INT. J. CONTROL, 2003, VOL. 76, NO. 6, 591–606

Taylor & Francis
Taylor & Francis Group

Modelling and optimal controller design of networked control systems with multiple delays

FENG-LI LIAN†*, JAMES MOYNE‡ and DAWN TILBURY§

In this paper we discuss the modelling and control of networked control systems (NCS) where sensors, actuators and controllers are distributed and interconnected by a common communication network. Multiple distributed communication delays as well as multiple inputs and multiple outputs (MIMO) are considered in the modelling algorithm. In addition, the asynchronous sampling mechanisms of distributed sensors are characterized to obtain the actual time delays between sensors and the controller. Due to the characteristics of a network architecture, piecewise constant plant inputs are assumed and discrete-time models of plant and controller dynamics are adopted to analyse the stability and performance of a closed-loop NCS. The analysis result is used to verify the stability and performance of an NCS without considering the impact of multiple time delays in the controller design. In addition, the proposed NCS model is used as a foundation for optimal controller design. The proposed control algorithm utilizes the information of delayed signals and improves the control performance of a control system encountering distributed communication delays. Several simulation studies are provided to verify the control performance of the proposed controller design.

1. Introduction

A major trend in modern industrial and commercial systems is to integrate computing, communication, and control into different levels of machine/factory operations and information processes. The traditional communication architecture for control systems, which has been successfully implemented in industry for decades, is point-to-point; that is, a wire connects the central control computer with each sensor or actuator point. However, expanding physical setups and functionality are pushing the limits of the point-to-point architecture. Hence, a traditional centralized point-to-point control system is no longer suitable to meet new requirements, such as modularity, decentralization of control, integrated diagnostics, quick and easy maintenance and low cost. The introduction of common-bus network architectures can improve the efficiency, flexibility and reliability of these integrated applications through reduced wiring and distributed intelligence, and reduce installation, reconfiguration and maintenance time and costs.

These types of distributed control systems are called networked control systems (NCS): sensors, actuators, and controllers are interconnected by one communication network. The change of communication architecture from point-to-point to common-bus, however, introduces different forms of time delay uncertainty between sensors, actuators and controllers. These time delays come from the time sharing of the communication medium as well as the computation time required for physical signal coding and communication processing. The characteristics of time delays can be constant, bounded or even random, depending on the network protocols adopted and the chosen hardware. It is well known in control systems that time delays can degrade a system's performance and even cause system instability. However, though the analysis and modelling of time-delay systems has been a progressive research area, existing methodologies cannot be directly applied due to the discrete and distributed nature of the many different time delays in NCSs.

Most NCS research has focused on two areas: communication protocols and controller design. A proper message transmission protocol is necessary to guarantee the network quality of service, whereas advanced controller design is desirable to guarantee the control quality of performance. In this paper we consider the modelling of an NCS with multiple communication delays and formulate and solve the optimal control problem based on the proposed discrete-time time-delay model. In § 2, we survey existing results on time-delay systems and networked control systems. In § 3, we discuss the problem formulation and assumptions used in the discrete-time modelling algorithm. In § 4, we address the delay impact on plant and controller dynamics. In § 5, we provide an example to illustrate the stability analysis of the proposed modelling approach. Based on the proposed modelling framework, we formulate an optimal controller in § 6. We present simulation result of the proposed controller design in § 7. In § 8, we summarize the results of the proposed modelling and controller design of networked control systems.

Received 27 August 2001. Revised 15 January 2003.

* Author for correspondence. e-mail: fengli@ntu.edu.tw

† Electrical Engineering 537, National Taiwan University, No. 1, Sec. 4, Roosevelt Road, Taipei, 106, Taiwan.

‡ Department of Electrical Engineering and Computer Science, University of Michigan, 1124 EECS Building, 1301 Beal Avenue, Ann Arbor, MI 48109-2122, USA. e-mail: moyne@umich.edu

§ Department of Mechanical Engineering, University of Michigan, 2250 G. G. Brown Building, 2350 Hayward Street, Ann Arbor, MI 48109-2125, USA. e-mail: tilbury@umich.edu

International Journal of Control ISSN 0020–7179 print/ISSN 1366–5820 online © 2003 Taylor & Francis Ltd
http://www.tandf.co.uk/journals
DOI: 10.1080/0020717031000098426

2. Related work

In this section we discuss related research on analysis of time-delay systems and modelling of control of networked control systems.

2.1. *Analysis of time-delay systems*

Modelling and control of NCSs is based on the time-delay systems analysis framework which has been studied for several decades. In general, delays occur in the transmission of signals or materials between different subsystems. Large-scale systems such as communication systems, manufacturing systems, transportation systems, power systems and teleoperation systems are typical examples of time-delay systems (Malek-Zavarei and Jamshidi 1987). There have been two approaches used to analyse the stability of time-delay systems: classical (frequency domain) and functional (time domain) approaches (Oğuztöreli 1966, Marshall 1979, Górecki *et al*. 1989) The classical approach utilizes analytical or graphical methods to find the roots of the characteristic equation of a dynamic system. Since delays appear in the characteristic equation as exponential functions, analytical techniques are developed to find solutions of a quasi-polynomial. Alternatively, the Padé approximation to the time delay can be used to obtain a pure polynomial. In addition, standard graphical methods such as the root locus, Bode plot and Nyquist diagram can be further modified to analyse delayed polynomials or transfer functions. For a discrete-time system with time delays, the system stability can also be easily verified by the root locus analysis in the z-domain (Franklin *et al*. 1998).

The functional approach, on the other hand, uses delay-differential equations to characterize systems with delays in the state, input or both. Multiple delays (Hsiao and Hwang 1997, Gu 1999) and time-varying delays (Goubet-Bartholoméüs *et al*. 1997, Gu *et al*. 1998) can be modelled in a similar setting. Based on standard algorithms from functional analysis, solutions of delay-differential equations can be constructed, and their stability can be analysed using Lyapunov's second methods such as Lyapunov–Krasovskii and/or Lyapunov–Razumikhin stability theorems (Dugard and Verrest 1998).

Since time delays in most applications are considered non-deterministic parameters, controller design for time-delay systems typically uses either a robust or stochastic control approach. Robust $\mathcal{H}_\infty$ controllers can be designed based on the known system structure or delay uncertainty. Uncertainties in both the system matrix and the input matrix can be included in the delay-differential dynamic equation. Moreover, using a linear matrix inequality (LMI) approach, system stability and performance can be analysed and guaranteed (Jeung *et al*. 1996, Li and de Souza 1997, Luo *et al*. 1998).

These modelling and analysis approaches developed for time-delay systems can only be used as a foundation for analysing the time-delay effects in NCSs. Since all devices in an NCS are distributed on one common-bus network, sample-and-hold and time skew among samples are typical and result in an inherent asynchronization in the system model. Also, due to the distribution of actuators and the network communication mechanism, the system inputs are piecewise constant with delays, rather than continuous. These properties have not been discussed in previous time-delay analyses; thus further research is needed to model and analyse NCSs.

2.2. *Modelling and control of networked control systems*

Research in NCSs is different from that in traditional time-delay systems. Because of the variability of network-induced time delays, the NCSs may be time-varying systems, making analysis and design more challenging. Wittenmark *et al*. (1995) discussed several timing issues such as communication and computation delays, processor jitter and transient errors existing in NCSs. These timing issues must be addressed when deriving a discrete-time state-space model. In their subsequent work, a discrete-time model with a single sensor-controller delay and a single controller-actuator delay was studied and a stochastic controller design was designed (Nilsson and Bernhardsson 1997, Nilsson *et al*. 1998). Nilsson (1998) studied the case with multiple sensor-controller and controller-actuator delays; only the case where the total maximum network delay is less than one controller sampling period was considered.

Recently research on the analysis and modelling of NCSs has been conducted using continuous-time and discrete-time models. It is more natural to analyse an NCS from the discrete-time point of view since in a typical NCS operation, physical signals (from sensors or to actuators) are sampled and then transmitted on the network medium after a short delay. For discrete-time models, most researchers assume that the network is synchronized and the sampling rates of sensors, controllers and actuators are the same. Halevi and Ray (1988) considered the case of a single time delay for sensor-controller and controller-actuator and a single time skew between sensor and controller sampling instants. They used the augmented state to include the past delayed signals and derived a closed-loop model for NCSs.

Krtolica *et al*. (1994) derived a discrete-time time-varying state-space representation of NCSs with

random network delays by using the augmented states to include past plant and controller states. The total number of states used depends on the possible range of sensor-controller and controller-actuator delays. Since the maximum possible number of network delays is bounded, and the closed-loop system is a discrete-time model, the system matrix can be viewed as a finite automaton with finite states. The stability analysis of such a system can be further described by a Markov chain with finite state transitions.

Branicky *et al.* (2000) considered a simplified NCS model where the sensor-controller and controller-actuator delays are lumped together. Their study only considered the case where the lumped time delay is less than one sampling period; the stability regions for the NCS model were investigated based on the lumped delay. In practical applications, however, sensor-controller and controller-actuator delays are different and time-varying at different networked devices due to the network transmission mechanism. Hence, a proper time delay profile in NCSs should be characterized based on the network transmission bandwidth and control system bandwidth.

Continuous-time NCS models were also considered by several researchers. Göktas *et al.* (1996, 1997) used a modified Padé approximation and considered the network delay as an uncertainty. They designed a robust controller to compensate for the uncertain delay in an ATM network. Kim *et al.* (1996) used a Lyapunov approach to obtain the maximum allowable delay bound for the stability of a network delayed system. A scheduling algorithm for determining the sampling rate and allocating bandwidth was also provided. Walsh *et al.* (1999 a, b) also adopted the Lyapunov approach on a continuous-time model to obtain the maximum allowable transfer interval and to analyse the stability of the closed-loop system. They further analysed the impact of different scheduling algorithms on the maximum allowable transfer interval. However, only a conservative delay bound was obtained. The impact of delay variance on control performance is discussed in these works, but is not formally characterized.

In practical applications, however, sensor-controller and controller-actuator delays are different and time-varying at different networked devices due to the network transmission mechanism. An NCS should be modelled based on the characteristics of network-induced delays and the consideration of network and control parameters. Furthermore, controllers should be designed based on the NCS model, taking into account the delay information. Therefore, in this paper, a discrete-time model of NCSs with multiple inputs and multiple outputs (MIMO) and multiple distributed communication delays is derived. In addition, the asynchronous sampling mechanisms of distributed sensors are characterized to obtain the actual time delays between sensors and the controller. Based on the proposed NCS model, the stability and performance of a closed-loop system with a standard controller are analysed, and a linear quadratic regulator (LQR) optimal control is formulated to compensate for the multiple time delays. A preliminary version of this work was presented in Lian *et al.* (2001 b, 2002 b).

3. Problem formulation and assumptions

Consider the block diagram of a networked control system with a single controller, but multiple sensors and multiple actuators as shown in figure 1. There are N states (x), M inputs (u), and R outputs (y) in the *plant dynamics model*, and Q states (z), R inputs (w), and M outputs (v), in the *controller dynamics model*, i.e. M actuators, R sensors and one controller, where N, M, R and Q are positive constant integers. We use $s_r, r = 1, 2, \ldots, R$ and $a_m, m = 1, 2, \ldots, M$ to represent the sensor-controller and controller-actuator delays, respectively. The variables w_r and u_m are the delayed y_r and v_m signals, respectively. The relationships between these variables will be addressed later. In the following discussion, we present the system models in continuous time and discrete time. In this paper, time is denoted by t for the continuous-time domain and k for the discrete-time domain.

In figure 1, the continuous-time, state-space model of the linear time-invariant plant dynamics G_p can be described by the standard form

$$\left.\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A}_p\mathbf{x}(t) + \mathbf{B}_p\mathbf{u}(t)\\ \mathbf{y}(t) &= \mathbf{C}_p\mathbf{x}(t)\end{aligned}\right\} \tag{1}$$

where $\mathbf{x}(t) \in \mathbb{R}^N, \mathbf{u}(t) \in \mathbb{R}^M, \mathbf{y}(t) \in \mathbb{R}^R$ and the constant matrices $\mathbf{A}_p, \mathbf{B}_p$ and $\mathbf{C}_p$ are of compatible dimensions. Since the controller is implemented at one digital computer, the controller is designed in discrete time with a sampling time T and the state-space model of the controller dynamics G_c can be expressed as

$$\left.\begin{aligned}\mathbf{z}_{k+1} &= \mathbf{F}\mathbf{z}_k + \mathbf{G}\mathbf{w}_k\\ \mathbf{v}_k &= \mathbf{H}\mathbf{z}_k + \mathbf{J}\mathbf{w}_k\end{aligned}\right\} \tag{2}$$

where $\mathbf{z}_k \triangleq \mathbf{z}(k) = \mathbf{z}(kT) \in \mathbb{R}^Q$, $\mathbf{w}_k \triangleq \mathbf{w}(k) = \mathbf{w}(kT) \in \mathbb{R}^R$, $\mathbf{v}_k \triangleq \mathbf{v}(k) = \mathbf{v}(kT) \in \mathbb{R}^M$ and the matrices $\mathbf{F}, \mathbf{G}, \mathbf{H}$ and $\mathbf{J}$ are of compatible dimensions. Note that $\mathbf{w}_k$ is a delayed version of the sensor output $\mathbf{y}(t)$ at some sampling instant, and similarly, $\mathbf{u}(t)$ is a delayed version of the controller output $\mathbf{v}_k$.

In practical applications of networked control systems, devices are distributed and have their own processing units and timing functions. Hence, synchronization of all devices is extremely difficult. In this paper, we assume that the network is not synchronized; each device may have a different time skew when related to

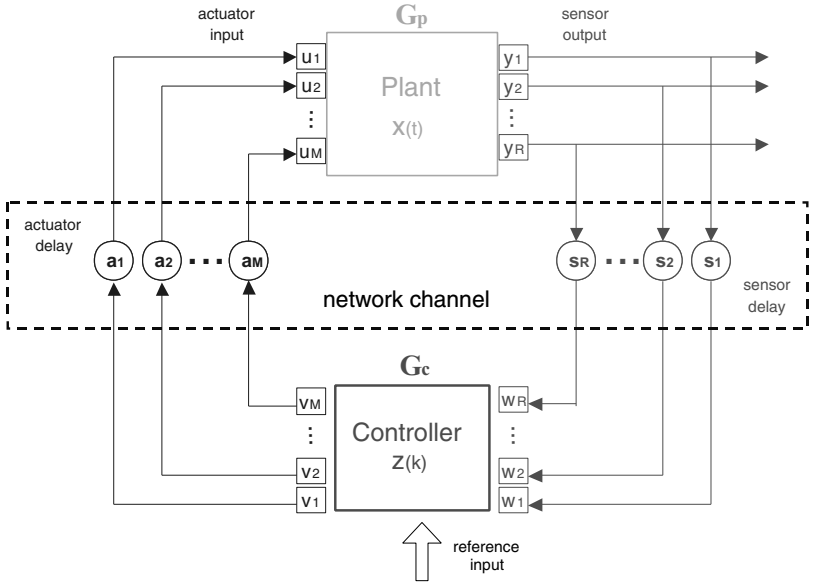


Figure 1. The block diagram of a networked control system. Sensors, actuators and controllers in an NCS are distributed and interconnected by communication networks. NCSs are flexible, reconfigurable and efficient. The information of these devices can be easily shared by other subsystems. However, time-delays between sensor-controller and controller-actuator are unavoidable because of the sharing of the communication medium.

the controller sampling instants. We also assume that the sensor and controller sampling times are the same, and that actuators respond to actuation commands immediately after receiving the information from the controller. The detailed assumptions and notations used in this paper are described as follows and are illustrated in figure 2.

1. The *buffer length* at the controller for each sensor and each actuator is equal to one. That is, the controller only uses the newest sensor messages and never sends a stale actuator command.
2. As shown in figure 2, the *periods* of all R sensors and one controller are identical and equal to T, but there may be R different *time skews*, denoted as $\Delta_r, r = 1, \ldots, R$, among these sensor sampling instants. The definition of Δ_r is the time difference between the sampling instant of the rth sensor and the sampling instant of the controller. We assume that Δ_rs are constant.
3. There are two types of time delays: *processing delay* and *communication delay*. Processing delays occur at the controller, sensors and actuators, and are denoted as $p^c(k), p_r^s(k)$ and $p_m^a(k)$, respectively. Communication delays between the sensors and the controller and between the controller and the actuators are denoted as $c_r^s(k)$ and $c_m^a(k)$, respectively. In this paper, we add the processing delays to the communication delays. The combined time delays are defined as follows. The combined sensor processing-communication delay is $c_r^s(k) + p_r^s(k)$ and the combined actuator-controller processing-communication delay is $a_m(k) = p^c(k) + c_m^a(k) + p_m^a(k)$. Hence, in the following discussion, we only consider the combined time delays. We also assume that these delays are bounded by one sampling period, that is, the cases of vacant sampling and message rejection are not considered in this paper. Note that $a_m(k)$s shown in figure 2 are the combined delays. Detailed discussions of characterizing communication delays and choosing a proper sampling period, guaranteeing control performance can be found in Lian *et al*. (2001 a, 2002 a), respectively.
4. As shown in figure 2, the *sensor-controller delay* $s_r(k)$ depends on both the time skew Δ_r and the sampling period T. If $c_r^s(k) + p_r^s(k) \leq \Delta_r$, then $s_r(k) = \Delta_r$. Otherwise, $s_r(k) = T + \Delta_r$. Here, we also assume that the time delay is bounded by the sampling period; that is, $\max\{s_r(k), r = 1, \ldots, R\} \leq 2T$. Note that this assumption is true for deterministic network protocols such as token passing or priority-based under normal traffic load. However, for those networks with a stochastic medium access control mechanism, this assumption might not be true.
5. Since the *controller-actuator delay* is $a_m(k)$, the kth controller sampling interval $[(k-1)T, kT)$, can be formulated as $[0, T) = [0, a_m(k)) \bigcup [a_m(k), T)$. Note that if the actuation delay is longer than one sampling period, then $a_m(k) = T$. This longer delay may be due

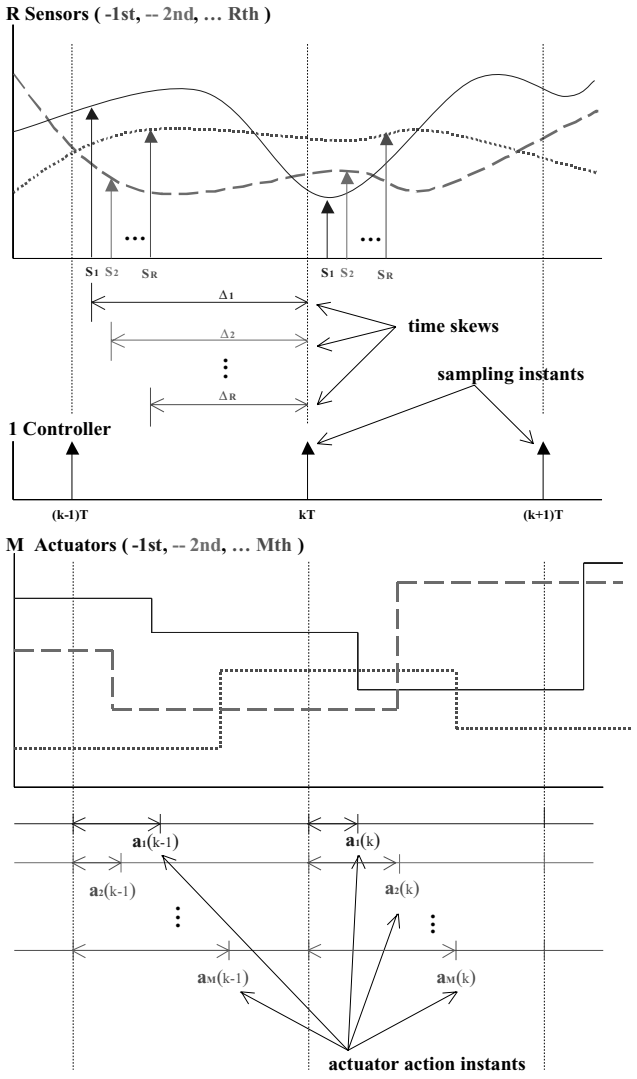


Figure 2. The timing diagram for sensors, controller, and actuators in an NCS.

to the blocking of message transmission or using too small of a sampling period. The first situation can be improved by further considering network parameters in system design. For the second situation, a larger T should be chosen. However, the control performance may degrade due to a long sampling period. On the other hand, a smaller sampling period increases the control performance as well as the system complexity due to the high network traffic load.

6. In deriving time-delay models, the sensor-controller and controller-actuator delays are assumed different and time-varying. However, for some cases of control networks, the message transmission times could be viewed as constant parameters as shown in figure 4 of Lian *et al*. (2002 a). Hence, for the controller design and illustrative examples in this paper, constant delays are further assumed.
7. For any variable θ, we will use $\bar{\theta} = \theta/T$ to denote its value in terms of sampling period. In addition, we split $\bar{\theta}$ into its integer and fractional parts as $\bar{\theta} = \hat{\theta} + \tilde{\theta}$, where $\hat{\theta} \in Z^+$ and $0 \leq \tilde{\theta} < 1$.

4. Delay impact on plant and controller dynamics

One of the main characteristics of NCSs is the different communication delays between the plant and controller, as shown in figure 1. That is, in general, $\mathbf{u}(t)|_{t=kT} \neq \mathbf{v}_k$ and $\mathbf{w}_k \neq \mathbf{y}(t)|_{t=kT}$, because of existing time delays among these signals. The actual values of these communication delays depend on the network protocol adopted as well as the network traffic load. In this section, standard linear systems theory is used to derive the relation between each pair of variables based on the assumptions described in § 3.

4.1. *Plant input and controller output*

We first study the relation between the controller outputs $\mathbf{v}_k$ and the plant inputs $\mathbf{u}(t)$. Since the actuators receive controller commands discontinuously, we assume that the actuator inputs are piecewise constant as shown in the actuator timing diagram in figure 2. Hence, for the mth actuator and $kT \leq t < (k+1)T$, the input signal $u_m(t)$ can be described as

$$u_m(t) = v_m(k-1)\mathbf{1}_{(0,a_m(k))}(t) + v_m(k)\mathbf{1}_{(a_m(k),T)}(t) \quad (3)$$

where

$$\mathbf{1}_{(a(k),b(k))}(t) = \begin{cases} 1 & \text{if } a(k) + kT \leq t < b(k) + kT \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

That is, $u_m(t)$ is the combination of piecewise constant functions. For example, consider the first actuator at the kth sampling instant shown in figure 2. The following relation holds: $u_1(t) = v_1(k-1)\mathbf{1}_{(0,a_1(k))} + v_1(k)\mathbf{1}_{(a_1(k),T)}$, for $kT \leq t < (k+1)T$. Then, $\mathbf{u}(t) = [u_1(t), \ldots, u_M(t)]^{\mathrm{T}}$ and $\mathbf{v}(k) = [v_1(k), \ldots, v_M(k)]^{\mathrm{T}}$.

4.2. *Plant model in discrete-time domain*

In order to analyse the closed-loop system in discrete-time, we use the following state-space solution of a first-order matrix differential equation to discretize the continuous-time plant dynamics model (Franklin *et al*. 1998)

$$\mathbf{x}(t) = \exp(\mathbf{A}_p(t-t_0))\mathbf{x}(t_0) + \int_{t_0}^{t} \exp(\mathbf{A}_p(t-q'))\mathbf{B}_p\mathbf{u}(q')\,\mathrm{d}q' \quad (5)$$

We first discretize the plant model at the controller sampling instants by applying (5) with $t_0 = kT$ and $t = (k+1)T$. For simplicity, we use the notation

$\mathbf{x}_k \triangleq \mathbf{x}(k) = \mathbf{x}(kT)$, $\mathbf{A} \triangleq \exp(\mathbf{A}_p T)$, $q = q' - kT$ and $\mathbf{\Gamma}(T,q) \triangleq \exp(\mathbf{A}_p(T-q))\mathbf{B}_p \in \mathbb{R}^{N\times M}$. Then

$$\begin{aligned}\mathbf{x}_{k+1} &= \mathbf{A}\mathbf{x}_k + \int_0^T \exp(\mathbf{A}_p(T-q))\mathbf{B}_p\mathbf{u}(kT+q)\,\mathrm{d}q \\ &= \mathbf{A}\mathbf{x}_k + \sum_{m=1}^{M}\int_0^T \mathbf{\Gamma}_m(T,q)u_m(kT+q)\,\mathrm{d}q \end{aligned} \tag{6}$$

where $\mathbf{\Gamma}_m \in \mathbb{R}^{N\times 1}$ and $u_m \in \mathbb{R}$, $m = 1,\ldots,M$ are the components of $\mathbf{\Gamma}$ and $\mathbf{u}$, respectively. Now, because the actuator commands arrive at different times within the sample interval, u_m is not constant over $[kT,\ (k+1)T]$. From (3), we can compute $\int_0^T \mathbf{\Gamma}_m(T,q)u_m(kT+q)\,\mathrm{d}q$ as

$$\begin{aligned}\int_0^T \mathbf{\Gamma}_m(T,q)u_m(kT+q)\,\mathrm{d}q &= \int_0^{a_m(k)} \mathbf{\Gamma}_m(T,q)v_m(k-1)\,\mathrm{d}q \\ &\quad + \int_{a_m(k)}^{T} \mathbf{\Gamma}_m(T,q)v_m(k)\,\mathrm{d}q \\ &\triangleq \mathbf{B}_m^1(k)v_m(k-1) \\ &\quad + \mathbf{B}_m^0(k)v_m(k) \end{aligned} \tag{7}$$

where we also use the timing relations in Assumption 5 and the following definitions: $\mathbf{B}_m^1(k) \triangleq \int_0^{a_m(k)} \mathbf{\Gamma}_m(T,q)\,\mathrm{d}q$ and $\mathbf{B}_m^0(k) \triangleq \int_{a_m(k)}^{T} \mathbf{\Gamma}_m(T,q)\,\mathrm{d}q$. Therefore, by applying (7) to (6), we have

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \sum_{m=1}^{M}\sum_{j=0}^{1} \mathbf{B}_m^j(k)v_m(k-j) \tag{8}$$

By further exchanging the order of summations and letting $\mathbf{B}_k^j = \left[\mathbf{B}_1^j(k)\ \mathbf{B}_2^j(k)\ \cdots\ \mathbf{B}_M^j(k)\right]$ and $\mathbf{v}_{k-j} = [v_1(k-j)\ v_2(k-j)\ \cdots\ v_M(k-j)\ ]^{\mathrm{T}}$, we have the derivation

$$\begin{aligned}\mathbf{x}_{k+1} &= \mathbf{A}\mathbf{x}_k + \sum_{j=0}^{1}\sum_{m=1}^{M} \mathbf{B}_m^j(k)v_m(k-j) \\ &= \mathbf{A}\mathbf{x}_k + \mathbf{B}_k^0\mathbf{v}_k + \mathbf{B}_k^1\mathbf{v}_{k-1} \end{aligned} \tag{9}$$

In (9), $\mathbf{A}$ is time-invariant because it is independent of the delay variables $a_m(k)$, but the $\mathbf{B}_k^j$s will depend on the values of $a_m(k)$s. Therefore, if these $a_m(k)$s depend on k, the NCS model will be time-varying.

4.3. *State value between sampling instants*

Because the sampling of sensors happens between the sampling instants of the controller, we need to derive the formula for the state values between sampling instants, i.e. $\mathbf{x}((k+1)T - \delta T)$, where $0 \le \delta < 1$. By using (5) again with $t_0 = kT$ and $t = (k+1-\delta)T$

$$\mathbf{x}((k+1-\delta)T) = \mathbf{A}_\delta\mathbf{x}_k + \mathbf{B}_{\delta k}^0\mathbf{v}_k + \mathbf{B}_{\delta k}^1\mathbf{v}_{k-1} \tag{10}$$

where $\mathbf{A}_\delta = \exp(\mathbf{A}_p(T-\delta T))$, $\mathbf{B}_\delta^j(k) \in \mathbb{R}^{N\times M}$ and $\mathbf{v}_{k-j} \in \mathbb{R}^{M\times 1}$. Note that $\mathbf{A}_\delta$ is not constant but depends on δ, in contrast to $\mathbf{A}$ in (6), and $\mathbf{B}_\delta^j(k)$ can be formulated similarly to $\mathbf{B}_k^j$. If $0 \le \delta T < T - a_m(k)$, then the mth elements of $\mathbf{B}_\delta^0(k)$ and $\mathbf{B}_\delta^1(k)$ are $\mathbf{B}_{\delta m}^0(k) \triangleq \int_{a_m(k)}^{T-\delta} \mathbf{\Gamma}_m(T,q)\,\mathrm{d}q$, and $\mathbf{B}_{\delta m}^1(k) \triangleq \int_0^{a_m(k)} \mathbf{\Gamma}_m(T,q)\,\mathrm{d}q$, respectively. If $T - a_m(k) \le \delta T < T$, then the mth elements of $\mathbf{B}_\delta^0(k)$ and $\mathbf{B}_\delta^1(k)$ are $\mathbf{B}_{\delta m}^0(k) \triangleq \mathbf{0}$, and $\mathbf{B}_{\delta m}^1(k) \triangleq \int_0^{\delta T} \mathbf{\Gamma}_m(T,q)\,\mathrm{d}q$, respectively.

4.4. *Plant output and controller input*

Similarly, for plant outputs $\mathbf{y}(t)$ and controller inputs $\mathbf{w}_k$, we have the following relation due to the communication delays between the sensors and the controller. By Assumption 6 and for simplicity, we have the following equation for the rth sensor-controller delay: $s_r(k) = [\ \bar{s}_r(k)\ ]T = [\ \hat{s}_r(k) + \tilde{s}_r(k)\ ]T$, where $\hat{s}_r$ is an integer and $0 \le \tilde{s}_r < 1$. Therefore, for any value of $s_r(k)$ and by (10)

$$\begin{aligned}\mathbf{x}(kT - s_r(k)) &= \mathbf{x}((k-\hat{s}_r(k))T - \tilde{s}_r(k)T) \\ &= \mathbf{A}_{\tilde{s}_r(k)}\mathbf{x}_{k-1-\hat{s}_r(k)} \\ &\quad + \sum_{j=0}^{1} \mathbf{B}_{\tilde{s}_r(k)}^j(k-1-\hat{s}_r(k)) \\ &\quad \times \mathbf{v}(k-1-\hat{s}_r(k)-j) \end{aligned} \tag{11}$$

where we let $\delta = \tilde{s}_r(k)$ in (10). Equation (11) can then be used to compute the value of the plant output. At the kth sampling instant, the values of the plant output received by the controller are (see figure 1)

$$\begin{aligned}\mathbf{w}_k &= \begin{bmatrix} w_1(kT)) \\ \vdots \\ w_R(kT) \end{bmatrix} = \begin{bmatrix} y_1(kT - s_1(k)) \\ \vdots \\ y_R(kT - s_R(k)) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{C}_1\mathbf{x}(kT - s_1(k)) \\ \vdots \\ \mathbf{C}_R\mathbf{x}(kT - s_R(k)) \end{bmatrix} = \begin{bmatrix} \mathbf{C}_1\mathbf{A}_{\tilde{s}_1(k)}\mathbf{x}_{k-1-\hat{s}_1(k)} \\ \vdots \\ \mathbf{C}_R\mathbf{A}_{\tilde{s}_R(k)}\mathbf{x}_{k-1-\hat{s}_R(k)} \end{bmatrix} \\ &\quad + \begin{bmatrix} \mathbf{C}_1\sum_{j=0}^{1}\mathbf{B}_{\tilde{s}_1(k)}^j(k-1-\hat{s}_1(k))\mathbf{v}(k-1-\hat{s}_1(k)-j) \\ \vdots \\ \mathbf{C}_R\sum_{j=0}^{1}\mathbf{B}_{\tilde{s}_R(k)}^j(k-1-\hat{s}_R(k))\mathbf{v}(k-1-\hat{s}_R(k)-j) \end{bmatrix} \\ &= \sum_{i=1}^{2}\mathbf{\Theta}_k^i\mathbf{x}_{k-i} + \sum_{i=1}^{3}\mathbf{\Phi}_k^i\mathbf{v}_{k-i} \end{aligned} \tag{12}$$

where

$$\boldsymbol{\Theta}_k^i = \begin{bmatrix} \boldsymbol{\Theta}_k^{i1} \\ \vdots \\ \boldsymbol{\Theta}_k^{iR} \end{bmatrix}, \qquad \boldsymbol{\Theta}_k^{ir} = \begin{cases} \mathbf{C}_r \mathbf{A}_{\tilde{s}_r(k)}, & \text{if } \hat{s}_r(k) = i-1 \\ 0, & \text{otherwise} \end{cases}$$

and

$$\boldsymbol{\Phi}_k^i = \begin{bmatrix} \boldsymbol{\Phi}_k^{i1} \\ \vdots \\ \boldsymbol{\Phi}_k^{iR} \end{bmatrix}, \qquad \boldsymbol{\Phi}_k^{ir} = \begin{cases} \mathbf{C}_r \mathbf{B}^0_{\tilde{s}_r(k)}(k-1) & \text{if } i=1 \\ \mathbf{C}_r[\mathbf{B}^1_{\tilde{s}_r(k)}(k-1) + \mathbf{B}^0_{\tilde{s}_r(k)}(k-2)], & \text{if } i=2 \\ \mathbf{C}_r \mathbf{B}^1_{\tilde{s}_r(k)}(k-2) & \text{if } i=3 \end{cases}$$

4.5. *Closed-loop model of NCSs*

In order to analyse the system property and provide guidelines for controller design, we will derive the closed-loop model that combines the discrete-time plant model (9), controller model (2), and (12).

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{A}\mathbf{x}_k + \mathbf{B}_k^0[\mathbf{H}\mathbf{z}_k + \mathbf{J}\mathbf{w}_k] + \mathbf{B}_k^1 \mathbf{v}_{k-1} \\ &= \mathbf{A}\mathbf{x}_k + \mathbf{B}_k^0 \mathbf{H}\mathbf{z}_k \\ &\quad + \mathbf{B}_k^0 \mathbf{J}\left[\sum_{i=1}^{2} \boldsymbol{\Theta}_k^i \mathbf{x}_{k-i} + \sum_{i=1}^{3} \boldsymbol{\Phi}_k^i \mathbf{v}_{k-i}\right] + \mathbf{B}_k^1 \mathbf{v}_{k-1} \end{aligned} \tag{13}$$

Also, the controller dynamics (2) can be further expressed as

$$\mathbf{z}_{k+1} = \mathbf{F}\mathbf{z}_k + \mathbf{G}\left[\sum_{i=1}^{2} \boldsymbol{\Theta}_k^i \mathbf{x}_{k-i} + \sum_{i=1}^{3} \boldsymbol{\Phi}_k^i \mathbf{v}_{k-i}\right] \tag{14}$$

$$\mathbf{v}_k = \mathbf{H}\mathbf{z}_k + \mathbf{J}\left[\sum_{i=1}^{2} \boldsymbol{\Theta}_k^i \mathbf{x}_{k-i} + \sum_{i=1}^{3} \boldsymbol{\Phi}_k^i \mathbf{v}_{k-i}\right] \tag{15}$$

By further combining (13)–(15) and defining

$$\mathbf{X}_k = \left[\mathbf{x}_k^{\mathrm{T}}\ \mathbf{x}_{k-1}^{\mathrm{T}}\ \mathbf{x}_{k-2}^{\mathrm{T}} \mid \mathbf{z}_k^{\mathrm{T}} \mid \mathbf{v}_{k-1}^{\mathrm{T}}\ \mathbf{v}_{k-2}^{\mathrm{T}}\ \mathbf{v}_{k-3}^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbb{R}^{3N+Q+3R}$$

we obtain the closed-loop dynamics as

$$\mathbf{X}_{k+1} = \boldsymbol{\Xi}_k \mathbf{X}_k \tag{16}$$

where

$$\boldsymbol{\Xi}_k = \left[\begin{array}{c|c|c} \boldsymbol{\Xi}_k(1,1) & \boldsymbol{\Xi}_k(1,2) & \boldsymbol{\Xi}_k(1,3) \\ \hline \boldsymbol{\Xi}_k(2,1) & \boldsymbol{\Xi}_k(2,2) & \boldsymbol{\Xi}_k(2,3) \\ \hline \boldsymbol{\Xi}_k(3,1) & \boldsymbol{\Xi}_k(3,2) & \boldsymbol{\Xi}_k(3,3) \end{array}\right] \in \mathbb{R}^{(3N+Q+3R)\times(3N+Q+3R)} \tag{17}$$

The variables $\boldsymbol{\Xi}(i, j)$ are defined as

$$\boldsymbol{\Xi}_k(1,1) = \left[\begin{array}{cc} \mathbf{A} & \mathbf{B}_k^0\mathbf{J}\boldsymbol{\Theta}_k^1 \quad \mathbf{B}_k^0\mathbf{J}\boldsymbol{\Theta}_k^2 \\ \hline \mathbf{I}_{2N\times 2N} & \mathbf{0}_{2N\times N} \end{array}\right] \in \mathbb{R}^{3N\times 3N}$$

$$\boldsymbol{\Xi}_k(1,2) = \left[\frac{\mathbf{B}_k^0\mathbf{H}}{\mathbf{0}_{2N\times Q}}\right] \in \mathbb{R}^{3N\times Q}$$

$$\boldsymbol{\Xi}_k(1,3) = \left[\frac{\mathbf{B}_k^0\mathbf{J}\boldsymbol{\Phi}_k^1 + \mathbf{B}_k^1 \quad \mathbf{B}_k^0\mathbf{J}\boldsymbol{\Phi}_k^2 \quad \mathbf{B}_k^0\mathbf{J}\boldsymbol{\Phi}_k^3}{\mathbf{0}_{2N\times 3R}}\right] \in \mathbb{R}^{3N\times 3R}$$

$$\boldsymbol{\Xi}_k(2,1) = \left[\mathbf{0}_{Q\times N} \quad \mathbf{G}\boldsymbol{\Theta}_k^1 \quad \mathbf{G}\boldsymbol{\Theta}_k^2\right] \in \mathbb{R}^{Q\times 3N}$$

$$\boldsymbol{\Xi}_k(2,2) = \mathbf{F} \in \mathbb{R}^{Q\times Q}$$

$$\boldsymbol{\Xi}_k(2,3) = \left[\mathbf{G}\boldsymbol{\Phi}_k^1 \quad \mathbf{G}\boldsymbol{\Phi}_k^2 \quad \mathbf{G}\boldsymbol{\Phi}_k^3\right] \in \mathbb{R}^{Q\times 3R}$$

$$\boldsymbol{\Xi}_k(3,1) = \left[\frac{\mathbf{0}_{R\times N} \quad \mathbf{J}\boldsymbol{\Theta}_k^1 \quad \mathbf{J}\boldsymbol{\Theta}_k^2}{\mathbf{0}_{2R\times 3N}}\right] \in \mathbb{R}^{3R\times 3N}$$

$$\boldsymbol{\Xi}_k(3,2) = \left[\frac{\mathbf{H}}{\mathbf{0}_{2R\times Q}}\right] \in \mathbb{R}^{3R\times Q}$$

$$\boldsymbol{\Xi}_k(3,3) = \left[\begin{array}{ccc} \mathbf{J}\boldsymbol{\Phi}_k^1 & \mathbf{J}\boldsymbol{\Phi}_k^2 & \mathbf{J}\boldsymbol{\Phi}_k^3 \\ \hline \mathbf{I}_{2R\times 2R} & \mathbf{0}_{2R\times R} \end{array}\right] \in \mathbb{R}^{3R\times 3R}$$

The closed-loop system could be time-varying since $\boldsymbol{\Xi}_k$ will depend on the network delay characteristics. If the network delays are constant, then the closed-loop system will be time-invariant.

At the first step of analysis procedures, we assume the controller has been designed. Then the bound for different network delays can be found based on the stability criterion that all the eigenvalues of $\boldsymbol{\Xi}_k$ are less than 1. However, in the MIMO case, there are $M + R$ different time delays and it is very difficult to determine the upper bound of delay values. The models (13)–(15) provide the system structure under network delays. For the controller design, although the exact value of the system matrices are unknown, an estimation algorithm can be developed to identify on-line the constant system parameters when the system has constant network delays. For random time delays due to a network protocol or random processing times, a stochastic controller may be used to guarantee stability and performance (Krtolica *et al*. 1994, Tsai and Ray 1997, Nilsson *et al*. 1998).

5. Illustrative example of NCS stability analysis

In this section, we consider a two-axis example of a three-axis milling machine tool. Each axis moves on a linear slide and is driven through a ball screw by a DC motor with a tachometer which provides an angular velocity measurement. The DC motor is driven by a PWM drive. Each axis also has a linear encoder that provides linear position measurement. Therefore, both position and velocity feedback are available. The two axes operate independently. The time constants τ (sec) for each axis are 0.055 (X) and 0.056 (Y) and the overall gains K ((mm/s)/PWM) are 28.346 (X) and 28.956 (Y), respectively.

Then, we define $x_1 = P_x, x_2 = V_x, x_3 = P_y, x_4 = V_y$, $u_1 = u_x$ and $u_2 = u_y$, where P_i and V_i are the position

and velocity variables of the i-axis. The state space form of the two-axis system can be expressed as

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\\ \dot{x}_4\end{bmatrix} = \begin{bmatrix}0 & 1 & 0 & 0\\ 0 & -18.18 & 0 & 0\\ 0 & 0 & 0 & 1\\ 0 & 0 & 0 & -17.86\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\\ x_4\end{bmatrix} + \begin{bmatrix}0 & 0\\ 515.38 & 0\\ 0 & 0\\ 0 & 517.07\end{bmatrix}\begin{bmatrix}u_1\\ u_2\end{bmatrix}$$

We further assume that $c_r^s(k) + p_r^s(k) \le \Delta_r$, $a_m(k) < T$ and $\max(a_m) < T - \max(s_r)$, for $r = 1, \ldots, 4$ and $m = 1, 2$, respectively. These assumptions can be achieved by properly selecting the sampling period T given sensing and actuation delays. Based on Assumptions 4 and 5 in § 3, we have the following relations: $s_r = \Delta_r$, for $r = 1, \ldots, 4$. Hence, from (3), the plant input signals $u_1(t)$ and $u_2(t)$ can be described as follows, for $kT \le t < (k+1)T$: $u_i(t) = v_i(k-1)\mathbf{1}_{(0,a_i)}(t) + v_i(k)\mathbf{1}_{(a_i,T)}(t)$, $i = 1, 2$. Although the model presented in § 4 is valid for time-varying delays, in this illustrative example, we assume that these actuator delays a_m are constant and known in advance, says $a_1 = 1$ ms and $a_2 = 2$ ms. By applying (9), we can obtain the discrete-time plant model (at $T = 10$ ms) as

$$\mathbf{x}_{k+1} = \begin{bmatrix}1 & 0.0091 & 0 & 0\\ 0 & 0.8338 & 0 & 0\\ 0 & 0 & 1 & 0.0092\\ 0 & 0 & 0 & 0.8365\end{bmatrix}\mathbf{x}_k + \begin{bmatrix}0.0198 & 0\\ 4.2788 & 0\\ 0 & 0.0158\\ 0 & 3.8547\end{bmatrix}\mathbf{v}_k + \begin{bmatrix}0.0045 & 0\\ 0.4336 & 0\\ 0 & 0.0086\\ 0 & 0.8807\end{bmatrix}\mathbf{v}_{k-1}$$

We can further calculate the signal $x_r(kT - s_r)$ by obtaining the sensing delays of s_r, say $s_1 = 3$, $s_2 = 4$, $s_3 = 5$ and $s_4 = 6$ (ms). For example, by applying (11), $x_1(kT - s_1)$ can be described as

$$x_1(kT - s_1) = [1 \quad 0.0066 \quad 0 \quad 0]\mathbf{x}_{k-1} + [0.0198 \quad 0]\mathbf{v}_{k-1} + [0.0045 \quad 0]\mathbf{v}_{k-2}$$

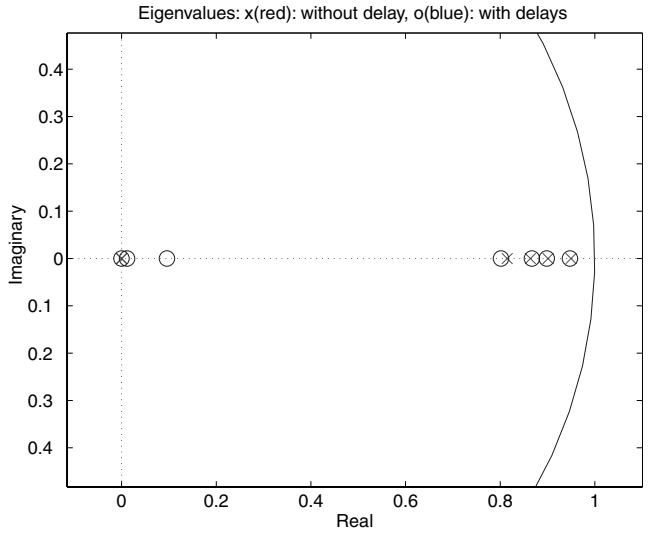
Therefore, (12) becomes

$$\mathbf{w}_k = \begin{bmatrix}1 & 0.0066 & 0 & 0\\ 0 & 0.8966 & 0 & 0\\ 0 & 0 & 1 & 0.0048\\ 0 & 0 & 0 & 0.9311\end{bmatrix}\mathbf{x}_{k-1} + \begin{bmatrix}0.0089 & 0\\ 2.4632 & 0\\ 0 & 0.0023\\ 0 & 1.0159\end{bmatrix}\mathbf{v}_{k-1} + \begin{bmatrix}0.0032 & 0\\ 0.4663 & 0\\ 0 & 0.0040\\ 0 & 0.9803\end{bmatrix}\mathbf{v}_{k-2}$$

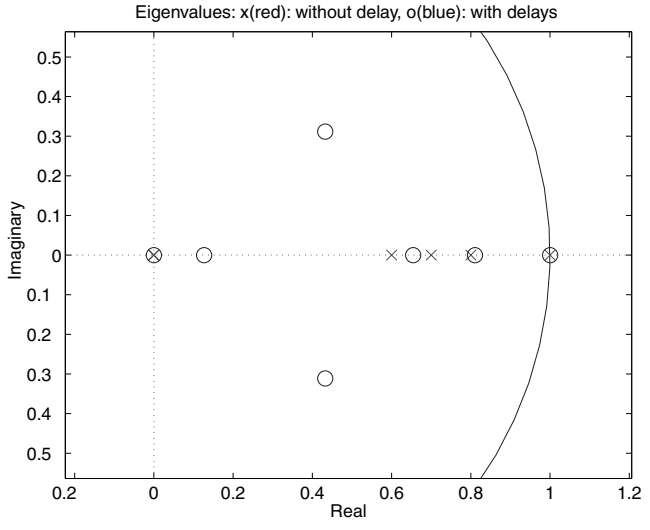
In order to validate the stability and performance of standard controller design, we first consider a memoryless state feedback controller, i.e. $\mathbf{u}(t) = -K\mathbf{x}(t)$, where K is designed based on the pole placement in continuous time domain. In this case, $\mathbf{F}, \mathbf{G}$ and $\mathbf{H}$ are zero matrices of compatible dimension, but $\mathbf{J} = -K$. Therefore, $\mathbf{X}_k = \left[\mathbf{x}_k^{\mathrm{T}} \ \mathbf{x}_{k-1}^{\mathrm{T}} \ \mathbf{z}_k^{\mathrm{T}} \ \mathbf{v}_{k-1}^{\mathrm{T}} \ \mathbf{v}_{k-2}^{\mathrm{T}}\right]^{\mathrm{T}}$. In this example, the system dimension is 13. However, if there are no time delays, then the system dimension becomes 7, i.e. $\mathbf{X}_k = \left[\mathbf{x}_k^{\mathrm{T}} \ \mathbf{z}_k^{\mathrm{T}} \ \mathbf{v}_{k-1}^{\mathrm{T}}\right]^{\mathrm{T}}$. The 13 eigenvalues ('$\circ$') of $\boldsymbol{\Xi}$ are plotted in figure 3 along with the 7 eigenvalues ('$\times$') of the closed-loop system without delays. Figures 3(a) and (b) show two different feedback gains K. The values of time delays (s_r, $r = 1, \ldots, 4$ and a_m, $m = 1, 2$) are identical in figures 3(a) and (b). In each plot, the dotted lines are the real and imaginary axes and the solid line is the unit circle. The '$\circ$' symbols are the locations of the eigenvalues of the closed-loop system with delays and the '$\times$' symbols are those without delays. However, only the four right-most points map to the eigenvalues of original closed-loop system, i.e. eig$(A_p - B_pK)$. From this comparison, we find that the closed-loop systems with multiple time delays will perform differently if the controller designer does not consider these time delays at the first design stage. In fact, at some combination of different time delays and sampling periods, the closed-loop systems could be unstable.

6. Formulation for optimal controller design

In this section, we formulate and solve the optimal controller design problem for the NCS model presented in § 4. The standard control algorithm utilizes the state values at the sampling instants as controller inputs. In an NCS, the sensor signals are not all sampled at the same instants. Hence, in order to apply the linear quadratic regulator (LQR) optimal controller design procedure, we develop a delay transformation which maps an NCS model into a standard control model with delayed states as state variables. Although the model



(a)



(b)

Figure 3. The location of eigenvalues of closed-loop systems. (a) Design 1, (b) Design 2.

presented in § 4 is valid for time-varying delays, for the ease of presentation, we only consider the case where sensor and actuator delays are constant and $\mathbf{y} = \mathbf{x}$, i.e. $N = R$.

The optimal control derivation proceeds as follows. First, we incorporate the sensing and actuation delays into the system model as described in § 4. This results in a delayed state-variable model with the same dimension, ($\mathbf{x} \in \mathbb{R}^N$) as the original system; however, the $\mathbf{A}$ and $\mathbf{B}$ matrices have changed to include the delays. If the delays are constant, as is assumed here, the delayed state-variable model is time-invariant. Standard optimal control techniques are then applied to this delayed state-variable model.

Recall that $\mathbf{x}(kT)$ represents the actual values of the system states at the controller sample times; this value is not measured. $x_r(kT - s_r)$, $r = 1, \ldots, R$ are the values of the rth state received at the controller. Suppose that the sensor delays are all less than one sampling period, i.e. $s_r < T$, $r = 1, \ldots, R$. As stated in (11), we use $\mathbf{x}(k - \tilde{s}_r)$, where $\tilde{s}_r = s_r/T$, to denote the true value of the continuous state at time $(k - \tilde{s}_r)T$. Using (9), we can express the value of the state at the sensor sampling instants, $\mathbf{x}(k + 1 - \tilde{s}_r)$ as

$$\mathbf{x}(k + 1 - \tilde{s}_r) = \mathbf{A}\mathbf{x}(k - \tilde{s}_r) + \sum_{j=0}^{2} \mathbf{B}_r^j \mathbf{v}(k - j) \quad (18)$$

where $\mathbf{B}_r^j = [\, \mathbf{B}_{1r}^j, \ldots, \mathbf{B}_{mr}^j, \ldots, \mathbf{B}_{Mr}^j \,]$ and $\mathbf{B}_{mr}^j(k)$s are defined as

If $T - s_r \geq a_m$, then

$$\mathbf{B}_{mr}^0(k) \triangleq \int_{a_m + s_r}^{T} \mathbf{\Gamma}_m(T, q)\, \mathrm{d}q$$

$$\mathbf{B}_{mr}^1(k) \triangleq \int_{0}^{a_m + s_r} \mathbf{\Gamma}_m(T, q)\, \mathrm{d}q \quad \text{and} \quad \mathbf{B}_{mr}^2(k) \triangleq \mathbf{0}$$

If $T - s_r < a_m$, then

$$\mathbf{B}_{mr}^0(k) \triangleq \mathbf{0}, \quad \mathbf{B}_{mr}^1(k) \triangleq \int_{a_m + s_r - T}^{T} \mathbf{\Gamma}_m(T, q)\, \mathrm{d}q \quad \text{and}$$

$$\mathbf{B}_{mr}^2(k) \triangleq \int_{0}^{a_m + s_r - T} \mathbf{\Gamma}_m(T, q)\, \mathrm{d}q$$

Note that the actuator delays are taken care of in the $\mathbf{B}$ matrix terms. The dependence of $\mathbf{B}_r^j$ on the time k is dropped because all delays are assumed constant in this section. Since the sensor delays are different for each measurement, we need to carefully extract the correct dynamics for each state. The rth element of the vector $\mathbf{x}(\cdot)$, i.e. the individual state $x_r(\cdot)$, can be described as

$$x_r(k + 1 - \tilde{s}_r) = \mathbf{A}^{\langle r,r \rangle} x_r(k - \tilde{s}_r) + \mathbf{A}^{\langle r,-r \rangle} \mathbf{x}_{\langle -r \rangle}(k - \tilde{s}_r) + \sum_{j=0}^{2} \mathbf{B}_r^{j\langle r,* \rangle} \mathbf{v}(k - j) \quad (19)$$

where $\mathbf{A}^{\langle r,r \rangle} \in \mathbb{R}$ is the (r,r)th element of matrix $\mathbf{A}$, $\mathbf{A}^{\langle r,-r \rangle} \in \mathbb{R}^{1 \times (R-1)}$ is the rth row with the rth element deleted, $\mathbf{B}^{\langle r,* \rangle} \in \mathbb{R}^{1 \times M}$ is the rth row of matrix $\mathbf{B}$ and $\mathbf{x}_{\langle -r \rangle}$ is the $\mathbf{x}$ vector with the rth element deleted. $x_r(k - \tilde{s}_r)$ are the sensed values of the rth state that are sampled and sent over the network.

In the meantime, by using (10) and letting $\delta = \tilde{s}_r$, we can also express the sensed state $\mathbf{x}(k + 1 - \tilde{s}_r)$,

$r = 1, \ldots, R$, in terms of the state of the system at the controller sample instant, $\mathbf{x}(k)$, as

$$\mathbf{x}(k+1-\tilde{s}_r) = \mathbf{A}_{\tilde{s}_r}\mathbf{x}(k) + \sum_{j=0}^{1}\mathbf{B}_{\tilde{s}_r}^{j}\mathbf{v}(k-j) \tag{20}$$

Then, we can separate the rth element from the rest of the vector $\mathbf{x}(\cdot)$ as

$$x_r(k+1-\tilde{s}_r) = \mathbf{A}_{\tilde{s}_r}^{\langle r,*\rangle}\mathbf{x}(k) + \sum_{j=0}^{1}\mathbf{B}_{\tilde{s}_r}^{j\langle r,*\rangle}\mathbf{v}(k-j), \tag{21}$$

and

$$\mathbf{x}_{\langle -r\rangle}(k+1-\tilde{s}_r) = \mathbf{A}_{\tilde{s}_r}^{\langle -r,*\rangle}\mathbf{x}(k) + \sum_{j=0}^{1}\mathbf{B}_{\tilde{s}_r}^{j\langle -r,*\rangle}\mathbf{v}(k-j) \tag{22}$$

In order to formulate the system dynamics in terms of measured states, we further define the new variables

$$\mathbf{x}_s(k) = \begin{bmatrix} x_1(k-s_1) \\ x_2(k-s_2) \\ \vdots \\ x_R(k-s_R) \end{bmatrix}, \qquad \mathbf{A}_{s*} = \begin{bmatrix} \mathbf{A}_{\tilde{s}_1}^{\langle 1,*\rangle} \\ \mathbf{A}_{\tilde{s}_2}^{\langle 2,*\rangle} \\ \vdots \\ \mathbf{A}_{\tilde{s}_R}^{\langle R,*\rangle} \end{bmatrix}$$

and

$$\mathbf{B}_{s*}^{j} = \begin{bmatrix} \mathbf{B}_{\tilde{s}_1}^{j\langle 1,*\rangle} \\ \mathbf{B}_{\tilde{s}_2}^{j\langle 2,*\rangle} \\ \vdots \\ \mathbf{B}_{\tilde{s}_R}^{j\langle R,*\rangle} \end{bmatrix}$$

$\mathbf{x}_s(k)$ will be the state variables of the new delayed state-variable model. By applying (21) for $r = 1, \ldots, R$, we have the equation

$$\mathbf{x}_s(k+1) = \mathbf{A}_{s*}\mathbf{x}(k) + \sum_{j=0}^{1}\mathbf{B}_{s*}^{j}\mathbf{v}(k-j) \tag{23}$$

Since $\mathbf{A}_{s*}$ is the combination of $\mathbf{A}_{\tilde{s}_r}$, i.e. $\exp(A_p\tilde{s}_r T)$, which is non-singular, we can multiply both sides of (23) by $\mathbf{A}_{s*}^{-1}$ and obtain the actual value of the state at the controller sample instant as a function of the sensed (delayed) values of the states and the inputs

$$\mathbf{x}(k) = \mathbf{A}_{s*}^{-1}\mathbf{x}_s(k+1) - \mathbf{A}_{s*}^{-1}\sum_{j=0}^{1}\mathbf{B}_{s*}^{j}\mathbf{v}(k-j) \tag{24}$$

We further apply (24) to (22) at time $k-1$ to find the values of the other system states at the instant when the rth state is sampled

$$\begin{aligned}\mathbf{x}_{\langle -r\rangle}(k-\tilde{s}_r) &= \mathbf{A}_{\tilde{s}_r}^{\langle -r,*\rangle}\left[\mathbf{A}_{s*}^{-1}\mathbf{x}_s(k) - \mathbf{A}_{s*}^{-1}\sum_{j=0}^{1}\mathbf{B}_{s*}^{j}\mathbf{v}(k-1-j)\right] \\ &\quad + \sum_{j=0}^{1}\mathbf{B}_{\tilde{s}_r}^{j<-r,*>}\mathbf{v}(k-1-j) \\ &= \mathbf{A}_{\tilde{s}_r}^{x_r}\mathbf{x}_s(k) + \sum_{j=0}^{1}\mathbf{B}_{\tilde{s}_r}^{x_r j}\mathbf{v}(k-1-j)\end{aligned} \tag{25}$$

where the first equality is from (24) at time $k-1$, and, in the second equality, we use the definitions $\mathbf{A}_{\tilde{s}_r}^{x_r} \triangleq \mathbf{A}_{\tilde{s}_r}^{\langle -r,*\rangle}\mathbf{A}_{s*}^{-1}$, and $\mathbf{B}_{\tilde{s}_r}^{x_r j} \triangleq \mathbf{B}_{\tilde{s}_r}^{j\langle -r,*\rangle} - \mathbf{A}_{\tilde{s}_r}^{\langle -r,*\rangle}\mathbf{A}_{s*}^{-1}\mathbf{B}_{s*}^{j}$. Hence, by further plugging (25) into (19), we have

$$\begin{aligned}x_r(k+1-\tilde{s}_r) &= \mathbf{A}^{\langle r,r\rangle}x_r(k-\tilde{s}_r) + \sum_{j=0}^{2}\mathbf{B}_r^{j\langle r,*\rangle}\mathbf{v}(k-j) \\ &\quad + \mathbf{A}^{\langle r,-r\rangle}\left\{\left[\mathbf{A}_{\tilde{s}_r}^{\langle -r,*\rangle}\mathbf{A}_{s*}^{-1}\right]\mathbf{x}_s(k)\right. \\ &\quad \left. + \sum_{j=0}^{1}\left[\mathbf{B}_{\tilde{s}_r}^{j\langle -r,*\rangle} - \mathbf{A}_{\tilde{s}_r}^{\langle -r,*\rangle}\mathbf{A}_{s*}^{-1}\mathbf{B}_{s*}^{j}\right]\mathbf{v}(k-1-j)\right\}\end{aligned} \tag{26}$$

Therefore, by using the definition of $\mathbf{x}_s(k)$, we can obtain the new delayed state-variable model as

$$\mathbf{x}_s(k+1) = \mathbf{A}_{x_s}\mathbf{x}_s(k) + \sum_{j=0}^{2}\mathbf{B}_{x_s}^{j}\mathbf{v}(k-j) \tag{27}$$

where

$$\mathbf{A}_{x_s} = \begin{bmatrix} \mathbf{A}^{(1,1)} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \mathbf{A}^{(R,R)} \end{bmatrix} + \begin{bmatrix} \mathbf{A}^{\langle 1,-1\rangle}\mathbf{A}_{\tilde{s}_1}^{\langle -1,*\rangle}\mathbf{A}_{s*}^{-1} \\ \vdots \\ \mathbf{A}^{\langle R,-R\rangle}\mathbf{A}_{\tilde{s}_R}^{\langle -R,*\rangle}\mathbf{A}_{s*}^{-1} \end{bmatrix}$$

and

$$B_{x_s}^{j} = \begin{cases} \begin{bmatrix} \mathbf{B}_1^{0\langle 1,*\rangle} \\ \vdots \\ \mathbf{B}_R^{0\langle R,*\rangle} \end{bmatrix} & j = 0 \\ \begin{bmatrix} \mathbf{B}_1^{j\langle 1,*\rangle} + \mathbf{A}^{\langle 1,-1\rangle}\left[\mathbf{B}_{\tilde{s}_1}^{j-1\langle -1,*\rangle} - \mathbf{A}_{\tilde{s}_1}^{\langle -1,*\rangle}\mathbf{A}_{s*}^{-1}\mathbf{B}_{s*}^{j-1}\right] \\ \vdots \\ \mathbf{B}_R^{j\langle R,*\rangle} + \mathbf{A}^{\langle R,-R\rangle}\left[\mathbf{B}_{\tilde{s}_R}^{j-1\langle -R,*\rangle} - \mathbf{A}_{\tilde{s}_R}^{\langle -R,*\rangle}\mathbf{A}_{s*}^{-1}\mathbf{B}_{s*}^{j-1}\right] \end{bmatrix}, & j = 1,2 \end{cases}$$

We further define $\mathbf{z}(k) = [\,\mathbf{x}_s(k)^{\mathrm{T}}, \mathbf{v}(k-1)^{\mathrm{T}}, \mathbf{v}(k-2)^{\mathrm{T}}\,]^{\mathrm{T}} \in \mathbb{R}^{N+2R}$. $\mathbf{z}(k)$ will be the new state variables in the standard optimal control design technique. Then, we have the state-space model for controller design

$$\mathbf{z}(k+1) = \mathbf{A}_z\mathbf{z}(k) + \mathbf{B}_z\mathbf{v}(k) \tag{28}$$

where

$$\mathbf{A}_z = \begin{bmatrix} \mathbf{A}_{x_s} & \mathbf{B}^1_{x_s} & \mathbf{B}^2_{x_s} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} \end{bmatrix} \in \mathbb{R}^{(N+2R)\times(N+2R)}$$

and

$$\mathbf{B}_z = \begin{bmatrix} \mathbf{B}^0_{x_s} \\ \mathbf{I} \\ \mathbf{0} \end{bmatrix} \in \mathbb{R}^{(N+2R)\times R}$$

For the controller design of tracking problem, we first let the desired trajectory $\mathbf{d}(t)$ or $\mathbf{d}(k)$ be described by the equation

$$\left.\begin{array}{ll} \dot{\mathbf{d}}(t) = \mathbf{A}^c_d \mathbf{d}(t), & \text{in continuous time} \\ \text{or} & \\ \mathbf{d}(k+1) = \mathbf{A}^d_d \mathbf{d}(k), & \text{in discrete time} \end{array}\right\} \quad (29)$$

where $\mathbf{d}(\cdot) = [d_1(\cdot), \ldots, d_R(\cdot)]^{\mathrm{T}}$. Furthermore, since the new state variables are measured at time $k - \tilde{s}_r$, we define the desired trajectory at each individual time instant $k - \tilde{s}_r$ as $\mathbf{d}_s(k) \triangleq [\, d_1(k - \tilde{s}_1), \ldots, d_R(k - \tilde{s}_R)\,]^{\mathrm{T}}$. For the optimal controller design, we then choose the cost function V

$$V = \sum_{k=0}^{N} [\mathbf{z}(k) - \mathbf{z}_d(k)]^{\mathrm{T}} \mathbf{Q} [\mathbf{z}(k) - \mathbf{z}_d(k)] + \mathbf{v}(k)^{\mathrm{T}} \mathbf{R} \mathbf{v}(k) \quad (30)$$

where $\mathbf{z}_d(k) \triangleq [\mathbf{d}_s(k)^{\mathrm{T}}, \mathbf{0}^{\mathrm{T}}]^{\mathrm{T}}$, i.e. the combined vector of desired trajectory and zero input, and $\mathbf{Q}$ and $\mathbf{R}$ are defined as

$$\mathbf{R} = \mathbf{R}_0, \quad \text{and} \quad \mathbf{Q} = \begin{bmatrix} \mathbf{Q}_s & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{R}_2 \end{bmatrix}$$

and $\mathbf{Q}_s$ and $\mathbf{R}_i$ are the weighting matrices for the original system states and inputs, respectively. Based on the standard LQR algorithm (Lewis 1986), the optimal controller law $\mathbf{v}^*(k)$ can be obtained as

$$\begin{aligned} \mathbf{v}^*(k) &= -\mathbf{K}(k)[\mathbf{z}(k) - \mathbf{z}_d(k)] \\ &= -\mathbf{K}_s(k)[\mathbf{x}_s(k) - \mathbf{d}_s(k)] - \sum_{i=1}^{2} \mathbf{K}^i_v(k) \mathbf{v}^*(k-i) \end{aligned} \quad (31)$$

where $\mathbf{K} = \left[\mathbf{K}_s,\ \mathbf{K}^1_v, \mathbf{K}^2_v\right]$. The new optimal controller utilizes the available measured states as well as past inputs as the controller inputs. The number of past inputs depends on the sensor and actuator delays as described in (14) and (15). An NCS-LQR design example for the illustrative example presented in § 5 will be presented in the next section.

7. Numerical examples of NCS-LQR

In this section, we revisit the two-axis example in § 5. Three cases of the optimal controller design will be compared. The first case applies the standard LQR controller for the original delay-free system. In the second case, the LQR controller that is designed based on delay-free system is used to control the delayed system. The final case considers the LQR design based on the delayed state-variable model discussed in § 6.

The actuation and sensing delays are assumed as follows: $a_1 = 1$, $a_2 = 2$, $s_1 = 3$, $s_2 = 4$, $s_3 = 5$ and $s_4 = 6$ (ms) and the sampling time is 10 ms. The choice of this delay characteristics is based on the performance analysis of one type of priority-based control networks such as DeviceNet™† (Lian *et al.* 2001 a). Also, based on the assumptions used in § 5, equation (27) has the form

$$\begin{aligned} \mathbf{x}_s(k+1) = & \begin{bmatrix} 1 & 0.0090 & 0 & 0 \\ 0 & 0.8338 & 0 & 0 \\ 0 & 0 & 1 & 0.0090 \\ 0 & 0 & 0 & 0.8365 \end{bmatrix} \mathbf{x}_s(k) \\ & + \begin{bmatrix} 0.0323 & 0 \\ 3.9787 & 0 \\ 0 & 0.0342 \\ 0 & 3.4630 \end{bmatrix} \mathbf{v}(k) \\ & + \begin{bmatrix} 0.0104 & 0 \\ 0.4032 & 0 \\ 0 & 0.0175 \\ -0.0088 & 0.7912 \end{bmatrix} \mathbf{v}(k-1) + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \mathbf{v}(k-2) \end{aligned}$$

Furthermore, the new delayed state-variable model (28) becomes

$$\begin{aligned} \mathbf{z}(k+1) = & \begin{bmatrix} 1 & 0.0090 & 0 & 0 & 0.0104 & 0 & 0 & 0 \\ 0 & 0.8338 & 0 & 0 & 0.4032 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0.0090 & 0 & 0.0175 & 0 & 0 \\ 0 & 0 & 0 & 0.8365 & -0.0088 & 0.7912 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \mathbf{z}(k) \\ & + \begin{bmatrix} 0.0323 & 0 \\ 3.9787 & 0 \\ 0 & 0.0342 \\ 0 & 3.4630 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \mathbf{v}(k) \end{aligned}$$

† DeviceNet is a sensor bus communication protocol commonly used as in an NCS. DeviceNet utilizes a non-destructive collision resolution scheme through message priority.

For the LQR optimal controller design, we first choose the weighting matrices

$$\mathbf{Q}_s = \mathrm{diag}(20, 0.05, 20, 0.05)$$

$$\mathbf{R}_0 = \mathrm{diag}(0.1, 0.1), \quad \mathbf{R}_1 = \mathrm{diag}(0.1, 0.1)$$

and

$$\mathbf{R}_2 = \mathrm{diag}(0.001, 0.001)$$

Most of the cost function is weighted upon position tracking. Since $\mathbf{v}(k), \mathbf{v}(k-1)$ and $\mathbf{v}(k-2)$ are not independent, we discount the weights on previous inputs. By using the discrete-time system matrix $\mathbf{A}$ and input matrix $\mathbf{B}$, and the weighting matrices $\mathbf{Q}_s$ and $\mathbf{R}_0$, the LQR state feedback gain, called $\mathbf{K}_{lqr}$, is computed as

$$\mathbf{K}_{lqr} = \begin{bmatrix} 3.6038 & 0.1829 & 0 & 0 \\ 0 & 0 & 3.5879 & 0.1827 \end{bmatrix}$$

The gain is the same as the standard LQR design for the delay-free systems. On the other hand, by using the NCS system matrix $\mathbf{A}_z$, input matrix $\mathbf{B}_z$ and the weighting matrices $\mathbf{Q}$ and $\mathbf{R}$, the NCS-LQR state feedback gain, called $\mathbf{K}_{lqr}^{ncs}$, is obtained as

$$\mathbf{K}_{lqr}^{ncs} = \begin{bmatrix} 3.7917 & 0.1802 & 0 & 0 & 0.1102 & 0 & 0 & 0 \\ 0.0002 & 0 & 4.0722 & 0.1881 & -0.0016 & 0.2145 & 0 & 0 \end{bmatrix}$$

$\mathbf{K}_{lqr}^{ncs}$ is computed based on the proposed NCS delayed state-variable model discussed in § 6.

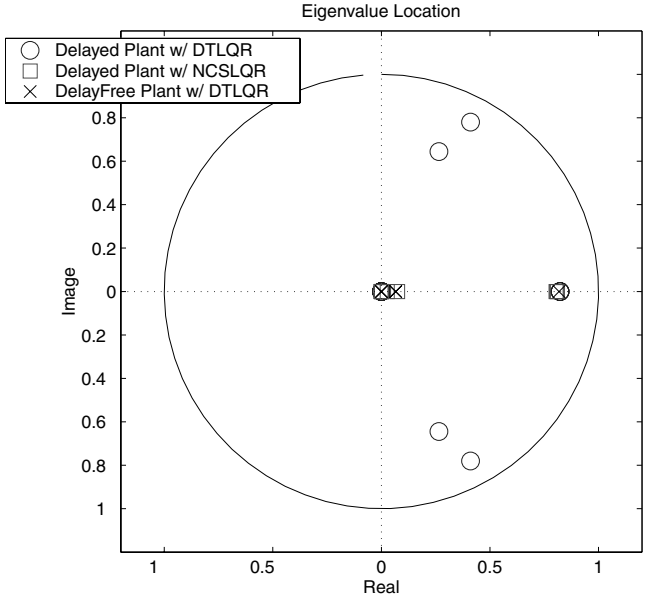


Figure 4. Eigenvalue location of three different closed-loop systems: $\mathbf{K}_{lqr}$ for delay-free system, $\mathbf{K}_{lqr}$ for delayed system and $\mathbf{K}_{lqr}^{ncs}$ for delayed system.

Simulation results of the first LQR design are shown in figures 4–6. Figure 4 shows the location of eigenvalues of three different closed-loop systems: (1) the discrete-time LQR (DT-LQR) controller for the delay-free system, (2) the DT-LQR controller for the delayed

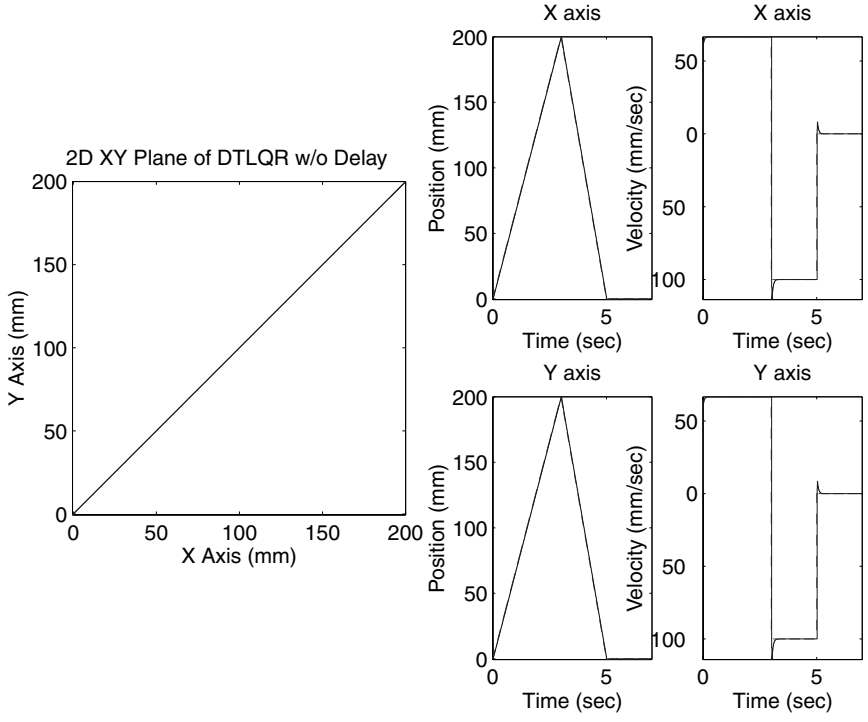


Figure 5. Standard LQR controller design: $\mathbf{K}_{lqr}$ for delay-free system.

system, and (3) the NCS-LQR controller for the delayed system. The eigenvalue locations for the NCS-LQR case are closer to those of the DT-LQR for the delay-free system than those of the DT-LQR for the delayed system.

Figure 5 shows the simulation results of the tracking performance of the two-axis system, i.e. X and Y axes, with the DT-LQR controller for the delay-free system. Here, two constant feed-rates for forward and backward motions are considered as the desired trajectory for both X and Y axes. For this standard LQR case where time delays are not considered, $\mathbf{v}(k) = -\mathbf{K}_{lqr}[\mathbf{x}(k) - \mathbf{d}(k)]$ is used as controller input. The plot on the left-hand side is the top-view trajectory, and the four plots on the right-hand side are the positions and velocities of X and Y axes. This case can be used as a basis for comparison. Figure 6 shows the summary of the tracking errors with different indexes. The left-hand plot is the root-mean-square tracking errors of (1) the entire simulation time (0–7 s), (2) the first second (0–1 s), and (3) the last two seconds (5–7 s). The right-hand plot is the result of the integral of the time multiplied by the absolute values of the error (ITAE). The NCS-LQR controller improves the control performance over the DT-LQR controller when considering the time delays.

The simulation results of two other desired trajectories, a circle with a radius of 100 mm, and constant trajectories for both position and velocity, are also shown in figures 7–10. The circular trajectory can be viewed as a time-varying trajectory; the constant trajectory is a regulation case for both position and velocity. For both types of trajectories, the proposed NCS-LQR guarantees better control performance than the DT-LQR. Especially, the 2-D top-view plots in figures 7

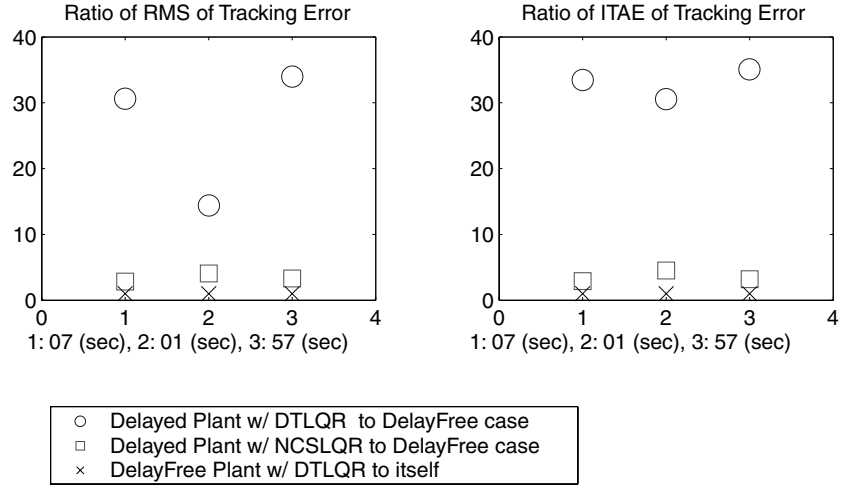


Figure 6. The summary of tracking errors: $\mathbf{K}_{lqr}$ for delay-free system, $\mathbf{K}_{lqr}$ for delayed system and $\mathbf{K}_{lqr}^{ncs}$ for delayed system. Left: RMS, right: ITAE.

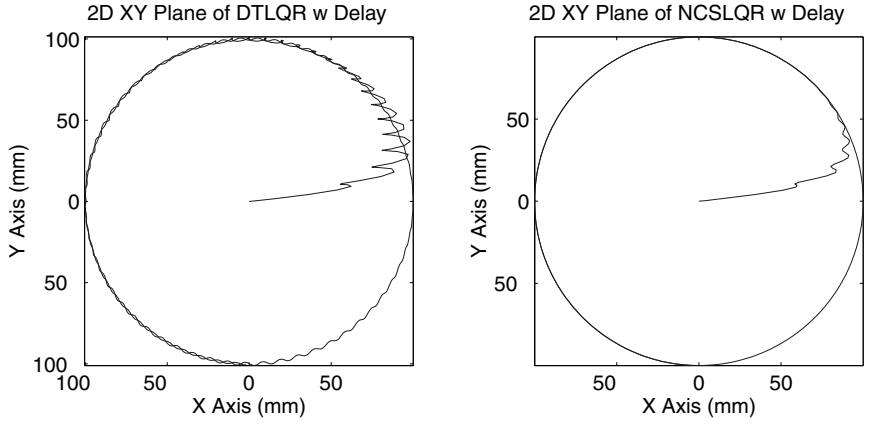


Figure 7. The 2-D top view of 'circular' X–Y axis system: Left: $\mathbf{K}_{lqr}$ for delayed system; right: $\mathbf{K}_{lqr}^{ncs}$ for delayed system.

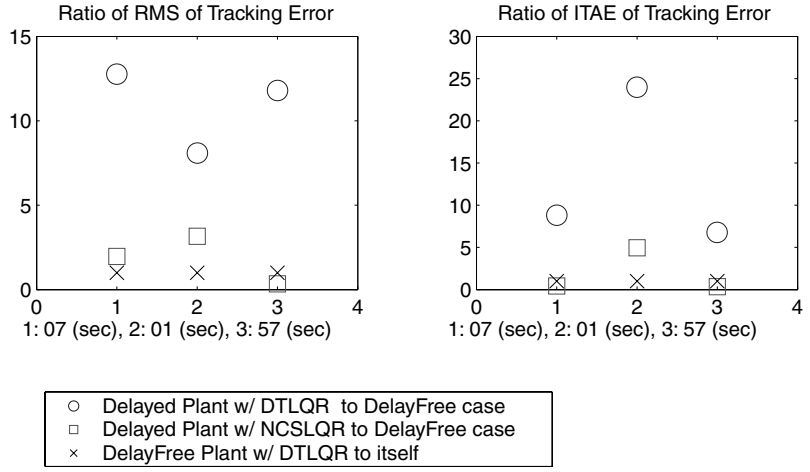


Figure 8. The summary of 'circular' tracking errors: $\mathbf{K}_{lqr}$ for delay-free system, $\mathbf{K}_{lqr}$ for delayed system and $\mathbf{K}_{lqr}^{ncs}$ for delayed system. Left: RMS, right: ITAE.

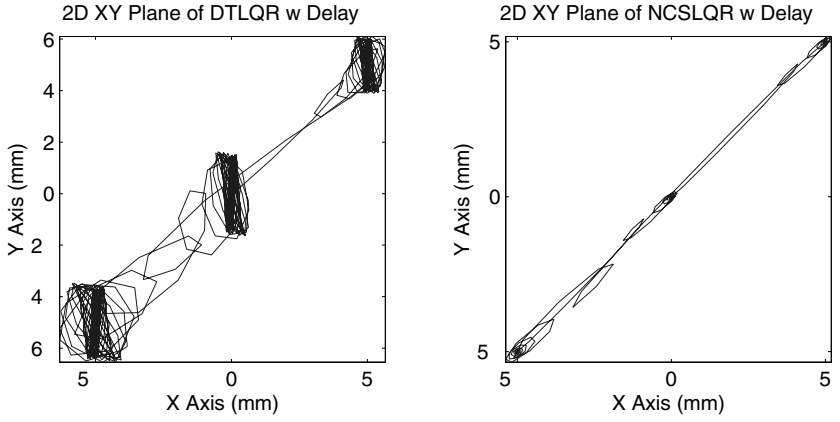


Figure 9. The 2-D top view of 'constant' X–Y axis system: left: $\mathbf{K}_{lqr}$ for delayed system; right: $\mathbf{K}_{lqr}^{ncs}$ for delayed system.

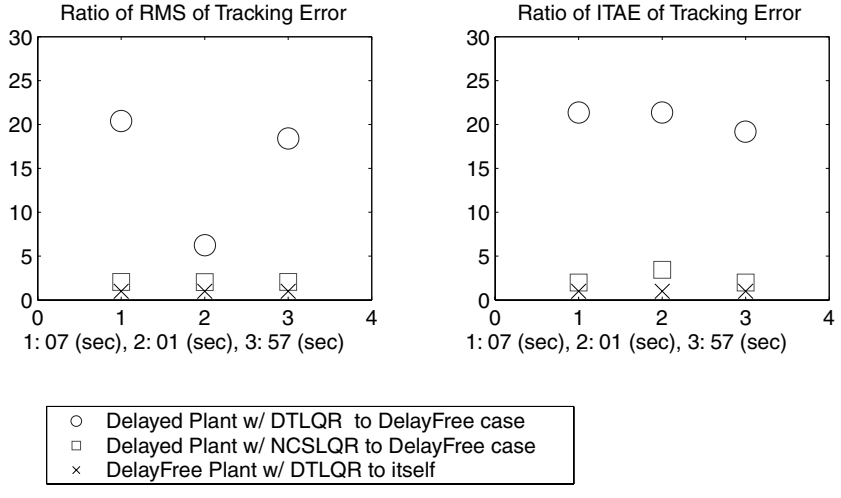


Figure 10. The summary of 'constant' tracking errors: $\mathbf{K}_{lqr}$ for delay-free system, $\mathbf{K}_{lqr}$ for delayed system and $\mathbf{K}_{lqr}^{ncs}$ for delayed system. Left: RMS, right: ITAE.

and 9 show the dramatic improvement of the proposed NCS-LQR controller design.

8. Summary and future work

In this paper we analysed and modelled a MIMO networked control system with multiple time delays. The time delays between sensor-controller and controller-actuator and the time skews at different devices' sampling instants were considered and included in the derivation of a discrete-time MIMO model. By including these time delay parameters, both the control system and network system designers can utilize this model to design networked control systems and optimize their overall performance.

The closed-loop NCS model discussed in § 4 only included a standard controller designed without considering the time delay effect *a priori*. Therefore, this closed-loop stability analysis was used as a verification tool for the stability and performance of an NCS. Based on the time-delay modelling algorithm and considering the case of constant communication delays, a delayed state-variable model was formulated for the standard LQR controller design in § 6. This NCS-LQR controller improved the control performance over the DT-LQR controller when considering the time delays. Simulation studies provided in §§ 5 and 7 demonstrated the utility of the proposed design algorithms for stability analysis and performance improvement.

The advantage of having an NCS architecture is that it provides the flexibility to quickly reconfigure the system architecture, and to easily share information with other subsystems. The change of system configuration may also change the time delay of a networked device even in a deterministic NCS. Hence, the advanced networked controller should be able to maintain a proper level of network traffic load and adaptively modify the controller algorithm. This slowly changing situation can be modelled as an adaptive control problem which can adjust on-line to the changing system parameters. Also, before a truly deterministic network is available for control applications, the time-varying delays should be included in the modelling algorithm, and a stochastic controller design could be adopted. For the uncertainties existing in an NCS such as the variance of time delays and system uncertainty, a robust controller design could then be applied. In addition, an intelligent controller design that incorporates both network and control parameters to further improve both network and control performance should be studied in the future. This intelligent networked controller can identify/estimate network performance and operate among multiple modes of control action, depending on network traffic or control performance.

Acknowledgements

The authors are grateful to the anonymous referees for their careful reading of the paper and for their pertinent suggestions. This research was supported in part by the NSF Engineering Research Center for Reconfigurable Machining Systems, University of Michigan, under the grant EEC95-92125, the DARPA MICA Program, the New Faculty Research Fund at National Taiwan University and the National Science Council, Taiwan, ROC, under the grant NSC 91-2213-E-002-133.

References

Branicky, M. S., Phillips, S. M., and Zhang, W., 2000, Stability of networked control systems: explicit analysis of delay. *Proceedings of 2000 American Control Conference*, Chicago, IL, USA, pp. 2352–2357.

Dugard, L., and Verrest, E. I., 1998, *Stability and Control of Time-delay Systems* (Berlin–Heidelberg–New York: Springer-Verlag).

Franklin, G. F., Powell, J. D., and Workman, M. L., 1998, *Digital Control of Dynamic Systems*, third edition (Menlo Park, CA: Addison-Wesley).

Göktas, F., Smith, J. M., and Bajcsy, R., 1996, μ-Synthesis for distributed control systems with network-induced delays. *Proceedings of the 35th Conference on Decision and Control*, Kobe, Japan, pp. 813–814.

Göktas, F., Smith, J. M., and Bajcsy, R., 1997, Telerobotics over communication networks. *Proceedings of the 36th Conference on Decision and Control*, San Diego, CA, USA, pp. 2399–2404.

Górecki, H., Fuksa, S., Grabowski, P., and Korytowski, A., 1989, *Analysis and Synthesis of Time Delay Systems* (New York: Wiley).

Goubet-Bartholoméüs, A., Dambrine, M., and Richard, J. P., 1997, Stability of perturbed systems with time-varying delays. *Systems and Control Letters*, **31**, 155–163.

Gu, K., 1999, Discretized Lyapunov functional for uncertain systems with multiple time-delay. *International Journal of Control*, **72**, 1436–1445.

Gu, Y., Wang, S., Li, Q., Cheng, Z., and Qian, J., 1998, On delay-dependent stability and decay estimate for uncertain systems with time-varying delay. *Automatica*, **34**, 1035–1039.

Halevi, Y., and Ray, A., 1988, Integrated communication and control systems: Part I—analysis. *ASME Journal of Dynamic Systems, Measurement, and Control*, **110**, 367–373.

Hsiao, F.-H., and Hwang, J.-D., 1997, Stability analysis of uncertain feedback systems with multiple time delays and series nonlinearities. *Journal of Franklin Institute*, **334**, 491–505.

Jeung, E. T., Oh, D. C., Kim, J. H., and Park, H. B., 1996, Robust controller design for uncertain systems with time delays: LMI approach. *Automatica*, **32**, 1229–1331.

Kim, Y. H., Kwon, W. H., and Park, H. S., 1996, Stability and a scheduling method for network-based control systems. *IECON Proceedings*, Taipei, Taiwan, pp. 934–939.

Krtolica, R., Özgüner, Ü, Chan, H., Göktas, H., Winkelman, J., and Liubakka, M., 1994, Stability of linear

feedback systems with random communication delays. *International Journal of Control*, **59**, 925–953.

Lewis, F. L., 1986, *Optimal Control* (New York: Wiley).

Li, X., and de Souza, C. E., 1997, Delay-dependent robust stability and stabilization of uncertain linear delay systems: a linear matrix inequality approach. *IEEE Transactions on Automatic Control*, **42**, 1144–1148.

Lian, F.-L., Moyne, J. R., and Tilbury, D. M., 2001 a, Performance evaluation of control networks: Ethernet, ControlNet, and DeviceNet. *IEEE Control Systems Magazine*, **21**, 66–83.

Lian, F.-L., Moyne, J. R., and Tilbury, D. M., 2001 b, Analysis and modeling of networked control systems: MIMO case with multiple time delays. *Proceedings of 2001 American Control Conference*, Arlington, VA, USA, pp. 4306–4312.

Lian, F.-L., Moyne, J. R., and Tilbury, D. M., 2002 a, Network design consideration for distributed control systems. *IEEE Transactions on Control Systems Technology*, **10**, 297–307.

Lian, F.-L., Moyne, J. R., and Tilbury, D. M., 2002 b, Optimal controller design and evaluation for a class of networked control systems with distributed constant delays. *Proceedings of 2002 American Control Conference*, Anchorage, AL, USA, pp. 3009–3014.

Luo, J. S., van den Bosch, P. P. J., Weiland, S., and Goldenberge, A., 1998, Design of performance robustness for uncertain linear systems with state and control delays. *IEEE Transactions on Automatic Control*, **43**, 1593–1596.

Malek-Zavarei, M., and Jamshidi, M., 1987, *Time-Delay Systems: Analysis, Optimization and Applications* (North-Holland: Amsterdam).

Marshall, J. E., 1979, *Control of Time-Delay Systems* (Petter Peregrinus: Stevenage UK).

Nilsson, J., 1998, Real-time control systems with delays. PhD thesis, Lund Institute of Technology, Lund, Sweden.

Nilsson, J., and Bernhardsson, B., 1997, LQG control over a Markov communication network. *Proceedings of the 36th Conference on Decision and Control*, San Diego, CA, USA, pp. 4586–4591.

Nilsson, J., Bernhardsson, B., and Wittenmark, B., 1998, Stochastic analysis and control of real-time systems with random time delays, *Automatica*, **34**, 57–64.

Oğuztöreli, M. N., 1966, *Time-lag Control Systems* (New York: Academic Press).

Tsai, N.-C., and Ray, A., 1997, Stochastic optimal control under randomly varying distributed delays. *International Journal of Control*, **68**, 1179–1202.

Walsh, G. C., Beldiman, O., and Bushnell, L., 1999 a, Asymptotic behavior of networked control systems. *Proceedings of the International Conference on Control Applications*, Hawaii, USA, pp. 1448–1453.

Walsh, G. C., Ye, H., and Bushnell, L., 1999 b, Stability analysis of networked control systems. In *Proceedings of the American Control Conference*, San Diego, CA, USA, pp. 2876–2889.

Wittenmark, B., Nilsson, J., and Torngren, M., 1995, Timing problems in real-time control systems. *Proceedings of American Control Conference*, Seattle, WA, USA, pp. 2000–2004.