Proceedings of the American Control Conference
Arlington, Virginia, June 25-27, 2001

Analysis and Modeling of Networked Control Systems: MIMO Case with Multiple Time Delays

Feng-Li Lian[1], James Moyne[2], Dawn Tilbury[1]
NSF-Engineering Research Center for Reconfigurable Machining Systems
1. Department of Mechanical Engineering
2. Department of Electrical Engineering and Computer Science
The University of Michigan, Ann Arbor, MI 48109-2125[1]/2101[2]
{fengli,moyne,tilbury}@umich.edu

Abstract

This paper discusses the modeling and analysis of Networked Control Systems (NCSs) where sensors, actuators, and controllers are distributed and interconnected by a common communication medium. Therefore, multiple distributed communication delays as well as multiple inputs and multiple outputs are considered in the modeling 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 analyze the stability and performance of a closed-loop NCS. The analysis result is used to verify the stability and performance of an NCS if the controller is designed without considering the impact of multiple time delays. Also, the NCS model provided can be used as a foundation for further controller design to compensate for the distributed communication delays.

1 Introduction

The trend of modern manufacturing systems such as Flexible Manufacturing Systems (FMS) and Reconfigurable Manufacturing Systems (RMS) is to integrate Computer, Communication and Control into different levels of information processes and factory operations [1]. For example, an integrated system might include computer-numerically-controlled (CNC) machines, computer-aided design (CAD) tools, supervisory controllers, and intelligent monitoring devices. The communication medium is a key component in these advanced manufacturing systems. The recent introduction of control network "bus" architectures provides a way to improve the efficiency, flexibility and reliability of these future manufacturing systems. With these new control architectures, all devices are interconnected by one common bus communication medium, and, thus, information can be easily exchanged. Network architectures provide the features of easy installation and reconfiguration and can also reduce the setup and maintenance costs. The health of a system can also be monitored by adding extra detecting tools or sensing devices.

These control systems where sensors, actuators, and controllers are interconnected by communication networks are called Networked Control Systems (NCSs). Research in NCSs is different from that in traditional time-delay systems where time delays are assumed to be constant or bounded. Because of the variability of network-induced time delays, the NCSs may be time-varying systems which makes analysis and design more difficult. Wittenmark et al. [12] discussed several timing issues such as communication and computation delays, processor jitter, and transient errors, existing in NCSs. Those 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 [8], [9]. Nilsson studied the case with multiple sensor-controller and controller-actuator delays [7]. However, only the case where the total maximum network delay is less than one controller sampling period was considered.

Recently research on the analysis and modeling of NCSs has been conducted using continuous-time and discrete-time models. It is more natural to analyze an NCS from the discrete-time point of view since, in 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 [3] considered the case of one single time delay for sensor-controller and controller-actuator and one 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.

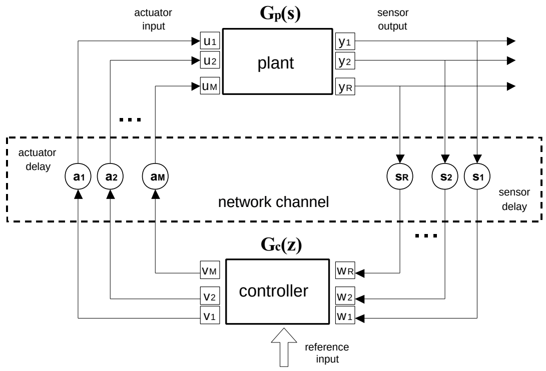

Figure 1: The closed-loop block diagram of a networked control system.

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. By further analyzing the network protocols, Lian et al. characterized the network delays for different industrial networks [5] and studied the inherent tradeoffs between network bandwidth and control sampling rates [4]. Hence, a proper time delay profile in NCSs can be characterized based on the network transmission bandwidth and control system bandwidth.

In this study, we will first focus on NCSs in which the closed-loop model can be represented by the block diagram shown in Fig. 1 [7]. This is a multi-input and multi-output (MIMO) system where every sensor and actuator encounters different network induced time delays and time skews related to the controller sampling instants. The characteristics of these time delays depend on the network protocol used. The time-delay characterization of typical control networks can be found in [5]. A fundamental model of the plant and controller in NCSs with delays will be derived in Section 2. The impact of network transmission delays on the relationship between the inputs and outputs of the plant and controller is discussed in Section 3. In Section 4 a closed-loop model and stability analysis is provided and used for controller design issue. Section 5 presents a short example, and concluding remarks are summarized in Section 6.

2 Fundamental Models of Plant and Controller

Consider the block diagram of a networked control system with a single controller, but multiple sensors and multiple actuators as shown in Fig. 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, Q are positive constant numbers. We use $s_1, ..., s_R$ and $a_1, ..., a_M$ 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 we will discuss the system models in continuous time and discrete time. Time in this paper is denoted by t for the continuous-time domain and k for the discrete-time domain.

In Fig. 1, the continuous-time, state-space model of the linear time-invariant plant dynamics, $G_p(s)$, can be described by the following standard form:

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

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 the discrete-time and the state-space model of the controller dynamics, $G_c(z)$, can be expressed as follows:

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

where $\mathbf{z}_k = \mathbf{z}(k) = \mathbf{z}(kT) \in \mathbb{R}^Q, \mathbf{w}_k = \mathbf{w}(k) = \mathbf{w}(kT) \in \mathbb{R}^R, \mathbf{v}_k = \mathbf{v}(k) = \mathbf{v}(kT) \in \mathbb{R}^M$ 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 study, we assume that the network is not synchronized; each device may have a different time skew when related to the controller sampling instants. We also assume that the sensor and controller sampling rates 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 illustrated in Fig. 2.

1. As shown in Fig. 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 $\Delta_r, r = 1, ..., R$, 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_r's are constant.

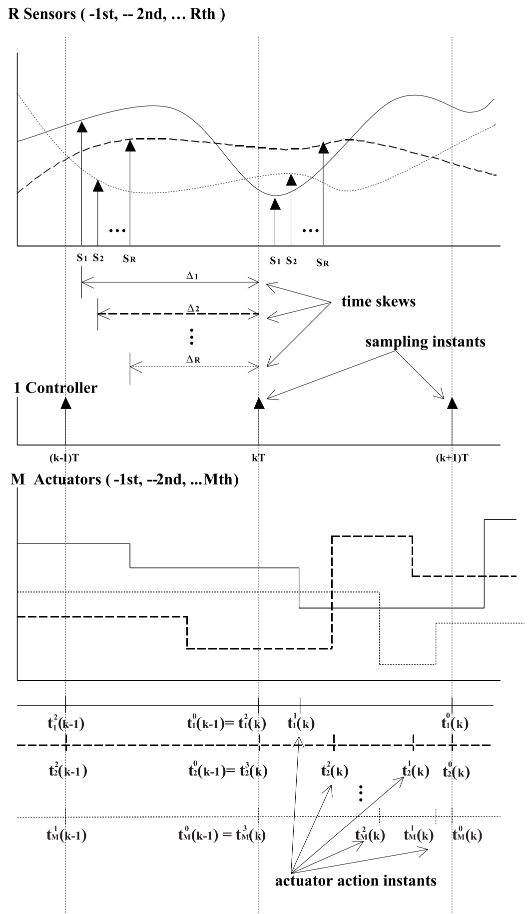

Figure 2: The timing diagram for sensors, controller, actuators in an NCS.

2. 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 $p^c, p_r^s(k), p_m^a(k)$, respectively. Communication delays between sensors and controller and between controller and actuators are denoted as $c_r^s(k)$, $c_m^a(k)$, respectively. In this paper, we assume that the processing delays are constant and can thus be added to the communication delays. The combined time delays are defined as follows. The combined sensor processing-communication delay is $t_r^s(k) = c_r^s(k) + p_r^s(k)$ and the combined actuator-controller processing-communication delay is $t_m^a(k) = p^c + c_m^a(k) + p_m^a(k)$. Hence, in the following discussion, we only consider the combined time delays, $t_r^s(k)$ and $t_m^a(k)$.

3. As shown in Fig. 2, the **sensor-controller delay** $s_r(k)$ depends on both the time skew Δ_r and the sampling period T. If $t_r^s(k) \leq \Delta_r$, then $s_r(k) = \Delta_r$. Otherwise, for $(j-1)T + \Delta_r \leq t_r^s(k) < jT + \Delta_r$, $s_r(k) = jT + \Delta_r$, where $j \in \mathbb{Z}^+$. Here, we also assume that $max\{s_r(k), r = 1, ..., R\} \leq n_s T$, where n_s is a constant integer. 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.

4. The series of $n_m (\in \mathbb{N})$ arriving instants at the mth actuator, during the kth controller sampling interval $[(k-1)T, kT]$, can be formulated as $0 = t_m^{n_m+1}(k) \leq ... \leq t_m^1(k) \leq t_m^0(k) = T$. Hence, the **controller-actuator delay** can be characterized as follows: $a_m(k-j+1) = t_m^a(k-j+1) = t_m^j(k) + (j-1)T$ for $j = 1, 2, ...n_m$. We also let n_a be $max\{n_m,\ n = 1, ..., M\}$. Note that n_m depends on the transmission and processing delays as well as the selected sampling time T. If all delays are constant and identical, then $n_m = 1$. If T is designed to be larger, then $n_m = 1,\ m = 1, ..., M$. However, the control performance may degrade due to the low sampling rate. On other hand, a smaller sampling time increases the control performance as well as the system complexity, i.e., there will be a large variance of n_m's due to the high network traffic load.

5. For any parameter 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 $\hat{\theta} \in \mathbb{Z}^+$ and $0 \leq \tilde{\theta} < 1$.

6. 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. Hence, vacant sampling or message rejection of sensing signals may happen at some controller sampling instants.

3 Delay impact on plant and controller dynamics

One of main features of NCSs is the different communication delays between the inputs and outputs of plant and controller, respectively, as shown in Fig. 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, we will derive the relation between each pair of variables based on the assumptions described above.

3.1 Plant input and controller output

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

$$u_m(t) = \sum_{j=0}^{n_m-1} v_m(k-j)\mathbf{1}_{(t_m^{j+1}(k),t_m^j(k))}(t), \quad (3)$$

where

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

and

$$[0,\ T) = \bigcup_{j=1}^{n_m} [t_m^j(k),\ t_m^{j-1}(k)). \quad (5)$$

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 Fig. 2. The following relation holds: $u_1(t) = v_1(k)\mathbf{1}_{(t_1^1(k),t_1^0(k))} + v_1(k-1)\mathbf{1}_{(t_1^2(k),t_1^1(k))}$ for $kT \leq t < (k+1)T$. Note that the time delays $t_m^j(k)$ are functions of the controller-actuator delays as defined in Assumption 4. Then, $\mathbf{u}(t) = [u_1(t), ..., u_M(t)]^T$, and $\mathbf{v}(k) = [v_1(k), ..., v_M(k)]^T$.

3.2 Plant model in discrete-time domain

In order to analyze the closed-loop system in discrete-time, we will use the following state-space solution of a set of first-order matrix differential equations to discretize the continuous-time plant dynamics model [2].

$$\mathbf{x}(t) = \exp(\mathbf{A}_p(t-t_0))\mathbf{x}(t_0) + \int_{t_0}^{t} \exp(\mathbf{A}_p(t-q'))\mathbf{B}_p\mathbf{u}(q')dq. \quad (6)$$

We first discretize the plant model at the controller sampling instants by applying Eq. (6) with $t = (k+1)T, t_0 = kT$. For simplicity, we let $\mathbf{x}_k := \mathbf{x}(k) = \mathbf{x}(kT)$, $\mathbf{A} := \exp(\mathbf{A}_pT)$, $q = q' - kT$, and $\Gamma(T,q) := \exp(\mathbf{A}_p(T-q))\mathbf{B}_p \in \mathbb{R}^{N\times M}$. Then,

$$\begin{aligned}
&\mathbf{x}_{k+1} \\
&= \mathbf{A}\mathbf{x}_k + \int_0^T \exp(\mathbf{A}_p(T-q))\mathbf{B}_p\mathbf{u}(kT+q)dq, \\
&= \mathbf{A}\mathbf{x}_k + \int_0^T \Gamma(T,q)\mathbf{u}(kT+q)dq, \\
&= \mathbf{A}\mathbf{x}_k \\
&+ \int_0^T [\Gamma_1(T,q), ..., \Gamma_M(T,q)] \begin{bmatrix} u_1(kT+q) \\ \vdots \\ u_M(kT+q) \end{bmatrix} dq, \\
&= \mathbf{A}\mathbf{x}_k + \sum_{m=1}^{M} \int_0^T \Gamma_m(T,q)u_m(kT+q)dq, \quad (7)
\end{aligned}$$

where $\Gamma_m \in \mathbb{R}^{N\times 1}$ and $u_m \in \mathbb{R}$, $m = 1, ..., M$. The first and second equalities are obtained by definition. The third and fourth equalities are used to explicitly describe Γ and $\mathbf{u}$ in terms of their elements, Γ_m's and u_m's, respectively, $m = 1, ..., M$. Now, because the actuator commands arrive at different times within the sample interval, u_m is not constant over $[kT,\ (k+1)T]$. From Eqs. (3)–(5), we can compute $\int_0^T \Gamma_m(T,q)u_m(kT+q)dq$ as follows:

$$\begin{aligned}
&\int_0^T \Gamma_m(T,q)u_m(kT+q)dq \\
&= \sum_{j=0}^{n_m} \int_{a_m(k-j)-jT}^{a_m(k-j+1)-(j-1)T} \Gamma_m(T,q)v_m(k-j)dq,
\end{aligned}$$

where we also use the relations, $t_m^j(k) = a_m(k-j+1)-(j-1)T$, $a_m(k-n_m)-n_mT = 0$, and $a_m(k+1)+T = T$, from Assumption 4. Since v_m's are constant, we define $B_m^j(k) = \int_{a_m(k-j)-jT}^{a_m(k-j+1)-(j-1)T} \Gamma_m(T,q)dq$, and further simplify the above equation as follows:

$$\int_0^T \Gamma_m(T,q)u_m(kT+q)dq = \sum_{j=0}^{n_m} B_m^j(k)v_m(k-j). \quad (8)$$

Therefore, by applying Eq. (8) to Eq. (7), we have:

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \sum_{m=1}^{M}\sum_{j=0}^{n_m} B_m^j(k)v_m(k-j). \quad (9)$$

By using the maximum number of n_m's, n_a, and exchanging the order of summations, we have the following derivation:

$$\begin{aligned}
&\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \sum_{j=0}^{n_a}\sum_{m=1}^{M} B_m^j(k)v_m(k-j) \\
&= \mathbf{A}\mathbf{x}_k + \sum_{j=0}^{n_a} \left[B_1^j(k)\ B_2^j(k)\ ...\ B_M^j(k)\right] \begin{bmatrix} v_1(k-j) \\ v_2(k-j) \\ \vdots \\ v_M(k-j) \end{bmatrix} \\
&= \mathbf{A}\mathbf{x}_k + \sum_{j=0}^{n_a} \mathbf{B}^j(k)\mathbf{v}(k-j) = \mathbf{A}\mathbf{x}_k + \sum_{j=0}^{n_a} \mathbf{B}_k^j\mathbf{v}_{k-j}. \quad (10)
\end{aligned}$$

In Eq. (10), $\mathbf{A}$ 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 the $a_m(k)$ depend on k, the NCS model will be time-varying.

3.3 $\mathbf{x}(kT - \delta T)$: state between sampling times

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., $\mathbf{x}(kT-\delta T)$, where $0 \leq \delta < 1$. By using Eq. (6) again with $t = kT - \delta T, t_0 = kT - T$, we obtain the following derivation for $\mathbf{x}(kT - \delta T)$.

$$\begin{aligned}
&\mathbf{x}(kT-\delta T) \\
&= \mathbf{A}_\delta\mathbf{x}_{k-1} + \sum_{j=0}^{n_a} \mathbf{B}_\delta^j(k-1)\mathbf{v}(k-1-j), \\
&= \mathbf{A}_\delta\mathbf{x}_{k-1} + \sum_{j=0}^{n_a} \mathbf{B}_{\delta k-1}^j\mathbf{v}_{k-1-j}, \quad (11)
\end{aligned}$$

where $\mathbf{B}_{\delta k}^j \in \mathbb{R}^{N\times M}$ and $\mathbf{v}_{k-j} \in \mathbb{R}^{M\times 1}$. Note that $\mathbf{B}_{\delta k}^j$ can be formulated similarly to $\mathbf{B}_k^j$ and $\mathbf{A}_\delta$ is not constant but depends on δ, in contrast to $\mathbf{A}$ in Eq. (7).

3.4 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 sensors and controller. By assumption 5 and for simplicity, we have the following equation for the rth sensor-controller delay: $s_r(k) = T[\ \bar{s}_r(k)\ ] = T[\ \hat{s}_r(k) + \tilde{s}_r(k)\ ]$ where $\hat{s}_r$ is an integer and $0 \le \tilde{s}_r < 1$. Therefore, for any value of $s_r(k)$ and by Eq. (11),

$$\begin{aligned}&\mathbf{x}(kT - s_r(k))\\ &= \mathbf{x}(kT - \bar{s}_r(k)T) = \mathbf{x}((k - \hat{s}_r(k))T - \tilde{s}_r(k)T)\\ &= \mathbf{A}_{\tilde{s}_r(k)}\mathbf{x}_{k-1-\hat{s}_r(k)}\\ &+ \sum_{j=0}^{n_a} \mathbf{B}^j_{\tilde{s}_r(k)}(k-1-\hat{s}_r(k))\mathbf{v}(k-1-\hat{s}_r(k)-j)),\end{aligned} \tag{12}$$

where we let $\delta = \tilde{s}_r(k)$ in Eq. (11). At the kth instant,

$$\begin{aligned}&\mathbf{w}_k = \mathbf{w}(kT)\\ &= \begin{bmatrix} w_1(kT)\\ \vdots \\ w_R(kT)\end{bmatrix} = \begin{bmatrix} y_1(kT - s_1(k))\\ \vdots \\ y_R(kT - s_R(k))\end{bmatrix}\\ &= \begin{bmatrix} \mathbf{C}_1\mathbf{x}(kT - s_1(k))\\ \vdots \\ \mathbf{C}_R\mathbf{x}(kT - s_R(k))\end{bmatrix}\\ &= \begin{bmatrix} \mathbf{C}_1(\mathbf{A}_{\tilde{s}_1(k)}\mathbf{x}_{k-1-\hat{s}_1(k)} \\ + \sum_{j=0}^{n_a}\mathbf{B}^j_{\tilde{s}_1(k)}(k-1-\hat{s}_1(k))\mathbf{v}(k-1-\hat{s}_1(k)-j)) \\ \vdots \\ \mathbf{C}_R(\mathbf{A}_{\tilde{s}_R(k)}\mathbf{x}_{k-1-\hat{s}_R(k)} \\ + \sum_{j=0}^{n_a}\mathbf{B}^j_{\tilde{s}_R(k)}(k-1-\hat{s}_R(k))\mathbf{v}(k-1-\hat{s}_R(k)-j))\end{bmatrix}\\ &= \sum_{i=1}^{n_s}\Theta^i_k\mathbf{x}_{k-i} + \sum_{i=1}^{n_a+n_s}\Phi^i_k\mathbf{v}_{k-i},\end{aligned} \tag{13}$$

where

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

and

$$\Phi^i_k = \begin{bmatrix}\Phi^{i1}_k \\ \vdots \\ \vdots \\ \Phi^{iR}_k\end{bmatrix},$$

$$\Phi^{ir}_k = \begin{cases}\mathbf{C}_r\mathbf{B}^{i-\hat{s}_r(k)}_{\tilde{s}_r(k)}(k-1-\hat{s}_r(k)), & \text{if } \hat{s}_r(k) \le i \le \hat{s}_r + n_a,\\ 0, & \text{otherwise.}\end{cases} \tag{14}$$

4 Closed-loop model

In order to analyze the system property and provide guidelines for controller design, we will derive the closed-loop model that combines the discrete-time plant model, Eq. (10), controller model, Eq. (2), and Eq. (13).

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \sum_{j=0}^{n_a}\mathbf{B}^j_k\mathbf{v}_{k-j},$$

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}^0_k\mathbf{v}_k + \sum_{j=1}^{n_a}\mathbf{B}^j_k\mathbf{v}_{k-j}$$

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}^0_k(\mathbf{H}\mathbf{z}_k + \mathbf{J}\mathbf{w}_k) + \sum_{j=1}^{n_a}\mathbf{B}^j_k\mathbf{v}_{k-j}$$

$$\begin{aligned}&\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}^0_k\mathbf{H}\mathbf{z}_k\\ &+\mathbf{B}^0_k\mathbf{J}(\sum_{i=1}^{n_s}\Theta^i_k\mathbf{x}_{k-i} + \sum_{i=1}^{n_a+n_s}\Phi^i_k\mathbf{v}_{k-i}) + \sum_{j=1}^{n_a}\mathbf{B}^j_k\mathbf{v}_{k-j}.\end{aligned} \tag{15}$$

Also, the controller dynamics, Eq. (2), can be further expressed as follows:

$$\mathbf{z}_{k+1} = \mathbf{F}\mathbf{z}_k + \mathbf{G}(\sum_{i=1}^{n_s}\Theta^i_k\mathbf{x}_{k-i} + \sum_{i=1}^{n_a+n_s}\Phi^i_k\mathbf{v}_{k-i}) \tag{16}$$

$$\mathbf{v}_k = \mathbf{H}\mathbf{z}_k + \mathbf{J}(\sum_{i=1}^{n_s}\Theta^i_k\mathbf{x}_{k-i} + \sum_{i=1}^{n_a+n_s}\Phi^i_k\mathbf{v}_{k-i}). \tag{17}$$

By further combining Eqs. (15)–(17), and defining $\mathbf{X}_k = [\mathbf{x}^T_k\ \mathbf{x}^T_{k-1} \ldots \mathbf{x}^T_{k-n_s} \mid \mathbf{z}^T_k \mid \mathbf{v}^T_{k-1} \ldots \mathbf{v}^T_{k-n_a-n_s}]^T$, we obtain the closed-loop dynamics as follows:

$$\mathbf{X}_{k+1} = \Xi_k\mathbf{X}_k, \tag{18}$$

where

$$\Xi_k = \left[\begin{array}{c|c|c}\Xi_k(1,1) & \Xi_k(1,2) & \Xi_k(1,3)\\ \hline \Xi_k(2,1) & \Xi_k(2,2) & \Xi_k(2,3)\\ \hline \Xi_k(3,1) & \Xi_k(3,2) & \Xi_k(3,3)\end{array}\right], \quad \text{and}$$

$$\Xi_k(1,1) = \left[\begin{array}{cccc}\mathbf{A} & \mathbf{B}^0_k\mathbf{J}\Theta^1_k & \ldots & \mathbf{B}^0_k\mathbf{J}\Theta^{n_s}_k\\ \hline \mathbf{I} & \mathbf{0} & \cdots & \mathbf{0}\end{array}\right],$$

$$\Xi_k(1,2) = \left[\begin{array}{c}\mathbf{B}^0_k\mathbf{H}\\ \hline \mathbf{0}\end{array}\right],$$

$$\Xi_k(1,3) = \left[\begin{array}{ccc}\mathbf{B}^0_k\mathbf{J}\Phi^1_k + \mathbf{B}^1_k & \ldots & \mathbf{B}^0_k\mathbf{J}\Phi^{n_a}_k + \mathbf{B}^{n_a}_k\\ \mathbf{B}^0_k\mathbf{J}\Phi^{n_a+1}_k & \ldots & \mathbf{B}^0_k\mathbf{J}\Phi^{n_a+n_s}_k\\ \hline & \mathbf{0} & \end{array}\right],$$

$$\Xi_k(2,1) = [\ 0 \quad \mathbf{G}\Theta^1_k \quad \ldots \quad \mathbf{G}\Theta^{n_s}_k\ ],$$

$$\Xi_k(2,2) = \mathbf{F},$$

$$\Xi_k(2,3) = [\ \mathbf{G}\Phi^1_k \quad \ldots \quad \mathbf{G}\Phi^{n_a}_k \quad \mathbf{G}\Phi^{n_a+1}_k \quad \ldots \quad \mathbf{G}\Phi^{n_a+n_s}_k\ ],$$

$$\Xi_k(3,1) = \left[\begin{array}{cccc}0 & \mathbf{J}\Theta^1_k & \ldots & \mathbf{J}\Theta^{n_s}_k\\ \hline & \mathbf{0} & & \end{array}\right],$$

$$\Xi_k(3,2) = \left[\begin{array}{c}\mathbf{H}\\ \hline \mathbf{0}\end{array}\right],$$

$$\Xi_k(3,3) = \left[\begin{array}{cccccc}\mathbf{J}\Phi^1_k & \ldots & \mathbf{J}\Phi^{n_a}_k & \mathbf{J}\Phi^{n_a+1}_k & \ldots & \mathbf{J}\Phi^{n_a+n_s}_k\\ \hline \mathbf{I} & \mathbf{0} & \cdots & \mathbf{0} & \cdots & \mathbf{0}\end{array}\right].$$

The closed-loop system could be time-varying since Ξ_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 of different network delays could be found based on the stability criterion that all the eigenvalues of Ξ_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, Eqs. (15)–(17), 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.

5 Numerical Examples

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 with control input between 0 (negative) and 255 (positive). 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 mathematical model of each axis between the PWM input (U) and the position output (P) is described by a second-order linear system:

$$G(s) = \frac{P(s)}{U(s)} = \frac{K}{s(\tau s+1)} \tag{19}$$

The time constants τ (sec) for each axis are 0.055 (X) and 0.056 (Y) and the overall gains K $((mm/sec)/PWM)$ 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 follows:

$$\begin{aligned}
\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -18.18 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -17.86 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \\
&+ \begin{bmatrix} 0 & 0 \\ 515.38 & 0 \\ 0 & 0 \\ 0 & 517.07 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \\
\text{or} \quad \dot{\mathbf{x}} &= \mathbf{A}_p \mathbf{x} + \mathbf{B}_p \mathbf{u}
\end{aligned} \tag{20}$$

We further assume that $t_r^s(k) \le \Delta_r$, $t_m^a(k) < T$, and $\max(a_m) < T$ - $\max(s_r)$, for $r = 1, ..., 4$ 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 3 and 4 in Section 2, we have the following relations: $s_r = \Delta_r$ and $a_m = t_m^a(k)$, for $r = 1, ..., 4$ and $m = 1, 2$, respectively. Hence, from Eq. (3), the plant input signals, $u_1(t)$ and $u_2(t)$, can be described as follows, for $kT \le t < (k+1)T$:

$$\begin{aligned}
u_1(t) &= v_1(k-1)\mathbf{1}_{(kT, kT+a_1)}(t) + v_1(k)\mathbf{1}_{(kT+a_1, (k+1)T)}(t) \\
u_2(t) &= v_2(k-1)\mathbf{1}_{(kT, kT+a_2)}(t) + v_2(k)\mathbf{1}_{(kT+a_2, (k+1)T)}(t)
\end{aligned}$$

If these actuator delays a_m are constant and known in advance, say $a_1 = 1$ ms and $a_2 = 2$ ms, by applying Eq. (10), we can obtain the discrete-time plant model (at $T = 10$ ms) as follows:

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

We can further calculate the signal $x_r(kT - s_r)$ by obtaining the sensing delays of s_r, says $s_1 = 3$, $s_2 = 4$, $s_3 = 5$, and $s_4 = 6$ (ms). For example, by applying Eq. (12), $x_1(kT - s_1)$ can be described as follows:

$$\begin{aligned}
x_1(kT - s_1) &= \begin{bmatrix} 1 & 0.0066 & 0 & 0 \end{bmatrix} \mathbf{x}_{k-1} \\
&+ \begin{bmatrix} 0.0198 & 0 \end{bmatrix} \mathbf{v}_{k-1} + \begin{bmatrix} 0.0045 & 0 \end{bmatrix} \mathbf{v}_{k-2}.
\end{aligned}$$

Therefore, Eq. (13) becomes:

$$\begin{aligned}
\mathbf{w}_k &= \begin{bmatrix} w_1(k) \\ w_2(k) \\ w_3(k) \\ w_4(k) \end{bmatrix} = \begin{bmatrix} x_1(kT - s_1) \\ x_2(kT - s_2) \\ x_3(kT - s_3) \\ x_4(kT - s4) \end{bmatrix} \\
&= \begin{bmatrix} 1 & 0.0066 & 0 & 0 \\ 0 & 0.8966 & 0 & 0 \\ 0 & 0 & 1 & 0.0048 \\ 0 & 0 & 0 & 0.9311 \end{bmatrix} \mathbf{x}_{k-1} \\
&+ \begin{bmatrix} 0.0089 & 0 \\ 2.4632 & 0 \\ 0 & 0.0023 \\ 0 & 1.0159 \end{bmatrix} \mathbf{v}_{k-1} + \begin{bmatrix} 0.0032 & 0 \\ 0.4663 & 0 \\ 0 & 0.0040 \\ 0 & 0.9803 \end{bmatrix} \mathbf{v}_{k-2}.
\end{aligned}$$

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, by letting $\mathbf{X}_k = [\mathbf{x}_k^T \ \mathbf{x}_{k-1}^T \ \mathbf{z}_k^T \ \mathbf{v}_{k-1}^T \ \mathbf{v}_{k-2}^T\ ]^T$ the closed-loop dynamics can be obtained as follows:

$$\mathbf{X}_{k+1} = \Xi \mathbf{X}_k \tag{21}$$

In this example, the system dimension is 13. However, if there are no time delays, then the system dimension becomes 7, i.e., $\mathbf{X}_k = [\mathbf{x}_k^T \ \mathbf{z}_k^T \ \mathbf{v}_{k-1}^T\ ]^T$. The 13 eigenvalues ('o') of Ξ are plotted in Fig. 3 along with the 7 eigenvalues ('x') of the closed-loop systems without delays. Fig. 3(a) and 3(b) show two different feedback gains K. The values of time delays ($s_r, r = 1, 2, 3, 4$ and $a_m, m = 1, 2$) are identical in Fig. 3(a) and 3(b). In each plot, the dotted lines are the real and imagery axes and the solid line is the unit circle. The 'o' symbols are the locations of the eigenvalues of the closed-loop system with delays and the 'x' symbols are those

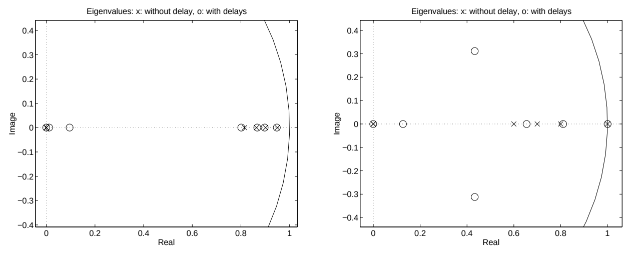

Figure 3: The location of eigenvalues of closed-loop systems.

without delays. However, only the four right-most ones map to the the eigenvalues of original closed-loop system, i.e., $\mathrm{eig}(A_p - B_p K)$. 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 Summary and Future Work

Although the introduction of control networks to send sensor and actuator data in control systems provides flexibility and reliability from a design point of view, it also induces asynchronous time delays among different devices. These time delays could degrade the system performance and may introduce instability in high bandwidth or complex systems.

In this paper, we analyzed and modeled 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 time-variant and included in the derivation of this 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 here only includes a standard controller designed without considering the time delay effect a priori. Therefore, this closed-loop stability analysis is used as a verification tool of the stability and performance of an NCS. If the closed-loop system does not meet the requirement, a new controller can be designed based on the NCS model provided. Our future work will focus on the design of controllers for NCS which take into account the time delays.

Acknowledgment

This research was supported in part by the NSF-ERC/RmS at the University of Michigan under grant EEC95-92125, the Open DeviceNet Vendor Association, and ControlNet International.

References

[1] H.A. Abdullah and C.R. Chatwin. Disctributed C^3 environment for small to medium-sized enterprises. *Integrated Manufacturing Systems*, 5(3):20–28, 1994.

[2] G.F. Franklin, J.D. Powell, and M.L. Workman. *Digital control of dynamic systems.* Addison-Wesley, third edition, 1998.

[3] Y. Halevi and A. Ray. Integrated communication and control systems: Part I — analysis. *ASME Journal of Dynamic Systems, Measurement, and Control*, 110(4):367–373, Dec. 1988.

[4] F.-L. Lian, J.R. Moyne, and D.M. Tilbury. Control performance study of a networked machining cell. In *Proceedings of 2000 American Control Conference*, pages 2337–2341, June 2000.

[5] F.-L. Lian, J.R. Moyne, and D.M. Tilbury. Performance evaluation of control networks: Ethernet, ControlNet, and DeviceNet. *IEEE Control Systems Maganize*, Vol. 21, No. 1, pp. 66-83, Feb. 2001.

[6] L.-W. Liou and A. Ray. Integrated communication and control systems: Part III— nonidentical sensor and controller sampling. *ASME Journal of Dynamic Systems, Measurement and Control*, 112(3):357–364, Sep. 1990.

[7] J. Nilsson. *Real-time control systems with delays.* PhD thesis, Lund Institute of Technology, 1998.

[8] J. Nilsson and B. Bernhardsson. LQG control over a Markov communication network. In *Proceedings of 36th Conference on Decision and Control*, pages 4586–4591, Dec. 1997.

[9] J. Nilsson, B. Bernhardsson, and B. Wittenmark. Stochastic analysis and control of real-time systems with random time delays,. *Automatica*, 34(1):57–64, Jan. 1998.

[10] A. Ray and Y. Halevi. Integrated communication and control systems: Part II — design considerations. *ASME Journal of Dynamic Systems, Measurement, and Control*, 110(4):374–381, Dec. 1988.

[11] G. Schickhuber and O. McCarthy. Distributed fieldbus and control network systems. *Computing & Control Engineering*, 8(1):21–32, Feb. 1997.

[12] B. Wittenmark, J. Nilsson, and M. Torngren. Timing problems in real-time control systems. In *Proceedings of American Control Conference*, pages 2000–2004, 1995.