Bing Ai
Department of Computer Science,
University of Texas at Austin,
Austin, TX 78712
e-mail: bingai@cs.utexas.edu

Luis Sentis[1]
Department of Mechanical Engineering,
University of Texas at Austin,
Austin, TX 78705
e-mail: lsentis@austin.utexas.edu

Nicholas Paine
Department of Mechanical Engineering,
University of Texas at Austin,
Austin, TX 78705
e-mail: npaine@utexas.edu

Song Han
Department of Computer Science and Engineering,
University of Connecticut,
Storrs, CT 06269
e-mail: song@engr.uconn.edu

Aloysius Mok
Department of Computer Science,
University of Texas at Austin,
Austin, TX 78712
e-mail: mok@cs.utexas.edu

Chien-Liang Fok
Department of Mechanical Engineering,
University of Texas at Austin,
Austin, TX 78705
e-mail: liangfok@utexas.edu

Stability and Performance Analysis of Time-Delayed Actuator Control Systems

Time delay is a common phenomenon in robotic systems due to computational requirements and communication properties between or within high-level and low-level controllers as well as the physical constraints of the actuator and sensor. It is widely believed that delays are harmful for robotic systems in terms of stability and performance; however, we propose a different view that the time delay of the system may in some cases benefit system stability and performance. Therefore, in this paper, we discuss the influences of the displacement-feedback delay (single delay) and both displacement and velocity feedback delays (double delays) on robotic actuator systems by using the cluster treatment of characteristic roots (CTCR) methodology. Hence, we can ascertain the exact stability interval for single-delay systems and the rigorous stability region for double-delay systems. The influences of controller gains and the filtering frequency on the stability of the system are discussed. Based on the stability information coupled with the dominant root distribution, we propose one nonconventional rule which suggests increasing time delay to certain time windows to obtain the optimal system performance. The computation results are also verified on an actuator testbed. [DOI: 10.1115/1.4032461]

1 Introduction

Due to data processing, data transition, and physical performance constraints of sensors and actuators, delay is certainly inevitable in almost all the robotic systems. The main sources of delay in these systems are sensing, actuation, and communication. The negative influence of the delay on the robotic systems has been studied for several decades, and it has been commonly believed that the delay is detrimental for system stability and performance [1–9]. However, the works in Refs. [10–14] have shown the useful effects of the system delay. Therefore, in this paper, we propose that for some robotic systems, increasing the feedback delay can stabilize and also improve the overall system performance. More specifically, we shall discuss the influences of time delay on robotic systems using a powerful tool called CTCR [15–20]. To validate the proposition, we will pursue the following studies.

We will construct a model of a robotic actuator and introduce the displacement-feedback delay and the velocity-feedback delay into our model.

We will obtain the exact stability interval for a displacement-feedback delay (single-delay) system and rigorous stability region for the system with both displacement-feedback delay and velocity-feedback delay (two/double delays) using the CTCR method.

We will discuss the performance of the system by changing the system delay(s) within the stability interval/region.

1.1 Related Work in Robotics Literature. One of the typical robotic systems is the bilateral teleoperation system, which has seen application in the areas of undersea exploration, outer space exploration, toxic material handling, and modern automation factories [21–25]. The bilateral teleoperation system enables robots to accomplish operations at a distance by the exchange of the information between a master robot and a slave robot. Ohnishi et al. [26,27] described bilateral telehaptics over a network to support human activities in remote environments and pointed out that the time delay compensation enables real world haptics to be utilized for various teleoperation applications like telesurgery. Since the communication network is a part of the system, Cheong et al. [28] gave a communication delay parameterized stability analysis of the bilateral teleoperation system.

Several methods were proposed to reduce the influence of the delay on the bilateral teleoperation system. The Smith predictor was adopted for the preknown constant delay of the system [29]. The passivity-based control principle [30–32] is another powerful tool to design the time delay robotic systems, based on which several approaches were proposed to reduce the influence of delays in single-master single-slave teleoperation systems [33] and single-master multiple-slave teleoperation systems [24,34]. Similar to

[1]Corresponding author.

Contributed by the Dynamic Systems Division of ASME for publication in the Journal of Dynamic Systems, Measurement, and Control. Manuscript received April 8, 2015; final manuscript received December 14, 2015; published online March 10, 2016. Assoc. Editor: M. Porfiri.

the Smith predictor or passivity-based control, Insperger et al. [35] proposed the act-and-wait control concept for a digital force control model with proportional feedback subject to a short, one sample unit feedback delay. By using this method, the proportional gain can be increased significantly without losing stability. By applying linear matrix inequalities and adaptive techniques, a fuzzy control strategy to reduce the influence of the stochastic delays was proposed in Ref. [36]. Bilateral teleoperation of holonomic constrained time delay robotic system was discussed in Ref. [37], where the partial feedback linearization method was used to decouple the delay dynamics from the local master/slave position/force unmodeled dynamics. Reference [38] gave an extensive study on Smith predictor-based control for haptic systems subject to distributed time delays. However, all of the above methods were concentrating on designing new control algorithms, which is cumbersome for most situations.

One of the most popular architectures in today's robotic systems is a distributed control architecture because of its modularity and high degree of flexibility [39–41]. A good example in this case is the hypothetical robot as shown in Fig. 1, which uses embedded systems located at each actuator to perform joint control. Typically, the high-level controller is responsible for generating reference trajectories to the distributed low-level controllers, but due to the computation density and communications with low-level controllers, the high-level controller is always subject to larger delays. On the contrary, although the low-level controllers lack information of the whole system they usually experience smaller delays because each low-level controller is only responsible for computation of its own actuator and there is no communications between different actuators through low-level controllers (more detailed reasons for using distributed control architecture in robotics can be found in Ref. [41]).

Distributed control architecture is also widely used in wireless sensor network where each sensor node only takes care of its own computation and communication with gateway [42]. Another example of distributed control design can be seen in high-mixed semiconductor manufacturing run-to-run process control where the same kinds of products are grouped together, and the control actions are made based on the output of the latest product of the same kind instead of the output of the previous product (which may be different) [43–48].

Our actuator testbed at the Human Centered Robotics Lab at the University of Texas at Austin (UT), Austin, TX serves as a good example of a robotic system with distributed controller architecture. Based on the Nyquist stability criterion, Zhao et al. [49] discussed the performance of the UT actuator testbed with different fixed time delays. Paine and Sentis [50] gave a closed-form solution for selecting maximum critically damped actuator impedance parameters for the known fixed delay of the system.

However, all of the above works focused on the stability problem of the system with a given delay or achieved conservative results for an unknown time delay. In this paper, we give the first exact analytical solution to characterize the maximum tolerable delay (MTD)/latency (the maximum delay of the system before it becomes unstable) of the system. The authors think that this is important especially in distributed systems because knowing the maximum latency of the system will ease the real-time hard scheduling problem by giving a lower priority for the task which has the larger maximum tolerable latency. The CTCR method which examines the stability property rigorously shines light on the unusual behavior associated with increased delays. Therefore, the main contributions of this paper are as follows:

(i) The maximum stability interval for a single-delay robotic system and the maximum stability region for a double-delay robotic system have been obtained by using the CTCR technique. One of the important properties of the obtained stability interval and region is that they are exact and not approximated.
(ii) Based on the stability information of the robotic system, we proposed one unconventional method to increase system delay within the maximum tolerable stability interval/region to improve the system overall performance.

Although our analysis will be mainly focused on analyzing time delay of the UT actuator testbed, the same procedures can be easily implemented to analyze other time-delayed robotic systems.

The remainder of the paper is organized as follows: In Sec. 2, we will summarize the CTCR methodology for both single time delay and multiple time delay systems (TDS). In Sec. 3, performance for TDS will be discussed. Sections 4 and 5 examine the influences of single time delay and double time delays on the UT actuator testbed and how to improve the system performance will also be addressed. Experimental validation of our analysis will be done in Sec. 6, and the concluding remarks are presented in Sec. 7.

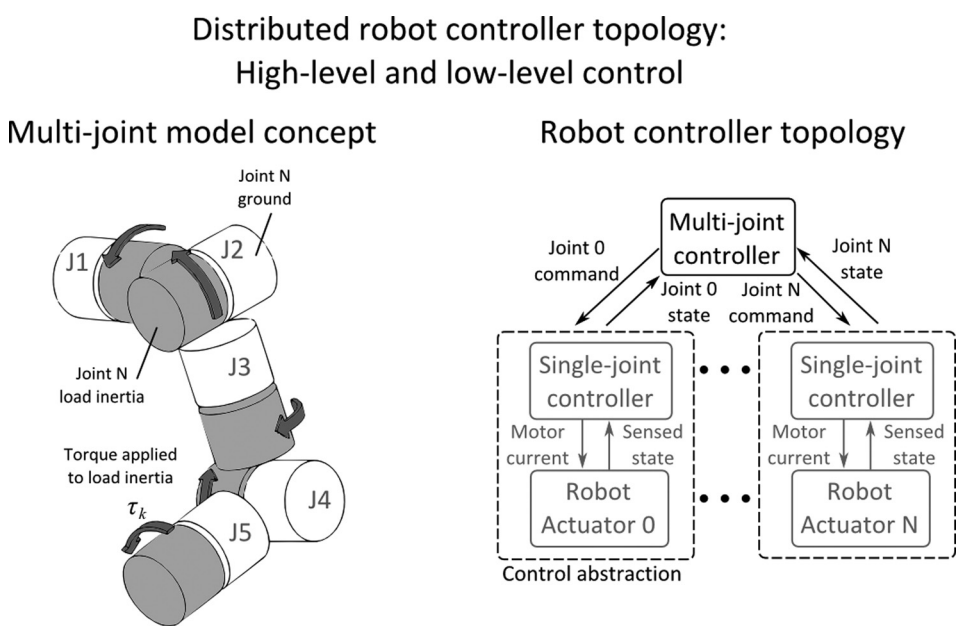


Fig. 1 Hypothetical robot control architecture

2 Stability Theorems for Time Delay Linear Time Invariant (LTI) Systems

Stability is one of the important properties of any real system. An unstable system should not be used in any case. In this section, one powerful stability tool for LTI systems with single delay and multiple delays will be summarized.

2.1 Single TDS. A general class of LTI-TDS is given by the form

$$\dot{\boldsymbol{X}}(t) = \boldsymbol{A}\boldsymbol{X}(t) + \boldsymbol{B}\boldsymbol{X}(t-\tau) \tag{1}$$

where $\boldsymbol{X} \in \mathbb{R}^n$ is the state of the system, and $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{n\times n}$ are the system matrix and the input matrix, respectively, both of which are constant. $\tau \in \mathbb{R}^+$ is the time delay of the system.

The stability of this kind of system is determined by the roots of its characteristic equation (CE), which is obtained by Laplace transform with Laplace variable s and a proper dimensional identity matrix $\boldsymbol{I}$, i.e., the CE of system (1) is

$$\mathrm{CE}(s,\tau) = \det(s\boldsymbol{I} - \boldsymbol{A} - \boldsymbol{B}e^{-\tau s}) = 0 \tag{2}$$

which can be expanded to obtain

$$\begin{aligned}\mathrm{CE}(s,\tau) &= a_n(s)e^{-n\tau s} + a_{n-1}(s)e^{-(n-1)\tau s} + \cdots \\ &+ a_1(s)e^{-\tau s} + a_0(s) = \sum_{j=0}^{n} a_j(s)e^{-j\tau s}\end{aligned} \tag{3}$$

where $a_j(s)$ are the polynomials in s with real coefficients of degree $n-j$. Interestingly enough, this kind of system is "retarded" due to the fact that the derivative term in Eq. (1) is not influenced by the time delay and s^n term only appears at $a_0(s)$. Notice that $a_0(s)$ in Eq. (3) is the only polynomial which contains the highest order, n, of s, and no delay term accompanying it [51].

It is well known in control theory that system (1) is asymptotically stable if and only if all of the characteristic roots of its CE are located in the left-half of the complex plane; each of these roots with negative real part pushes $\boldsymbol{X}(t)$ to 0 as $t \to +\infty$. However, the CE of system (1) in form (2) (or in its generic form (3)) is transcendental, having infinitely many roots. As discussed in Refs. [51–54], for the retarded LTI-TDS, there are an infinite number of roots in the left-half plane and a finite number of roots in the right-half plane (if any exist); the number of purely imaginary roots has an upper bound [17].

One of the possible root distributions in the complex plane for system (1) is shown in Fig. 2, where the stable root (blue) represents the complex root with negative real part. An unstable root (red) refers to a complex root with positive real part. The imaginary roots on the imaginary axis are depicted in black.

The examination of the location of infinite many roots is a complex task, and several methods have been proposed in Refs. [55–57], but unfortunately none of them resolved the problem completely. In Refs. [58–61], the researchers proposed simplified tests for determining the ranges of stability which involve the introduction of a pseudodelay T in the form of

$$e^{-\tau s} = \frac{1-Ts}{1+Ts} T \in \mathbb{R} \tag{4}$$

which only holds on the imaginary axis, i.e., $s = \pm\omega i$ and $i = \sqrt{-1}$ (since the root loci are symmetric with respect to the real axis, we only concentrate on $s = +\omega i$ and ignore the case of $s = -\omega i$). The existence of Eq. (4) was guaranteed by the D-subdivision method (or the "continuity argument"), which states that there are regions within which the number of unstable (NU) root of Eq. (2) is fixed, and there are at least one pair of purely imaginary roots for each τ on the boundaries which separate these regions [62]. By using the D-subdivision method, one can develop a complete description of the stability regions in the delay space. The application of the D-subdivision method for LTI-TDS can be found in Refs. [63–66]. The substitution of the transcendental term with a rational function in the form of Eq. (4) is often called the *Rekasius substitution* in the control and math literature.

It is obvious that the angular mapping condition of the Rekasius substitution is

$$\tau = \frac{2}{\omega}\left[\tan^{-1}(\omega T) \pm l\pi\right] \quad l = 0, 1, \ldots, +\infty \tag{5}$$

from which one knows that there are an infinite number of values of τ that can be mapped onto the same ω.

Substituting Eq. (4) into Eq. (3), one obtains

$$\sum_{j=0}^{n} a_j(s)\left(\frac{1-Ts}{1+Ts}\right)^j = 0 \tag{6}$$

multiplying Eq. (6) with $(1+Ts)^n$ and grouping by powers of s, one gets

$$\overline{\mathrm{CE}}(s,T) = \sum_{j=0}^{2n} b_j(T)s^j \tag{7}$$

which is a polynomial of degree $2n$ with coefficients b_j (polynomials in T). Its pure imaginary roots coincide with the pure imaginary roots of Eqs. (2) and (3). It is important to note that the Rekasius substitution is an exact method, unlike the Pade approximant or the Kautz formula methods.

By the Rekasius substitution, the transcendental CE (2) or (3), which has an infinite number of solutions, has been reduced to a finite polynomial equation and hence possesses only a finite number of solutions. This in return makes detection of the stability of the LTI-TDS analytically tractable. Since T has an infinite range, it is important to note that the imaginary crossings are not lost with the dimensional reduction from Eqs. (3) to (7). The systematic detection of the stability of LTI-TDS is introduced first in a method called CTCR by Olgac and Sipahi [15,16].

The main philosophy behind the CTCR paradigm is the "clustering" of all possible imaginary crossing frequencies in delay domain. This method starts with exhaustively finding these crossing frequencies and then determining the root tendencies at each crossing. The following is an outline of the CTCR paradigm:

(i) The LTI-TDS only has a finite number of possible imaginary characteristic roots $\omega_c i$ for all $\tau \in \mathbb{R}^+$, where ω_c is given in a set

$$\omega_c = \{\omega_c^1, \quad \omega_c^2, \ldots, \quad \omega_c^m\} \quad m < +\infty \tag{8}$$

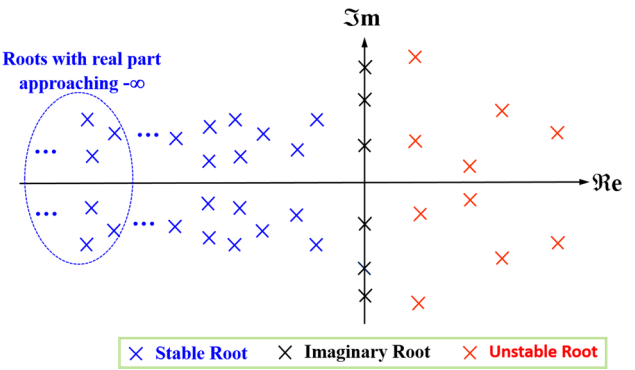


Fig. 2 Possible root distribution of retarded LTI-TDS

and subscript c refers to "crossing" the imaginary axis, the superscript m means the mth crossing.

(ii) For each imaginary crossing, $s = \omega_c^k i,\ k = 1, 2, \ldots, m$, there are infinite many periodically spaced τ values, i.e.,

$$\tau_k = \{\tau_k^0, \tau_k^1, \ldots, \tau_k^j, \ldots, \tau_k^\infty\} \tag{9}$$

where the subscript k of τ_k corresponds to delay for the ω_c^k, and the superscript j refers to the jth solution in τ_k. From the angular mapping condition of the Rekasius substitution, i.e., Eq. (5), one obtains that $\tau_k^j = 2/\omega_c^k\left[\tan^{-1}\left(\omega_c^k T\right) \pm j\pi\right]$ and $\tau_k^{l+1} - \tau_k^l = 2\pi/\omega_c^k$. Since the time delay for the real system cannot be negative, one always only consider the positive time delays.

(iii) The root tendency (RT) associated with each purely imaginary characteristic root, $s = \omega_c^k i$, is defined as

$$\mathrm{RT}|_{s=\omega_c^k i;\tau=\tau_k} \triangleq \mathrm{sgn}\left[\mathrm{Re}\left(\frac{ds}{d\tau}\bigg|_{s=\omega_c^k i;\tau=\tau_k}\right)\right] \tag{10}$$

is invariant at each purely imaginary characteristic root, $s = \omega_c^k i$, and corresponding infinite delays τ_k, i.e.,

$$\mathrm{RT}|_{s=\omega_c^k i;\tau=\tau_k} = \mathrm{RT}|_{s=\omega_c^k i;\tau=\tau_k^j} \tag{11}$$

where $ds/d\tau$ can be obtained from Eq. (2), i.e.,

$$\frac{ds}{d\tau} = -\frac{\partial \mathrm{CE}(s,\tau)}{\partial \tau} \Big/ \frac{\partial \mathrm{CE}(s,\tau)}{\partial s} \tag{12}$$

Note that RT describes the direction of transition of the roots at $s = \omega_c^k i$ as τ_k^l increases from $\tau_k^l - \varepsilon$ to $\tau_k^l + \varepsilon$, $0 < \varepsilon \le 1$. If $\mathrm{RT} = +1$, which means the roots cross the imaginary axis from the left-half plane to the right-half plane, the corresponding delays will bring the instability into the system (Fig. 3(*a*)). On the other hand, if $\mathrm{RT} = -1$, the roots cross the imaginary axis from the right-half plane to the left-half plane, and the corresponding delays will possibly stabilize the system (Fig. 3(*b*)).

The imaginary crossing frequencies can be efficiently obtained by using *Routh array* properties (the Routh array for Eq. (7) is listed in Table 1) [15,16].

The following is a brief review of the properties of the Routh stability criterion.

If Eq. (7) has a pair of purely imaginary roots, then the next two conditions must be satisfied:

(i) The only term on the row corresponding to s^1, call it $R_{11}(T)$, must be zero.

(ii) The two terms on the row corresponding to s^2, call them $R_{21}(T)$ and $R_{22}(T)$, must satisfy the so-called "auxiliary equation"

$$R_{21}(T)s^2 + R_{22}(T) = 0 \tag{13}$$

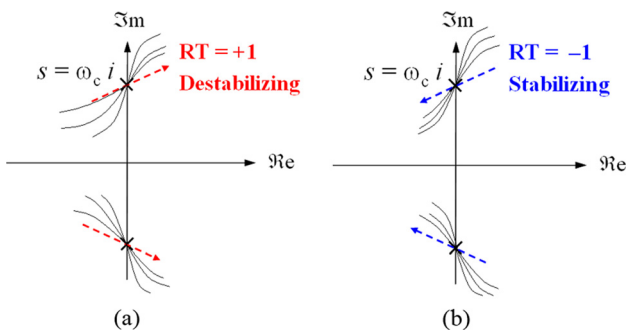


Fig. 3 RT of retarded LTI-TDS

and the sign agreement (to exclude the asymmetric real roots case) between these coefficients must be satisfied, i.e.,

$$R_{21}(T) \cdot R_{22}(T) > 0 \tag{14}$$

and whenever Eq. (14) holds, the corresponding frequency is

$$\omega_c = \sqrt{R_{22}(T)/R_{12}(T)} \tag{15}$$

When implementing the CTCR method, one needs to find each T_c^k which satisfies $R_{11}\left(T_c^k\right) = 0$ and then the corresponding frequencies $s = \omega_c^k i$ can be determined from Eq. (15). By using Eq. (9), the infinite number of delays for each ω_c^k can be found exhaustively, and therefore, their RT can be examined directly by using Eq. (10). Schematically, the correspondence is as follows:

$$T_c^k \overset{\text{generates}}{\longrightarrow} \omega_c^k \overset{\text{generates}}{\longrightarrow} \tau_k, \quad k = 1, 2, \ldots, m \tag{16}$$

From the above analysis, one can establish $\left\{\omega_c^k, \tau_k\right\}$, and the stability of the LTI-TDS is determined by the following steps [15]:

(i) Form a table of τ_k^l, where $k = 1, 2, \ldots, m$, $l = 1, 2, \ldots, +\infty$, and corresponding $\mathrm{RT}|_{s=\omega_c^k i;\tau=\tau_k}$ in ascending order of τ_k^l.

(ii) Obtain the NU roots for $\tau = 0$ from the Routh stability criterion.

(iii) Calculate the NU roots by using the information of $\mathrm{RT}|_{s=\omega_c^k i;\tau=\tau_k}$. If $\mathrm{RT}|_{s=\omega_c^k i;\tau=\tau_k} = +1$, then increase NU by 2; if $\mathrm{RT}|_{s=\omega_c^k i;\tau=\tau_k} = -1$, decrease NU by 2.

(iv) Repeat the previous step for the next τ_k^l until the τ_k of interest is reached.

(v) Identify those regions in τ_k, where $\mathrm{NU}(\tau) = 0$ as stable and others as unstable.

Remark. It is impossible to form a table with infinite many delays. In the real implementation of CTCR, it proves out that it is unnecessary to consider all the delays. What one needs to do is calculate the NU for the first few delays until there are delays for each imaginary crossing. Repeat this step several times, and note the pattern of NU based on the RT and τ_k^j. The stability of the system can be easily judged based on the pattern of NU.

2.2 Multiple TDS. As described in Ref. [67], the stability problem for multiple TDS is *N-P* hard due to the lack of a numerically efficient algorithm to solve it. Implementing the same idea as CTCR, this problem becomes tractable but still *N-P* hard [18–20]. For the LTI system with multiple (P) independent time delays in the form of Eq. (17), we summarize the main results obtained by implementing the CTCR methodology

$$\dot{\boldsymbol{X}}(t) = \boldsymbol{A}\boldsymbol{X}(t) + \sum_{J=1}^{P} \boldsymbol{B}_J \boldsymbol{X}(t - \tau_J) \tag{17}$$

where the capital J for subscript is used here to differentiate from the single time delay case.

The CE of this kind of system is

$$\begin{aligned}\mathrm{CE}(s,\boldsymbol{\tau}) &= \mathrm{CE}(s,\tau_1,\tau_2,\ldots,\tau_P) \\ &= \det\left(s\boldsymbol{I} - \boldsymbol{A} - \sum_{J=1}^{P} \boldsymbol{B}_J e^{-\tau_J s}\right) = 0\end{aligned} \tag{18}$$

The same as single time delay LTI systems, by implementing the Rekasius substitution on each delay τ_J, one obtains the pseudo delay T_J which satisfies

$$e^{-\tau_J s} = \frac{1 - T_J s}{1 + T_J s} \quad T_J \in \mathbb{R} \tag{19}$$

and substituting Eq. (19) into Eq. (18), the resulting equivalent CE on imaginary axis is

$$\overline{\overline{\mathrm{CE}}}\,(s, \boldsymbol{T}) \triangleq \overline{\overline{\mathrm{CE}}}(s, T_1, T_2, \ldots, T_P) = \sum_{J=0}^{2n} b_J(T_1, T_2, \ldots, T_P) s^J \tag{20}$$

In order to examine the influences of delays on the imaginary crossings, we map the solution of Eq. (20) for each crossing $s = \omega_c^K i$ (capital K for superscript refers to the Kth crossing) into delay space by using Eq. (19) and denote its solution as $\boldsymbol{\tau}^K = \{\tau_{10}^K + (2\pi/\omega_c^K)K_1, \tau_{20}^K + (2\pi/\omega_c^K)K_2, \ldots, \tau_{P0}^K + (2\pi/\omega_c^K)K_P\}$ with $0 \le \tau_{J0}^K < 2\pi/\omega_c^K$, $K = 1, 2, \ldots, M$, and $K_1, K_2, \ldots, K_P = 0, 1, 2, \ldots, +\infty$. Furthermore, we define the smallest positive delay set, $\boldsymbol{\tau}_{\mathrm{kernel}}^K \triangleq \boldsymbol{\tau}^K|_{K_1 = K_2 = \cdots = K_P = 0}$, and the remaining sets as $\boldsymbol{\tau}_{\mathrm{offspring}}^K \triangleq \boldsymbol{\tau}^K - \boldsymbol{\tau}_{\mathrm{kernel}}^K$. The corresponding kernel hyperplanes and offspring hyperplanes consist of the delays at all the imaginary crossings: $\wp_{\mathrm{kernel}} = \{\boldsymbol{\tau}_{\mathrm{kernel}}^1, \boldsymbol{\tau}_{\mathrm{kernel}}^2, \ldots, \boldsymbol{\tau}_{\mathrm{kernel}}^M\}$ and $\wp_{\mathrm{offspring}} = \{\boldsymbol{\tau}_{\mathrm{offspring}}^1, \boldsymbol{\tau}_{\mathrm{offspring}}^2, \ldots, \boldsymbol{\tau}_{\mathrm{offspring}}^M\}$.

The following results are held:

(i) The offspring hyperplanes are uniquely determined by kernel hyperplanes.

(ii) The RT

$$\mathrm{RT}|_{s=\omega_c^K i;\tau=\tau_{J0}^K} \triangleq \mathrm{sgn}\left[\mathrm{Re}\left(\frac{ds}{dt}\bigg|_{s=\omega_c^K i;\tau=\tau_{J0}^K}\right)\right]$$

is invariant as long as the grid points on different offspring hyperplanes are selected, while the $\tau_{Q0}^K + (2\pi/\omega_c^K)K_Q$ (with $Q = 1, 2, \ldots, J-1, J+1, \ldots, P$ and $K_Q = 0, 1, \ldots, +\infty$) are kept unchanging, i.e., for a specific crossing frequency, the RT on offspring hyperplanes is the same as the RT on the corresponding kernel hyperplanes $\left(\mathrm{RT}|_{s=\omega_c^K i;\tau=\tau_{J0}^K} = \mathrm{RT}|_{s=\omega_c^K i;\tau=\tau_{J0}^K + (2\pi/\omega_c^K)K_J}\right)$.

(iii) The possible stability switch can only take place on the kernel hyperplanes and offspring hyperplanes.

The same as in the single time delay case, for the multiply time delay case, we claim that the regions with $\mathrm{NU}(\boldsymbol{\tau}) = \mathrm{NU}(\tau_1, \tau_2, \ldots, \tau_P) = 0$ are stable for the system, and the regions with $\mathrm{NU}(\boldsymbol{\tau}) = \mathrm{NU}(\tau_1, \tau_2, \ldots, \tau_P) > 0$ are unstable for the system.

Detailed proofs can be found in Refs. [18–20].

Since Eqs. (19) and (20) have multiple variables $T_J \in \{T_1, T_2, \ldots, T_P\}$ in them, one cannot use Routh array properties to calculate the crossing frequencies. Instead the half-angle substitution method can be adopted. To simplify our illustration, we use the two-TDS as an example and the angular mapping condition can be obtained from Eq. (19)

$$T_J \omega = \tan\left(\frac{\xi_J}{2}\right) = Z_J \tag{21}$$

where $\xi_J = \tau_J \omega \in [0, 2\pi)$ and $J = 1, 2$. By substituting Eq. (21) into Eq. (20), one can obtain the CE of the two TDS

Table 1 Routh array for CE of Eq. (7)

s^{2n}	$b_{2n}(T)$	$b_{2n-2}(T)$	$\cdots$	b_0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
s^2	$R_{21}(T)$	$R_{22}(T) = b_0$		
s^1	$R_{11}(T)$			
s^0	$R_{01}(T) = b_0$			

$$q(s, Z_1, Z_2) = \sum_{L=0}^{n} C_L(Z_1, Z_2) s^L = 0 \tag{22}$$

with coefficients of C_L (polynomials in both Z_1 and Z_2). In order for Eq. (22) to hold, both imaginary and real parts of $q(s, Z_1, Z_2)$ with $s = \omega i$ should be 0, i.e.,

$$\begin{cases} \mathrm{Re}(q(s, Z_1, Z_2)) = \sum_{L=0}^{n} f_L(Z_1, Z_2)\omega^L = 0 \\ \mathrm{Im}(q(s, Z_1, Z_2)) = \sum_{L=0}^{n} g_L(Z_1, Z_2)\omega^L = 0 \end{cases} \tag{23}$$

where f_L and g_L are the coefficients of the real part and imaginary part of $q(\omega i, Z_1, Z_2)$, respectively. The common root of Eq. (23) can be obtained by using the Sylvester method [68]

$$\boldsymbol{M} \cdot \boldsymbol{W}^{\mathrm{T}} = 0 \tag{24}$$

where $\boldsymbol{W}^{\mathrm{T}} = (\omega^n \ \omega^{n-1} \ \cdots \ \omega^1 \ \omega^0)^{\mathrm{T}}$ and

$$M = \begin{pmatrix} f_n & f_{n-1} & \cdots & f_0 & 0 & \cdots & 0 & 0 \\ 0 & f_n & \cdots & f_1 & f_0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \ldots & f_n & f_{n-1} & \ldots & f_0 & 0 \\ 0 & 0 & \ldots & 0 & f_n & \ldots & f_1 & f_0 \\ g_n & g_{n-1} & \ldots & g_0 & 0 & \ldots & 0 & 0 \\ 0 & g_n & \ldots & g_1 & g_0 & \ldots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \ldots & g_n & g_{n-1} & \ldots & g_0 & 0 \\ 0 & 0 & \ldots & 0 & g_n & \ldots & g_1 & g_0 \end{pmatrix} \tag{25}$$

the arguments (Z_1, Z_2) of both functions $f(Z_1, Z_2)$and$g(Z_1, Z_2)$ have been omitted for brevity. Equation (24) has solution only if the determinant of the Sylvester's resultant matrix ($\boldsymbol{M}$) is singular [69].

One can scan ξ_1 through $[0, 2\pi)$ and calculate ξ_2 from Eq. (24), i.e., for each fixed ξ_1, the ξ_2 can be evaluated based on the singularity condition of Eq. (25), and the corresponding ω value can be obtained from Eq. (23). Note that each (ξ_1, ξ_2) will result in a pair of $\pm\omega$. With the information of (ξ_1, ξ_2, τ), and the kernel hyperplanes $\wp_{\mathrm{kernel}}$ and the offspring hyperplanes $\wp_{\mathrm{offspring}}$ can be obtained directly; therefore, the stability of the system can be checked on the kernel and corresponding offspring hyperplanes.

For the system with arbitrary number of delays, Refs. [70] and [71] provide advanced methodologies to find the kernel and offspring hyperplanes.

3 System Performance for LTI-TDS

If a system is proved to be stable, one should then inspect the next important property performance. In this section, we will discuss the influences of the root of the CE on the system performance and propose a delay selection rule to improve the system performance for LTI-TDS.

For a second-order delay-free system, the specifications like rise time, settling time, overshoot, and peak time for the step response are often examined during control system design. Linear quadratic regulation (LQR) method and model approximation approach are widely adopted in higher-order system designs. The LQR method always gives an exact solution, while the model approximation approach usually renders approximate solutions by reducing a higher-order system into a second-order system. Let us

define DI as the distance between the dominant root and the imaginary axis. In order to obtain a good solution, the *model approximation* method requires that all the other roots of the CE of the closed-loop system must be at least five times DI from the dominant (rightmost) root. However, it is impossible to do the rigorous performance analysis for generic LTI-TDS because of the infinitely many roots of its CE and the fact that there is no explicit formula solution for a general equation with an order higher than four (also known as the *Abel–Ruffini theorem*).

In the real applications, one is more concerned about the time required for the transient to decay to a small value so that the desired output is almost in the steady-state. The time required is mainly determined by the dominant roots of the CE of both LTI-TDS or delay-free LTI system. Therefore, we will use the real part of the dominant root as the main factor to analyze the system performance especially for the settling time in this section. The possible use of LQR and pole placement methods for LTI-TDS are subjects for future research.

If there are infinitely many dominant roots in the complex plane, then it is impossible for us to analyze the system performance. Fortunately, the following lemma guarantees a finite number of dominant roots:

LEMMA 1. *For any $\rho \in \mathbb{R}$, the number of roots of the CE of the retarded LTI-TDS in the half plane of $\mathrm{Re}(s) > \rho$ is finite [72,73,74].*

A mapping-based algorithm called *QPMR algorithm* is widely used to find the roots of the CE (see Refs. [74–76] for details). In implementing the QPMR algorithm, one substitutes $s = \alpha + \beta i$ with $\alpha, \beta \in \mathbb{R}$ into the quasi-polynomial (i.e., the CE of the system) and separates this quasi-polynomial into real and imaginary parts. The intersection points of these two parts are the solution of the quasi-polynomial, i.e., the root of the CE.

By adopting the CTCR method, one can always find the MTD for each stable interval of the system with a single-delay and maximum tolerable delay region (MTDR) for each stable region of the system with multiple delays. It is worth pointing out that, for some systems, there are more than one stable interval/region. The MTD here means the maximum delay that the system can bear before it changes from stable to unstable for the single-delay system. The MTDR means the maximum delay combination within which the system is stable for multiple delays case.

The MTD/MTDR information coupled with Lemma 1 may enable us to improve the performance of the system by choosing proper delay/delay-combination which is no greater than MTD or within MTDR. We propose the following rule for selecting the latency of the system.

3.1 Delay Selection Rule

(1) If the real part of the rightmost characteristic root $\mathrm{Re}\left(s_{\mathrm{dom}}^{\mathrm{first}}\right)$ is five times DI from other roots, we choose the delay τ which leads to the smallest $\mathrm{Re}\left(s_{\mathrm{dom}}^{\mathrm{first}}\right)$.

(2) If the real part of the rightmost characteristic root $\mathrm{Re}\left(s_{\mathrm{dom}}^{\mathrm{first}}\right)$ is close (less than five times DI) to the real part of its nearest root $\mathrm{Re}\left(s_{\mathrm{dom}}^{\mathrm{second}}\right)$, we choose the delay τ which maximizes the following function:

$$\max_{\tau}\left\{-\lambda_1 \cdot \mathrm{Re}\left(s_{\mathrm{dom}}^{\mathrm{first}}\right) + \lambda_2 \cdot \mathrm{Im}\left(s_{\mathrm{dom}}^{\mathrm{second}}\right)\right\} \tag{26}$$

subject to

$$\lambda_1 + \lambda_2 = 1, \quad \lambda_1, \lambda_2 \in \mathbb{R}^+ \tag{27}$$

where λ_1 and λ_2 are the weights determined by real application requirements.

Since the settling time is approximately four times of the system time constant $-1/\mathrm{Re}\left(s_{\mathrm{dom}}^{\mathrm{first}}\right)$, when $\mathrm{Re}\left(s_{\mathrm{dom}}^{\mathrm{first}}\right)$ and $\mathrm{Re}\left(s_{\mathrm{dom}}^{\mathrm{second}}\right)$ are close to each other, one should try to increase the weight of $\mathrm{Im}\left(s_{\mathrm{dom}}^{\mathrm{second}}\right)$ to accelerate the peak time of the response.

Note that the delay selection rule does not necessarily generate the delay value of zero. In other words, increasing the delay of the system purposely may give the system better performance than a delay-free case.

Although the system may have multiple stable intervals/regions, one is likely more interested in the MTD/MTDR for the first stable interval/region, because the system is prone to achieve better performance by choosing a delay within this interval/region, this is more evident when the second stable interval/region is far away from the first stable interval/region.

The following example is used to explain the reason why the system is prone to have better performance in the first stable region.

Taking the single-delay case as an example; if the system is stable in delay spaces $(0, 5)$ and $(10, 20)$, then the optimal delays will fall within these intervals. Given the optimal delays are 3 s and 11 s and the settling times of the system are 10 s and 5 s, then the entire process takes 13 s and 16 s. Although the settling time for the 11 s optimal delay is shorter, we prefer using the optimal delay in the first stable interval because the entire process takes only 13 s opposed to 16 s.

4 Robotic Actuator Control Systems With Displacement-Feedback Delay

In this section, we will discuss the influences of filtering frequency and controller gains on the MTD of the UT actuator testbed with displacement-feedback delay by using the CTCR tool. Based on the knowledge of MTD, the system performance will be discussed for different time delays within MTD. In addition, the possible ways of improving system performance will be discussed.

4.1 System Model. As mentioned in Ref. [50], most typical actuators (see Fig. 4) can be modeled as a force (F_{ext}) on a mass-(m) damper (b) system. The total system impedance, which represents the relationship between force and the displacement (X) of the actuator, is composed of both the impedance of the system (Z_{passive}) and the impedance caused by the force (F) supplied by the actuator (Z_{active}), i.e.,

$$Z(s) = \frac{F_{\mathrm{ext}}(s)}{X(s)} = Z_{\mathrm{passive}}(s) + Z_{\mathrm{active}}(s) \tag{28}$$

where $Z_{\mathrm{passive}} = -ms^2 - bs$ and $Z_{\mathrm{active}} = F(s)/X(s)$.

If the proportional-derivative (PD) controller with derivative filter is used to generate actuator force, then one obtains

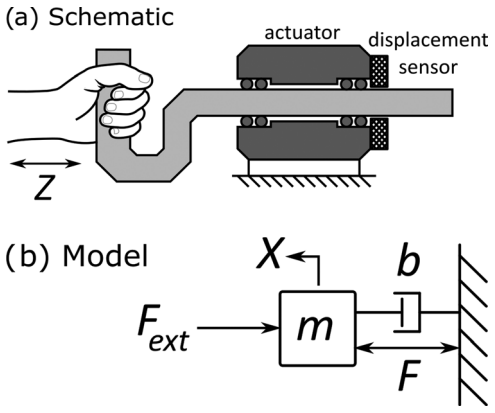


Fig. 4 Actuator model with external force and actuator force: (*a*) schematic and (*b*) model

$$F(s) = K_D \cdot (X_d(s) - X(s)) + K_v \cdot s \cdot (X_d(s) - X(s)) \cdot Q_v(s) \tag{29}$$

where K_D is the P-controller gain, K_v is the D-controller gain, X_d is the desired actuator displacement, and $Q_v = 2\pi f_v/(s + 2\pi f_v)$ is the first-order filter with the frequency f_v.

In real applications, however, there is always a delay in the actuator system due to computation and communication time requirements. The block diagram for the UT actuator testbed with displacement-feedback delay of value τ_D is depicted in Fig. 5. In the figure, the actuator force is transformed into motor current $(I(s))$ by a constant $G(s) = 1/(NC\eta)$, where N is the speed reduction, C is the motor torque constant, and η is drivetrain efficiency. The transfer function between the motor current and displacement is denoted by $P(s)$ and $P(s) = NC\eta/(ms^2 + bs)$.

The system with PD-controller coupled with a first-order filter described above is with single time delay, and the closed-loop transfer function for this system is

$$G_{\text{cl}}^{\text{single-delay}}(s) = \frac{K_v s + K_D}{ms^2 + bs + (K_v Q_v(s)s + K_D)e^{-\tau_D s}} = \frac{K_v s^2 + (2\pi f_v K_v + K_D)s + 2\pi f_v K_D}{ms^3 + (2\pi f_v m + b)s^2 + 2\pi f_v bs + [(2\pi f_v K_v + K_D)s + 2\pi f_v K_D]e^{-\tau_D s}} \tag{30}$$

It is evident that if $\tau_D = 0$ and $Q_v(s) = 1$, then the above system is exactly second-order delay-free mass–damper system. In the UT actuator testbed, for the full actuator experiment, $m = 256$ kg and $b = 1250\,\text{N} \cdot \text{s/m}$ are used.

4.2 MTD Analysis. We first use a simple example to illustrate how to obtain the MTD of the system by using the CTCR method. In this computation, the values of $f_v = 50$, $K_D = 363833$, and $K_v = 18051$ are randomly chosen, the CE of closed-loop system is

$$\text{CE}(s, \tau_D) = 256s^3 + 81,634s^2 + 392,500s + (6,031,847s + 114,243,562)e^{-\tau_D s} \tag{31}$$

The pioneering researches in computing the imaginary crossings and delays for the problem in form of Eq. (31) can be found in Refs. [77–82].

The characteristic roots of the delay-free system are -25.23, -84.60, and -209.6, all of which are located on the left-half plane. Therefore, the delay-free system is stable, i.e., $\text{NU}(0) = 0$. The maximum delay this system can bear under this set of parameters can be obtained by the following steps.

By applying the Rekasius substitution in Eq. (31), one obtains

$$\overline{\text{CE}}(s, T) = 256Ts^4 + (256 + 81,634T)s^3 + (81,634 - 5,639,347T)s^2 + (6,424,347 - 114,243,562T)s + 114,243,562 \tag{32}$$

By using the Routh stability criterion in Eq. (32), the auxiliary equation can be obtained

$$R_{21}(T)s^2 + 114,243,562 = 0 \tag{33}$$

and the first row of the Routh table is

$$R_{21}(T)(6,424,347 - 114,243,562T) - (256 + 81,634T)14,243,562 = 0 \tag{34}$$

where

$$R_{21}(T) = \frac{(81,634 - 5,639,347 \cdot T)(256 + 81,634 \cdot T) - 256T(6,424,347 - 114,243,562T)}{256 + 81,634T}$$

Solving Eq. (34), one obtains three T values, represented as $T_c^1 = -0.0039$, $T_c^2 = 0.0087$, and $T_c^3 = 0.0753$, where T_c^j refers to the jth solution. In order to satisfy the sign agreement for the particular case of the Routh stability criterion (see Eq. (14)), i.e., $R_{21}(T) \cdot 114,243,562 > 0$, only $T_c^2 = 0.0087$ is selected from these three T values, and the corresponding crossing frequency which is calculated from Eq. (15) is $\omega_c = 75.1054\,\text{rad/s}$. Substituting ω_c and T_c^2 into Eq. (5), one obtains all delays $\tau_D^l = 0.0154 + 0.0836l$ with $l = 0, 1, 2, \ldots, +\infty$ for crossing frequency 75.1054 rad/s. In addition, from Eq. (10), one obtains $\text{RT}|_{s=\omega_c i;\tau=\tau_D^l} = +1$, which means for each τ_D^l value, the NU roots of the closed-loop CE of the TDS will be increased by two, and the only stable interval of the delay for the system is $[0, 0.0154)$. The detailed NU roots in each delay interval are given in Table 2.

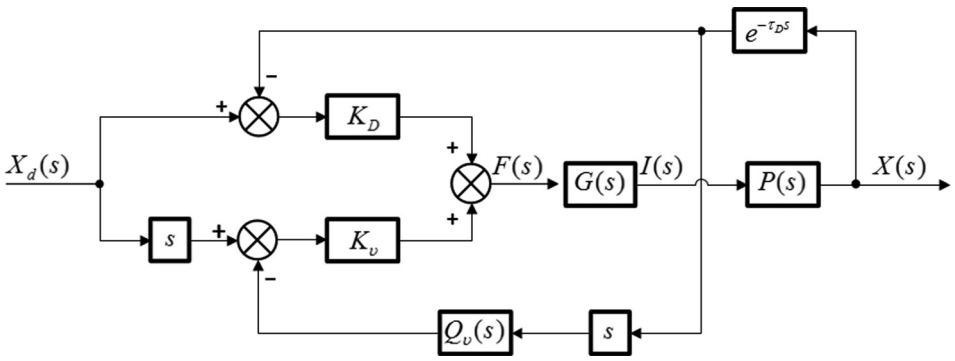


Fig. 5 UT actuator testbed with displacement delay

Table 2 Stability interval (shaded) and the NU root of closed-loop CE

τ_D (s)	RT	NU	ω (rad/s)	T
		0		
0.0154	1	2	75.1054	0.0087
0.0990	1	4	75.1054	0.0087
0.1827	1	6	75.1054	0.0087
⋮	⋮	⋮	⋮	⋮
0.6010	1	16	75.1054	0.0087
⋮	⋮	⋮	⋮	⋮

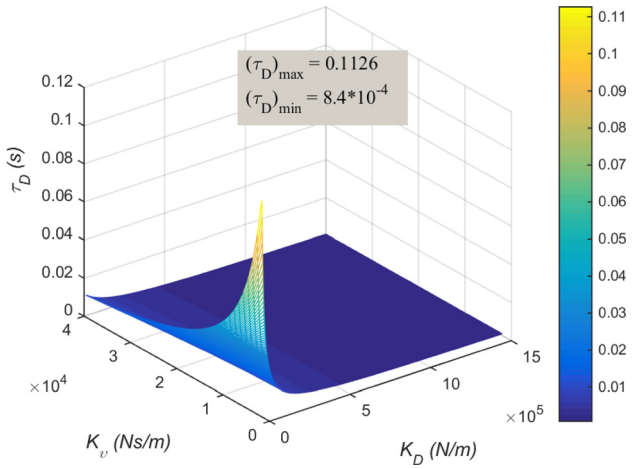


Fig. 6 Maximum tolerable latency of the system with filtering frequency of 1 Hz

Following the above schedules, the MTD with different (K_D, K_v) combinations is obtained under fixed filtering frequencies $f_v = 1, 25, 50$, and 150 Hz in Figs. 6, 7, 8, and 9, respectively. In our computations, we choose the P-controller gain $K_D \in [1.0 \times 10^4, 1.5 \times 10^6]$ and the D-controller gain $K_v \in [2.0 \times 10^3, 4.0 \times 10^4]$, both with step size 1000. There are two considerations when we choose these ranges for the controller gains: low gains are not able to generate meaningful output motion and high gains will saturate the current supply of the actuator's power source [50].

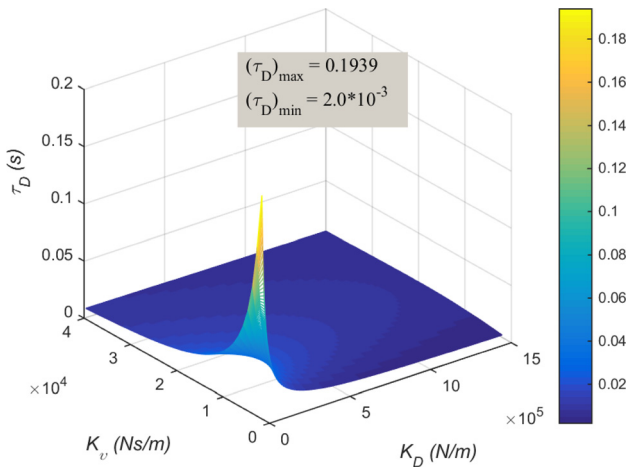


Fig. 7 Maximum tolerable latency of the system with filtering frequency of 25 Hz

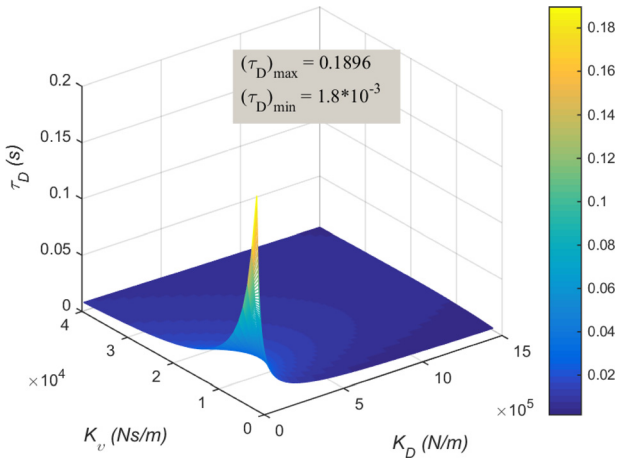


Fig. 8 Maximum tolerable latency of the system with filtering frequency of 50 Hz

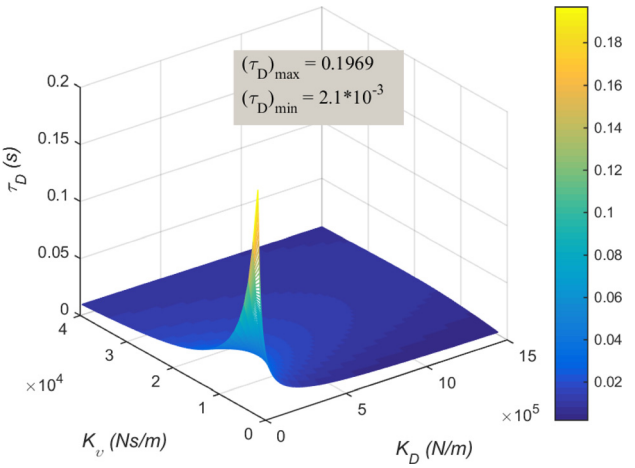


Fig. 9 Maximum tolerable latency of the system with filtering frequency of 150 Hz

All of these four figures show that the MTD can be achieved by choosing smaller K_D and K_v. With the increase of K_D, the MTD will decrease for all the cases for every fixed K_v; however, for the fixed K_D, if one increases the K_v, the MTD may increase, especially when the filtering frequency is large; some comparison examples for this case with different filtering frequencies can be found in Table 3. The delay margin [83] which is the smallest time delay required to make the system unstable is also calculated for comparison purpose in this table. In all the 12 examples, the

Table 3 Examples of MTD with different filtering frequencies

f_v (Hz)	$(K_D,\ K_v)$	MTD	Delay margin
1	$(6.35 \times 10^5,\ 3.0 \times 10^3)$	1.99×10^{-3}	2.0×10^{-3}
	$(6.35 \times 10^5,\ 1.7 \times 10^4)$	2.0×10^{-3}	2.0×10^{-3}
	$(6.35 \times 10^5,\ 3.4 \times 10^4)$	1.95×10^{-3}	2.0×10^{-3}
25	$(1.006 \times 10^6,\ 3.0 \times 10^3)$	3.51×10^{-3}	3.50×10^{-3}
	$(1.006 \times 10^6,\ 1.2 \times 10^4)$	7.17×10^{-3}	7.20×10^{-3}
	$(1.006 \times 10^6,\ 3.3 \times 10^4)$	6.0×10^{-3}	6.0×10^{-3}
50	$(5.15 \times 10^5,\ 1.6 \times 10^4)$	1.42×10^{-2}	1.42×10^{-2}
	$(5.15 \times 10^5,\ 2.3 \times 10^4)$	1.18×10^{-2}	1.18×10^{-2}
	$(5.15 \times 10^5,\ 9.0 \times 10^3)$	1.4×10^{-2}	1.4×10^{-2}
125	$(9.82 \times 10^5,\ 3.0 \times 10^3)$	4.2×10^{-3}	4.2×10^{-3}
	$(9.82 \times 10^5,\ 1.8 \times 10^4)$	1.2×10^{-2}	1.14×10^{-2}
	$(9.82 \times 10^5,\ 3.5 \times 10^4)$	8.9×10^{-3}	8.7×10^{-3}

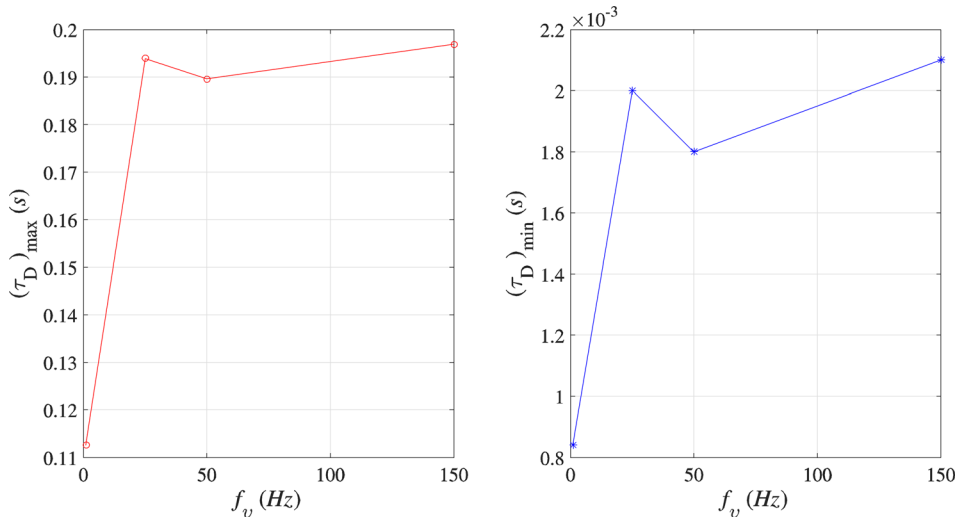


Fig. 10 Optimal tolerable latency of the system with different filtering frequencies

delay margins are equivalent to their MTDs since the system can only be stable in the first stable interval. The numerical results of the MTDs and the delay margins in Table 3 are in full agreement with each other with slight computational errors.

In addition to the above results, from Fig. 10, one also notices that with the increase of the filtering frequency, the maximum and minimum MTD of the system do not necessarily change monotonically. However, for the specific controller gain combination, the increase of filtering frequency will result in the increase of MTD as shown in Fig. 11, and this phenomenon is more evident when filtering frequencies are less than 60 Hz.

4.3 System Performance. In this section, the system performance of the UT actuator testbed with displacement-feedback delay will be discussed based on the delay selection rule proposed in Sec. 3. For simplicity in the remainder of this paper, we will refer to the optimal dominant root. The optimal dominant root of the system is the lowest-valued root of all rightmost roots for varying delays within each MTD interval. The optimal dominant root's corresponding delay is the optimal delay.

Figure 12 is the optimal dominant root of the system for different filtering frequencies using four different controller gain combinations. From Fig. 12, one notices that for the same controller gains, the optimal dominant root of the system will decrease with the increase of the filtering frequency. However, the optimal dominant root will not change too much when the filtering frequency becomes high and the controller gains are relatively small. This result and the stability result obtained in Sec. 4.2 demonstrate that increasing the filtering frequency improves both stability and system performance, especially when the filtering frequency is low.

Figure 13 shows the optimal dominant root of systems using four different filtering frequencies, i.e., $f_v = 5$, 50, 100, and 400 Hz, with controller gains in the ranges we are interested in. For all the four cases, if the P-controller gain is fixed, an increase in the D-controller gain will decrease the optimal dominant root first and then increase the optimal dominant root especially when the P-controller gain is small. For the fixed D-controller gain, increasing the P-controller gain will always decrease the dominant root when the filtering frequency is high; however, if the filtering frequency is low, increasing the P-controller gain will decrease the dominant root first and then increase the dominant root.

Another result one can draw is that for the low filtering frequency, the dominant root will arrive at optimal when both the controller gains are the largest; while for the high filtering frequency, this result does not hold anymore and this tells us that in the real application, increasing the controller gains does not necessarily improve the system performance.

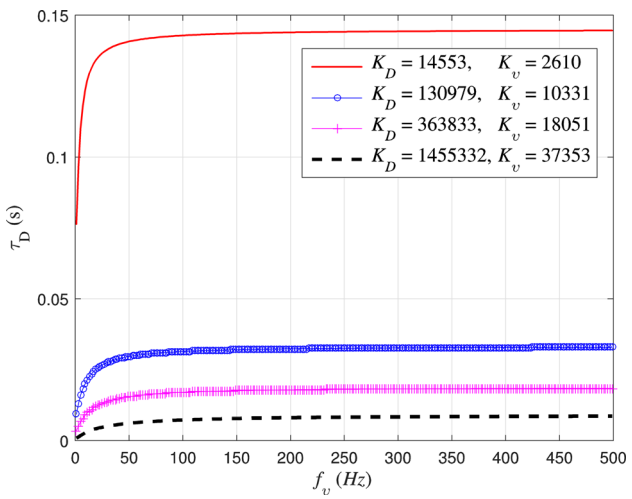


Fig. 11 Maximum tolerable latency of the system with different filtering frequencies

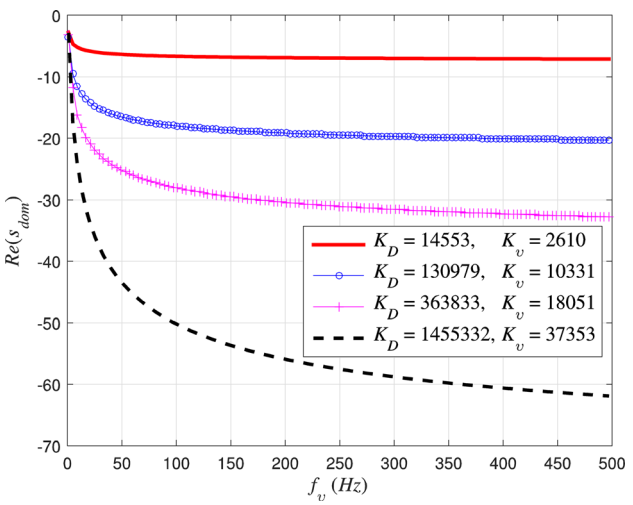


Fig. 12 Optimal dominant root of the system

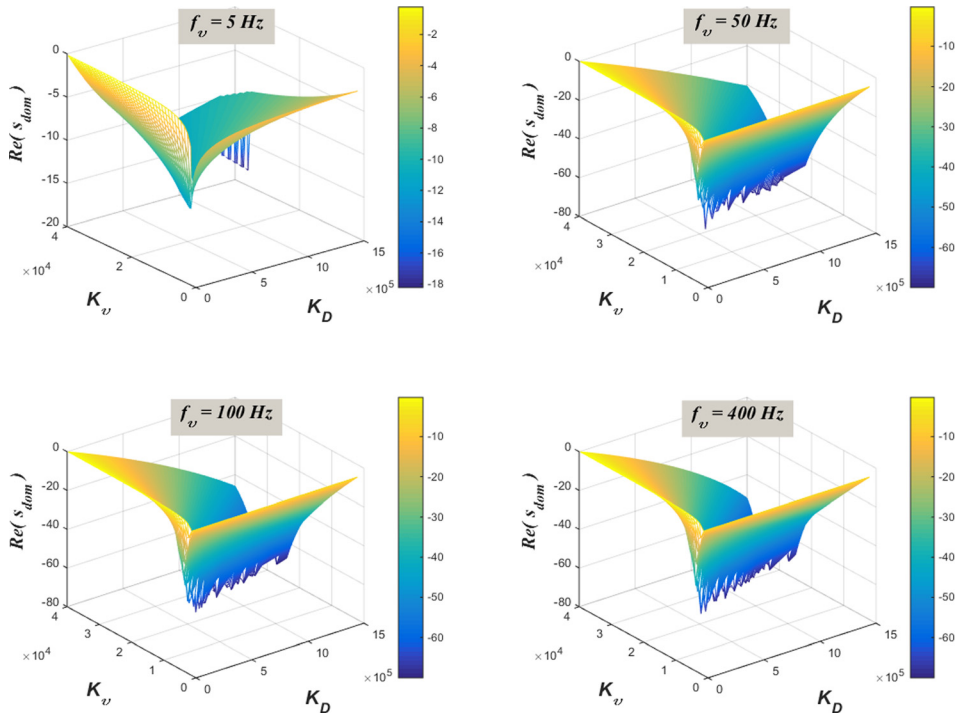


Fig. 13 Optimal dominant root of the system using different filtering frequencies

Figure 14 shows the corresponding optimal delay values when the systems in Fig. 13 have the optimal dominant roots. One may experience certain counterintuitive results regarding system performance. For example, optimal delay values are not necessarily 0, especially when P-controller gain is small for all the four cases. Also, the optimal dominant root may happen at $\tau_D > 0$ for some controller gain combinations with high filtering frequencies.

These are interesting results because one may improve the system performance by enlarging the delay of the system to certain time windows purposely without changing other parameters of the system.

Figures 15 and 16 are the step responses of the system with randomly chosen controller gains $(K_D, K_v) = (14, 553, 2610)$ and filtering frequencies $f_v = 50$ Hz and $f_v = 121$ Hz, respectively. Each figure shows the system's varying step response when there is applied no delay, optimal delay, shorter than optimal delay, and longer than optimal delay. From both figures, one can see that when the systems are delay-free, their responses are the worst. When the filtering frequency is 50 Hz, the system has the best performance for the latency 0.051 s, where the corresponding largest dominant root of the closed-loop CE of the system is −4.6832. For the case with filtering frequency of 121 Hz, the largest

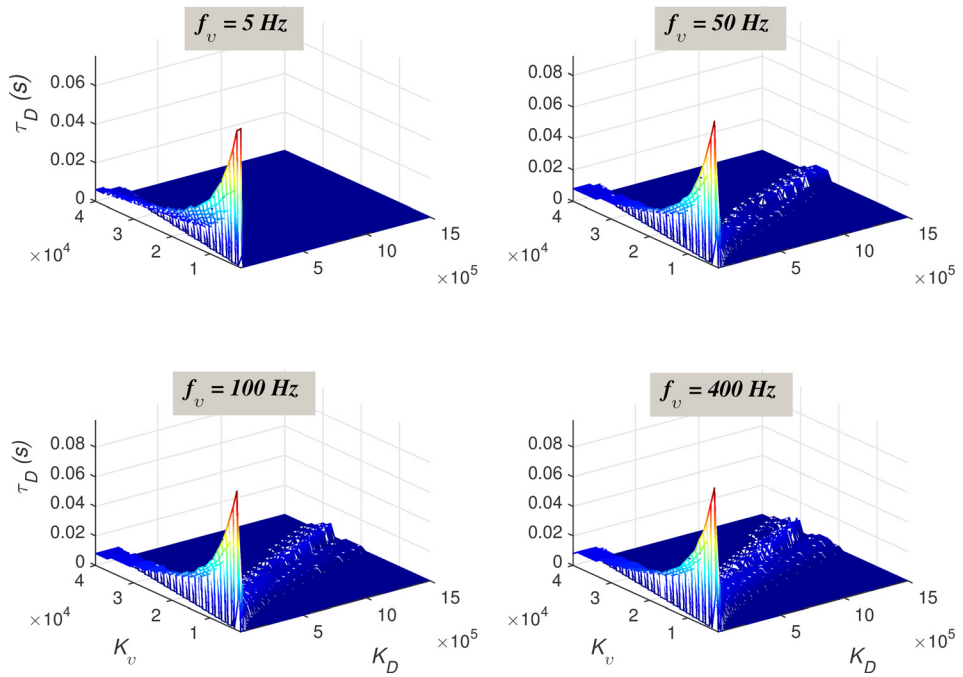


Fig. 14 Optimal delay of the system using different filtering frequencies

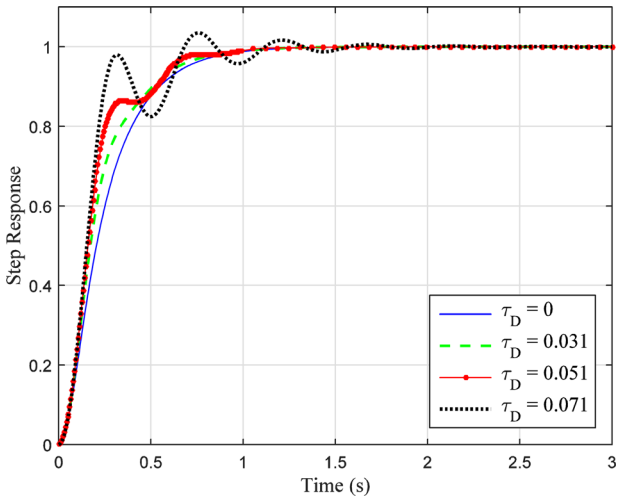


Fig. 15 Step response of the system using different latencies with $(K_D, K_v, f_v) = (14, 553, 2610, 50)$

dominant root of the closed-loop CE of the system is -4.6832 if the latency is 0.036 s; however, the optimal system response among the four cases is achieved when the latency is 0.056 s with the maximum dominant root -5.9963. Since -5.9963 is greater than -6.1244, the system with 0.036 s latency should have better performance than that of the system with 0.056 s latency, but Fig. 16 shows the opposite result. The reason for this result is because of the distribution of the root of the system which is depicted in Fig. 17.

From Fig. 17, one notices that even though -5.9963 is greater than -6.1244, the secondary dominant root of the system with latency $0.056s$ is $-11.4844 + 15.8276i$ which is much further away from the real axis than the one (i.e., $-25.5165 + 3.9178i$) with latency 0.036 s. The secondary dominant root with the imaginary part which is further away from the real axis will accelerate the response to achieve the peak value. Therefore, the second root of the system should be taken into consideration based on the required specifications. Further, these simulation results coincide with the delay selection rule proposed in Sec. 3.

5 Robotic Actuator Systems With Displacement and Velocity Feedback Delays

Following the same line as Sec. 4, in this section, we will discuss the influences of filtering frequency and controller gains on the stability of the UT actuator testbed with double delays. Based on the knowledge of stability map, the system performance will be discussed for different time delays within the stable region. However, due to the computational costs we will only choose some representative parameter sets which are usually used on the UT actuator testbed to discuss its property.

5.1 System Model. If the velocity feedback (low-level control) is achieved through an embedded damping servo loop and the displacement feedback (high-level) is accomplished by an embedded stiffness servo loop, then the UT actuator testbed possesses distributed control architecture. Figure 18 is the block diagram for this system with velocity-feedback delay τ_v and displacement-feedback delay τ_D.

If all the other parameters are the same as single-delay in the UT actuator testbed, the closed-loop transfer function for this system is

$$G_{cl}^{\text{two-delay}}(s) = \frac{K_v s + K_D}{ms^2 + (b + K_v Q_v(s)e^{-\tau_v s})s + K_D e^{-\tau_D s}} = \frac{K_v s^2 + (2\pi f_v K_v + K_D)s + 2\pi f_v K_D}{ms^3 + (2\pi f_v m + b)s^2 + 2\pi f_v bs + 2\pi f_v K_v s e^{-\tau_v s} + (K_D s + 2\pi f_v K_D)e^{-\tau_D s}} \tag{35}$$

It is clear that if $\tau_v = \tau_D$, then the double-delay system is equivalent to the single-delay system described in Fig. 5.

5.2 Maximum Tolerable Latency Analysis. In this section, we will first use an example to show how to obtain the stability map for the UT actuator testbed using specific controller gains. The influences of filtering frequency and controller gains on the stability of the system will be examined thereafter.

If we choose $(K_D, K_v) = (14, 553, 2610)$ and filtering frequency $f_v = 50$ Hz, then the CE of the system is

$$\begin{aligned} \text{CE}(s, \tau_v, \tau_D) = {} & 256s^3 + 81634s^2 + 392500s \\ & + (14553s + 4569642)e^{-\tau_D s} \\ & + 819540se^{-\tau_v s} \end{aligned} \tag{36}$$

the roots of the *CE* without any delay are -6.38, -9.22, and -303.28, all of which are on the left-half plane, therefore the system is stable if there is no delay.

By applying two Rekasius substitutions: $e^{-\tau_D s} = (1 - T_D s)/(1 + T_D s)$ and $e^{-\tau_v s} = (1 - T_v s)/(1 + T_v s)$ with $T_D, T_v \in \mathbb{R}$ in Eq. (36), one obtains

$$\begin{aligned} \overline{\overline{\text{CE}}}(s, T_v, T_D) = {} & 256T_v T_D s^5 + (256T_v + 256T_D + 81,634T_v T_D)s^4 \\ & + (81,634T_v + 81,634T_D - 441,593T_v T_D + 256)s^3 \\ & + (-412,487T_D + 1,197,487T_v - 4,569,642T_v T_D \\ & + 81,634)s^2 + (4,569,642T_v - 4,569,642T_D \\ & + 1,226,593)s + 4,569,642 \end{aligned} \tag{37}$$

Implementing half-angle substitutions $T_v \triangleq \tan(\tau_v \omega/2)/\omega$ and $T_D \triangleq \tan(\tau_D \omega/2)/\omega$ in Eq. (37), a new equation, which is a function of the two delays and ω, can be obtained, i.e.,

$$\overline{\overline{\text{CE}}}(\omega, \tau_v, \tau_D) = \text{Re}\left\{\overline{\overline{\text{CE}}}(\omega, \tau_v, \tau_D)\right\} + i \cdot \text{Im}\left\{\overline{\overline{\text{CE}}}(\omega, \tau_v, \tau_D)\right\} \tag{38}$$

Notice that Eq. (38) does not contain any transcendental terms, and it is equivalent to Eq. (36) only if $s = \omega i$. The solution for Eq. (38) can be obtained by using the Sylvester method where $\tau_v \omega$ changes from 0 to 2π.

The graphical solution for Eq. (38) is shown in Fig. 19, and the stability property of this system is totally determined by this

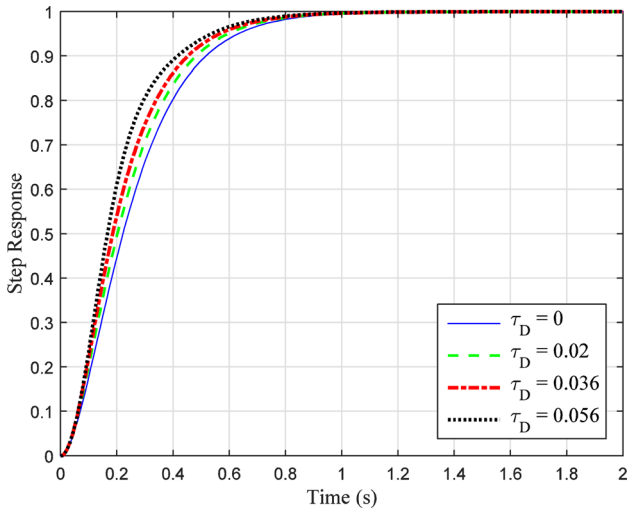


Fig. 16 Step response of the system using different latencies with $(K_D, K_v, f_v) = (14, 553, 2610, 121)$

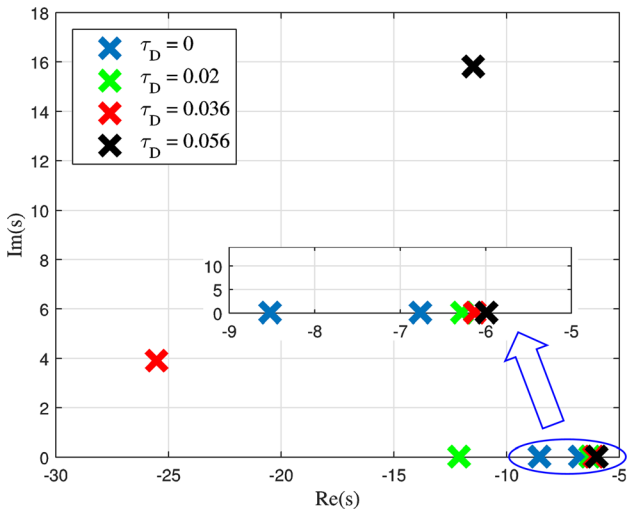


Fig. 17 Root distribution of the system using different latencies with $(K_D, K_v, f_v) = (14, 553, 2610, 121)$

solution. Figure 20 is a two-dimensional visualization of Fig. 19, where the kernel of the delays is shown. The corresponding crossing frequencies and root tendencies with respect to τ_v (RT_{τ_v}) and τ_D (RT_{τ_D}) of the ten points in Fig. 20 are listed in Table 4. From Fig. 20, the following conclusions can be made:

(1) Any delay combination (τ_v, τ_D) which is slightly outside (inside) the region confined by the axes and arc ABCD will bring about two unstable (stable) roots to the CE of the closed-loop system.
(2) Any delay combination (τ_v, τ_D) which is slightly on the lower (upper) side of the arc EFG will bring about two stable (unstable) roots to the CE of the closed-loop system.
(3) Any delay combination (τ_v, τ_D) which is slightly on the lower (upper) side of the arc HIJ will bring two stable (unstable) roots to the CE of the closed-loop system.

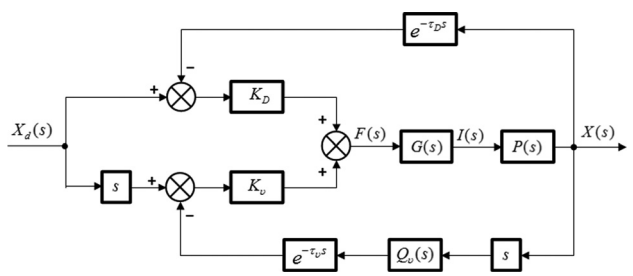


Fig. 18 UT actuator testbed with displacement and velocity delays

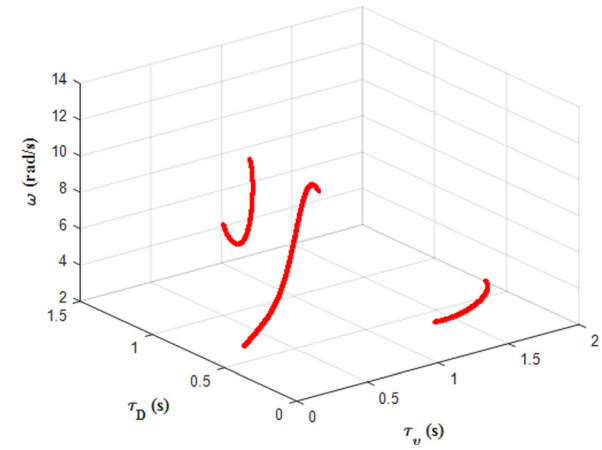


Fig. 19 Three-dimensional graphic solution for the two-delay system with $(K_D, K_v, f_v) = (14, 553, 2610, 50)$

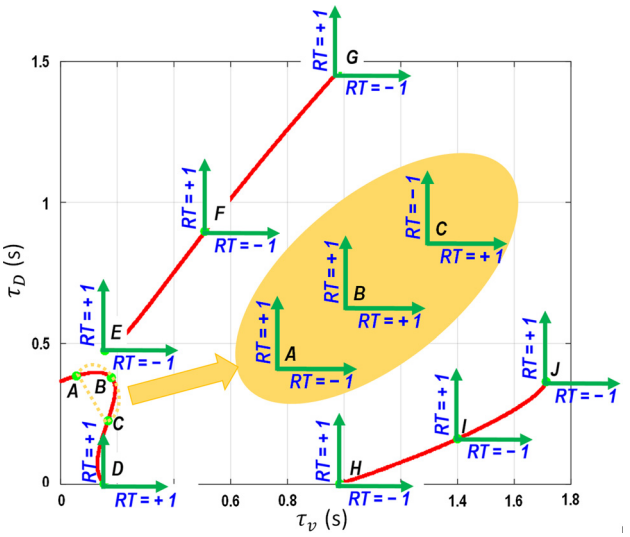


Fig. 20 The kernel and its root tendencies of the two-delay system with $(K_D, K_v, f_v) = (14, 553, 2610, 50)$

Table 4 Detailed information of the points on the kernel

Location	(τ_v, τ_D)	ω (rad/s)	RT_{τ_v}	RT_{τ_D}
A	(0.0557, 0.3845)	3.8149	−1	+1
B	(0.1801, 0.3781)	4.8555	+1	+1
C	(0.1673, 0.2260)	8.1538	+1	−1
D	(0.1533, 0.0)	13.235	+1	+1
E	(0.1569, 0.4744)	13.1197	−1	+1
F	(0.5101, 0.8966)	6.3121	−1	+1
G	(0.9770, 1.4620)	4.2951	−1	+1
H	(0.9841, 0.0019)	4.2791	−1	+1
I	(1.40, 0.1602)	3.6244	−1	+1
J	(1.7120, 0.3649)	3.670	−1	+1

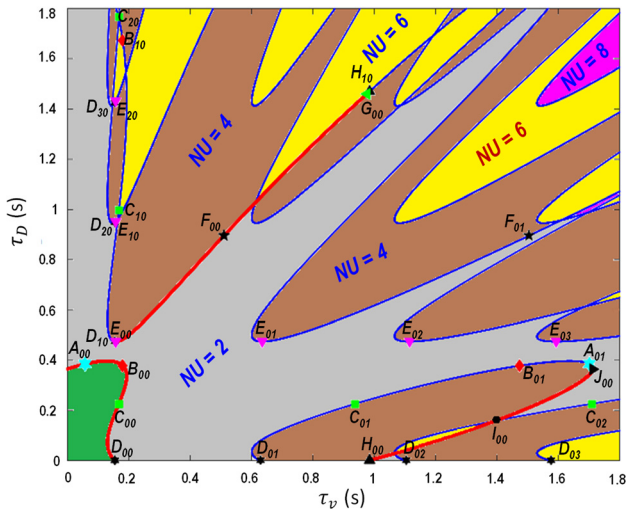


Fig. 21 Stability map and the NU roots for the two-delay system with $(K_D, K_v, f_v) = (14, 553, 2610, 50)$

Since the delay-free system is stable, i.e., $\mathrm{NU}(0,0) = 0$, from the first conclusion above, one knows that the system will be stable wherever the delay combination (τ_v, τ_D) is inside this region, which we name as main stable area (MSA), i.e., $\mathrm{NU}(\tau_v, \tau_D) = 0$, for $(\tau_v, \tau_D) \in$ MSA. The MSA in this example is equivalent to the MTDR of the first stable region. In order to obtain all the stability regions, i.e., stability map of the system, one needs to use both kernel and offspring information.

Figure 21 is the stability map for this system using this specific parameter set within region 1.8×1.8. The ten points on the *kernel* of stability region in Fig. 20 are redrawn in this figure with different notations, i.e., every point is attached with subscript 00 to denote the kernel, and its offsprings with respect to τ_v and τ_D axes are denoted as $01, 02, \ldots$ and $10, 20, \ldots$, respectively. Take the point $C_{00} = (0.16773, 0.226)$ with crossing frequency $\omega = 8.1538\mathrm{rad/s}$ on the kernel as an example, its first offsprings with respect to τ_v and τ_D axes are $C_{01} = (0.5526, 0.226)$ and $C_{10} = (0.1673, 0.9966)$, the second offsprings are $C_{02} = (0.9379, 0.226)$ and $C_{20} = (0.1673, 1.7672)$. Note that C_{00} and all of its offsprings are sharing the same crossing frequency. Since the delay-free system is stable, i.e., $\mathrm{NU}(0,0) = 0$, the NU roots only change on the kernel and its offsprings, $\mathrm{NU}(\tau_v, \tau_D) = 0$ until

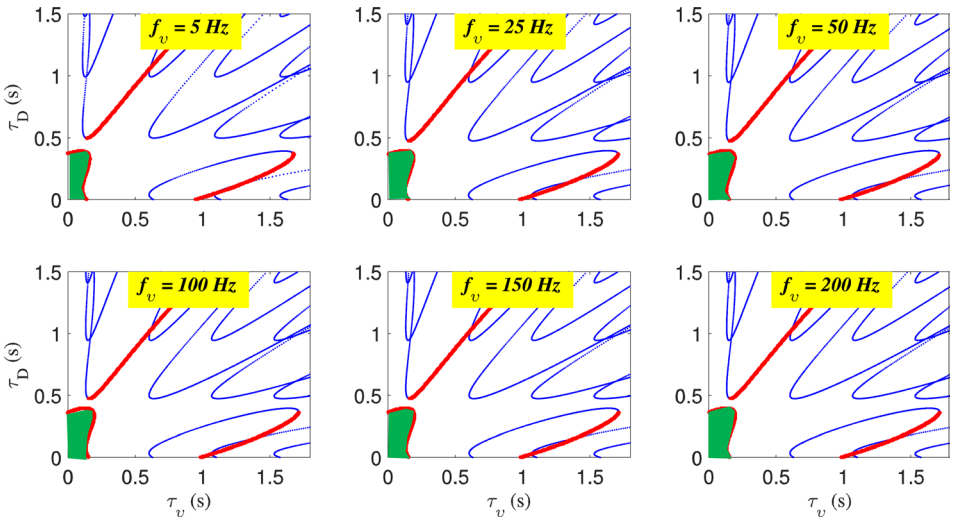


Fig. 22 Stability maps for the system with different filtering frequencies with controller gains $(K_D, K_v) = (14, 553, 2610)$

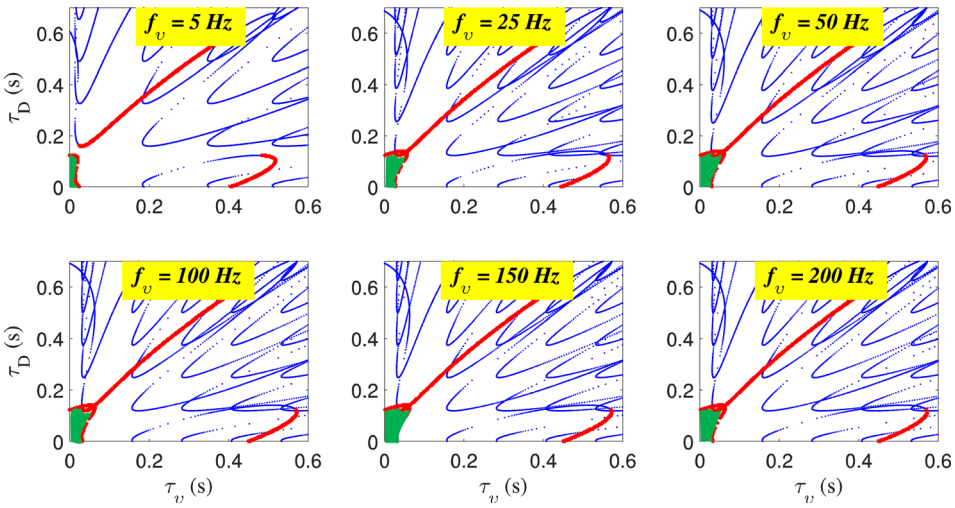


Fig. 23 Stability maps for the system with different filtering frequencies with controller gains $(K_D, K_v) = (130, 979, 10, 331)$

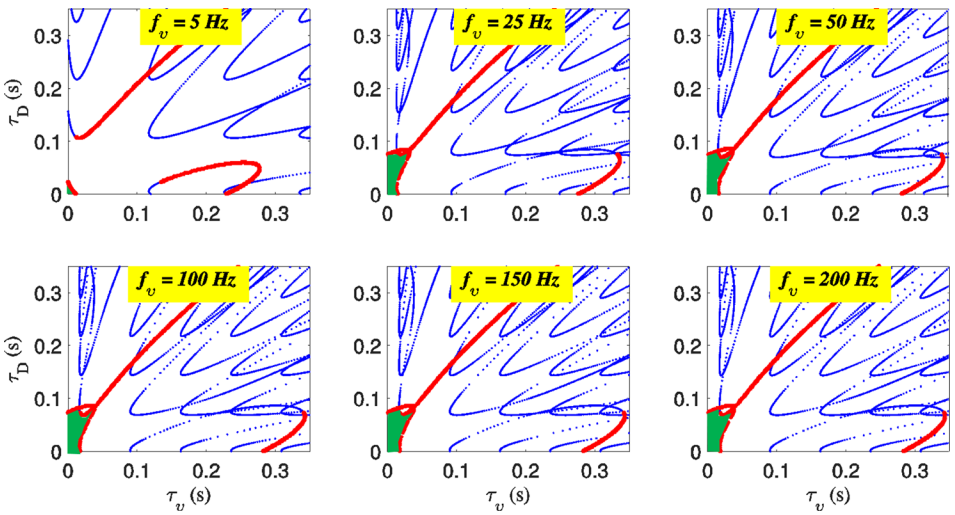


Fig. 24 Stability maps for the system with different filtering frequencies with controller gains $(K_D, K_v) = (363, 833, 18, 051)$

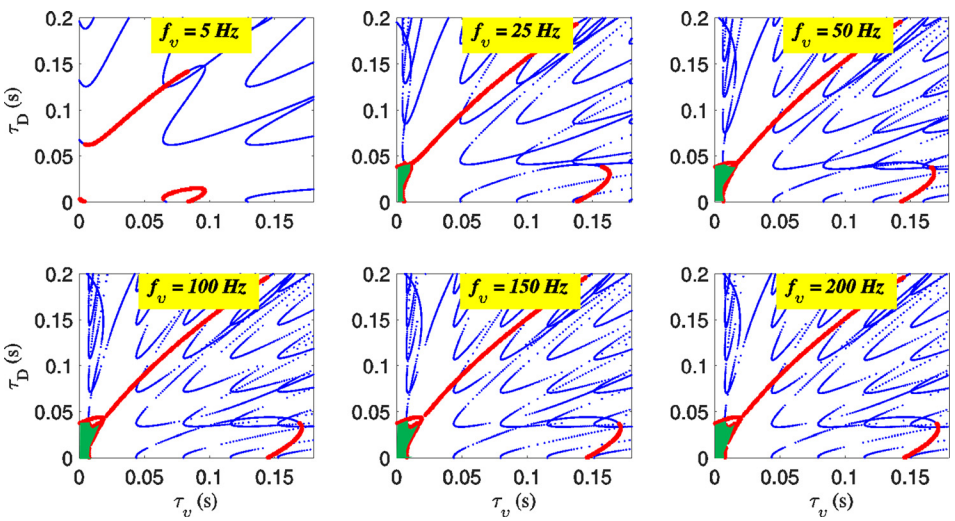


Fig. 25 Stability maps for the system with different filtering frequencies with controller gains $(K_D, K_v) = (1, 455, 332, 37, 353)$

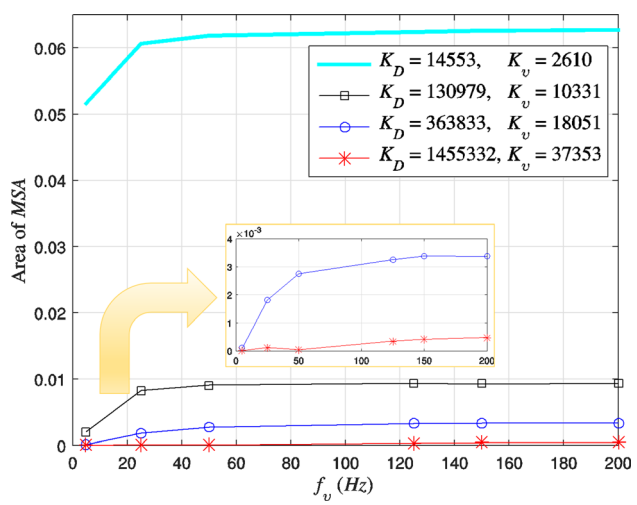


Fig. 26 Area of MSA of the system for varying filtering frequency with different control gain combinations

(τ_v, τ_D) arrives at its smallest kernel. With the information of points on the kernel and their root tendencies, the stability map can be obtained through the following procedures.

We start from the region where τ_v and τ_D have small values, and increasing τ_v while keeping τ_D fixed (the same analysis can be done for the case of changing τ_D while fixing τ_v) until the arc $B_{00}C_{00}D_{00}$ is met. Increasing τ_v such that (τ_v, τ_D) crosses arc $B_{00}C_{00}D_{00}$ from left to right, because $\mathrm{RT}_{\tau_v} = +1$ on the arc $B_{00}C_{00}D_{00}$, there will be two unstable roots brought by this crossing, and $\mathrm{NU}(\tau_v, \tau_D) = 2$ immediately after this increase of τ_v. In the same way, one knows that $\mathrm{NU}(\tau_v, \tau_D) = 2$ for all the region right outside MSA. We continue increasing τ_v while keeping τ_D fixed until we meet arc $D_{01}C_{01}B_{01}A_{01}$. Since the root tendencies of (τ_v, τ_D) on this arc are all $\mathrm{RT}_{\tau_v} = +1$, the $\mathrm{NU}(\tau_v, \tau_D) = 4$ for any (τ_v, τ_D) which crosses this arc from left to right before crossing any other arc. For example, when the (τ_v, τ_D) crosses arc $I_{00}J_{00}$ from left to right with $\mathrm{RT}_{\tau_v} = -1$ and $\mathrm{RT}_{\tau_D} = +1$, then $\mathrm{NU}(\tau_v, \tau_D) = 4$ should be changed into $\mathrm{NU}(\tau_v, \tau_D) = 2$. We continue doing the same processes to obtain $\mathrm{NU}(\tau_v, \tau_D)$ until all of the crossing arcs have been crossed, then the stability regions for the system will be the regions where $\mathrm{NU}(\tau_v, \tau_D) = 0$.

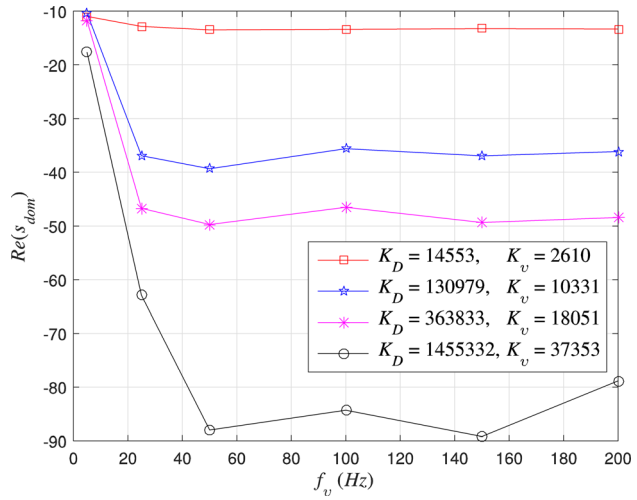


Fig. 27 Relationship between the optimal dominant root and filtering frequency

The detailed information of $\mathrm{NU}(\tau_v, \tau_D)$ is shown in Fig. 21 where the arguments (τ_v, τ_D) are omitted. From the figure, one knows that the system is stable only in the region shadowed with green color (gray for $\mathrm{NU}(\tau_v, \tau_D) = 2$, brown for $\mathrm{NU}(\tau_v, \tau_D) = 4$, yellow for $\mathrm{NU}(\tau_v, \tau_D) = 6$, and magenta for $\mathrm{NU}(\tau_v, \tau_D) = 8$). From the shape of MSA, one notices that increasing τ_D or τ_v may stabilize the system as are the cases for $C_{00} = (0.16773, 0.226)$ and $A_{00} = (0.0557, 0.3845)$.

Increasing the delay to stabilize the system is counterintuitive but one of the application advantages.

Figures 22–25 are stability maps for the system using six different filtering frequencies (i.e., f_v are 5, 25, 50, 100, 150, and 200 Hz) with four different controller gain combinations of $(K_D, K_v) = (14, 553, 2610)$, $(K_D, K_v) = (130, 979, 10, 331)$, $(K_D, K_v) = (363, 833, 18, 051)$, and $(K_D, K_v) = (1, 455, 332, 37, 353)$. In all of the figures, the shaded regions are the regions with property of MSA, i.e., $\mathrm{NU}(\tau_v, \tau_D) = 0$. The following conclusions can be made based on the information of the figures:

(1) For the same frequency, with the increase of controller gains (K_D, K_v), the stability region will decrease.
(2) For the small controller gains (K_D, K_v), the filtering frequency does not have much influence on the system stability as can be seen in Fig. 22.
(3) For the larger controller gains (K_D, K_v), with the increase of filtering frequency, the stability region will increase especially when the filtering frequency is less than 50 Hz. However, once the filtering frequency is greater than 50 Hz, its influence on stability is not evident.

More quantitative view of the relationship between filtering frequency and the area of MSA for different controller gain combinations is shown in Fig. 26.

The above conclusions tell us that in the real application, if one chooses the large controller gains, the filtering frequency should not be too low, otherwise the system will most likely be unstable (see Figs. 24 and 25). In addition, if a low filtering frequency is chosen, then the controller gains should also be small to guarantee the stability of the system (see $f_v = 5$ Hz in the figures).

5.3 System Performance. In this section, the performance of the system using different controller gains and filtering frequencies will be discussed. As in Sec. 4.3, the optimal dominant root of the system refers to the lowest-valued root of all the rightmost roots for varying time delays of τ_v and τ_D within each MTDR. The optimal dominant root's corresponding delay combination (τ_v, τ_D) is the optimal delay combination.

Figure 27 shows the relationship between the optimal dominant root and filtering frequency. For each fixed controller gains (K_D, K_v), with the increase of filtering frequency, the optimal dominant root of the CE will decrease and this is more evident when the filtering frequency is low. However, when the filtering frequency is high, the optimal dominant root of the CE does not change too much. From the figure, one also notices that for each fixed filtering frequency, with the increase of controller gains

Table 5 Dominant root of the CE of the system

(K_v, K_D, f_v)	$(\tau_v, \tau_D, \mathrm{Re}(s_{\mathrm{dom}}))^{\mathrm{opt}}$ $(\tau_v, \tau_D, \mathrm{Re}(s_{\mathrm{dom}}))$	(K_v, K_D, f_v)	$(\tau_v, \tau_D, \mathrm{Re}(s_{\mathrm{dom}}))^{\mathrm{opt}}$ $(\tau_v, \tau_D, \mathrm{Re}(s_{\mathrm{dom}}))$
(14,553, 2610, 5)	(0.0, 0.056, −10.9730) (0.0, 0.0, −4.6625)	(130,979, 10,331, 5)	(0.0, 0.010, −10.3664) (0.0, 0.0, −9.586)
(14,553, 2610, 25)	(0.048, 0.082, −12.8558) (0.0, 0.0, −5.9871)	(130,979, 10,331, 25)	(0.008, 0.028, −36.8974) (0.0, 0.0, −14.7601)
(14,553, 2610, 50)	(0.048, 0.076, −13.4801) (0.0, 0.0, −6.3819)	(130,979, 10,331, 50)	(0.014, 0.032, −39.3525) (0.0, 0.0, −16.5324)
(14,553, 2610, 100)	(0.052, 0.080, −13.3787) (0.0, 0.0, −6.6885)	(130,979, 10,331, 100)	(0.016, 0.032, −35.6223) (0.0, 0.0, −17.9935)
(14,553, 2610, 150)	(0.05, 0.076, −13.2613) (0.0, 0.0, −6.8330)	(130,979, 10,331, 150)	(0.014, 0.028, −36.9463) (0.0, 0.0, −18.710)
(14,553, 2610, 200)	(0.054, 0.082, −13.322) (0.0, 0.0, −6.9221)	(130,979, 10,331, 200)	(0.018, 0.034, −36.1339) (0.0, 0.0, −19.1606)
(K_v, K_D, f_v)	$(\tau_v, \tau_D, \mathrm{Re}(s_{\mathrm{dom}}))^{\mathrm{opt}}$ $(\tau_v, \tau_D, \mathrm{Re}(s_{\mathrm{dom}}))$	(K_v, K_D, f_v)	$(\tau_v, \tau_D, \mathrm{Re}(s_{\mathrm{dom}}))^{\mathrm{opt}}$ $(\tau_v, \tau_D, \mathrm{Re}(s_{\mathrm{dom}}))$
(363,833, 18,051, 5)	(0.0, 0.0, −11.7509) (0.0, 0.0, −11.7509)	(1,455,332, 37,353, 5)	(0.0, 0.0, −17.695) (0.0, 0.0, −9.3018)
(363,833, 18,051, 25)	(0.0, 0.014, −46.6728) (0.0, 0.0, −21.8915)	(1,455,332, 37,353, 25)	(0.002, 0.0140, −62.9212) (0.0, 0.0, −35.9471)
(363,833, 18,051, 50)	(0.008, 0.022, −49.7102) (0.0, 0.0, −25.2359)	(1,455,332, 37,353, 50)	(0.0, 0.008, −88.0004) (0.0, 0.0, −43.4786)
(363,833, 18,051, 100)	(0.006, 0.016, −46.4943) (0.0, 0.0, −28.0802)	(1,455,332, 37,353, 100)	(0.004, 0.010, −84.2912) (0.0, 0.0, −50.1882)
(363,833, 18,051, 150)	(0.010, 0.022, −49.3312) (0.0, 0.0, −29.5058)	(1,455,332, 37,353, 150)	(0.02, 0.032, −89.2236) (0.0, 0.0, −53.6608)
(363,833, 18,051, 200)	(0.008 0.018, −48.4033) (0.0, 0.0, −30.4137)	(1,455,332, 37,353, 200)	(0.0, 0.020, −78.8183) (0.0, 0.0, −55.9124)

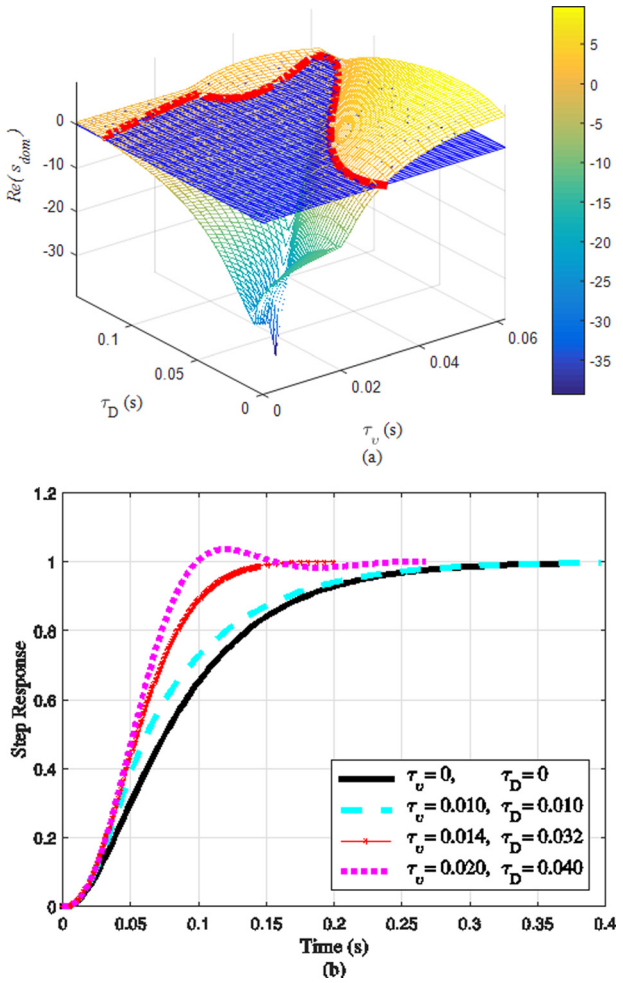


Fig. 28 Dominant root and step responses of the system with $(K_D, K_v, f_v) = (130, 979, 10, 331, 50)$

(K_D, K_v), the optimal dominant root of the CE also decreases, especially when the filtering frequency is high.

These two results coupled with the stability results obtained in Sec. 5.2 give us some guidance during our controller design which indicates that increasing the controller gains will decrease the stability region, but can improve the system performance; however, increasing the filtering frequency not only increases the stability region but also improves the system performance, especially when the filtering frequency is low.

Table 5 gives the detailed information of the dominant root of the CE of the system using different controller gains and filtering frequencies. The optimal delay combination (τ_v, τ_D) and the corresponding optimal dominant root $\mathrm{Re}(s_{\mathrm{dom}})$ are denoted as $(\tau_v, \tau_D, \mathrm{Re}(s_{\mathrm{dom}}))^{\mathrm{opt}}$. From the table, one can see that for most of

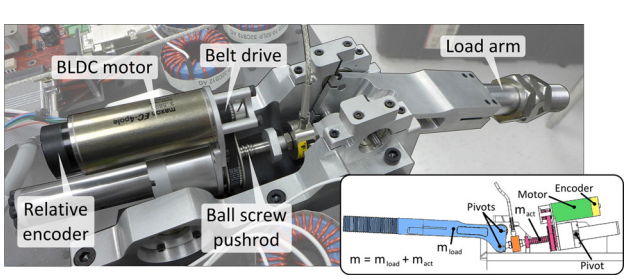


Fig. 29 UT actuator testbed. A motor drives a ball screw via a belt reduction. A sensor measures the linear displacement of the actuator.

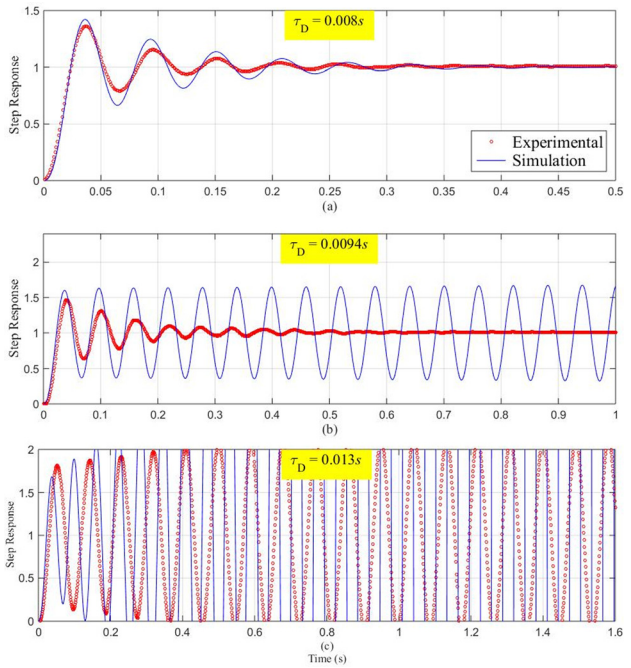


Fig. 30 Stability of the system under different displacement-feedback delays with $(K_D, K_v, f_v) = (1, 290, 000, 22, 000, 100)$

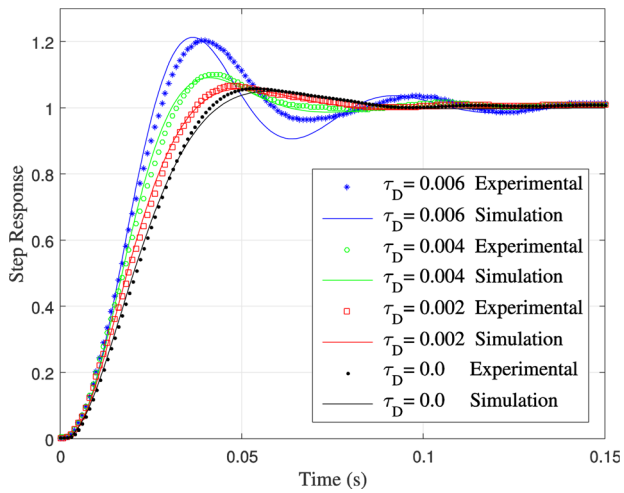


Fig. 31 Step responses of the system under different displacement-feedback delays with $(K_D, K_v, f_v) = (1, 290, 000, 22, 000, 100)$

the cases, the optimal dominant root happens when the delay combination $(\tau_v, \tau_D) \neq (0, 0)$ which is counterintuitive and implies that enlarging the delay may not only increase the stability but also improve the system performance.

In the following, one example with parameter set $(K_D, K_v, f_v) = (130, 979, 10, 331, 50)$ is given to show the system

Table 6 Dominant root of the system with displacement delay

τ_D (s)	Root
0	$-53.2479 + 56.3548i$
0.002	$-60.1953 + 67.5279i$
0.004	$-60.8563 + 95.4715i$
0.006	$-32.71 + 112.36i$; -100.84
0.008	$-10.35 + 109.29i$; -85.7

performance with different delay combinations (see Fig. 28). Figure 28(*a*) shows the dominant roots of the CE for different delays, and the optimal delay combination is $(\tau_D, \tau_v) = (0.014, 0.032)$ which results in the optimal dominant root $-39.3525 + 25.2292i$. Figure 28(*b*) is the step response for this system with different delays. It is clear that the system has the worst performance when it is delay-free. When $(\tau_D, \tau_v) = (0.014, 0.032)$, the system has the shortest settling time of 0.102 s, which coincides with the dominant root result. For the case of $(\tau_D, \tau_v) = (0.020, 0.040)$, which is within the stable region but slightly larger than the optimal delay combination, although it has the shortest rising time, it oscillates and takes longer to enter the error band.

Generally speaking, for many systems, increasing the system delay will most likely deteriorate system performance. However, there might be some areas within which the system performance can be improved by increasing of system delay. The reason for this counterintuitive phenomenon is that the delay within this area will push the roots of the CE further away from the imaginary axis, and this, in return, results in a better system performance.

6 Experimental Validation

In this section, we will test the stability and performance results obtained in Secs. 4 and 5 on the UT actuator testbed (see Fig. 29). We will first test the stability and performance of the system with displacement-feedback delay and then test the stability and performance of the system using both displacement-feedback and velocity-feedback delays. Since it is hard to generate an impulse signal in our testbed, to make the experimental result and simulated result comparable, the numerators of the close-loop transfer function for both the single and double delays in Eqs. (30) and (35) will be replaced by $2\pi f_v K_D$ in our simulation.

We measured the actuator's parameters in the following ways. Actuator inertia was measured by observing the relationship between force and velocity for a position tracking chirp test. An inertia model was fit to the measured data using the transfer function $V(s)/F(s) = 1/\text{ms}$. To measure damping, we analyzed the position response of a proportional position feedback controller. The fact that a proportional controller was stable indicated the presence of damping. To quantify the amount of damping and with the actuator inertia known, we measured the overshoot of a step response and fit the actuator damping parameter to the experimental data.

6.1 Experiments With Displacement-Feedback Delay. For the case when the UT actuator testbed uses only displacement-feedback delay, we choose the filtering frequency $f_v = 100$ Hz and controller gain combination $(K_D, K_v) = (1, 290, 000, 22, 000)$. Following the same procedures in Sec. 4.2, we know that the maximum tolerable latency for the system is 0.0094 s.

Figure 30 shows the step responses of the system with different displacement delays. In Fig. 30(*a*), since the delay $\tau_D = 0.008 < 0.0094$, the system is stable and the simulated result coincides very well with the experimental result.

Figure 30(*c*) illustrates the scenario where the system is unstable due to the delay greater than the MTD.

Figure 30(*b*) shows the marginal stable case since the system delay $\tau_D = 0.0094$ is equivalent to its maximum tolerable latency. While the experimental result shows the convergence of the system, this is contradictory to the theoretical result. If the estimated model is accurate enough, another reason for this contradictory result could be due to the sensing precision and sensing speed of the UT actuator testbed. The actuator testbed we used has a cycle time of 1 ms. Delays were simulated by implementing a first-in first-out (FIFO) data structure and passing sensor data through this FIFO before feeding the data to the feedback control loop. The length of this FIFO is proportional to the simulated delay value. For simulating larger delays, the 1 ms cycle time places a restriction on the minimum step size of the simulated delay value. Therefore, the actual delay used in this experiment is 0.009 s instead of 0.0094 s. Since 0.009 is less than the maximum tolerable latency, the system is stable which agrees with the theoretical result.

Figure 31 shows the simulated and experimental results of step responses of the system using different displacement delays. Since the system is stable as long as the displacement-feedback delay is less than the maximum tolerable latency 0.0094 s, we try to find the optimal delay within this stable interval. The characteristic roots of the system with different delays that have a real part greater than -150 are listed in Table 6. Based on the delay selection rule proposed in Sec. 3, one knows that the optimal delay of the system should be 0.004 s which will result in a settling time of 0.0657 s. For this experiment, one also notices that if the delay of the system is 0.002 s, then the system would have the settling time 0.0665 s, which is almost the same as when the delay is 0.004 s. However, since the imaginary part of the dominant root of the system with delay 0.004 s is 95.4715, which is greater than 67.5279 (the imaginary part of the dominant root of the system with delay 0.002 s), the system with latency 0.004 s should also have a shorter rising time than the system with latency 0.002 s and therefore has the best overall performance. One can also notice that the delay-free system has the longest rising time due to the relatively small value of the imaginary part of its characteristic root. From

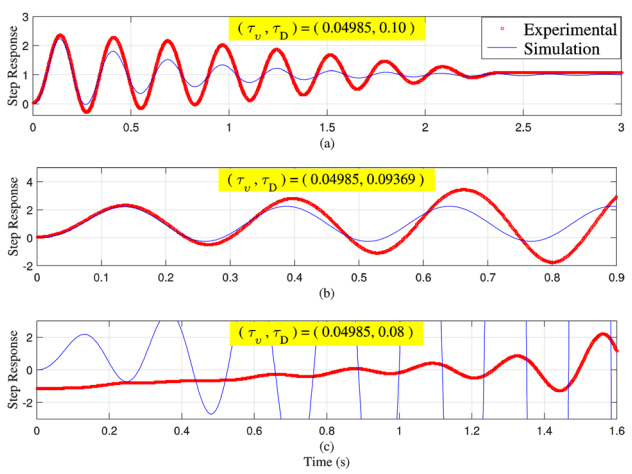


Fig. 32 Stability of the system under different displacement and velocity feedback delays with $(K_D, K_v, f_v) = (130, 979, 10, 331, 50)$

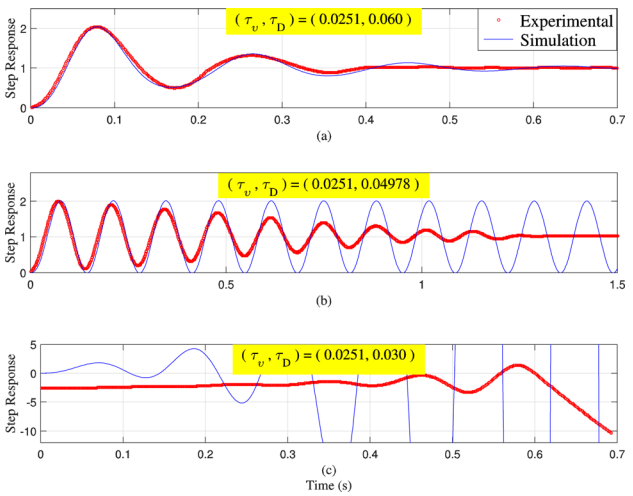
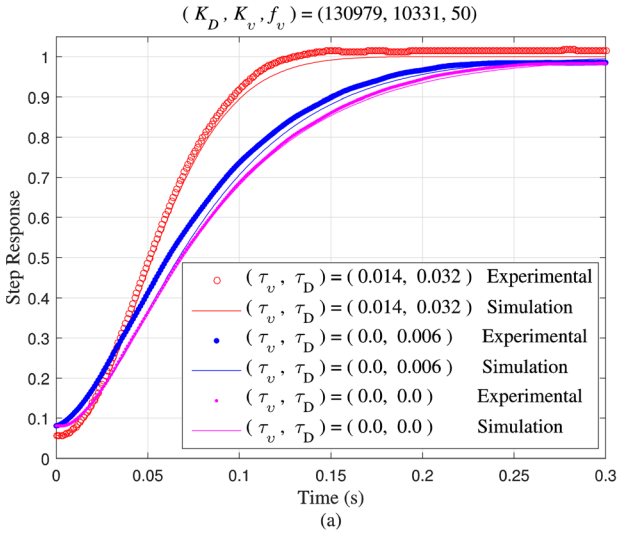
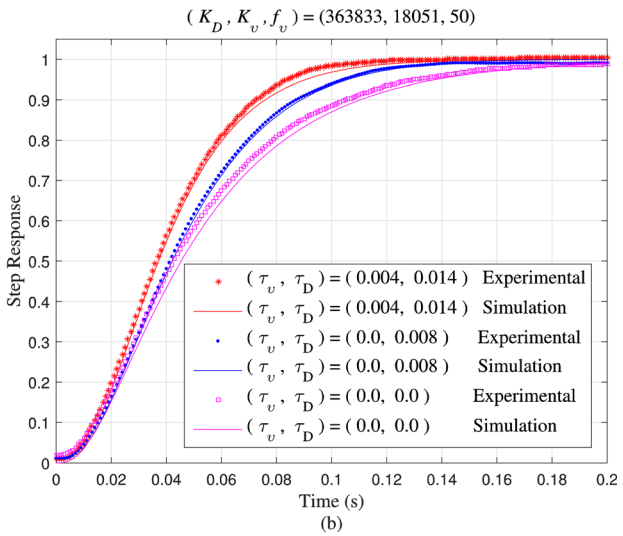


Fig. 33 Stability of the system under different displacement and velocity feedback delays with $(K_D, K_v, f_v) = (363, 833, 18, 051, 50)$



(a)



(b)

Fig. 34 Step responses of the system under different latency combinations with $f_v = 50$ Hz

the figure, it is obvious that the simulated results and the experimental results agree with each other.

6.2 Experiments With Displacement and Velocity Feedback Delays. For the UT actuator testbed with both displacement and velocity feedback delays, we will run the experiments using different system parameter sets but with the same filtering frequency, $f_v = 50$ Hz.

Figures 32 and 33 are the step responses of the system with different delay combinations using control gains $(K_D, K_v) = (130, 979, 10, 331)$ and $(K_D, K_v) = (363, 833, 18, 051)$, respectively. In each of the figures, both the simulated and experimental data are plotted for comparison purpose. Figures 32(*a*) and 33(*a*) illustrations show the situation in which the system is stable, and Figs. 32(*b*) and 33(*b*) show the scenario where the system should be marginally stable, while Figs. 32(*c*) and 33(*c*) demonstrate the case when the system is unstable.

Following the same procedures described in Sec. 5.2, one knows that the system with control gains $(K_D, K_v) = (130, 979, 10, 331)$ is stable when delay combination is $(\tau_v, \tau_D) = (0.04985, 0.10)$, and the system with control gains $(K_D, K_v) = (363, 833, 18, 051)$, delay combination $(\tau_v, \tau_D) = (0.0251, 0.060)$ is also stable. Both Figs. 32(*a*) and 33(*a*) verified this assertion.

Similarly, the system with control gains $(K_D, K_v) = (130, 979, 10, 331)$ will be marginally stable with delay combination $(\tau_v, \tau_D) = (0.04919, 0.09369)$ (see Fig. 32(*b*)). If the control gains are $(K_D, K_v) = (363, 833, 18, 051)$, and delay combination is $(\tau_v, \tau_D) = (0.0251, 0.04978)$, then the system will also be marginally stable (see Fig. 33(*b*)). However, from Figs. 32(*b*) and 33(*b*), one may notice that the experimental data does not match the simulated data very well; one of the possible reasons is due to the discrepancy between the real model and the estimated model. In these two experiments, even though the experimental and simulation results are different in the end, they behave similarly for a certain amount of time.

When the system is marginally stable, two interesting results are that if one increases the velocity delay, the system will become stable (compare Fig. 32(*b*) with Fig. 32(*a*) and Fig. 33(b) with Fig. 33(*a*)); however, if one decreases the velocity delay, the system will become unstable (compare Fig. 32(*b*) with Fig. 32(*c*) and Fig. 33 (*b*) with Fig. 33(*c*)).

Comparing Figs. 32(*a*) and 33(*a*), one also observes that the system with higher control gains can decrease the discrepancy between the experimental results and simulated results.

In general, the disparity between the simulated results and experimental results is only obvious when the system is marginally stable because any factor (noise, unmodeled dynamics, etc.) will change a marginally stable system into either a stable or unstable system.

Figure 34 shows the step responses of the system with different control gains and delay combinations. In Fig. 34(*a*), we choose control gains $(K_D, K_v) = (130, 979, 10, 331)$; from the discussion in Sec. 5.3, we know that with the delay combination $(\tau_D, \tau_v) = (0.014, 0.032)$, the system has the best performance with settling time 0.102 s, since the dominant root of the system is $-39.3525 + 25.2292i$; however, if the delay combination $(\tau_D, \tau_v) = (0.0, 0.006)$, then the dominant root becomes $-24.3802 + 1.0475i$ and the system has settling time 0.1641 s. For the delay-free case, the roots of the system are -16.5324 and -36.5571; therefore, the settling time of the system is 0.2419 s. In Fig. 34(*b*), the control gain combination is $(K_D, K_v) = (363, 833, 18, 051)$, and the delay combinations (τ_D, τ_v) are $(0.004, 0.014)$, $(0.0, 0.008)$, and $(0.0, 0.0)$. For the different delay combinations, the corresponding dominant roots are -49.5338, $-41.1205 + 13.3555i$, and -25.2359; therefore, the settling times are 0.0808 s, 0.0973 s, and 0.1585 s, respectively. The results obtained from the experimental data match the results obtained from simulation, and both the simulated results and experimental results coincide with the theoretical results.

One thing should be pointed out: in Fig. 34, both the experimental and simulation data are not starting from zero; this is because it is hard to set the initial condition of our testbed exactly at its origin due to inertia.

7 Conclusions

This paper discussed the influences of displacement-feedback delay as well as both displacement and velocity feedback delays on the UT actuator testbed stability and performance. The following conclusions can be made.

7.1 Stability. For a displacement-feedback delay system, the MTD of the system can be achieved by choosing smaller controller gains or a higher filtering frequency. The stability interval will decrease with an increase of the P-controller gain or if the filtering frequency is increased.

For the system with both displacement and velocity feedback delays, the filtering frequency does not have too much influence on the stability of the system if the controller gains are small, however, increasing controller gains will shrink the stability region tremendously with the low filtering frequency. Therefore, a high filtering frequency is required for large controller gains, and

the small controller gains are required for a low filtering frequency to guarantee the stability of the system.

7.2 Performance. For a displacement-feedback delay system, if the filtering frequency is high, increasing the P-controller gain will always improve the system performance, while increasing the D-controller gain will improve the system performance at first and then deteriorate system performance, especially for a small P-controller gain.

For a system with both displacement-feedback and velocity-feedback delays, increasing the controller gains will always improve the system performance.

For both single time delay and double TDS, increasing the filtering frequency not only increases the stability robustness of the system but also improves the performance of the system.

The other important finding for both single time delay and double TDS is that the optimal system performance might be achieved when the system is not delay free, in other words, for the UT actuator testbed, we can enlarge system delay purposely to achieve better performance; this finding is contrary to what was previously believed for many real systems.

Although the conclusions are made for the UT actuator testbed, a similar analysis can be done for any LTI-TDS by following the same procedures proposed in this paper.

It is worth mentioning that although this work shows that optimal values of delays exist for system stability and performance, delays are generally still very detrimental. Using delays should only be pursued within the design specifications outlined in this paper to attain the expected gains on performance.

Acknowledgment

The authors thank the three anonymous reviewers for their constructive criticism and helpful guidance which have significantly improved both the quality and the presentation of this paper. The first author would also like to thank Abraham Munoz, Scott Hornberger, Donald Karg, and Alex Collins for editorial assistance.

References

[1] Anderson, R., and Spong, M., 1989, "Bilateral Control of Teleoperators With Time Delay," IEEE Trans. Autom. Control, **34**(5), pp. 494–501.
[2] Bejczy, A., Kim, W., and Venema, W., 1990, "The Phantom Robot: Predictive Displays for Teleoperation With Time Delay," IEEE International Conference on Robotics and Automation, Cincinnati, OH, May 13–18, pp. 546–551.
[3] Leung, G., Francis, B., and Apkarian, J., 1995, "Bilateral Controller for Teleoperators With Time Delay Via μ-Synthesis," IEEE Trans. Rob. Autom., **11**(1), pp. 105–116.
[4] Kelly, J., Roy, N., and Sukhatme, G., 2014, "Determining the Time Delay Between Inertial and Visual Sensor Measurements," IEEE Trans. Rob., **30**(6), pp. 1514–1523.
[5] Hulin, T., Albu-Schaffer, A., and Hirzinger, G., 2014, "Passivity and Stability Boundaries for Haptic Systems With Time Delay," IEEE Trans. Control Syst. Technol., **22**(4), pp. 1297–1309.
[6] Ai, B., Zheng, Y., Wong, D., and Jang, S., 2011, "Stability Analysis of EWMA Run-to-Run Controller Subject to Stochastic Metrology Delay," 18th IFAC World Congress, Milan, Italy, Aug. 28–Sept. 2, pp. 12354–12359.
[7] Ai, B., 2012, "Stability and Performance Analysis of Semiconductor Manufacturing Process With Exponentially Weighted Moving Average Controllers," Ph.D. thesis, Huazhong University of Science and Technology, Wuhan, Hubei.
[8] Ai, B., Wong, D., and Jang, S., 2015, "Stability Analysis of Semiconductor Manufacturing Process With EWMA Run-to-Run Controllers," e-print arXiv:1510.08946[cs.SY].
[9] Imaida, T., and Senda, K., 2015, "Performance Improvement of the PD-Based Bilateral Teleoperators With Time Delay by Introducing Relative D-Control," Adv. Rob., **29**(6), pp. 385–400.
[10] Suh, I., and Bien, Z., 1979, "Proportional Minus Delay Controller," IEEE Trans. Autom. Control, **24**(2), pp. 370–372.
[11] Pyragas, K., 1992, "Continuous Control of Chaos by Self-Controlling Feedback," Phys. Lett. A, **170**(6), pp. 421–428.
[12] Abdallah, C., Dorato, P., Benites-Read, J., and Byrne, R., 1993, "Delayed Positive Feedback Can Stabilize Oscillatory Systems," 1993 American Control Conference, pp. 3106–3107.
[13] Kokame, H., Hirata, K., Konishi, K., and Mori, T., 2001, "Difference Feedback Can Stabilize Uncertain Steady States," IEEE Trans. Autom. Control, **46**(12), pp. 1908–1913.
[14] Ulsoy, A., 2015, "Time-Delayed Control of SISO Systems for Improved Stability Margins," ASME J. Dyn. Syst., Meas., Control, **137**(4), p. 041014.
[15] Olgac, N., and Sipahi, R., 2002, "An Exact Method for the Stability Analysis of Time-Delayed Linear Time-Invariant (LTI) Systems," IEEE Trans. Autom. Control, **47**(5), pp. 793–797.
[16] Olgac, N., and Sipahi, R., 2006, "An Improved Procedure in Detecting the Stability Robustness of Systems With Uncertain Delay," IEEE Trans. Autom. Control, **51**(7), pp. 1164–1165.
[17] Olgac, N., and Sipahi, R., 2005, "The Cluster Treatment of Characteristic Roots and the Neutral Type Time-Delayed Systems," ASME J. Dyn. Syst., Meas., Control, **127**(1), pp. 88–97.
[18] Sipahi, R., and Olgac, N., 2006, "A Unique Methodology for the Stability Robustness of Multiple Time Delay Systems," Syst. Control Lett., **55**(10), pp. 819–825.
[19] Sipahi, R., and Olgac, N., 2006, "Stability Map of Systems With Three Independent Delays," 2006 American Control Conference, Minneapolis, MN, June 14–16, pp. 2451–2456.
[20] Fazelinia, H., Sipahi, R., and Olgac, N., 2007, "Stability Robustness Analysis of Multiple Time-Delayed Systems Using "Building Block" Concept," IEEE Trans. Autom. Control, **52**(5), pp. 799–810.
[21] Penin, L., Matsumoto, K., and Wakabayashi, S., 2000, "Force Reflection for Time-Delayed Teleoperation of Space Robots," IEEE International Conference on Robotics, San Francisco, CA, Apr. 24–28, pp. 3120–3125.
[22] Suzuki, T., Sekine, T., Fujii, T., Asama, H., and Endo, I., 2000, "Cooperative Formation Among Multiple Mobile Robot Teleoperation in Inspection Task," 39th IEEE International Conference on Decision and Control, pp. 358–363.
[23] Lee, D., Martinez-Palafox, O., and Spong, M. W., 2005, "Bilateral Teleoperation of Multiple Cooperative Robots Over Delayed Communication Networks: Application," IEEE International Conference on Robotics, Apr. 18–22, pp. 366–371.
[24] Lee, D., and Spong, M., 2005, "Bilateral Teleoperation of Multiple Cooperative Robots Over Delayed Communication Networks: Theory," IEEE International Conference on Robotics, Apr. 18–22, pp. 360–365.
[25] Qiu, T., Hamel, W., and Lee, D., 2014, "Preliminary Experiments of Kinesthetic Exploration in a 6 DOF Teleoperation System," IEEE International Conference on Robotics, Hong Kong, May 31–June 7, pp. 5959–5964.
[26] Ohnishi, K., Shimono, T., and Natori, K., 2009, "Haptics for Medical Applications," Artif. Life Rob., **13**(2), pp. 383–389.
[27] Ohnishi, K., 2010, "Real World Haptics and Telehaptics for Medical Applications," IEEE International Conference on Industrial Electronics, Bari, Italy, July 4–7, pp. 11–14.
[28] Cheong, J., Niculescu, S., Annaswamy, A., and Srinivasan, M., 2005, "Motion Synchronization in Virtual Environments With Shared Haptics and Large Time Delays," IEEE International Conference on Haptics, Mar. 18–20, pp. 277–282.
[29] Smith, O., 1957, "Closer Control of Loops With Dead Time," Chem. Eng. Prog., **53**(5), pp. 217–219.
[30] Takegaki, M., and Arimoto, S., 1981, "A New Feedback Method for Dynamic Control of Manipulators," ASME J. Dyn. Syst., Meas., Control, **103**(2), pp. 119–125.
[31] Koditschek, D., 1984, "Natural Motion for Robot Arms," 23rd IEEE International Conference on Decision and Control, pp. 733–735.
[32] Ortega, R., and Spong, M., 1989, "Adaptive Motion Control of Rigid Robots: A Tutorial," Automatica, **25**(6), pp. 877–888.
[33] Li, Z., Xia, Y., and Cao, X., 2013, "Adaptive Control of Bilateral Teleoperation With Unsymmetrical Time-Varying Delays," Int. J. Innovations Comput. Inf. Control, **9**(2), pp. 753–767.
[34] Hokayem, P., Stipanović, D., and Spong, M., 2009, "Semiautonomous Control of Multiple Networked Lagrangian Systems," Int. J. Robust Nonlinear Control, **19**(18), pp. 2040–2055.
[35] Insperger, T., Kovacs, L., Galambos, P., and Stepan, G., 2010, "Increasing the Accuracy of Digital Force Control Process Using the Act-and-Wait Concept," IEEE/ASME Trans. Mechatronics, **15**(2), pp. 291–298.
[36] Li, Z., Xia, Y., and Sun, F., 2014, "Adaptive Fuzzy Control of Multilateral Cooperative Teleoperation for Multiple Robotic Manipulators Under Random Time Delays," IEEE Trans. Fuzzy Syst., **22**(2), pp. 437–450.
[37] Li, Z., Cao, X., Tang, Y., Li, R., and Ye, W., 2013, "Bilateral Teleoperation of Holonomic Constrained Robotic Systems With Time-Varying Delays," IEEE Trans. Instrum. Meas., **62**(4), pp. 752–765.
[38] Seuret, A., Ozbay, H., Bonnet, C., and Mounier, H., 2014, *Low-Complexity Controllers for Time-Delay Systems*, Vol. 2, Springer, Berlin.
[39] Diftler, M., Mehling, J., Abdallah, M., Radford, N., Bridgwater, L., and Sanders, M., 2011, "Robonaut 2: The First Humanoid Robot in Space," IEEE International Conference on Robotics, Shanghai, China, May 9–13, pp. 2178–2183.
[40] Cheng, G., Hyon, S., Morimoto, J., Ude, A., Hale, J., and Colvin, G., 2007, "CB: A Humanoid Research Platform for Exploring Neuroscience," Adv. Rob., **21**(10), pp. 1097–1114.
[41] Gu, G., Zhu, L., Xiong, Z., and Ding, H., 2010, "Design of a Distributed Multiaxis Motion Control System Using the IEEE-1394 Bus," IEEE Trans. Ind. Electron., **57**(12), pp. 4209–4218.
[42] Ai, B., Fan, Z., and Gao, R., 2014, "Occupancy Estimation for Smart Building by an Auto-Regressive Hidden Markov Model," American Control Conference, Portland, OR, June 4–6, pp. 2234–2239.
[43] Ai, B., Zheng, Y., Wang, Y., Jang, S., and Tao, S., 2010, "Cycle Forecasting EWMA (CF-EWMA) Approach for Drift and Fault in Mixed-Product Run-to-Run Process," J. Process Control, **20**(5), pp. 689–708.
[44] Zheng, Y., Ai, B., Wong, D., Jang, S., and Zhang, J., 2010, "An EWMA Algorithm With a Cycled Resetting (CR) Discount Factor for Drift and

Fault of High-Mix Run-to-Run Control," IEEE Trans. Ind. Inf., **6**(2), pp. 229–242.
[45] Ai, B., Zheng, Y., Jang, S., Wang, Y., Ye, L., and Zhou, C., 2009, "The Optimal Drift-Compensatory and Fault Tolerant Approach for Mixed-Product Run-to-Run Control," J. Process Control, **19**(8), pp. 1401–1412.
[46] Ai, B., Zheng, Y., Zhang, H., Wang, Z., and Zhang, Z., 2009, "Cycle Prediction EWMA Run-to-Run Controller for Mixed-Product Drifting Process," 48th IEEE International Conference on Decision and Control, Shanghai, China, Dec. 15–18, pp. 1908–1913.
[47] Ai, B., Zheng, Y., and Zhang, Z., 2009, "A Fault-Tolerant Algorithm With Cycled Resetting Discount Factor in Semiconductor Manufacturing Industry," 7th IEEE International Conference on Control and Automation, Christchurch, New Zealand, Dec. 9–11, pp. 483–488.
[48] Zheng, Y., Ai, B., Wang, Y., and Zhang, H., 2009, "The dEWMA Fault Tolerant Approach for Mixed Product Run-to-Run Control," IEEE International Conference on Industrial Electronics, Seoul, Korea, July 5–8, pp. 155–160.
[49] Zhao, Y., Paine, N., Kim, K., and Sentis, L., 2015, "Stability and Performance Limits of Latency-Prone Distributed Feedback Controllers," IEEE Trans. Ind. Electron., **62**(11), pp. 7151–7162.
[50] Paine, N., and Sentis, L., 2015, "A Closed-Form Solution for Selecting Maximum Critically Damped Actuator Impedance Parameters," ASME J. Dyn. Syst., Meas., Control, **137**(7), p. 041011.
[51] Stépán, G., 1989, *Retarded Dynamical Systems: Stability and Characteristic Functions*, Vol. 200, Longman Scientific & Technical Essex, UK and Wiley, New York.
[52] Hale, J., 1993, *Introduction to Functional Differential Equations*, Vol. 99, Springer, Berlin.
[53] Hale, J., and Lunel, S., 2002, "Strong Stabilization of Neutral Functional Differential Equations," IMA J. Math. Control Inf., **19**(1–2), pp. 5–23.
[54] Hu, G., 1996, "Some Simple Criteria for Stability of Neutral Delay-Differential Systems," Appl. Math. Comput., **80**(2–3), pp. 257–271.
[55] Chen, J., Gu, G., and Nett, C., 1994, "A New Method for Computing Delay Margins for Stability of Linear Delay Systems," 33rd IEEE International Conference on Decision and Control, Lake Buena Vista, FL, Dec. 14–16, pp. 433–437.
[56] Niculescu, S., 2001, *Delay Effects on Stability: A Robust Control Approach*, Vol. 269, Springer, Berlin.
[57] Park, P., 1999, "A Delay-Dependent Stability Criterion for Systems With Uncertain Time-Invariant Delays," IEEE Trans. Autom. Control, **44**(4), pp. 876–877.
[58] Rekasius, Z., 1980, "A Stability Test for Systems With Delays," Joint Automation Control Conference, Paper No. TP9–A.
[59] Thowsen, A., 1981, "An Analytic Stability Test for a Class of Time-Delay Systems," IEEE Trans. Autom. Control, **26**(3), pp. 735–736.
[60] Hertz, D., Jury, E., and Zeheb, E., 1984, "Simplified Analytic Stability Test for Systems With Commensurate Time Delays," IEE Proc. D (Control Theor. Appl.), **131**(1), pp. 52–56.
[61] MacDonald, N., 1985, "Comments on a Simplified Analytical Stability Test for Systems With Delay," IEE Proc. D (Control Theor. Appl.), **132**(5), pp. 237–238.
[62] Kolmanovskiĭ, V., 1986, *Stability of Functional Differential Equations*, Vol. 180, Elsevier, New York.
[63] Zhang, L., Yang, C., Chajes, M., and Cheng, A., 1993, "Stability of Active-Tendon Structural Control With Time Delay," J. Eng. Mech. ASCE, **119**(5), pp. 1017–1024.
[64] Chiasson, J., and Loiseau, J., 2007, *Applications of Time Delay Systems*, Vol. 352, Springer, Berlin.
[65] Insperger, T., and Stépán, G., 2011, *Semi-Discretization for Time-Delay Systems: Stability and Engineering Applications*, Vol. 178, Springer, Berlin.
[66] Michiels, W., and Niculescu, S., 2014, *Stability, Control, and Computation for Time-Delay Systems: An Eigenvalue-Based Approach*, Vol. 27, SIAM, Philadelphia.
[67] Toker, O., and Özbay, H., 1996, "Complexity Issues in Robust Stability of Linear Delay-Differential Systems," Math. Control Signals, **9**(4), pp. 386–400.
[68] Akritas, A., Gennadi, M., and Vigklas, P., 2014, "Sturm Sequences and Modified Subresultant Polynomial Remainder Sequences," Serdica J. Comput., **8**(1), pp. 29–46.
[69] Barnett, S., 1983, *Polynomials and Linear Control Systems*, Marcel Dekker, New York.
[70] Sipahi, R., and Delice, I. I., 2009, "Extraction of 3D Stability Switching Hypersurfaces of a Time Delay System With Multiple Fixed Delays," Automatica, **45**(6), pp. 1449–1454.
[71] Sipahi, R., and Delice, I., 2011, "Advanced Clustering With Frequency Sweeping Methodology for the Stability Analysis of Multiple Time-Delay Systems," IEEE Trans. Autom. Control, **56**(2), pp. 467–472.
[72] Michiels, W., and Niculescu, S., 2007, *Stability and Stabilization of Time-Delay Systems: An Eigenvalue-Based Approach*, Vol. 12, SIAM, Philadelphia.
[73] Michiels, W., Vyhlídal, T., and Zítek, P., 2010, "Control Design for Time-Delay Systems Based on Quasi-Direct Pole Placement," J. Process Control, **20**(3), pp. 337–343.
[74] Michiels, W., and Vyhlídal, T., 2005, "An Eigenvalue Based Approach for the Stabilization of Linear Time-Delay Systems of Neutral Type," Automatica, **41**(4), pp. 991–998.
[75] Vyhlídal, T., and Zítek, P., 2003, "Quasipolynomial Mapping Based Rootfinder for Analysis of Time Delay Systems," 4th Proceedings of the IFAC Workshop on Time-Delay Systems, pp. 227–232.
[76] Vyhlídal, T., and Zítek, P., 2009, "Mapping Based Algorithm for Large-Scale Computation of Quasi-Polynomial Zeros," IEEE Trans. Autom. Control, **54**(1), pp. 171–177.
[77] Cooke, K., and Driessche, P., 1996, "Analysis of an SEIRS Epidemic Model With Two Delays," J. Math. Boil., **35**(2), pp. 240–260.
[78] Cooke, K., Driessche, P., and Zou, X., 1999, "Interaction of Maturation Delay and Nonlinear Birth in Population and Epidemic Models," J. Math. Boil., **39**(4), pp. 332–352.
[79] Cooke, K., and Driessche, P., 1986, "On Zeroes of Some Transcendental Equations," Funkcialaj Ekvacio, **29**(1), pp. 77–90.
[80] Walton, K., and Marshall, J., 1984, "Closed Form Solution for Time Delay Systems' Cost Functionals," Int. J. Control, **39**(5), pp. 1063–1071.
[81] Walton, K., and Marshall, J., 1987, "Direct Method for TDS Stability Analysis," IEE Proc. D, **134**(2), pp. 101–107.
[82] Loiusell, J., 2001, "A Matrix Method for Determining the Imaginary Axis Eigenvalues of a Delay System," IEEE Trans. Autom. Control, **46**(12), pp. 2008–2012.
[83] Aström, K., and Murray, R., 2010, *Feedback Systems: An Introduction for Scientists and Engineers*, Princeton, Englewood Cliffs, NJ.