Non-Conservative Distributed Control over Real-Time Network Links

Babak Tavassoli

Abstract—Communication technologies are being continuously developed to provide improved support of real-time links for an industrial internet of things. This work considers a general and flexible distributed control architecture over real-time links. No bounds are assumed on the value of delays that are modeled as random variables. An accurate modeling and analysis framework is provided and the exact value of a control performance cost function is calculated. The results provide the required tools for optimization-based design of distributed networked control systems. A case study is presented in which networked controllers are designed based on the consensus algorithm for coordination of multiple parallel power electronic converters. It is shown that consideration of the communication effects in the controller design process is crucial for achieving a superior performance.

Index Terms—Real-time communication, Networked control systems, Unbounded random delay, Performance cost function.

I. INTRODUCTION

One of the current challenges in the field of automatic control is dealing with networked control systems (NCSs) in which communication networks are used for transferring information between distributed controllers, sensors, and actuators. In general, the control performance in an NCS is affected by the networking constraints which must be taken into account. Considerable research has been devoted to various networked control problems [1], [2], [3] and NCSs now appear in a wide spectrum of applications ranging from simple industrial control loops to complex cyber-physical systems [5], [4].

The networking constraints can be categorized into the constraints on interconnection graph of the NCS subsystems [6], [7], and communication constraints in each of the network links. There are various forms of communication constraints where many of them such as network-induced delays and packet losses can be modeled and treated as upper bounded time varying delays [8], [9], [10], [2]. The hybrid and switching systems frameworks are also capable of handling many of the NCS problems [11], [12]. Other forms of communication constraints include limited bitrate [13], [14] and limited channel access [15], [1]. In some of the works, additional components such as predictor [16], [17] or event-triggering mechanism [12], [18], [19] are used to improve control or communication performance.

Two main obstacles of applying the existing works to practical NCS problems are: (1) conservativeness of the results and (2) usage of control structures that are different from the commonly used control algorithms such as PID (proportional integral derivative). The conservativeness in existing NCS results stems from two main reasons. The first reason is simplified and inaccurate modeling of the communication effects (e.g. using bounded delay models [8]). The second reason is conservativeness of the methods applied to the complex NCS models (e.g. deriving the results based on inequalities and boundings [8], [11]). The most accurate models have been obtained for special cases of NCS in the form of stochastic discrete-time systems. Some examples are, delays bounded by sampling period [20], bounded delays that are multiples of the sampling period [21], delays in CSMA networks [22], and packet loss modeling in [23], [24]. Accurate modeling of an NCS with time-varying delays belonging a real interval which spans multiple sampling periods is very complex, even if there is only a single communication link [25].

In this work, a general distributed control structure over real-time network links is considered. A network link is said to be real-time if it is possible to specify deadlines for delivery of data packets [26], [27]. Establishment of real-time links is quite straightforward over single segment networks such as CAN and PROFIBUS [28]. It is also remarkable that the support for real-time links is expected to be added in common data networks due to the rapid spreading of the internet of things (IOT) and its industrial version (IIOT) [29], [30], [31]. The contributions of this work can be itemized as below.

- The communication delays in each link are modeled as random variables without any assumption on the probability distribution of delays. It is possible for example to model loss of data packets as infinite delays.
- A general distributed control structure is considered such that a desirable control algorithms like PID can be used as a building block of the NCS.
- An accurate stochastic model is obtained for the distributed NCS over real-time links such that a non-conservative analysis is made possible.
- The exact value of a quadratic cost function for the NCS step response is computed which can be used for optimization based controller design. Provisions are made for efficient computation of the cost function.

To demonstrate an application of the results, a case study is also provided in which a distributed NCS is designed based on the consensus algorithm for coordinating three parallel converters by taking the effects of communication into account.

The paper is organized as follows. NCS structure and components are described in Section 2. An integrated model of NCS is developed in Section 3. The cost-function is calculated in Section 4. The case study is presented in Section 5, whilst Section 6 draws conclusions.

Notation: The sets of real numbers, non-negative real numbers, integers, and non-negative integers are respectively...
denoted by $\mathbb{R}$, $\mathbb{R}_{\geq 0}$, $\mathbb{Z}$, $\mathbb{Z}_{\geq 0}$. An integer interval \( \{i \in \mathbb{Z} : a \leq i \leq b\} \) is also denoted as \( \{a, b\} \). The expected value of a random variable \( v \) is denoted by \( E\{v\} \). For a symmetric matrix \( M \in \mathbb{R}^{n \times n} \), eigenvalues are denoted by \( \lambda_i(M) \) for \( i \in \{1..n\} \) such that \( \lambda_1(M) \leq \lambda_j(M) \) if \( i \leq j \). Also, the spectral radius of \( M \) is denoted by \( \rho(M) = \lambda_n(M) \). For a Boolean variable \( a \), we define \( \text{real}\{a\} = 1 \) if \( a \) is true and \( \text{real}\{a\} = 0 \) if \( a \) is false.

II. SYSTEM DESCRIPTION

The general configuration of the NCS which is considered in this work is depicted in Fig. 1. The figure shows multiple controllers indexed by \( I_c = \{1..m\} \) that receive feedback data from sensors indexed by \( I_s = \{1..r\} \) through a network in order to control a plant. For simplicity, it is assumed that each controller sends its output to an actuator such that the number of actuators is also \( I_s \). The controllers indexed by \( I_c \) receive data from the set of indices for sensors that send data to the controller \( i \in I_c \) is denoted by \( \Gamma_i \subseteq I_s \) and the set of indices for controllers that receive data from the \( j \)th sensor for \( j \in I_s \) is denoted by \( \Gamma_j = \{i \in I_c : \Gamma_i \subseteq I_s \} \).

Accurate modeling of the NCS in Fig. 1 is a complex task due to the several delays in the communication links between the sensors, controllers, and actuators. To model each flying data packets (that are sent but not yet received), a new state variable should be introduced. Evolution of these states with time and their interaction with other states increase the complexity. A simplification can be made by modelling each communication link as a bounded delay that can vary arbitrarily with time [8]. But, this modeling approach is inefficient, because communication delays are usually random variables with long-tailed probability distributions. Hence, one has to consider a large delay bound to capture some delay values that occur rarely.

A. Real-time communication links

Practical control networks are not as complex as the general case. Many of the industrial and non-industrial networking protocols support real-time communication services. In a real-time communication link, deadlines can be defined such that a data packet is discarded if its deadline is approached [29], [26]. Discard of data packets can take place either at source, router, or destination nodes. Moreover, many of the industrial fieldbuses are composed of a single segment such that each data packet is received at the same time by all of the network nodes [28]. In single segment networks, no special services are required for establishing real-time links. There are no routers and if the access to the communication medium is not acquired within the deadline, it is only needed that the source node ignores sending the data packet. In this work, we consider an NCS with real-time links while making the following assumptions.

A1. In the case that a real-time link has multiple destination nodes, they are all connected to the same segment of the network. Hence, the data packets in that link are received at the same time by all of the destination nodes.

A2. The network nodes are time synchronized. The sensors take their \( k \)th sample at the time \( kh \) where \( h \) is the sampling period. Also, the \( k \)th execution for the \( i \)th controller starts at \( kh + \bar{\tau}_c \) for some \( \bar{\tau}_c < h \) and \( i \in I_c \).

A3. If \( \{i_1, i_2\} \subseteq \Gamma_j \) for some \( j \in I_s \), then \( \bar{\tau}_{i_1} = \bar{\tau}_{i_2} \).

A4. The packet transmission deadlines in all of the links are set to \( h \) based on the fact that new data is generated by the sensors and controllers with period \( h \) in a way that the old data is not useful anymore.

In spite of the above assumptions, asynchrony of the real-time deadlines in the links established by sensors and controllers results in modeling complexities.

The networked communication is described as follows. The \( j \)th measured output channel for \( j \in I_s \) denoted by \( y_j(t) \in \mathbb{R}^{n_j} \) is sampled by the \( j \)th sensor where the \( k \)th sample is denoted by \( y_j^k \). This sample is sent to all of the controllers indexed by \( \Gamma_j \) with the same transmission delay denoted by \( \bar{\tau}_{c,j} \) according to the Assumption A1. The execution time instants of the receiving controllers are also the same according to the Assumption A3 and the same sample from the \( j \)th sensor denoted by \( y_j^k \) is available to all of the controllers indexed by \( \Gamma_j \) at their \( k \)th execution. The \( i \)th controller starts to compute \( u_i^k \) using the latest received samples from the sensors at \( kh + \bar{\tau}_c \). It is assumed that the computation of \( u_i^k \) completes at \( kh + \bar{\tau}_c^i \) for some \( \bar{\tau}_c^i \in [\bar{\tau}_c, h) \). Then, \( u_i^k \) is sent to the \( i \)th actuator with a transmission delay denoted by \( \bar{\tau}_{c,i} \). The \( i \)th plant input channel denoted by \( \hat{u}_i(t) \) is set to the latest value of the \( i \)th controller output which is received until the time \( t \).

The following two remarks show that more general classes of NCSs can be presented in the framework of the NCS problem formulation in this section.

**Remark 1.** The controllers indexed by \( \Gamma_j^i \) for \( j \in I_s \) must be connected to the same segment of network according to the Assumption A1. But, this assumption is not restrictive since if the samples of the \( j \)th sensor should be sent to controllers on different segments, then one can virtually consider duplicates of the \( j \)th sensor in order to define different links to each of the network segments.

**Remark 2.** In distributed and decentralized control systems, each actuator is usually driven by one controller. However, consider a more general case in which there are \( m' \neq m \) plant input channels denoted by \( v(t) \) for \( t \in \{1..m\} \) driven by \( m' \) actuators. The \( i \)th controller for \( i \in I_c \) sends its output
to multiple actuators indexed by $\bar{I}$. Each actuator applies a linear combination of the received control signals to the plant. If the Assumption A1 is satisfied, then the same value of the $i$th controller output denoted by $\bar{u}_i(t)$ is received by the actuators indexed by $\bar{I}$. Hence, the dependency of the actuated inputs on the control signals can be written as $\bar{u}_i(t) = \sum_{i=1}^{m} a_i \bar{u}_i(t)$. Then, $\bar{u}_i(t)$ for $i \in I_c$ can be considered as a new set of inputs by a change of variables such that there are virtually $m$ actuators paired with the $m$ controllers.

B. Equations of the NCS components

The plant is described by the continuous-time model

$$\dot{x}(t) = \bar{A}_p x(t) + \sum_{i=1}^{m} \bar{B}_p^i \bar{u}_i(t), \quad (1a)$$

$$y^i(t) = C^i_p x(t) \quad \forall \ j \in I_s \quad (1b)$$

in which $x(t) \in \mathbb{R}^{n_x}$ is the state of the plant.

The controllers are described by the recursive equations

$$\xi^i_{k+1} = A^i_c \xi^i_k + \sum_{j=1}^{r} B^i_{c,j} y^j_k, \quad (2a)$$

$$u^i_k = C^i_c \xi^i_k + \sum_{j=1}^{r} D^i_{c,j} y^j_k, \quad (2b)$$

in which $i \in I_c$ and $\xi^i_k \in \mathbb{R}^{n_c}$ is the state of the $i$th controller at the $k$th step.

It is assumed that the matrix coefficients in (2) depend on a set of $q$ tuning parameters $\beta_{\mu} \in \mathbb{R}$, $\mu \in \{1..q\}$ as

$$A^i_c = \sum_{\mu=0}^{q} A^i_{c,\mu} \beta_{\mu}, \quad C^i_c = \sum_{\mu=0}^{q} C^i_{c,\mu} \beta_{\mu}, \quad (3a)$$

$$B^i_{c,j} = \sum_{\mu=0}^{q} B^i_{c,j,\mu} \beta_{\mu}, \quad D^i_{c,j} = \sum_{\mu=0}^{q} D^i_{c,j,\mu} \beta_{\mu} \quad (3b)$$

for $i \in I_c, j \in I_s$ in which $\beta_0 = 1$ (defined for compactness of the formulas).

It is assumed that $\tau_{k+1}^c$ and $\tau_{k}^{c,i}$ for $j \in I_s$, $i \in I_c$ are independent and identically distributed (i.i.d) random variables and their probability density functions (PDFs) are given by the functions $f^c_\tau$ and $f^{c,i}_\tau$ according to the following equations.

$$P\{\tau^c_{k+1} \leq c\} = \int_{0}^{c} f^c_\tau(\alpha) \, d\alpha, \quad (4a)$$

$$P\{\tau^{c,i}_{k} \leq c\} = \int_{0}^{c} f^{c,i}_\tau(\alpha) \, d\alpha \quad (4b)$$

Remark 3. According to the Assumption A4, the shapes of PDFs in (4) are important only over the interval $[0, h]$ and integrations of the PDFs over $[h, \infty)$ only determine the probabilities that the sensor samples or controller outputs are discarded.

III. NCS Modeling

In this section, an integrated stochastic model of the NCS is obtained.

A. Combining the plant model with communication effects

The effects of communication can be combined with the plant model in (1) during the time-discretization process. According to the Assumption A4, $u_k^i$ for $i \in I_c$ may be received only within $[kh + \tau^c_i, kh + h + \tau^c_i]$. Hence, the $i$th actuated input $\bar{u}_i$ may be updated up to two times during the $k$th sampling interval $[kh, kh + h]$. The first update may occur if $u_{k-1}^i$ is received during $[kh, kh + \tau^c_i]$. The second update may occur if the $i$th controller receives $u_{k-1}^i$ before $kh + h$. An example timing diagram is shown in Fig. 2 in which two updates occur during the $k$th sampling interval but no update occurs during the $(k - 1)$th sampling interval.

Since the $i$th actuator may receive $u_{k-1}^i$ in the $k$th sampling interval, the time difference between the instant at which $u_{k-1}^i$ is received and the time $kh$ is defined as a variable in the following.

$$\tau_{k}^{p,i} = \tau_{k}^{c,i} - (h - \tau^c_i) \quad (5)$$

Defining $\bar{u}_k^i = \lim_{t \to kh} \bar{u}_i(t)$, the update equation for $\bar{u}_k^i$ can be written as

$$\bar{u}_{k+1}^i = \lambda_k^{a,i} u_k^i + \lambda_k^{b,i} u_{k-1}^i + \lambda_k^{c,i} \bar{u}_k^i \quad i \in I_c, \quad (6a)$$

$$\lambda_k^{a,i} = \text{real}\{\tau_k^{c,i} < h - \tau^c_i\}, \quad (6b)$$

$$\lambda_k^{b,i} = (1 - \lambda_k^{a,i}) \text{real}\{\tau_k^{c,i} < \tau^c_i\}, \quad (6c)$$

$$\lambda_k^{c,i} = 1 - \lambda_k^{a,i} - \lambda_k^{b,i}. \quad (6d)$$

During the $k$th sampling interval $[kh, kh + h]$, we have $\bar{u}_i(t) = \bar{u}_k^i$ for $t \in [kh, kh + \tau_{k}^{p,i}]$, $\bar{u}_i(t) = u_{k-1}^i$ for $t \in [kh + \tau_{k}^{p,i}, kh + \tau_{k}^{c,i}]$, and $\bar{u}_i(t) = u_k^i$ for $t \in [kh + \tau_{k}^{c,i}, kh + h]$. Consequently, the following update equation is obtained for the time-discretized plant state $x_k = x(kh)$ by solving (1a) as a differential equation over $t \in [kh, kh + h]$.

$$x_{k+1} = A_p x_k + \sum_{i=1}^{m} B_{p,i} \tau_{k}^{p,i} \bar{u}_k^i + \sum_{i=1}^{m} B_{c,i} \tau_{k}^{c,i} u_{k-1}^i + \sum_{i=1}^{m} B_{c,i} \tau_{k}^{c,i} \bar{u}_k^i \quad (7a)$$

$$y_k^j = C_p^j x_k \quad i \in I_s \quad (7b)$$

$$A_p = \exp(\bar{A}_p h) \quad (7c)$$

$$B_{p,i}^\tau(\tau^p, \tau^c) = \int_{0}^{h} \exp(\bar{A}_p h) B_{p,i}^\tau_d h \, dh \quad (7d)$$

$$B_{c,i}^\tau(\tau^p, \tau^c) = \int_{h - \tau^p}^{h} \exp(\bar{A}_p h) B_{c,i}^\tau_d h \, dh \quad (7e)$$

$$B_{c,i}^\tau(\tau^p, \tau^c) = \int_{h - \tau^p}^{h} \exp(\bar{A}_p h) B_{c,i}^\tau_d h \, dh \quad (7f)$$

Fig. 2. An example timing diagram of the NCS communication.
Despite the fact that $\tau_{k-1}^{c,i}$ depends on $\tau_{k-1}^{c,i}$ according to (5), one can virtually consider $\tau_{k-1}^{c,i}$ as a random variable which is independent of $\tau_{k-1}^{c,i}$ and conforms to the PDF $f_p$ given by (8) in the following. To show this, one can consider two cases. If $u_{k-1}^j$ is not received before $kh$, then we have $\tau_{k-1}^{c,i} \geq 0$ by the definition (5). The conditional PDF of $\tau_{k-1}^{c,i}$ given $\tau_{k-1}^{c,i} \geq 0$ is easily calculated from the PDF of $\tau_{k-1}^{c,i}$ as in (8). Otherwise, $u_{k-1}^j$ is received before $kh$ such that we have $u_{k-1}^j = u_{k-1}$ and it does not make a difference whether another update of the $i$th actuated input to $u_{k-1}$ occurs after $kh$ or not. Therefore, it will be assumed that (8) holds in all of the cases.

\[
P(\tau_{k-1}^{c,i} \leq c) = \int_0^c f_p(\alpha) d\alpha = \frac{\int_{h-\tau_k^c}^{h+c} f_p(\alpha) d\alpha}{\int_{h-\tau_k^c}^{h+c} f_p(\alpha) d\alpha}
\]

According to A1, the same sample from the $\tau_{k-1}^{c,i}$ is received by all of the controllers indexed by $\Gamma$ given by (8) in all of the cases. The conditional PDF of $\tau_{k-1}^{c,i}$ given $\tau_{k-1}^{c,i} \geq 0$ is easily calculated from the PDF of $\tau_{k-1}^{c,i}$ as in (8). Otherwise, $u_{k-1}^j$ is received before $kh$ such that we have $u_{k-1}^j = u_{k-1}$ and it does not make a difference whether another update of the $i$th actuated input to $u_{k-1}$ occurs after $kh$ or not. Therefore, it will be assumed that (8) holds in all of the cases.

Using the above definitions, the set of all equations that construct the overall model can be represented as

\[
x_{k+1} = A_p x_k + B_p (\theta_k^e) u_k + B_c (\theta_k^e) u_k \hat{u}_k + B_a (\theta_k^e) u_k + B_b u_k,
\]

\[
\hat{y}_{k+1} = \Lambda (\theta_k^e) y_k + (I - \Lambda (\theta_k^e)) \hat{y}_k,
\]

\[
\hat{u}_{k+1} = \Lambda (\theta_k^e) u_k + \Lambda (\theta_k^e) u_k - \Lambda (\theta_k^e) \hat{u}_k,
\]

\[
y_k = C_p x_k,
\]

\[
\xi_k = A_c x_k + B_c \theta_k^e + B_c \sigma_k^e + C_c \theta_k^e + C_c \sigma_k^e + \xi_{k-1} + \epsilon_k,
\]

\[
y_k = C_k x_k + D_k \theta_k^e + D_k \sigma_k^e + \epsilon_k.
\]

with the matrix coefficients defined in the following for every $\alpha \in \{a, b, c\}$ and $\nu \in \{w, e\}$.

\[
B(\theta_k^e) = [B_1^{a} (\tau_k^{p,1}, \tau_k^{c,1}) \cdots B_m^{a} (\tau_k^{p,m}, \tau_k^{c,m})]
\]

\[
\Lambda (\theta_k^e) = \text{diag} (\lambda^1_k I_{n_c} \cdots \lambda^m_k I_{n_c})
\]

\[
\Lambda (\theta_k^e) = \text{diag} (\lambda^{\nu,1}_k I_{n_c} \cdots \lambda^{\nu,m}_k I_{n_c})
\]

\[
C_p = \begin{bmatrix} C_{p}^{1,T} & \cdots & C_{p}^{q,T} \end{bmatrix}
\]

\[
A_c = \sum_{\mu=0}^{q} A_{c,\mu} \beta_{\mu}, \quad B_c = \sum_{\mu=0}^{q} B_{c,\mu} \beta_{\mu}
\]

\[
C_c = \sum_{\mu=0}^{q} C_{c,\mu} \beta_{\mu}, \quad D_c = \sum_{\mu=0}^{q} D_{c,\mu} \beta_{\mu}
\]

\[
A_{c,\mu} = \text{diag} (A_{c,\mu}^{1} \cdots A_{c,\mu}^{m}), \quad 0 \leq \mu \leq q
\]

\[
B_{c,\mu} = \begin{bmatrix} B_{1,\mu}^{1} & \cdots & B_{1,\mu}^{m} \\ \vdots & \ddots & \vdots \\ B_{m,\mu}^{1} & \cdots & B_{m,\mu}^{m} \end{bmatrix}, \quad 0 \leq \mu \leq q
\]

\[
C_{c,\mu} = \text{diag} (C_{c,\mu}^{1} \cdots C_{c,\mu}^{m}), \quad 0 \leq \mu \leq q
\]

\[
D_{c,\mu} = \begin{bmatrix} D_{1,\mu}^{1} & \cdots & D_{1,\mu}^{m} \\ \vdots & \ddots & \vdots \\ D_{m,\mu}^{1} & \cdots & D_{m,\mu}^{m} \end{bmatrix}, \quad 0 \leq \mu \leq q
\]

By defining the augmented controller state $\xi_k$ and the total NCS state $\eta_k$ as

\[
\eta_k = \begin{bmatrix} x_k \\ y_k \\ u_k \end{bmatrix}, \quad \xi_k = \begin{bmatrix} \xi_k \\ \hat{y}_k \end{bmatrix},
\]

the equations (11a) through (11d) can be combined as (15a) and the equations (11e) through (11g) can be combined as (15b) in the following with the matrix coefficients in (16).

\[
\eta_{k+1} = F(\theta_k^e, \theta_k^e) \eta_k + G_c (\theta_k^e) \begin{bmatrix} \xi_{k+1} \\ u_k \end{bmatrix}
\]

\[
\begin{bmatrix} \xi_{k+1} \\ u_k \end{bmatrix} = K G_s (\theta_k^e) \eta_k
\]
\[ F(\theta_k^*, \theta_k^c) = \begin{bmatrix} A_p & 0 & 0 & B_c(\theta_k^c) & B_o(\theta_k^c) \\ 0 & 0 & 0 & 0 \\ \Lambda^w(\theta_k^c)C_p & 0 & I - \Lambda^w(\theta_k^c) & 0 \\ 0 & 0 & 0 & \Lambda^e(\theta_k^c) \\ 0 & 0 & 0 & 0 \\ \end{bmatrix} \] (16a)

\[ G_s(\theta_k^c) = \begin{bmatrix} 0 & I & 0 & 0 & 0 \\ \Lambda^e(\theta_k^c)C_p & 0 & I - \Lambda^w(\theta_k^c) & 0 & 0 \\ \end{bmatrix} \] (16b)

\[ G_c(\theta_k^c) = \begin{bmatrix} 0 & B_c(\theta_k^c) \\ I & 0 \\ 0 & 0 \\ 0 & \Lambda^e(\theta_k^c) \\ 0 & I \\ \end{bmatrix} \] (16c)

\[ K = \sum_{q=0}^{m} K_i \beta_q \] (16d)

\[ K_q = \begin{bmatrix} A_{c,q} & B_{c,q} \\ C_{c,q} & D_{c,q} \end{bmatrix}, \quad 0 \leq q \leq m \] (16e)

The equations in (15) are used for controller design in the next sections. By replacing (15b) in (15a), the overall NCS model is obtained as

\[ \eta_{k+1} = A(\theta_k)\eta_k \] (17)

with \( \theta_k \) and \( A(\cdot) \) defined as

\[ \theta_k = \begin{bmatrix} \theta_k^* \\ \theta_k^c \end{bmatrix}, \] (18a)

\[ A(\theta_k) = F(\theta_k^*, \theta_k^c) + G_c(\theta_k^c)KG_s(\theta_k^c). \] (18b)

We denote by \( \Theta^* \subset \mathbb{R}^{2}, \Theta^c \subset \mathbb{R}^{2m}, \) and \( \Theta = \Theta^* \times \Theta^c \) the Lebesgue measurable sets that contain all possible valuations of \( \theta_k^*, \theta_k^c, \) and \( \theta_k \) in (18a) respectively. According to (4), (8), and independence of the random variables, the PDF of \( \theta_k \) denoted by \( f_\theta \) satisfies (19) in the following for every Lebesgue measurable set \( M \subseteq \Theta. \)

\[ P(\theta_k \in M) = \int_M f_\theta(\theta) d\theta \] (19a)

\[ f_\theta([\tau_s^T \theta^c T]) = f_\theta(\theta^c) = \prod_{j=1}^{r} f_\theta^j(\tau_s^j) \] (19b)

\[ f_\theta([\tau_s^1 \cdots \tau_s^r] T) = \prod_{j=1}^{r} f_\theta^j(\tau_s^j), \] (19c)

\[ f_\theta([\tau_s^1 \cdots \tau_s^r; \tau_c^1 \cdots \tau_c^m] T) = \prod_{i=1}^{m} f_{\tau c}^i(\rho_{i}), \prod_{j=1}^{r} f_{\tau s}^j(\rho_{j}) \] (19d)

**Remark 4.** If the delays are zero, it can be easily shown that the total NCS model in (15) reduces to a closed loop model composed of the ordinary discrete-time model of the plant in (1) and the controllers in (2). Given \( j \in \mathbb{I}_e, \) if \( \tau_s^j = 0 \) for every \( k, \) then \( \lambda_{w,j} = \lambda_{c,j} = 1. \) As a result, \( \hat{y}_k^j \) is omitted from (9a) and (9b) such that it can be also eliminated from the total state in (14) and the total NCS model in (15). Similarly, if \( \tau_c^i \) and \( \tau_s^i \) are zero for some \( i \in \mathbb{I}_e, \) then \( \lambda_c^{b,j}, \lambda_s^{b,j}, B_s^i, \) and \( B_c^i \) become zero. As a result, \( u_{k-1}^j \) and \( \hat{u}_k^j \) are omitted from (6a) and (7a) such that they can be also omitted from the total state (14) and the total NCS model (15).

**IV. COST-BASED ANALYSIS**

In this section, a performance analysis framework is presented for the closed loop system (17) by calculating the cost function

\[ J = E\left\{ \sum_{k=0}^{\infty} \gamma^k (Q_{k} \eta_k + u_k^T U u_k) \right\} \] (20)

in which \( Q \) and \( U \) are positive definite matrices and \( \gamma \geq 1. \)

Using (15) to eliminate \( u_k, \) the cost (20) is written as

\[ J = E\left\{ \sum_{k=0}^{\infty} \gamma^k \bar{Q}(\theta_k) \eta_k \right\} \] (21)

with \( \bar{Q}(\cdot) \) defined as

\[ \bar{Q}(\theta_k) = Q + G_c^T(\theta_k^c)K^T U K G_s(\theta_k^c). \] (22)

A. Convergence of the cost

Convergence of the cost function \( J \) as a series can be studied in terms of the following notion of mean square stability.

**Definition 1.** Given \( \gamma \geq 1, \) the system described by (17) is said to be mean square stable (MSS) with rate \( \gamma \) if

\[ \lim_{k \to \infty} \gamma^k E\{\eta_k^T \eta_k\} = 0. \] (23)

Calculation of \( J \) is based on the following result.

**Theorem 1.** Considering the system described by (17) and the cost function \( J \) in (21) with \( \gamma \geq 1, \) if \( \theta_k \in \Theta, k \in \mathbb{Z}_{\geq 0} \) is a sequence of i.i.d. random vectors with PDF \( f_\theta : \Theta \to \mathbb{R}_{\geq 0} \) and there exist \( \sigma > 0 \) such that \( Q(\theta) \geq \sigma I \) for every \( \theta \in \Theta, \) then the following statements are equivalent:

(a) The cost function \( J \) in (21) converges to a finite value.

(b) The system in (17) is MSS with rate \( \gamma. \)

(c) There exist a positive definite matrix \( P \) which uniquely solves the following equation.

\[ P = \int_{\Theta} f_\theta(\theta)[\bar{Q}(\theta) + \gamma A^T(\theta)PA(\theta)]d\theta \] (24)

If any of the above statements hold, then the value of \( J \) is calculated as below given the initial condition \( \eta_0. \)

\[ J = E\{\eta_0^T P \eta_0\} \] (25)

**Proof.** The result is established in three steps by showing that the statement (a) implies (b) in the first step. Showing that (b) implies (c) and the Equation (25) in the second step. Then, showing that (c) implies (a) in the third step which completes the proof.

**First step:** Finiteness of the value of \( J \) in (21) as a series,
requires that the sequence under summation $\gamma^k \eta_k^T \bar{Q}(\theta_k) \eta_k$, $k \in \mathbb{Z}_{\geq 0}$ which has non-negative elements converges as

$$\lim_{k \to \infty} \gamma^k \eta_k^T \bar{Q}(\theta_k) \eta_k = 0 \quad (26)$$

Since $\bar{Q}(\theta) \geq \sigma I$, one can write $0 \leq \gamma^k \eta_k^T \bar{Q}(\theta_k) \eta_k$ which gives

$$0 \leq \gamma^k \eta_k^T \bar{Q}(\theta_k) \eta_k \leq \gamma^k \eta_k^T \eta_k \leq \lim_{k \to \infty} \gamma^k \eta_k^T \eta_k \leq \lim_{k \to \infty} \gamma^k \eta_k^T \bar{Q}(\theta_k) \eta_k$$

The above relationships together with (26) imply (23) which shows that (17) is MSS with rate $\gamma$.

**Second step:** The dimension of $\eta_k$ is denoted by $n_\eta$. Considering an arbitrary symmetric matrix $W$, one can write $\lambda_1(W) I \leq W \leq \lambda_{n_\eta}(W) I$. Multiplying this relationship from left by $\gamma^k \eta_k^T$ and from right by $\eta_k$, tending $k$ to infinity, and taking expectation we have $\lambda_1(W) \lim_{k \to \infty} \gamma^k E\{\eta_k^T \eta_k\} \leq \lim_{k \to \infty} \gamma^k E\{\eta_k^T W \eta_k\} \leq \lambda_{n_\eta}(W) \lim_{k \to \infty} \gamma^k E\{\eta_k^T \eta_k\}$ which according to (23) implies

$$\lim_{k \to \infty} \gamma^k E\{\eta_k^T W \eta_k\} = 0$$

One can recursively substitute $\eta_k$ in the above equation from (17) to obtain

$$\eta_0^T A_w \eta_0 = 0 \quad (27a)$$

$$A_w = \lim_{k \to \infty} \gamma^k E\left\{ \prod_{j=0}^{k-1} A^T(\theta_{k-1-j}) W \prod_{i=0}^{k-1} A(\theta_i) \right\} \quad (27b)$$

The matrix under the expectation operation in (27b) is symmetric. Hence, $A_w$ is also symmetric and diagonalizable. Since (27a) holds for every $\eta_0 \in \mathbb{R}^{n_\eta}$, one can set $\eta_0$ to the eigenvectors of $A_w$ to conclude that all of its eigenvalues are zero which means $A_w = 0$. Applying the svec operator [32, Appendix E] to the equation $A_w = 0$, using the symmetric Kronecker mathematics, and the i.i.d. property of $\theta_k$ we have

$$\lim_{k \to \infty} \gamma^k \prod_{i=0}^{k-1} E\{A^T(\theta_{k-1-i}) \otimes_s A^T(\theta_{k-1-i})\} \times svec\{W\} = 0 \quad \Rightarrow \quad \lim_{k \to \infty} \gamma^k A_k svec\{W\} = 0, \quad (28)$$

$$A_s = \int_\Theta f_0(\theta) [A^T(\theta) \otimes_s A^T(\theta)] d\theta. \quad (29)$$

The vector $svec\{W\}$ has $n_\eta(n_\eta + 1)/2$ elements and can be set arbitrarily by the choice of the symmetric matrix $W$. Hence, (28) implies (30) which results in (31).

$$\lim_{k \to \infty} \gamma^k A_k = 0 \quad (30)$$

$$\rho(\gamma A_s) < 1 \quad (31)$$

Substituting $\eta_k$ recursively from (17), the cost (21) can be calculated as

$$J = E\{\eta_0^T \bar{P} \eta_0\} \quad (32)$$

$$P = E\left\{ \sum_{k=0}^{\infty} \gamma^k \prod_{j=0}^{k-1} A^T(\theta_{k-1-j}) \bar{Q}(\theta_k) \prod_{i=0}^{k-1} A(\theta_i) \right\} \quad (33)$$

The matrix $\bar{P}$ is positive definite due to the structure of the right hand side of (33) and positive definiteness of $\bar{Q}$ in (22).

Applying the svec operator to (33), using the i.i.d. property of $\theta_k$, and (29) we have

$$svec\{\bar{P}\} = E\left\{ \sum_{k=0}^{\infty} \gamma^k svec\left[ \prod_{i=0}^{k-1} A(\theta_i) \bar{Q}(\theta_k) \prod_{i=0}^{k-1} A(\theta_i) \right] \right\}$$

$$svec\{\bar{P}\} = \sum_{k=0}^{\infty} \gamma^k A_k^s q_s \quad (34a)$$

$$q_s = \int_\Theta f_0(\theta) svec\{\bar{Q}(\theta)\} d\theta \quad (34b)$$

Multiplying (34a) by $I - \gamma A_s$ the result is (35) which in combination with (30) gives (36).

$$(I - \gamma A_s) svec\{\bar{P}\} = (I - \lim_{k \to \infty} \gamma^k A_k^s) q_s \quad (35)$$

$$(I - \gamma A_s) svec\{\bar{P}\} = q_s \quad (36)$$

Applying the svec operator, using (29) and (34b), the Equation (24) can be represented as

$$svec\{P\} = \int_\Theta f_0(\theta) svec\{\bar{Q}(\theta) + \gamma A^T(\theta) PA(\theta)\} d\theta \Rightarrow svec\{P\} = q_s + \gamma A_s svec\{P\} \Rightarrow (I - \gamma A_s) svec\{P\} = q_s \quad (37)$$

The matrix $I - \gamma A_s$ is nonsingular due to (31) which implies that $svec\{P\}$ and thus $P$ can be uniquely obtained from (37). This uniqueness requires that $P$ is equal to $\bar{P}$ in (36) which is positive definite as explained under the Equation (33). Hence, the statement (c) is resulted from (b). Also, the fact that $\bar{P} = P$ together with (32) gives (25).

**Third step:** If (24) is satisfied, then uniqueness of its solution $P$ obtained from (37) implies that $P = \bar{P}$ with $\bar{P}$ in (36) which result in (32). Therefore, $J$ converges to a finite value. □

**B. Calculation of the cost**

For calculation of the cost $J$ in (25), it is more convenient to use the ordinary Kronecker mathematics (instead of the symmetric one). Applying the vec operator ([32, Appendix E]) to the Equation (24), one can write

$$\begin{align*}
(I - \gamma A_s) \text{vec}\{P\} &= q_v, \quad (38a) \\
A_v &= \int_\Theta f_0(\theta) [A^T(\theta) \otimes s A^T(\theta)] d\theta, \quad (38b) \\
q_v &= \int_\Theta f_0(\theta) \text{vec}\{\bar{Q}(\theta)\} d\theta. \quad (38c)
\end{align*}$$

To calculate the value of $J$, Equation (25) is written as $J = E\{\eta_0^T \bar{P} \eta_0\} = \text{trace}\{X_0\} = \text{vec}\{X_0\} \text{vec}\{P\}$ with $X_0 = E\{\eta_0^T \eta_0\}$ which results in

$$J = \text{vec}\{X_0\}^T \text{vec}\{P\}, \quad (39)$$

Based on the equations (39) and (38), the value of $J$ can be minimized in order to obtain a set of optimal values for the controller coefficients. However, the optimization algorithms are iterative and the integration operations in (38b) and (38c) need to be performed repeatedly during the iterations in order
to optimize $J$ which can increase the load of computations. To avoid these repeated integration operations, the equations (39) and (38) for calculation of $J$ are reformulated in the following such that the integration operations are needed to be performed only once before starting the optimization iterations. First, the matrix $F$ in (16a) is decomposed as

$$F(\theta^s, \theta^p) = F_c(\theta^p)F_s(\theta^s),$$

(40a)

$$F_c(\theta^p) = \begin{bmatrix} A_p & 0 & 0 & B_c(\theta^p) & B_0(\theta^p) \\ C_p & I & 0 & 0 & 0 \\ 0 & 0 & 0 & \Lambda^c(\theta^p) & \Lambda^b(\theta^p) \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

(40b)

$$F_s(\theta^s) = \begin{bmatrix} I & 0 & 0 & 0 & 0 \\ 0 & I & 0 & 0 & 0 \\ [\Lambda^w(\theta^s) - I]C_p & 0 & I - \Lambda^w(\theta^s) & 0 & 0 \end{bmatrix}. \quad (40c)$$

Replacing $A$ in (38b) from (18b), using (19b) and (40a), we have

$$A_v = \int_{\Theta^s} \int_{\Theta^p} f_{\theta^s}(\theta^p)f_{\theta^p}(\theta^p) \left( [F_c(\theta^p)F_s(\theta^s) + G_c(\theta^p)KG_s(\theta^s)]T \otimes [F_c(\theta^p)F_s(\theta^s) + G_c(\theta^p)KG_s(\theta^s)] + \right) d\theta^s d\theta^p. \quad (41a)$$

Expanding the right hand side of the above equation and using the identity $(A_1A_2) \otimes (B_1B_2) = (A_1 \otimes B_1)(A_2 \otimes B_2)$, one can obtain

$$A_v = (\Omega_{FP} + \Omega_{FG} + \Omega_{GG}) + \Omega_{GG}(K^T \otimes I) + \Omega_{GG}(K^T \otimes K) \Omega_{GG}. \quad (41b)$$

$$\Omega_{MN} = \int_{\Theta^s} f_{\theta^s}(\alpha) [M_i^s(\alpha) \otimes N_i^s(\alpha)]d\alpha \quad \forall M, N \in \{F, G\}, i \in \{s, c\}. \quad (41b)$$

Also, using (22), the Equation (38c) can be written as

$$q_v = \text{vec}(Q) + \Omega_{GG} \text{vec}(K^T\bar{U}K). \quad (42)$$

By calculation of $J$ based on (38a), (39), (41), and (42), the integration operations should be performed only for calculating the matrix coefficients in (41b) before the start of an optimization algorithm. It will not be required to calculate the integration operations in (38b) and (38c) at each iteration.

C. Including setpoints

The controller equations in (2) can be modified to include setpoints $y_{v,j}^{eq,j}$ for $j \in I_3$ by replacing $y_v^{eq,j}$ with $y_{v,j}^{eq,j} - y_v^{eq,j}$. Assuming that the setpoints are constant and concatenating them into a vector $y^{eq}$, the equations (11e) and (11f) are modified to

$$\xi_{k+1} = A_c\xi_k + B_c(y_{v,k} - y^{eq}), \quad \text{equation (11e)} \quad (43a)$$

$$u_k = C_c\xi_k + D_c(y_{v,k} - y^{eq}), \quad \text{equation (11f)} \quad (43b)$$

The modified version of the NCS equations in (17) can be also written as

$$\eta_{k+1} = A(\theta_k)\eta_k - G_c(\theta^p_k)K[0 \ I] y^{eq} \quad (44)$$

with $\eta_k, G_c(\theta^p_k)$, and $A(\theta_k)$ defined in (14), (16c), and (18b) respectively.

It is not trivial to say that the system (44) admits a constant equilibrium state due to the time varying matrix coefficients $A(\theta_k)$ and $G_c(\theta^p_k)$. The following proposition states that such an equilibrium state exists.

**Proposition 1.** The system described by (44) admits a constant equilibrium state $\eta_{eq}$ which satisfies the set of equations in (45) with the matrix coefficients given in Section III.

$$\eta_{eq} = [x_{eq}^{T} \xi_{eq}^{T} y_{eq}^{T} u_{eq}^{T} u_{eq}^{T}]^{T} \quad (45a)$$

$$x_{eq} = A_p x_{eq} + B_p u_{eq} \quad (45b)$$

$$y_{eq} = C_p x_{eq} \quad (45c)$$

$$\xi_{eq} = A_c \xi_{eq} + B_c(y_{eq} - y^{eq}) \quad (45d)$$

$$u_{eq} = C_c \xi_{eq} + D_c(y_{eq} - y^{eq}) \quad (45e)$$

$$B_p = \int_0^\infty \exp(\bar{A}_p\alpha)[\bar{B}_p^1 \cdots \bar{B}_p^m]d\alpha \quad (45f)$$

**Proof.** The result is proved by showing that $\eta_k = \eta_{eq}$ implies $\eta_{k+1} = \eta_k$. If $\eta_k = \eta_{eq}$, then we have the equations (46a) through (46e) in the following by the definition of $\eta_k$ in (14). Then, (11d), (46a) and (45c) imply (46f) which together with (45d), (11b), and (11g) implies (46g) and (46h). The equations (45d), (45e), (43a), (43b), and (46h) imply (46i) and (46j). Also, (46f), (46e), (46d), and (11c) imply (46k). Using (7d) through (7f), (45f), and (12a), it can be easily verified that $B_s(\theta^p_k) + B_0(\theta^p_k) + B_c(\theta^p_k) = B_p$ which together with (46j), (46e), and (46d) implies (46l).

$$x_k = x_{eq} \quad (46a)$$

$$\dot{y}_{k+1} = y_{eq} \quad (46g)$$

$$\xi_k = \xi_{eq} \quad (46b)$$

$$\bar{y}_k = y_{eq} \quad (46c)$$

$$\dot{\xi}_{k+1} = \xi_{eq} \quad (46g)$$

$$\bar{u}_k = u_{eq} \quad (46d)$$

$$\bar{u}_{k+1} = u_{eq} \quad (46k)$$

$$y_{k+1} = y_{eq} \quad (46f)$$

$$\bar{y}_{k+1} = x_{eq} \quad (46l)$$

Then, (46l), (46i), (46g), (46k), and (46j) can be combined into $\eta_{k+1} = \eta_k$ based on the definition of $\eta_k$ in (14). □

Since the equilibrium state $\eta_{eq}$ is non-zero in general, \(\lim_{k \to \infty} \eta_{eq}^T Q(\theta_k) \eta_k\) is also non-zero, the cost $J$ in (21) does not converge, and the Theorem 1 cannot be applied directly. In this case the modified cost $J$ defined in the following with $\eta_0 = 0$ can be considered as a performance measure for the response to step changes in setpoints.

$$\dot{J} = E \left\{ \sum_{k=0}^{\infty} \gamma^k (\eta_k - \eta_{eq})^T Q(\theta_k)(\eta_k - \eta_{eq}) \right\} \quad (47)$$

To calculate of the above cost function based on the Theorem 1, a change of variable $\hat{\eta}_k = \eta_k - \eta_{eq}$ can be applied to translate the equilibrium state back to the origin such that the system equations are represented as (17). Then,
the Theorem 1 can be applied to compute $\dot{J}$ with $\eta_0 = 0$ as $\dot{J} = \dot{\eta}_0^T P \dot{\eta}_0 = \eta_{eq}^T P \eta_{eq}$ with $P$ obtained from (38a). However, convergence of the state for a specific initial condition does not imply the stability of NCS. This can cause stability issues when $\dot{J}$ should be optimized for controller design. To avoid instability, one can consider a random initial condition $\dot{\eta}_0$ with mean value $\eta_{eq}$ and covariance matrix $W_v$ as a design parameter. Then, $\dot{J}$ can be computed as

$$\dot{J} = \text{trace}\{\eta_{eq}^T \eta_{eq} + W_v\} P. \quad (48)$$

The covariance matrix $W_v$ can be selected as

$$W_v = w_v \text{diag}\{\eta_{eq}\}^2 \quad (49)$$

such that the ratio of standard deviation to mean value of $[\dot{\eta}_0]$, is a constant $w_v \geq 0$ for $i \in [1..n_q]$. The value of $w_v$ compromises between the capabilities of $\dot{J}$ for representing the step response performance and the overall NCS stability.

V. CASE STUDY

In this part, design of an NCS which is shown in Fig. 3 is considered. In this NCS, three DC power sources equipped with DC/DC converters are connect in parallel to supply electric power to a load modeled by a resistor $R_L$. The control objective is to regulate the load voltage denoted by $v_L$ while sharing the load current between the sources equally.

Each converter is composed of a switching circuit connected to an LC filter. The input voltage of the filters are denoted by $v_{in,i}$ for $i \in \{1, 3\}$ whose average values can be set by the switching circuit. The values of the inductor, the parasite resistor of inductor, and the capacitor for the $i$th converter ($i \in \{1, 3\}$) are denoted by $L_i, r_i, C_i$ respectively. The output of the $i$th converter (its capacitor voltage) is connected to the load via a resistor $R_i$. The system is described by the following set of equations in which $i^c, i^e$ and $i^l$ are the average values of the inductor current, the converter output current, and the capacitor voltage for the $i$th filter respectively.

$$L_i \frac{d}{dt} i^c_i + r_i i^e_i = v_{in,i} - i^l_i \quad (50a)$$

$$C_i \frac{d}{dt} i^e_i = i^c_i - i^e_i \quad (50b)$$

$$i^l_i = (v^c_i - v_L)/R_i \quad (50c)$$

$$\sum_{i=1}^3 i^l_i = v_L/R_L \quad (50d)$$

The $i$th converter for $i \in \{1, 3\}$ is controlled by a local proportional-integral (PI) controller to make its output voltage $v^i$ track a setpoint $v^{*i}$ according to the following equations.

$$\dot{\varphi}_i = v^{*i} - v^i \quad (51a)$$

$$v^i_{in} = k_i^f \varphi_i + k_i^l (v^{*i} - v^i) \quad (51b)$$

The equations (50) and (51) can be augmented to a set of equations in the form of (1). These equations are then discretized as in (7) with sampling period $h$ and controller execution time offsets $\bar{\tau}_i, i \in \{1, 3\}$. The values of system parameters are given in Table I. The converter controller gains in (51) are obtained by minimizing $v^2 + 0.1 v^2_{in,i}$ with $\bar{\tau}_i = 7.4A$, $v^c(0) = 40V$, $v^{*i} = 0$ and $3R_L$ resistive load.

A. Networked distributed control

Consider a continuous-time system with input $u(t) \in \mathbb{R}^n$ and outputs $y(t) \in \mathbb{R}^n$, $z(t) \in \mathbb{R}$ which is controlled by $n$ distributed controllers according to the consensus algorithm

$$\dot{u} = KLy + b(z - z_d) \quad (52)$$

in which $K \in \mathbb{R}^{n \times n}$ is diagonal, $L$ is the Laplacian matrix of the controller interconnections directed graph $G$, and $b \in \mathbb{R}^n$.

The consensus algorithm in (52) reduces to the ordinary consensus algorithm if $b = 0$ [6]. It can be easily shown that if the closed loop system is asymptotically stable and $G$ is a connected graph, then we have $z(t) = z_d$ and $y_j(t) = y_j(t)$ for $i, j \in \{1..n\}$ as $t \to \infty$. A discrete-time version of (52) for control of the converters in Fig. 3 can be written as

$$v^{*i}_{k+1} = v^{*i}_{k} + \left[ k_i \sum_{j \neq i} \phi \left( \varphi^c_{o,k} - \varphi^e_{o,k} \right) + b_i (v^c_{L,k} - v^c_{L,k}) \right] \quad (53)$$

![Fig. 3. Networked control of parallel converters (gray blocks indicate the distributed controllers).](image)

### TABLE I

| Name | Value |
|------|-------|
| $L_1$ | 1.1 mH |
| $L_2$ | 1.5 mH |
| $L_3$ | 2.3 mH |
| $C_1$ | 60 $\mu$F |
| $C_2$ | 40 $\mu$F |
| $C_3$ | 70 $\mu$F |
| $r_1$ | 0.07 $\Omega$ |
| $r_2$ | 0.12 $\Omega$ |
| $r_3$ | 0.09 $\Omega$ |
| $R_1$ | 1.527 |
| $R_2$ | 3343.7 |
| $R_3$ | 1959.8 |
| $k_{P1}$ | 1.207 |
| $k_{P2}$ | 1.807 |
| $k_{P3}$ | 1.749 |
| $k_{I1}$ | 2562.6 |
| $k_{I2}$ | 3433.7 |
| $h$ | 10 ms |
| $\tau_c$ | 4 ms |
in which the feedback signals from $i_o^i$ for $i \in \{1..3\}$, and $u_L$ are indicated by an ‘$e$’ in the superscript according to the notations introduced in sections II and III.

It is assumed that the distributed controllers are interconnected by a complete graph. The $i$th distributed controller in Fig. 3 implements the consensus algorithm in (53) for $i \in \{1..3\}$. The controller equations in (53) can be augmented to a set of equations in the form of (2).

The delay PDFs in (4) are assumed to be given as a discrete PDF $f(\tau) = \sum_{i=0}^{N} p_i \delta(\tau - ih/N)$ for all network links where $\delta$ is the Dirac delta function. The function $f$ is plotted in the Fig. 4 such that $p_i \delta(\tau - ih/N)$ is shown as a vertical bar of height $p_i$ horizontally centered at $ih/N$. The plot shows a typical probability distribution over a controller area network (CAN) which is shared between the NCS and several other applications [33]. The PDF is plotted over the delay range $\tau \in [0, h]$ according to the Remark 3. The integral of $f$ over $[h, \infty]$ is 0.37 which is also indicated in the figure.

**B. Design and simulation**

The information in previous part can be used to obtain an NCS model in the form of (15). In order to apply the NCS analysis method of Section IV, the following cost function

$$J = \sum_{k=0}^{\infty} \left[ (y_{L,k} - y^{opt}_{L})^2 + w_u \sum_{j=1}^{4} (v_{k,j}^* - v_{eq,j})^2 \right] \quad (54)$$

The distributed controllers in Fig. 3 are designed by optimizing $J$ for the NCS response to a setpoint $y^{opt}_{L} = 40$ from zero initial conditions. The method of the Subsection IV-C is applied for calculation of $J$ with $w_u$ in (49) set to 0.1. The fminsearch function from the GNU Octave software is used for performing the optimization. Two sets of controller coefficients are designed and presented in Table II. The resulting set of controller coefficients for the delay PDF $f(\tau)$ in the Fig. 4 is denoted as the networked tuning. Also, the set of coefficients obtained for zero delay PDF $f(\tau) = \delta(\tau)$ is denoted as the no-delay tuning.

The simulated time response of the load voltage $v_L$ and the voltage command to the first converter $v_{eq,1}$ are shown in Fig. 5 for different conditions. For simulating the networked communication, the delays are generated randomly according to the delay PDF of Fig. 4. The same sequence of delay values is used in simulation of the NCS response for both networked and no-delay tunings.

According to Fig. 5, the no-delay tuning results in the fastest convergence if there are no delays. However, the NCS becomes oscillatory in the presence of delays. The networked tuning shows to be robust to the delays at the cost of a little performance loss in absence of delays. Due to the randomness of delays, the NCS response is also random and the plots of Fig. 5 for the case of networked communications are samples of the random NCS response. The more precise criterion for comparison of the different cases is the averaged cost value $J$ in (54) which is given in Table III for the different conditions.

**VI. CONCLUSIONS**

A non-conservative design framework has been proposed in this work for distributed control systems over real-time network links. The motivation for considering real-time links is the continuous development of real-time capabilities in the commonly available data networks. An accurate model of the NCS is obtained without putting bounds on the communication delays. The result is a stochastic state space model of the

| Coefficient name | Optimal coefficient value |
|------------------|---------------------------|
| $k_1$            | 0.991                     | 1.143                     |
| $k_2$            | 0.888                     | 0.719                     |
| $k_3$            | 0.912                     | 0.707                     |
| $b_v$            | 0.492                     | 0.292                     |
NCS with a fixed number of states independently of the delay magnitudes. The value of a quadratic cost function is calculated for such a stochastic model. The cost function can be adopted to a performance measure for step tracking response which allows for optimization based control design. A method is proposed for drawing the heavier part of the cost function calculations out of the optimization iterations and perform them only once at the beginning of optimization. The results were applied to a case study in which three parallel converters are controlled by three distributed controllers based on the consensus algorithm to control the load voltage while sharing the load current appropriately. Evaluations and comparisons show that a considerable performance improvement is achieved by applying the proposed design framework.

REFERENCES

[1] J. P. Hespanha, P. Naghshtabrizi, and Y. Xu, “A survey of recent results in networked control systems,” Proceedings of the IEEE, vol. 95, no. 1, pp. 138–162, Jan 2007.

[2] L. Hetel, C. Fiter, H. Omran et al., “Recent developments on the stability of systems with aperiodic sampling: An overview,” Automatica, vol. 76, pp. 309 – 335, 2017.

[3] X. Zhang, Q. Han, and X. Yu, “Survey on recent advances in networked control systems,” IEEE Transactions on Industrial Informatics, vol. 12, no. 5, pp. 1740–1752, Oct 2016.

[4] M. Li, Y. Chen, A. Zhou et al., “Adaptive tracking control for networked control systems of intelligent vehicle,” Information Sciences, vol. 503, pp. 493 – 507, 2019.

[5] M. S. Mahmoud and Y. Xia, Networked Control Systems: Cloud Control and Secure Control. Butterworth-Heinemann, 2019.

[6] F. L. Lewis, H. Zhang, K. Hengster-Movric et al., Cooperative Control of Multi-Agent Systems: Optimal and Adaptive Design Approaches, 1st ed., ser. Communications and Control Engineering. Springer-Verlag London, 2014.

[7] D. Mukherjee and D. Zelazo, “Consensus of higher order agents: Robustness and heterogeneity,” IEEE Transactions on Control of Network Systems, vol. 6, no. 4, pp. 1323–1333, Dec 2019.

[8] K. Liu, A. Selivanov, and E. Fridman, “Survey on time-delay approach to networked control,” Annual Reviews in Control, vol. 48, pp. 57 – 79, 2019.

[9] B. Tang, J. Wang, and Y. Zhang, “A delay-distribution approach to stabilization of networked control systems,” IEEE Transactions on Control of Network Systems, vol. 2, no. 4, pp. 382–392, Dec 2015.

[10] M. Rouamel, S. Gherbi, and F. Bourahala, “Robust stability and stabilization of networked control systems with stochastic time-varying network-induced delays,” Transactions of the Institute of Measurement and Control, 2020. Early access.

[11] W. P. M. H. Heemels, A. R. Teel, N. van de Wouw et al., “Networked control systems with communication constraints: Tradeoffs between transmission intervals, delays and performance,” IEEE Transactions on Automatic Control, vol. 55, no. 8, pp. 1781–1796, Aug 2010.

[12] V. S. Dolik, D. P. Borgers, and W. P. M. H. Heemels, “Output-based and decentralized dynamic event-triggered control with guaranteed $L_2$-gain performance and zero-freeze,” IEEE Transactions on Automatic Control, vol. 62, no. 1, pp. 34–49, Jan 2017.

[13] A. Da Silva and C. Kawan, “Robustness of critical bit rates for practical stabilization of networked control systems,” Automatica, vol. 93, pp. 397 – 406, 2018.

TABLE III
THE VALUES OF THE COST FUNCTION $J$ IN (54) FOR DIFFERENT CASES.

| Controller tuning | Communication | Value of $J$ |
|-------------------|--------------|-------------|
| No-delay          | No-delay     | 12805       |
| No-delay          | Networked   | 113665      |
| Networked         | No-delay     | 17926       |
| Networked         | Networked   | 20147       |

Babak Tavassoli received the BS in Electronics Engineering in 1998, MS and Ph.D. in Control Engineering in 2001 and 2009, all from the University of Tehran, Iran. From 2009 to 2010 he was with the Research Institute of Petroleum Industry working on control and monitoring in refineries. Since 2010, he has been with the K. N. Toosi University of Technology as an assistant professor. His activities include teaching BS and MS courses (more than 10 titles) and supervising student theses or projects. His research interests are hybrid dynamical systems, networked control systems, and model predictive control. He is also interested in industrial automation technologies and industrial communication protocols.