-1)). \tag{10}$$

From the above, the control input is calculated by the following equation:

$$\begin{aligned} u(t) &= \alpha^{-1}(-W_F(q)(py(t) - \alpha u(t-1)) \\ &\quad + py_r(t)) + C(\varepsilon(t)) \\ &= W_F(q)u(t-1) + \alpha^{-1}(py_r(t) - W_F(q)py(t)) \\ &\quad + K^T\psi(q)\varepsilon(t). \end{aligned} \tag{11}$$

Remark 1: From (3) and the block diagram shown in Fig. 1, we can interpret $-\alpha^{-1}\hat{F}(t)$, $\alpha^{-1}py_r(t)$, and $C(\varepsilon(t))$ as corresponding to the disturbance observer, feedforward control, and feedback control, respectively. That is, we can consider that MFC is an extended form of the classical PID controller [20].

3) STABILITY ANALYSIS

Here, we describe the stability analysis [25]. Substituting the control law shown in (3) into the ultra-local model shown in (2), we obtain

$$py(t) = \hat{F}(t) + py_r(t) + \alpha C(\varepsilon(t)) + F(t). \tag{12}$$

The error between the real and the estimated value of $F(t)$ is given by

$$F_{dlt}(t) = F(t) - \hat{F}(t). \tag{13}$$

Then, the error equation becomes

$$\begin{aligned} p\varepsilon(t) &= -\alpha C(\varepsilon(t)) - F_{dlt}(t) \\ &= -\alpha\left(K_p + K_i\frac{1}{p} + K_d p\right)\varepsilon(t) - F_{dlt}(t). \end{aligned} \tag{14}$$

Let us assume that the error estimate is bounded. In other words,

$$\|F_{dlt}(t)\|_\infty < M. \tag{15}$$

The system is stable when the roots of the characteristic equation in z-domain given by

$$c_0 + c_1 z^{-1} + c_2 z^{-2} = 0 \tag{16}$$

with

$$\begin{aligned} c_0 &= 1 + \alpha(K_p T_s + K_i T_s^2 + K_d) \\ c_1 &= -2 - \alpha K_p T_s - 2\alpha K_d \\ c_2 &= 1 + \alpha K_d \end{aligned}$$

are inside the unit circle.

Remark 2: System stability is not guaranteed if (15) is not satisfied, even when the characteristic equation is a Hurwitz equation. Since the system is implemented in a digital controller, it is not possible to set the sampling period to zero; thus, an estimation delay must occur. If the control input fluctuates significantly over a large sampling period, the error in the estimate of F increases, which may cause (15) no longer to be satisfied.

B. PROBLEM SETTING

In MFC, design parameters need to be tuned through repeated experiments. Fig. 2 shows a block diagram of model-referenced control with MFC. We consider the direct tuning of the MFC design parameters w such that the transfer characteristic from the target value (setpoint) y_r to the output y matches the reference model M_d determined by the designer. The cost function is given by

$$J_{MR} = \frac{1}{N}\sum_{t=1}^{N}(y(t, w) - M_d y_r(t))^2, \tag{17}$$

where $y(t, w)$ is the closed-loop response and N is the data length. The tuning parameters w are the MFC design parameters (α and PID gain). If the model to be controlled is known, we can obtain the optimized parameters from the above nonlinear cost function; however, identification of the plant model may be difficult for industrial systems. Thus, in this paper, we consider a direct data-driven approach, which does not need the model to be controlled.

III. PROPOSED METHOD

Here, we develop a method for tuning the MFC design parameters directly using the VRFT approach, which is a data-driven control method.

A. VRFT

VRFT [5]–[7] is a method for tuning directly the control parameters of a controller with a pre-specified structure from input/output data in an open-loop system without requiring system identification. Furthermore, it is a model-referenced data-driven control that automatically tunes the

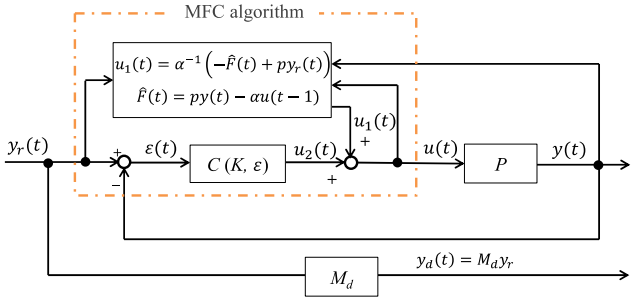

FIGURE 2. Model-referenced control by an MFC algorithm based on an ultra-local model.

control parameter w so that the closed-loop transfer characteristic from the target value (setpoint) y_r to the output y matches the reference model M_d determined by the designer. Fig. 3 shows the structure of VRFT, where C, M_d, and P are the controller, reference model, and controlled object (plant), respectively; $u(t)$ and $y(t)$ are the inputs and outputs, respectively; ρ is the controller parameter; and $y_{rv}(t)$ and $u_v(t)$ are the virtual reference signals and virtual control input, respectively. The VRFT procedure is briefly described below:

[Step 1] The reference model M_d is set. The input and output data for the plant $D = \{u(t), y(t) | t = 1, \ldots, N\}$ are acquired in a test.

[Step 2] If $y(t)$ is regarded as the output of the reference model, the reference signal that generates $y(t)$ is determined as

$$y_{rv}(t) = M_d^{-1} y(t). \tag{18}$$

[Step 3] This signal is considered the reference input for the closed-loop system shown in Fig. 2. In this case, the virtual control input is given by

$$u_v(t) = C(\rho, q)\left(y_{rv}(t) - y(t)\right). \tag{19}$$

[Step 4] If the data for this virtual control input and the actual control input are close, the closed-loop system can be considered close to the reference model. That is, the cost function to be minimized is

$$J_{VR}(\rho) = \frac{1}{N}\sum_{t=1}^{N}\left(u(t) - u_v(t)\right)^2. \tag{20}$$

Substituting the virtual control input into (20) yields

$$J_{VR}(\rho) = \frac{1}{N}\sum_{t=1}^{N}\left(u(t) - C(\rho, q)\,\varepsilon_v(t)\right)^2 \tag{21}$$

with

$$\varepsilon_v(t) = y_{rv}(t) - y(t). \tag{22}$$

[Step 5] A prefilter L is introduced.

The terms in (21) may include a nonproper factor because of the inverse of the reference model. The addition of a prefilter helps to avoid the nonproper property and to bring

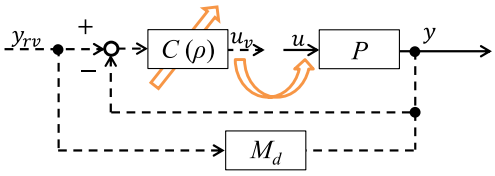

FIGURE 3. The concept of VRFT.

the VRFT cost function (21) close to the model reference cost function (17) [6], [7]. Adding the prefilter to (21) yields the following equation:

$$J_{VR}(\rho) = \frac{1}{N}\sum_{t=1}^{N}\left(u_L(t) - C(\rho, q)\,e_L(t)\right)^2 \tag{23}$$

with

$$u_L(t) = L(q)u(t), \quad e_L(t) = L(q)\varepsilon_v(t). \tag{24}$$

B. DERIVATION OF THE COST FUNCTION

The virtual control input for MFC can be obtained from (11) as follows:

$$\begin{aligned} u_v(t) &= \left(1 - W_F(q)\,q^{-1}\right)^{-1}\Big\{\alpha^{-1}\left(py_{rv}(t) - W_F(q)\,py_0(t)\right) \\ &\qquad + K^T\psi(q)\,\varepsilon_v(t)\Big\} \\ &= \left(1 - W_F(q)\,q^{-1}\right)^{-1} \\ &\quad \times \begin{bmatrix}\alpha^{-1} & K^T\end{bmatrix}\begin{bmatrix} p\left(M_d^{-1}(q) - W_F(q)\right) \\ \psi(q)\left(M_d^{-1}(q) - I\right)\end{bmatrix} y_0(t). \end{aligned} \tag{25}$$

Using the obtained virtual control input, the VRFT cost function for the MFC can be obtained as

$$J_{VR}(w) = \frac{1}{N}\sum_{t=1}^{N}\left(d(t) - w^T\xi(t)\right)^2 \tag{26}$$

with

$$d(t) = L(q)u_0(t) \tag{27}$$

$$w = \begin{bmatrix}\alpha^{-1} & K^T\end{bmatrix}^T \tag{28}$$

$$\begin{aligned} \xi(t) &= L(q)\left(1 - W_F(q)\,q^{-1}\right)^{-1} \\ &\quad \times \begin{bmatrix} p\left(M_d^{-1}(q) - W_F(q)\right) \\ \psi(q)\left(M_d^{-1}(q) - I\right)\end{bmatrix} y_0(t). \end{aligned} \tag{29}$$

Since the cost function is convex with respect to the control parameter vector w, the least-squares (LS) method yields the optimal solution:

$$w^* = \left(Z^T Z\right)^{-1} Z^T D \tag{30}$$

with

$$Z = \begin{bmatrix}\xi(1) & \xi(2) & \cdots & \xi(N)\end{bmatrix}^T \tag{31}$$

$$D = \begin{bmatrix}d(1) & d(2) & \cdots & d(N)\end{bmatrix}^T. \tag{32}$$

C. DIRECT PARAMETER TUNING BY L_2-NORM REGULARIZATION

L_2-norm regularization is a technique that suppresses both overtraining and the sizes of parameters. In the field of machine learning, it is employed to reduce prediction errors while preventing overfitting. Here, L_2-norm regularization is introduced to find the optimal solution of (26). This optimal solution is expected to prevent overfitting of the weight coefficients, thereby suppressing excessively large control inputs and system instability due to the overfitting. L_2-norm regularization, which is the cost function obtained by adding the L_2 norm term to (26), is as follows:

$$J(w) = \left\| D - Z^T w \right\|_2^2 + \Lambda \|w\|_2^2, \tag{33}$$

$$\Lambda = diag\left(\frac{1}{\lambda}, \lambda, \lambda, \lambda\right), \tag{34}$$

where $diag()$ denotes the diagonal matrix. λ, a parameter that tunes the relative strength between the regularization term and the term of the sum of squares of the errors, is a positive constant. By changing the value of λ, it is possible to tune the degree of suppression of overlearning. The L_2-norm regularization (33) yields the analytical solution:

$$w^* = \left(Z^T Z + \Lambda\right)^{-1} Z^T D. \tag{35}$$

In this paper, cross-validation [31] is used to determine the optimal regularization parameter λ. Fig. 4 briefly explains the steps in cross-validation.

1. Divide the data into k blocks. This division is called a fold.

2. Fold 1 is the test set, and the remaining folds 2–5 are the training set. The model is trained and evaluated, respectively, using those sets.

3. The model is trained using fold 2 as the test set, and is evaluated using the remaining folds 1 and 3–5 as the training set.

4. This process is repeated for test folds 3, 4, and 5.

The average of the obtained J_k is used as the evaluation value J of the model.

Cross-validation is performed for several λ set by the designer, and the evaluated value J for each λ is used to obtain the optimal λ and optimal parameters with the lowest evaluated value.

Remark 3: The reason for taking the reciprocal of λ in (34) is explained. From the MFC control law, a larger value of α results in smaller control input. Since the value directly tuned by VRFT is α^{-1}, it is necessary to decrease the magnitude of α^{-1} to increase the suppression of overlearning. Therefore, the reciprocal of λ is taken in the part concerning α^{-1}.

Remark 4: L_2-norm regularization was introduced (33) to suppress overlearning. L_1-norm regularization can also be applied to obtain a sparse controller, as in the literature [27]. However, since there are only four MFC design parameters, we employ L_2-norm regularization, with which we can obtain an analytical solution.

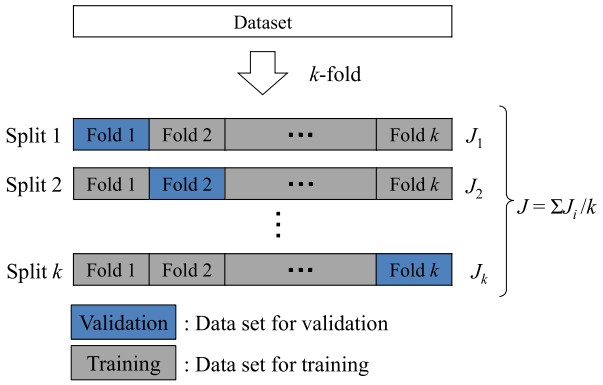

FIGURE 4. k-fold cross-validation.

D. ALGORITHM

The algorithm for direct-tuning method of the MFC design parameters is shown below:

[Step 1] Obtain the input/output data in a test.

[Step 2] Set the reference model and a prefilter. The prefilter in Step 2 is used as follows: [32, 33]

$$L(q) = M_d(q). \tag{36}$$

[Step 3] After calculating Z and D shown in (31) and (32) using the data obtained in Step 1 and setting the model in Step 2, use (35) to find the weight coefficients (MFC design parameters) that minimize the cost function.

In implementing the controller, we use the intelligent PID controller shown in (11), with the tuned parameters.

Remark 5: In references [28], [29], a direct-tuning method for the MFC design parameters was proposed using VRFT; however, the cost function is not convex. In reference [30], FRIT has been adopted; however, this approach cannot obtain a convex cost function for PID controllers, except for a special case like the I-PD controller. Additionally, full MFC is not targeted. Hence, the following issues arise: the high computational cost, the need to set hyperparameters related to the optimization solver, and the fact that the solution is not uniquely determined. However, the proposed method is useful in practical applications because the optimal parameters can be obtained by convex analysis. The optimal parameters uniquely determine the solution and significantly reduce the optimization time. Furthermore, the parameter-tuning method based on L_2-norm regularization suppresses overlearning. There are no previous studies that consider the suppression of overlearning.

Remark 6: As a hand-tuning guideline for iP control, reference [25] proposed to set the P gain to 0 and then gradually decrease α to a value in which overshoot does not occur. Afterward, the P gain is gradually increased. However, the need for fine-tuning remains, and iPID (full MFC) is not targeted. Thus, automatic tuning of the MFC parameters is important. In references [28, 29], a direct-tuning method for the MFC parameters was proposed; however, α was assumed to be known, and only the PID gain was treated as a tuning

parameter. In contrast, this paper can tune design parameters, including α, which were not addressed in the prior literature.

Remark 7: The VRFT approach does not guarantee closed-loop stability. However, it remains an attractive approach when the system is complex or models are unavailable due to development costs (engineering time and required hardware) [25]. Instead, in the proposed method, L_2-norm regularization can improve the obtained solution and suppress overlearning, as shown next in Section IV. This leads to improved closed-loop stability [34].

IV. SIMULATION STUDIES

The systems to be controlled are the Hammerstein model and the linear parameter-varying (LPV) system, which are widely used as models for describing nonlinear systems. The simulation is implemented using a PC (CPU: core i5-8250U 1.6 GHz; RAM: 16GB). MATLAB® is used as the programming language. The algorithm described in Section III-D is implemented. That is, the MFC parameters are tuned from the initial data, the user-defined reference model, and the user-defined prefilter. After MFC parameter tuning, a closed-loop response test is conducted.

A. EXAMPLE 1

1) SYSTEM FORMULATION

Here, the system formulation, including the plant and reference model, is the same as in the previous literature [32, 35]. The sampling period of the simulation is set to 1 s. The Hammerstein model [36] is given as

$$\begin{aligned} y(t) &= 0.6y(t-1) - 0.1y(t-2) + 1.2x(t-1) \\ &\quad - 0.1x(t-2) + v(t) \\ x(t) &= 1.5u(t) - 1.5u^2(t) + 1.5u^3(t), \end{aligned} \tag{37}$$

where v is white noise. The setpoint values at each time are set as follows:

$$r(t) = \begin{cases} 1.0 \ (0 < t \leq 100) \\ 3.0 \ (100 < t \leq 200) \\ 0.5 \ (200 < t \leq 300) \\ 2.0 \ (300 < t \leq 400). \end{cases} \tag{38}$$

The reference model is set to

$$M_d\left(q^{-1}\right) = \frac{0.399q^{-1}}{1 - 0.736q^{-1} + 0.135q^{-2}}. \tag{39}$$

2) RESULTS

Although an open-loop test is recommended in standard VRFT, it is sometimes difficult to perform an open-loop test in a real system. Therefore, initial input/output data with open-loop and closed-loop tests are considered in this simulation. Additionally, both the noiseless (ideal) case and the noisy case are examined. In this section, an iPD controller is used for comparison with a classical PID controller, since the structure of the iPD corresponds to that of a PID controller [12]. Additionally, the proposed method is compared with a conventional method [28], [29] in which the gravitational search algorithm (GSA) [37] is used as an optimization method [28], [29]. We use the same GSA hyperparameters as in the previous literature [37], [38]. For increased accuracy, here, the number of agents and the maximum number of iterations are increased to 30 and 100, respectively.

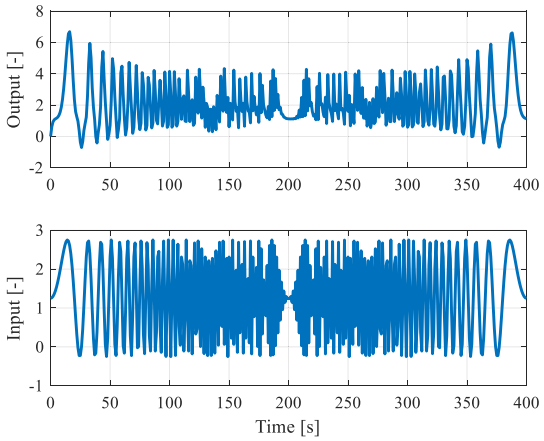

FIGURE 5. Time-series data of initial input and output under open-loop system.

[Use of open-loop test data]

First, we consider a noiseless case, which is the ideal case. Let v be white noise with variance 0. Fig. 5 shows the initial input/output data measured in the open-loop test. The input is a chirp-sine signal, and the input/output data is measured. Figs. 6 and 7 show the time-series data when the MFC (iPD controller) design parameters (α and PD gain) are obtained from the input/output data using the LS method (MFC-VRFT-LS) and L_2-norm regularization (MFC-VRFT-Regularization), respectively. For comparison, Fig. 8 shows the time-series data with a classical PID controller. The PID gains ($K_p = 0.059$, $K_i = 0.058$, and $K_d = 0.0038$) were obtained by the classically famous CHR method [35]. In Figs. 6, 7, and 8, the top and bottom images show the output and input, respectively. This confirms that the proposed method is more responsive than the CHR method. The overshoot in the case of MFC-VRFT-Regularization is less than that in the case of MFC-VRFT-LS. TABLE 1 shows the optimized parameters, cost function values, and the calculation time for optimization. The obtained λ is 10. The table includes the results of the conventional method [28, 29] for comparison. The magnitude of α in MFC-VRFT-Regularization is higher than that in MFC-VRFT-LS. Although the tracking error of the conventional method is almost same as that of MFC-VRFT-Regularization, the calculation time for optimization in MFC-VRFT-Regularization is much smaller than that of the conventional method.

Next, we examine the noisy case. Let v be white noise with variance 1×10^{-3}. As input, the same chirp-sine signal as in Fig. 5 is applied, and the input/output data are measured. Figs. 9 and 10 show the time-series data when the MFC design parameters (α and PID gain) are obtained from

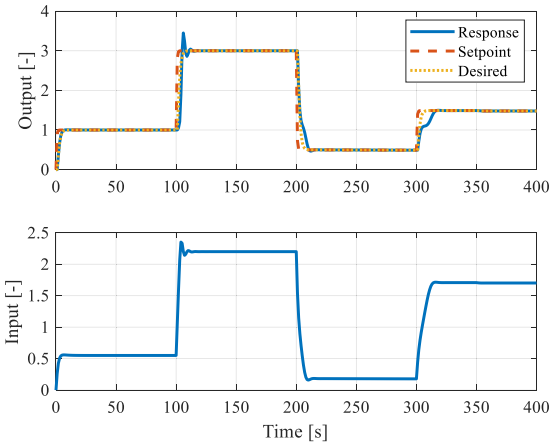

FIGURE 6. Time-series data with parameters tuned using MFC-VRFT-LS under ideal conditions with open-loop test data.

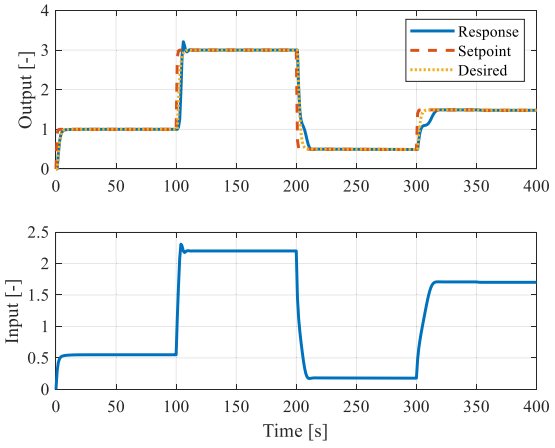

FIGURE 7. Time-series data with parameters tuned using MFC-VRFT-Regularization under ideal conditions with open-loop test data.

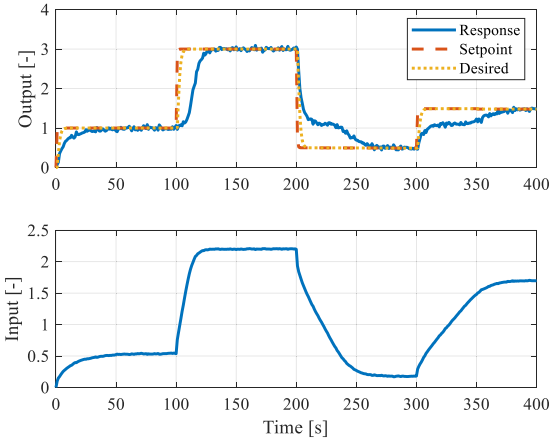

FIGURE 8. Time-series data with parameters tuned using the CHR method.

the input/output data using the LS method (MFC-VRFT-LS) and L_2-norm regularization (MFC-VRFT-Regularization), respectively. In Figs. 9 and 10, the top and bottom images

TABLE 1. Results under ideal conditions (open-loop system).

	LS	Regularization	Conventional
α	-0.0244	14.433	18.531
K_p	0.2679	0.2577	0.2678
K_d	40.963	0.0007	0.0129
J_{VR}	0.1063	0.10381	0.1008
J_{MR}	1.3211×10^{-2}	1.2288×10^{-2}	9.7838×10^{-3}
Time [s]	5.3640×10^{-4}	0.3712	3.4369×10^{3}

TABLE 2. Results under noisy conditions (open-loop system).

	LS	Regularization	Conventional
α	0.0557	14.659	26.190
K_p	0.2679	0.2579	0.2679
K_d	-17.867	0.0007	0.0284
J_{VR}	1.601×10^{-2}	8.5248×10^{-2}	0.1015
J_{MR}	1.7073×10^{-2}	1.8201×10^{-2}	1.1392×10^{-2}
Time [s]	5.6020×10^{-4}	0.5611	2.9721×10^{3}

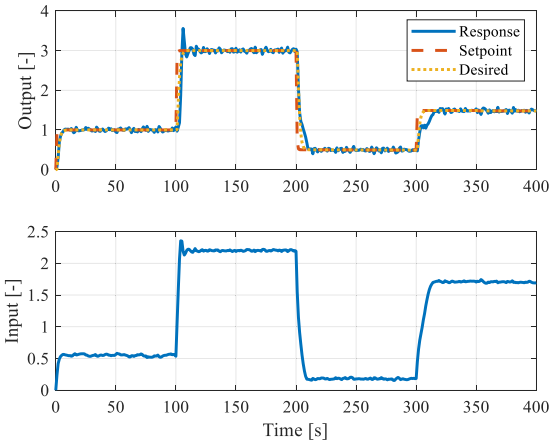

FIGURE 9. Time-series data with parameters tuned using MFC-VRFT-LS under noisy conditions with open-loop test data.

show the output and input, respectively. Additionally, these figures confirm that the proposed method has higher responsiveness than the CHR method. The overshoot in the case of MFC-VRFT-Regularization is smaller than that in the case of MFC-VRFT-LS. This difference in overshoot occurs because the fluctuations of the control input became smaller in the case of MFC-VRFT-Regularization because of suppressed overlearning. TABLE 2 shows the optimized parameters and cost-function values. The obtained λ is 1. When L_2-norm regularization is used, the magnitude of α is large compared to the result obtained when the LS method is used. This implies that overfitting was suppressed and that more stable MFC parameters were obtained. The table includes the results of the conventional method [28, 29] for comparison. The magnitude of α in MFC-VRFT-Regularization is higher than that in MFC-VRFT-LS. Although the tracking error of

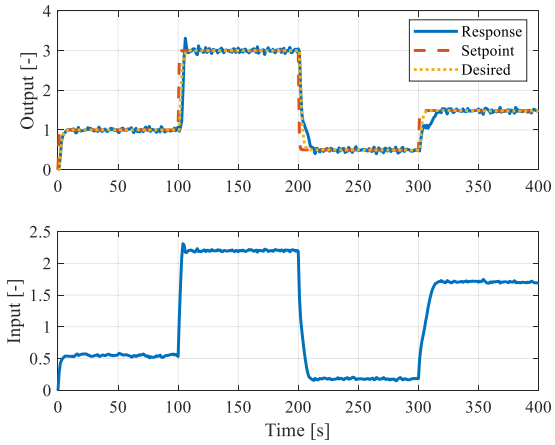

FIGURE 10. Time-series data with parameters tuned using MFC-VRFT-Regularization under noisy conditions with open-loop test data.

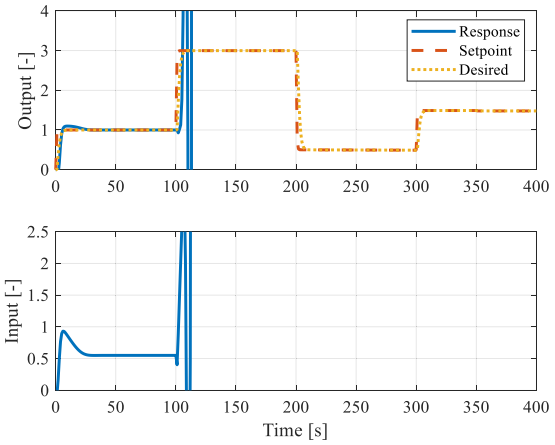

FIGURE 12. Time-series data with parameters tuned using MFC-VRFT-LS under ideal conditions with closed-loop test data.

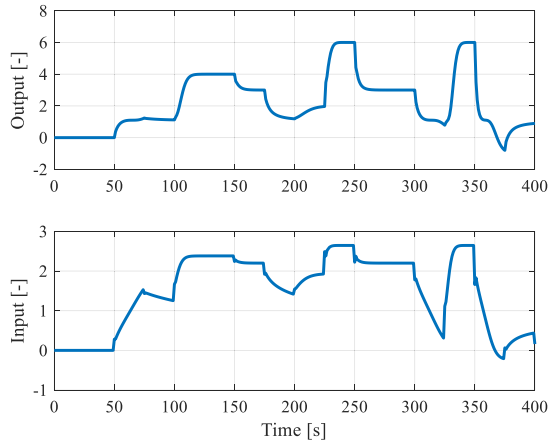

FIGURE 11. Time-series data of the initial input and output in a closed-loop system.

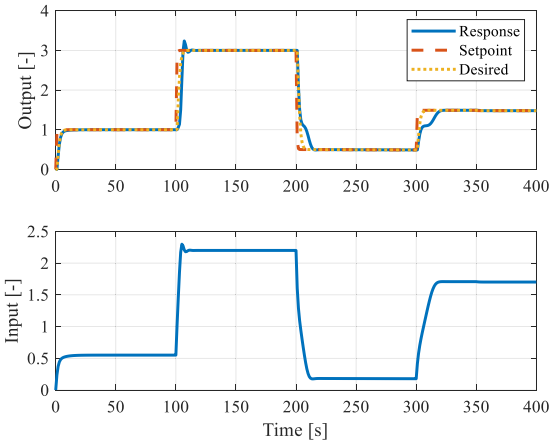

FIGURE 13. Time-series data with parameters tuned using MFC-VRFT-Regularization under ideal conditions with closed-loop test data.

the conventional method is almost same as that of MFC-VRFT-Regularization, the calculation time for optimization in MFC-VRFT-Regularization is much smaller than that of the conventional method. These tendencies are equivalent to those in the ideal case (without noise).

[Use of closed-loop test data]

First, we will consider an ideal situation, i.e., without noise. Let v be white noise with variance 0. Fig. 11 shows the initial input/output data measured in the closed-loop test. For the set value, a rectangular-wave random signal (minimum value -1, maximum value 5) is applied, and the input/output data are measured. Figs. 12 and 13 show the time-series data when the MFC design parameters (α and PID gain) are obtained from the input/output data using the LS method (MFC-VRFT-LS) and L_2-norm regularization (MFC-VRFT-Regularization), respectively. In these figures, the top and bottom images show the output and input, respectively. When MFC-VRFT-LS is used, the system is divergent. This situation is thought to occur because of overfitting. In contrast, when MFC-VRFT-Regularization is used, it can be confirmed that the response follows the target. Additionally, higher control performance is obtained compared to the classically famous CHR method, as in the case using open-loop test data. TABLE 3 shows the optimized parameters and the cost-function values. The obtained λ is 10. When the L_2-norm regularization is used, the magnitude of α is large, and the PD gain is small compared to the result when the LS method is used. This implies that overfitting was suppressed. The calculation time for optimization in MFC-VRFT-Regularization is much smaller than that required by the conventional method. In addition, in the conventional method, the closed-loop response became unstable because the parameters fell into a local solution due to nonlinear optimization.

Next, we examine the noisy case. Let v be white noise with variance 1×10^{-3}. A rectangular-wave random signal (minimum value -1, maximum value 6) is applied, as shown in Fig. 11, and the input/output data are measured. Figs. 14 and 15 show the time-series data when the MFC design parameters (α and PID gain) are obtained from the input/output data using the LS method (MFC-VRFT-LS)

TABLE 3. Results under ideal conditions (closed-loop system).

	LS	Regularization	Conventional
α	0.00015	24.292	2.1365
K_p	0.2777	0.2203	0.6444
K_d	−6621.262	6.4322×10^{-6}	0.7705
J_{VR}	0.2625	0.36976	5.6191
J_{MR}	unstable	1.6818×10^{-2}	unstable
Time [s]	4.6830×10^{-4}	0.3372	2.5907×10^{3}

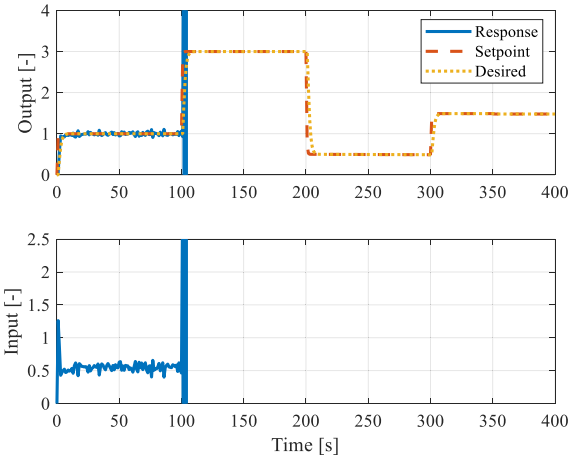

FIGURE 14. Time-series data with parameters tuned using MFC-VRFT-LS under noisy conditions with closed-loop test data.

and L_2-norm regularization (MFC-VRFT-Regularization), respectively. In these figures, the top and bottom images show the output and input, respectively. When MFC-VRFT-LS is used, the system is divergent. This situation is thought to occur because of overfitting. In contrast, when MFC-VRFT-Regularization is used, the response can follow the target. Furthermore, higher responsiveness is obtained compared to the classically famous CHR method. TABLE 4 shows the optimized parameters and cost-function values. The obtained λ is 1000. When L_2-norm regularization is used, the magnitude of α is large, and the PD gain is small compared to the result when the LS method is used. This implies that overfitting was suppressed. The calculation time for optimization in MFC-VRFT-Regularization is much smaller than that required by the conventional method. In addition, in the conventional method, the closed-loop response became unstable because the parameters fell into a local solution due to nonlinear optimization. This implies that overfitting was suppressed and that more stable MFC parameters were obtained.

B. EXAMPLE 2

Here, we provide a numerical example for a noisy case. In this section, iPID control is adopted. Further, we apply only the cost function (35) because we confirmed the effectiveness of MFC-VRFT-Regularization in the previous session. Additionally, the proposed method is compared with the conventional method [28], [29]. The hyperparameters are the same as in the previous section.

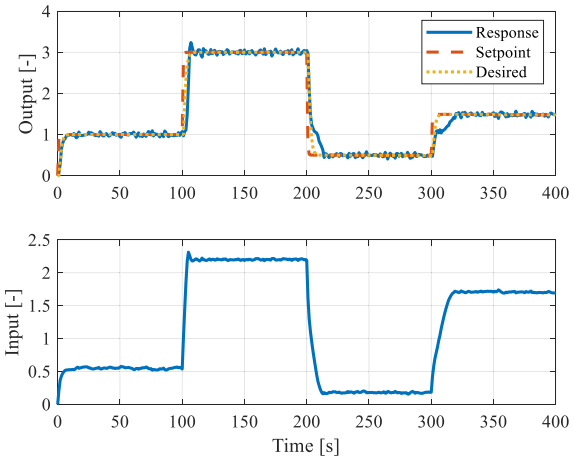

FIGURE 15. Time-series data with parameters tuned using MFC-VRFT-Regularization under noisy conditions with closed-loop test data.

TABLE 4. Results under noisy conditions (closed-loop system).

	LS	Regularization	Conventional
α	-5.412×10^{-5}	26.5967	30.0878
K_p	0.2778	0.2192	0.5071
K_d	18532.36	3.7599×10^{-6}	−2.7787
J_{VR}	0.4564	0.36931	3.2874
J_{MR}	unstable	1.7619×10^{-2}	unstable
Time [s]	4.9730×10^{-4}	0.3418	2.2835×10^{3}

1) SYSTEM FORMULATION

Here, the system formulation, including the plant and reference model, is the same as in the previous literature [32]. Fig. 16 shows the system to be controlled, which is a spring-mass LPV system. The quantities m, c, k, and y represent the mass, viscosity coefficient, spring stiffness, and system response, respectively. The system response determines the changes in the mass, spring stiffness, and viscosity coefficient. The controlled object is a system in which the following equation of motion is discretized:

$$m(y,t)\frac{d^2y(t)}{dt^2}+c(y,t)\frac{dy(t)}{dt}+k(y,t)\,y(t)=u(t)+v(t) \quad (40)$$

with

$$\begin{aligned} m(y,t) &= 1+0.2y(t) \\ k(y,t) &= 5+2y+y^2(t) \\ c(y,t) &= 2+0.5y(t), \end{aligned} \quad (41)$$

where v is white noise with variance 1×10^{-4}. The setpoint at each time is given as

$$r(t)=\begin{cases} 0.75\ (0<t\le 10) \\ 2.0\ (10<t\le 25) \\ 1.25\ (25<t\le 40) \\ 0.5\ (40<t\le 50). \end{cases} \quad (42)$$

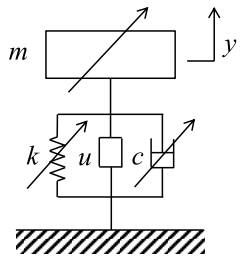

FIGURE 16. Spring-mass system with time-varying parameters.

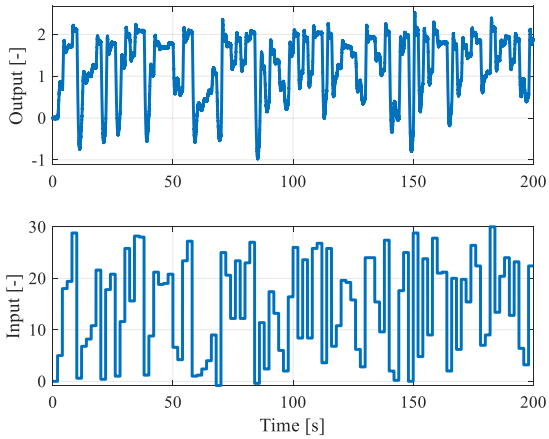

FIGURE 17. Time-series data for the initial input and output of the open-loop system.

The reference model, which has a time constant of 1 s, is set to

$$M_d(q) = \frac{0.01q}{1.01q - 1}. \tag{43}$$

The sampling period of the controller is 10 ms. A low-pass filter for F, which has time the constant 0.02 s, is given by

$$W_F(q) = \frac{0.01q}{0.03q - 0.02}. \tag{44}$$

2) RESLUTS

[Use of open-loop test data]

Fig. 17 shows the initial input/output data measured in the open-loop test. A rectangular-wave random signal (minimum value -1, maximum value 30) is applied, and the input/output data are measured. Fig. 18 shows that the time-series data when the MFC design parameters (α and PID gain) are obtained from the input/out data using L_2-norm regularization (MFC-VRFT-Regularization). The top and bottom images show the output and input, respectively. When MFC-VRFT-Regularization is used, the response follows the target. The optimized parameters and cost function are as follows: $\lambda = 1 \times 10^4$, $\alpha = 8.0973$, $K_p = 0.7185$, $K_i = -0.0001$, $K_d = 0.0024$, and $J_{VR} = 3.0958$, $J_{MR} = 3.8734 \times 10^{-3}$. Additionally, when MFC-VRFT-LS is used, the system becomes divergent. This is thought to occur because of overfitting. In the conventional method [28, 29], the system

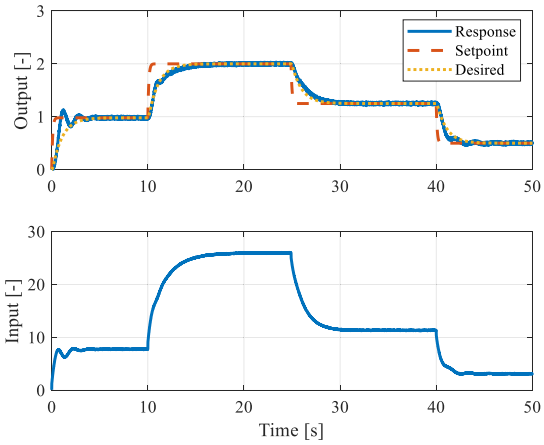

FIGURE 18. Time-series data with parameters tuned using MFC-VRFT-Regularization under noisy conditions with open-loop test data.

became unstable with the obtained parameters: $\alpha = 23.1229$, $K_p = -0.7265$, $K_i = -0.0034$, $K_d = -0.4885$, and $J_{VR} = 30.1717$. The calculation time was 3.1800×10^3 s.

[Use of closed-loop test data]

Fig. 19 shows the initial input/output data measured in the closed-loop test. A rectangular random signal (minimum value -0.5, maximum value 2.5) is applied as a set-point and the input/output data are measured. Fig. 20 shows the time-series data when the MFC design parameters (α and PID gain) are obtained from the input/output data using L_2-norm regularization (MFC-VRFT-Regularization). The top and bottom images show the output and input, respectively. When MFC-VRFT-Regularization is used, the response follows the target. The optimized parameters and cost function are as follows: $\lambda = 1 \times 10^5$, $\alpha = 11.5657$, $K_p = 0.5875$, $K_i = 0.0003$, $K_d = -0.0004$, and $J_{VR} = 4.0526$, $J_{MR} = 4.0099 \times 10^{-3}$. However, when MFC-VRFT-LS is used, the system becomes divergent. This situation is thought to occur because of overfitting. In the conventional method [28, 29], the system became unstable with the obtained parameters: $\alpha = 28.9101$, $K_p = 0.2889$, $K_i = 0.0012$, $K_d = -0.0466$, and $J_{VR} = 17.488$. The calculation time was 2.9805×10^3 s.

C. DISCUSSION

The proposed method was applied to two nonlinear systems. We confirmed that the proposed method enables the MFC design parameters to be tuned from only single-experiment data. The parameters obtained using MFC-VRFT-LS and MFC-VRFT-Regularization enabled high model-matching performance in both ideal and noisy cases. Additionally, we confirmed that the parameters tuned using MFC-VRFT-LS may make a closed-loop system unstable because of overlearning (see Figs. 12 and 14). In contrast, with MFC-VRFT-Regularization, the closed-loop system became stable. The tuned α using MFC-VRFT-Regularization is larger than that obtained using

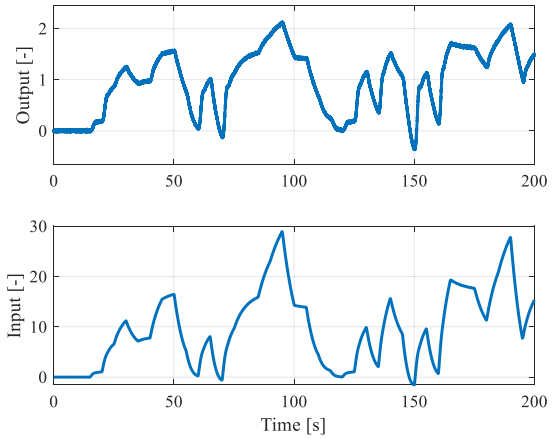

FIGURE 19. Time-series data for the initial input and output of the closed-loop system.

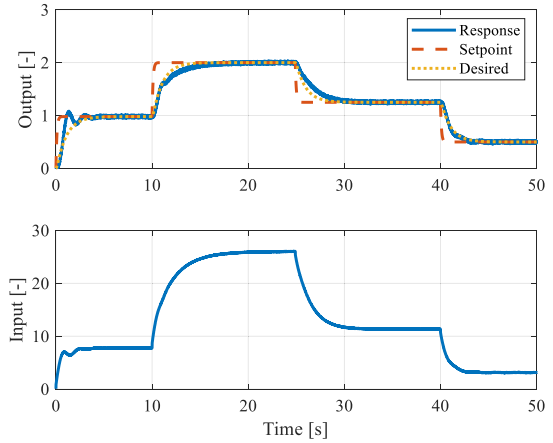

FIGURE 20. Time-series data with parameters tuned using MFC-VRFT-Regularization under noisy conditions with closed-loop test data.

MFC-VRFT-LS (see Tables 1–4). This result shows that overlearning was suppressed by introducing L_2-norm regularization. Additionally, the proposed method was compared with the conventional method. The results showed that the calculation time was approximately one hour and that the obtained parameters fell into a local solution in the conventional method. On the other hand, using the proposed method, the calculation time was less than one second, and the proposed method does not suffer from being trapped in a local minimum. Furthermore, there were some differences in the parameters optimized using the open-loop and closed-loop test data. This may be due to differences in the input/output data, which is understandable from the results [32].

V. CONCLUSION

In this paper, we have proposed a direct data-driven tuning method to obtain the MFC design parameters. This method is based on VRFT, which is a data-driven control method. That is, the MFC design parameters are obtained directly from a set of input/output data without requiring a model for the system to be controlled. In the proposed method, a cost function that is convex with respect to the tuning parameters was introduced, and the LS method can be applied. Additionally, L_2-norm regularization was introduced to suppress overlearning. The effectiveness of this method was examined using simulations of two nonlinear systems. The results showed that the MFC design parameters can be tuned from the data without trial-and-error iterative tuning. Moreover, overlearning can be avoided by introducing L_2-norm regularization, and the instability of a closed-loop system due to overlearning can be suppressed. Hence, the proposed method eliminates the need for trial-and-error parameter tuning for MFC design. In the future, applications to automotive systems will be considered.

REFERENCES

[1] K. Åström and T. Hägglund, "The future of PID control," *Control Eng. Pract.*, vol. 9, no. 11, pp. 1163–1175, 2001.
[2] H. Zhang, W. Assawinchaichote, and Y. Shi, "New PID parameter autotuning for nonlinear systems based on a modified monkey–multiagent DRL algorithm," *IEEE Access*, vol. 9, pp. 78799–78811, 2021.
[3] Z.-S. Hou and Z. Wang, "From model-based control to data-driven control: Survey, classification and perspective," *Inf. Sci.*, vol. 235, pp. 3–35, Jun. 2013.
[4] K. Prag, M. Woolway, and T. Celik, "Toward data-driven optimal control: A systematic review of the landscape," *IEEE Access*, vol. 10, pp. 32190–32212, 2022.
[5] G. O. Guardabassi and S. M. Savaresi, "Virtual reference direct design method: An off-line approach to data-based control system design," *IEEE Trans. Autom. Control*, vol. 45, no. 5, pp. 954–959, May 2000.
[6] M. C. Campi, A. Lecchini, and S. M. Savaresi, "Virtual reference feedback tuning: A direct method for the design of feedback controllers," *Automatica*, vol. 38, no. 8, pp. 1337–1346, 2002.
[7] M. C. Campi and S. M. Savaresi, "Direct nonlinear control design: The virtual reference feedback tuning (VRFT) approach," *IEEE Trans. Autom. Control*, vol. 51, no. 1, pp. 14–27, Jan. 2006.
[8] S. Soma and O. Kaneko, "A new method of controller parameter tuning based on input-output data–fictitious reference iterative tuning (FRIT)," *IFAC Proc. Volumes*, vol. 37, no. 12, pp. 789–794, 2004.
[9] O. Kaneko, "Data-driven controller tuning: FRIT approach," *IFAC Proc. Volumes*, vol. 46, no. 11, pp. 326–336, 2013.
[10] Z. Hou and S. Jin, "Data-driven model-free adaptive control for a class of MIMO nonlinear discrete-time systems," *IEEE Trans. Neural Netw.*, vol. 22, no. 12, pp. 2137–2188, Nov. 2011.
[11] M. Fliess and C. Join, "Intelligent PID controllers," in *Proc. 16th Medit. Conf. Control Autom.*, Ajaccio, France, Jun. 2008, pp. 326–331.
[12] M. Fliess and C. Join, "Model-free control," *Int. J. Control*, vol. 86, no. 12, pp. 2228–2252, 2013.
[13] R.-C. Roman, R.-E. Precup, and E. M. Petriu, "Hybrid data-driven fuzzy active disturbance rejection control for tower crane systems," *Eur. J. Control*, vol. 58, pp. 373–387, Mar. 2021.
[14] Z. Zhu, Y. Pan, Q. Zhou, and C. Lu, "Event-triggered adaptive fuzzy control for stochastic nonlinear systems with unmeasured states and unknown backlash-like hysteresis," *IEEE Trans. Fuzzy Syst.*, vol. 29, no. 5, pp. 1273–1283, May 2021.
[15] K. Deng, F. Li, and C. Yang, "A new data-driven model-free adaptive control for discrete-time nonlinear systems," *IEEE Access*, vol. 7, pp. 126224–126233, 2019.
[16] X. Zhang, H. Ma, X. Zhang, and Y. Li, "Compact model-free adaptive control algorithm for discrete-time nonlinear systems," *IEEE Access*, vol. 7, pp. 141062–141071, 2019.
[17] V. Milanes, J. Villagra, J. Godoy, and C. Gonzalez, "Comparing fuzzy and intelligent PI controllers in stop-and-go manoeuvres," *IEEE Trans. Control Syst. Technol.*, vol. 20, no. 3, pp. 770–778, May 2012.
[18] J. T. Agee, Z. Bingul, and S. Kizir, "Tip trajectory control of a flexible-link manipulator using an intelligent proportional integral (iPI) controller," *Trans. Inst. Meas. Control*, vol. 36, no. 5, pp. 673–682, Jul. 2014.
[19] J. T. Agee, S. Kizir, and Z. Bingul, "Intelligent proportional-integral (iPI) control of a single link flexible joint manipulator," *J. Vib. Control*, vol. 21, no. 11, pp. 2273–2288, 2015.

[20] A. Baciu and C. Lazar, "Model-free iPD control design for a complex nonlinear automotive system," in *Proc. 24th Int. Conf. Syst. Theory, Control Comput. (ICSTCC)*, Sinaia, Romania, Oct. 2020.
[21] M. Nakamoto, "An application of the virtual reference feedback tuning method to a multivariable process control," *IFAC Proc. Volumes*, vol. 38, no. 1, pp. 237–242, 2005.
[22] S. Formentin, G. Panzani, and S. M. Savaresi, "VRFT for LPV systems: Theory and braking control application," in *Robust Control and Linear Parameter Varying Approaches*. Heidelberg, Germany: Springer-Verlag, 2013, pp. 289–309.
[23] T. E. Passenbrunner, S. Formentin, S. M. Savaresi, and L. del Re, "Direct multivariable controller tuning for internal combustion engine test benches," *Control Eng. Pract.*, vol. 29, pp. 115–122, Aug. 2014.
[24] E. Madadi, Y. Dong, and D. Söffker, "Comparison of different model-free control methods concerning real-time benchmark," *J. Dyn. Syst., Meas., Control*, vol. 140, no. 12, Dec. 2018.
[25] P. Polack, S. Delprat, and B. d'Andréa-Novel, "Brake and velocity model-free control on an actual vehicle," *Control Eng. Pract.*, vol. 92, Nov. 2019, Art. no. 104072.
[26] S. Yahagi, I. Kajiwara, and T. Shimozawa, "Slip control during inertia phase of clutch-to-clutch shift using model-free self-tuning proportional-integral-derivative control," *Proc. Inst. Mech. Eng., D, J. Automobile Eng.*, vol. 234, no. 9, pp. 2279–2290, Aug. 2020.
[27] S. Yahagi and I. Kajiwara, "Direct tuning of gain-scheduled controller for electro-pneumatic clutch position control," *Adv. Mech. Eng.*, vol. 13, no. 7, pp. 1–12, 2021.
[28] R.-C. Roman, M.-B. Radac, and R.-E. Precup, "Mixed MFC-VRFT approach for a multivariable aerodynamic system position control," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Budapest, Hungary, Oct. 2016, pp. 002615–002620.
[29] R.-C. Roman, M.-B. Radac, R.-E. Precup, and E. Petriu, "Virtual reference feedback tuning of model-free control algorithms for servo systems," *Machines*, vol. 5, no. 4, p. 25, Oct. 2017.
[30] T. Sato, S. Miyake, and O. Kaneko, "An extension of intelligent PID control system to I-PD typed control system and parameter tuning by FRIT," *IEEJ Trans. Electron., Inf. Syst.*, vol. 142, no. 3, pp. 299–306, 2022.
[31] M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. New York, NY, USA: Springer-Verlag, 2006.
[32] S. Yahagi and I. Kajiwara, "Direct tuning method of gain-scheduled controllers with the sparse polynomials function," *Asian J. Control*, pp. 1–16, Aug. 2021, doi: 10.1002/asjc.2657.
[33] S. Formentin, S. M. Savaresi, and L. D. Re, "Non-iterative direct data-driven controller tuning for multivariable systems: Theory and application," *IET Control Theory Appl.*, vol. 6, no. 9, pp. 1250–1257, 2012.
[34] S. Formentin and A. Karimi, "Enhancing statistical performance of data-driven controller tuning via L_2-regularization," *Automatica*, vol. 50, no. 5, pp. 1514–1520, 2014.
[35] T. Yamamoto, K. Takao, and T. Yamada, "Design of a data-driven PID controller," *IEEE Trans. Control Syst. Technol.*, vol. 17, no. 1, pp. 29–39, Sep. 2009.
[36] Z. Lang, "On identification of the controlled plants described by the Hammerstein system," *IEEE Trans. Autom. Control*, vol. 39, no. 3, pp. 569–573, Mar. 1994.
[37] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, "GSA: A gravitational search algorithm," *J. Inf. Sci.*, vol. 179, no. 13, pp. 2232–2248, 2009.
[38] R.-E. Precup, M.-C. Sabau, and E. M. Petriu, "Nature-inspired optimal tuning of input membership functions of Takagi-Sugeno-Kang fuzzy models for anti-lock braking systems," *Appl. Soft Comput.*, vol. 27, pp. 575–589, Feb. 2015.

SHUICHI YAHAGI received the B.S. degree in engineering from the Shibaura Institute of Technology, Japan, in 2011, and the M.S. and Ph.D. degrees in engineering from Hokkaido University, Japan, in 2013 and 2021, respectively. He joined ISUZU Advanced Engineering Center Ltd., in 2013, where he is currently a Senior Research Engineer at the 6th Research Department. His research interests include automobile control, data-driven control, model-free control, and PID control. His research has led to over 18 patents.

ITSURO KAJIWARA received the B.S. degree in engineering from Tokyo Metropolitan University, in 1986, and the M.S. and Ph.D. degrees in engineering from the Tokyo Institute of Technology, in 1988 and 1993, respectively. From 1990 to 2000, he was an Assistant Professor with the School of Engineering, Tokyo Institute of Technology, where he was an Associate Professor with the Graduate School of Engineering, from 2000 to 2008. Since 2009, he has been a Professor with the Graduate School of Engineering, Hokkaido University. His research interests include vibration, control, structural health monitoring, and laser application.