INTERNATIONAL JOURNAL OF ROBUST AND NONLINEAR CONTROL
Int. J. Robust Nonlinear Control 2010; **20**:1529–1548
Published online 20 November 2009 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/rnc.1535

Robust flow control in data-communication networks with multiple time-delays

Hakkı Ulaş Ünal[1], Banu Ataşlar-Ayyıldız[2], Altuğ İftar[1,*,†] and Hitay Özbay[3]

[1]*Department of Electrical and Electronics Engineering, Anadolu University, Eskişehir 26470, Turkey*
[2]*Department of Electronics and Communication Engineering, Yıldız Technical University, İstanbul 34349, Turkey*
[3]*Department of Electrical and Electronics Engineering, Bilkent University, Ankara 06800, Turkey*

SUMMARY

Robust controller design for a flow control problem where uncertain multiple time-varying time-delays exist is considered. Although primarily data-communication networks are considered, the presented approach can also be applied to other flow control problems and can even be extended to other control problems where uncertain multiple time-varying time-delays exist. Besides robustness, tracking and fairness requirements are also considered. To solve this problem, an $\mathscr{H}^\infty$ optimization problem is set up and solved. Unlike previous approaches, where only a suboptimal solution could be found, the present approach allows to design an optimal controller. Simulation studies are carried out in order to illustrate the time-domain performance of the designed controllers. The obtained results are also compared to the results of a suboptimal controller obtained by an earlier approach. Copyright © 2009 John Wiley & Sons, Ltd.

Received 4 April 2007; Revised 15 July 2009; Accepted 14 September 2009

KEY WORDS: time-delay systems; multiple time-delays; robust control; $\mathscr{H}^\infty$-based control; flow control; data-communication networks

1. INTRODUCTION

In data communication networks, network providers should satisfy the desired Quality of Service (QoS) to the users. Most important problem that hinders QoS is congestion. Congestion occurs at a node of the network, when the total incoming flow to that node exceeds the capacity of the outgoing link of that node. In such a situation, long queuing delays may result and/or buffers may overflow, which would result in the loss of data. To avoid such undesirable behaviour, congestion control mechanisms must be implemented. One such mechanism is the flow control, which regulates data sending rate of the sources. In general, there are two flow control methods: *rated-based* [1–3] and *window-based* [4, 5]. Although window-based control is widely used for end-to-end congestion control in TCP/IP networks, rate-based control is preferred for edge-to-edge control in newer generation networks [6, 7].

In the rate-based flow control method, the controller is implemented at the *bottleneck node* to regulate the rate of data packets sent from the sources, which feed this node. The existence of time-delays in the network makes the flow control problem challenging. Furthermore, these time-delays are also uncertain and

*Correspondence to: Altuğ İftar, Department of Electrical and Electronics Engineering, Anadolu University, Eskişehir 26470, Turkey.
†E-mail: aiftar@anadolu.edu.tr

time-varying. Moreover, since there usually are more than one source feeding a bottleneck, there are multiple time-delays.

There are a number of different controller design methods for systems that involve time-delays (e.g. see [8] and references therein). The main difficulty in designing controllers for time-delay systems is that, such systems are infinite-dimensional. Toker and Özbay [9] used the operator theory [10, 11] to formulate an $\mathscr{H}^\infty$-optimal controller design approach for single-input single-output (SISO) infinite-dimensional systems. Nagpal and Ravi [12] and Tadmor [13] used state-space methods and Meinsma and Zwart [14] used J-spectral factorizations to solve the same problem for systems that involve a single delay. Mirkin and Raskin [15] considered parameterization of controllers, which stabilize a linear time-invariant (LTI) system with a single delay. The general solution of the $\mathscr{H}^\infty$-optimal controller design problem for systems which involve multiple delays, however, was not available up until recently. Meinsma and Mirkin [16] formulated a solution to this problem by splitting the problem into a nested sequence of simpler problems each with a single delay.

An $\mathscr{H}^\infty$-based controller design approach for the rate-based flow control problem was proposed in [17] by using the design techniques in [9]. The implementation of this controller was later illustrated in [18]. In [17, 18], however, the uncertain delays were assumed to be time-invariant. Furthermore, since the design approach in [9] is for SISO systems, the controller was designed for the multiple delays considering the longest delay and equalizing the delays in the other channels to the longest one. The case of uncertain time-varying multiple time-delays was later considered in [19], where a rate-based flow controller was designed, which is robust to variations in such delays. However, in [19], the controller was obtained by defining separate $\mathscr{H}^\infty$ control problems for each channel. The solutions to these problems were then weighted and blended to obtain the overall controller. Therefore, the overall solution presented in [19] is not optimal, but suboptimal in the $\mathscr{H}^\infty$ sense. To find an optimal solution to this problem, the approach of [16] was first considered in [20]. Then, in [21], where the general framework of the present work was reported, the approach of [16] was used to obtain an $\mathscr{H}^\infty$-optimal solution to the problem presented in [19]. In [21], however, it was assumed that the uncertain parts of the time-delays are always non-negative. This was achieved by using the minimum possible time-delays as the *nominal time-delays*, which introduces two disadvantages: (i) the best performance is obtained, not for the plant with most probable time-delays, but for the plant with minimum time-delays; (ii) robustness range must be larger, since the absolute value of the maximum allowable variation on the time-delays must now be twice compared to the case when the nominal time-delays are chosen as the average of the minimum and maximum possible time-delays. The reason for using the minimum possible time-delays as the nominal time-delays in [21] was to ensure the causality of the uncertainty block. The necessity of using causal uncertainty blocks stems from the fact that the *small-gain theorem* [22, 23], which is needed during the design process, is not in general valid for non-causal systems (see Example 1 in [24]). Recently, however, Ünal and İftar [24] showed that, under certain conditions, a small-gain theorem is valid for systems that involve non-causal blocks. Extending this result, Ünal and İftar [25] also showed that non-causal uncertainty blocks can indeed be used in the robust flow controller design for networks with multiple time-delays. Although an alternative approach (see Remark 3 at the end of Section 2) exists, in the present work, we remove the assumption of non-negativity on the uncertain parts of the time-delays, by allowing the uncertainty block to be non-causal. The problem is presented in Section 2. Its solution is given in Section 3. A number of simulations are presented in Section 4, to illustrate the performance of the controller in a number of typical cases. Concluding remarks are given in the last section.

The mathematical model considered in this work may also appear in flow control problems in areas other than data-communication networks. Typical examples are transportation networks, material transport systems (e.g. oil or gas pipelines), manufacturing systems, and process control. Therefore, the controller design approach presented here may also be applied in those areas (e.g. see Chapters 2 and 7 of [8]). In fact, this approach may be extended to the control of any

DOI: 10.1002/rnc

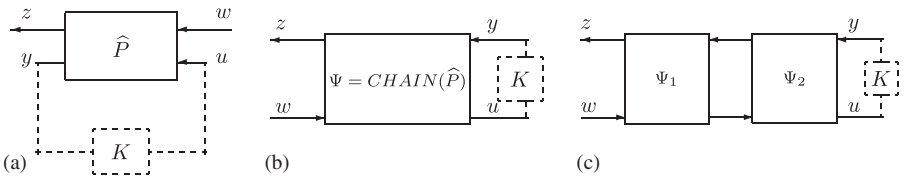

Figure 1. (a) Input–output relation of a 4-block system; (b) it's chain-scattering representation; and (c) cascade connection of two systems in chain-scattering representation.

integrating system with multiple uncertain time-varying time-delays in its input and/or output channels.

1.1. Notation and preliminaries

Throughout, I and 0 respectively denote an identity matrix and a zero matrix of appropriate dimensions. For a positive integer k, I_k denotes the $k\times k$ identity matrix and $\mathbf{1}_k$ denotes the $1\times k$ matrix of all 1's. For a vector w, n_w denotes the dimension of w. For two vectors z and w, $J_{zw}:=\text{blockdiag}(I_{n_z}, -I_{n_w})$ is a signature matrix. For a matrix M, M^{T} denotes its transpose, M^{-1} denotes its inverse, and $M^{-\mathrm{T}}$ denotes the transpose of its inverse. An $\mathscr{H}^\infty$ mapping Q is called *contractive* if $\|Q\|_\infty<1$.

$$\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right]$$

denotes the transfer function matrix (TFM) $C(sI-A)^{-1}B+D$. For a system with TFM $\mathrm{e}^{-hs}C(sI-A)^{-1}B$, π_h denotes the *completion operator*, which is defined as

$$\pi_h\left(\mathrm{e}^{-hs}\left[\begin{array}{c|c} A & B \\ \hline C & 0 \end{array}\right]\right)=\left[\begin{array}{c|c} A & B \\ \hline C\mathrm{e}^{-Ah} & 0 \end{array}\right]-\mathrm{e}^{-hs}\left[\begin{array}{c|c} A & B \\ \hline C & 0 \end{array}\right],$$

whose impulse response, $g(t)$, is limited to the time interval $[0,h)$:

$$g(t)=\begin{cases} Ce^{A(t-h)}B, & 0\leqslant t<h \\ 0, & \text{otherwise} \end{cases}. \tag{1}$$

Consider a system with TFM

$$\widehat{P}=\begin{bmatrix} \widehat{P}_{11} & \widehat{P}_{12} \\ \widehat{P}_{21} & \widehat{P}_{22} \end{bmatrix}$$

with input $\left[\begin{smallmatrix} w \\ u \end{smallmatrix}\right]$ and output $\left[\begin{smallmatrix} z \\ y \end{smallmatrix}\right]$, and a feedback connection $u=Ky$ as depicted in Figure 1(a). The closed-loop TFM from w to z is given as $F_l(\widehat{P},K):=\widehat{P}_{11}+\widehat{P}_{12}K(I-\widehat{P}_{22}K)^{-1}\widehat{P}_{21}$, where $F_l(\cdot,\cdot)$ denotes the *lower linear fractional transformation* (lower-LFT) [26].

Note that, for the system in Figure 1(a),

$$z=\widehat{P}_{11}w+\widehat{P}_{12}u, \tag{2}$$

$$y=\widehat{P}_{21}w+\widehat{P}_{22}u. \tag{3}$$

If $\widehat{P}_{21}^{-1}$ exists, then (3) can be written as

$$w=-\widehat{P}_{21}^{-1}\widehat{P}_{22}u+\widehat{P}_{21}^{-1}y. \tag{4}$$

By replacing w in (2) by (4), we obtain

$$z=(\widehat{P}_{12}-\widehat{P}_{11}\widehat{P}_{21}^{-1}\widehat{P}_{22})u+\widehat{P}_{11}\widehat{P}_{21}^{-1}y,$$

$$w=-\widehat{P}_{21}^{-1}\widehat{P}_{22}u+\widehat{P}_{21}^{-1}y.$$

Therefore, if we define

$$\Psi:=\begin{bmatrix} \widehat{P}_{12}-\widehat{P}_{11}\widehat{P}_{21}^{-1}\widehat{P}_{22} & \widehat{P}_{11}\widehat{P}_{21}^{-1} \\ -\widehat{P}_{21}^{-1}\widehat{P}_{22} & \widehat{P}_{21}^{-1} \end{bmatrix}=:\begin{bmatrix} \Psi_{11} & \Psi_{12} \\ \Psi_{21} & \Psi_{22} \end{bmatrix},$$

the system in Figure 1(a) can be represented as in Figure 1(b), where Ψ is called the *chain-scattering representation* of $\widehat{P}$ and denoted as $\Psi=\text{CHAIN}(\widehat{P})$ [27]. When the feedback connection $u=Ky$ is made, then the closed-loop TFM from w to z in Figure 1(b) is given as $HM(\Psi,K):=(\Psi_{11}K+\Psi_{12})(\Psi_{21}K+\Psi_{22})^{-1}$, where $HM(\cdot,\cdot)$ denotes the *homographic transformation* [27]. From Figure 1(a), the same TFM can also be written as $F_l(\widehat{P},K)$. Thus, $F_l(\widehat{P},K)=HM(\Psi,K)$. The main reason for using the chain scattering representation is for its simplicity in representing cascade

DOI: 10.1002/rnc

connections. The cascade connection of two chain scattering representations Ψ_1 and Ψ_2, as shown in Figure 1(c), is represented as the product $\Psi_1\Psi_2$ of each chain scattering representation. Furthermore, the closed-loop TFM in Figure 1(c), from w to z, is obtained as

$$HM(\Psi_1, HM(\Psi_2, K)) = HM(\Psi_1\Psi_2, K). \tag{5}$$

Moreover, if Ψ in Figure 1(b) is invertible, and the closed-loop TFM $Q := HM(\Psi, K)$ is known, then K is easily obtained as [27]:

$$K = HM(\Psi^{-1}, Q). \tag{6}$$

2. PROBLEM STATEMENT

2.1. Network model

In this study, we consider the flow control problem in a data communication network with n sources feeding a single bottleneck node. The flow controller, which is to be designed, is implemented at the bottleneck node. The controller calculates a rate command for each source to adjust the rate of data it sends to the bottleneck node in order to regulate the queue length at the bottleneck node so that congestion is avoided.

In an actual data-communication network, data flow consists of discrete entities, since data packets are handled individually. Since a model that reflects this behaviour would be very complicated, many researchers have used continuous flow models, which are customarily called *fluid-flow* models (e.g. see Chapters 5 and 6 of [28] and references therein). Therefore, here we will also use a fluid-flow model for the purpose of controller design. For the simulation studies in Section 4, however, we will use a more realistic discrete model and show that a controller that is designed based on a fluid-flow model can also work well when the actual flow is discrete.

According to our fluid-flow model, the dynamics of the queue length are given as [19]:

$$\dot{q}(t) = \sum_{i=1}^{n} r_i^b(t) - c(t), \tag{7}$$

where

$q(t)$ is the queue length at the bottleneck node at time t,
$r_i^b(t)$ is the rate of data received by the bottleneck node at time t from the ith source, $i = 1, \ldots, n$,
$c(t)$ is the outgoing rate of data from the bottleneck node at time t, which equals to the capacity of the outgoing link assuming that $q(t)$ is non-zero [19].

The total amount of data received at the bottleneck node from the ith source, $i = 1, \ldots, n$, by time t is given as [19]:

$$\int_0^t r_i^b(\varphi)\,\mathrm{d}\varphi = \begin{cases} \displaystyle\int_0^{t-\tau_i^f(t)} r_i^s(\varphi)\,\mathrm{d}\varphi, & t-\tau_i^f(t) \geqslant 0 \\ 0, & t-\tau_i^f(t) < 0 \end{cases}, \tag{8}$$

where

$r_i^s(t)$ is the rate of data sent from the ith source at time t and is assumed to be (for controller design purposes) equal to $r_i^c(t)$,
$r_i^c(t) := r_i(t-\tau_i^b(t))$ is the rate command received by the ith source at time t, and
$r_i(t)$ is the rate command for the ith source issued by the controller at time t.

By taking the derivative of both sides of (8) and using $r_i^s(t) = r_i^c(t) = r_i(t-\tau_i^b(t))$, the rate of data received by the bottleneck node, $r_i^b(t)$, is given in terms of the rate command, $r_i(t)$, as follows:

$$r_i^b(t) = \begin{cases} (1-\dot{\delta}_i^f(t))r_i(t-\tau_i(t)), & t-\tau_i^f(t) \geqslant 0 \\ 0, & t-\tau_i^f(t) < 0 \end{cases}. \tag{9}$$

Here, $\tau_i(t) = \tau_i^b(t) + \tau_i^f(t)$ is the *round-trip time-delay*, where

$\tau_i^b(t) = h_i^b + \delta_i^b(t)$ is the *backward time-delay* at time t, which is the time required for the rate command to reach the ith source. Here, h_i^b is the nominal time-invariant known backward time-delay, and $\delta_i^b(t)$ is the time-varying backward time-delay uncertainty,
$\tau_i^f(t) = h_i^f + \delta_i^f(t)$ is the *forward time-delay* at time t, which is the time required for the data sent

DOI: 10.1002/rnc

from the ith source to reach the bottleneck node. Here, h_i^f is the nominal time-invariant known forward time-delay, and $\delta_i^f(t)$ is the time-varying forward time-delay uncertainty.

The nominal round-trip time-delay for the ith channel of the system is $h_i = h_i^b + h_i^f$, and the time-varying round-trip time-delay uncertainty is $\delta_i(t) = \delta_i^b(t) + \delta_i^f(t)$. It is assumed that the uncertainties are bounded as follows:

$$|\delta_i(t)| < \delta_i^+, \quad |\dot{\delta}_i(t)| < \beta_i, \quad |\dot{\delta}_i^f(t)| < \beta_i^f \tag{10}$$

for some bounds $\delta_i^+ > 0$ and $0 < \beta_i^f \leqslant \beta_i < 1$. It is further assumed that $\delta_i(t)$ is such that $\tau_i(t) \geqslant 0$ at all times. In a real application, there also exist some *hard constraints*, such as non-negativity constraints and upper bounds on the queue length and data rates. In this work, however, we assume that such constraints are always satisfied for the purpose of controller design. We will consider such constraints in Section 4, while running simulations.

Remark 1
The term $\dot{\delta}_i^f$ in (9), which results from the differentiation of (8), represents the *jitter* effect [29] and is a characteristic of networks with a time-varying delay. Note that the jitter effect appears only due to the variations in the forward time-delays and not in the backward time-delays, since the variations in the backward time-delays does not induce any variations on the data flow.

It should also be remarked that, in a data-communication network, the round-trip time-delay for an individual data packet can be measured after this packet travels to its destination and its notification comes back to the source. That is the round-trip time-delay, $\tau_i(t)$, can be measured at time $t + \tau_i(t)$; i.e. after a time-delay which equals to itself. Therefore, this measurement cannot be used by the controller. Of course, $\tau_i(t)$ can be estimated based on such measurements (e.g. see [30]). However, good estimates cannot be obtained when variations on $\tau_i(t)$ are rather fast and random [31]. In such a case, to guarantee stability and obtain good performance, a robust controller design approach is needed as it was undertaken in [8, 32–36], among other places.

2.2. Control problem

Our aim is to design a controller, for the above described system, to regulate the queue length $q(t)$. The controlled system must be robustly stable against all time-varying uncertainties in the time-delays which satisfy (10). Assuming that $\lim_{t\to\infty} c(t) = c_\infty$ exists, the nominal system must satisfy the *tracking* requirement:

$$\lim_{t\to\infty} q(t) = q_d, \tag{11}$$

and the *weighted fairness* [19] requirement:

$$\lim_{t\to\infty} r_i(t) = \alpha_i c_\infty, \quad i = 1, \ldots, n. \tag{12}$$

Here, q_d is the *desired queue length*, which is chosen as some positive number (typically half the buffer size) and $\alpha_i > 0$, $i = 1, \ldots, n$, are the *fairness weights* [19], which satisfy $\sum_{i=1}^n \alpha_i = 1$.

To obtain an *uncertainty model*, we define the *uncertainty in the queue length* as $\delta_q(t) := q(t) - q_0(t)$, where

$$q_0(t) := \int_0^t \left[\sum_{i=1}^n r_i(v - h_i) - c(v) \right] \mathrm{d}v + q(0)$$

is the *nominal queue length*. By defining $r_i^h(t) := r_i(t - h_i)$, $i = 1, \ldots, n$, and proceeding as in [19] (by using r_i^h instead of r_i), we obtain $\delta_q(t) = \sum_{i=1}^n \delta_q^i(t)$, where $\delta_q^i(t)$ is the output of the system shown in Figure 2. In Figure 2, $M_\diamond$ represents multiplication by $\diamond$. The difference between Figure 2 and Figure 10 of [19] is that $r_i(t)$ and $\tau_i(t)$ in [19] are now respectively replaced by $r_i^h(t)$ and $\delta_i(t)$. This makes the system in Figure 2 not necessarily causal, since we may have $\delta_i(t) < 0$. In this case, the delay blocks in Figure 2 are in fact time-advance blocks and the integral is a non-causal integral. The reason for this difference is that, here, unlike in [19], we would like to take the nominal delays outside the plant (see Figure 3) in order to apply the approach of [16]. This will allow us to design an optimal controller, opposed to [19], where a suboptimal controller was designed. Following steps similar to those in [19], we choose

$$\varphi_{i,1} = \sqrt{2}\,\frac{\beta_i + \beta_i^f}{\sqrt{1 - \beta_i}} \quad \text{and} \quad \varphi_{i,2} = 2\sqrt{2}\,\delta_i^+,$$

DOI: 10.1002/rnc

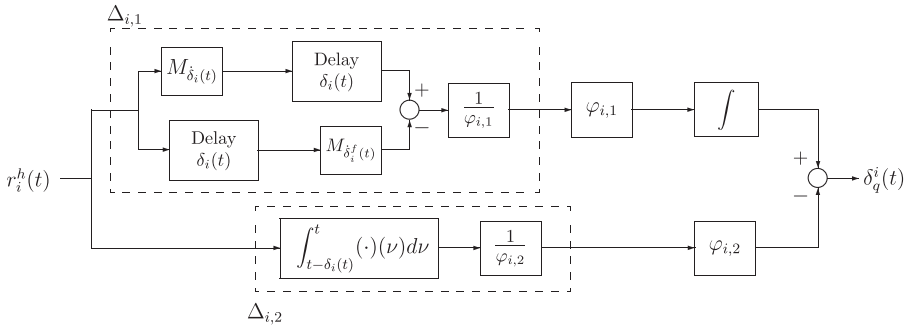

Figure 2. Uncertainty model.

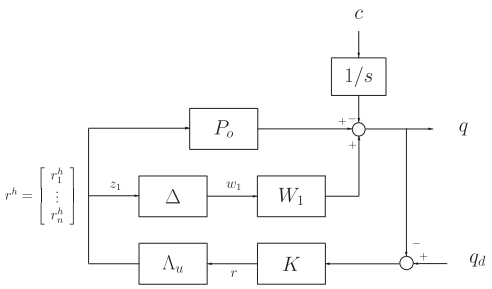

Figure 3. Overall system.

so that the $\mathscr{L}^2$-induced norms of the LTV systems $\Delta_{i,1}$ and $\Delta_{i,2}$ are both less than $1/\sqrt{2}$.

Remark 2

We note that $\varphi_{i,2}$ could be taken as $\sqrt{2}\delta_i^+$, as shown in [25]. Here, however, we let $\varphi_{i,2}=2\sqrt{2}\delta_i^+$ as in [19], so that we can compare our results to those of [19].

Without the loss of generality, let us assume that $h_1 \geqslant h_2 \geqslant \cdots > h_n \geqslant 0$. Let N be the number of distinct h_i's and let us rename the nominal time-delays as $\bar{h}_1 > \bar{h}_2 > \cdots > \bar{h}_N \geqslant 0$ so that all $\bar{h}_i$'s are distinct. For this, let $\bar{h}_1 = h_1$, $\bar{h}_2 = h_{i_2}$, where i_2 is the smallest index such that $h_{i_2} < h_1$, $\bar{h}_3 = h_{i_3}$, where i_3 is the smallest index such that $h_{i_3} < h_{i_2}$, and so on. Also let l_i $(i=1,\ldots,N)$ be the number of channels with nominal round trip time-delay $\bar{h}_i$. Then, $\sum_{i=1}^{N} l_i = n$.

Now, we can describe the overall system as shown in Figure 3, where $P_o(s) = (1/s)\mathbf{1}_n$ is the *nominal plant*, K is the controller to be designed, $\Lambda_u(s) = \text{blockdiag}(e^{-\bar{h}_1 s} I_{l_1}, \ldots, e^{-\bar{h}_N s} I_{l_N})$ represents the nominal time-delays, which are taken outside the plant in order to apply the approach of [16], $W_1(s) = [\overline{W}_1(s) \cdots \overline{W}_n(s)]$, where $\overline{W}_i(s) = [(\varphi_{i,1}/s)\ \varphi_{i,2}]$, and

$$\Delta = \text{blockdiag}\left(\begin{bmatrix}\Delta_{1,1}\\ \Delta_{1,2}\end{bmatrix}, \ldots, \begin{bmatrix}\Delta_{n,1}\\ \Delta_{n,2}\end{bmatrix}\right)$$

represents the uncertainties in the system. Note that, since the $\mathscr{L}^2$-induced norms of the LTV systems $\Delta_{i,1}$ and $\Delta_{i,2}$ $(i=1,\ldots,n)$ are made less than $1/\sqrt{2}$, Δ is a LTV system whose $\mathscr{L}^2$-induced norm is less than 1. Furthermore, since $\Delta_{i,1}$ and $\Delta_{i,2}$ $(i=1,\ldots,n)$ may be non-causal, Δ is a non-causal system in general. However, using the result of [25], we can still apply the small-gain theorem as long as $\tau_i(t) := h_i + \delta_i(t) \geqslant 0$, $\forall t \geqslant 0$, $\forall i$, which is naturally satisfied since round-trip time-delays cannot actually be time-advances. By Theorem 2 of [25] (which is an extension of the well-known small-gain theorem [26]), if we choose K to stabilize the system with $\Delta = 0$ and make the $\mathscr{L}^2$-induced norm of the system from w_1 to z_1 in Figure 3 less than 1, then the overall system is robustly stable for all uncertainties satisfying (10). Alternatively, if K stabilizes the system with $\Delta = 0$ and make the $\mathscr{L}^2$-induced norm of the system from w_1 to z_1 in Figure 3 less than some $\gamma > 0$, then the overall system is robustly stable for all Δ with $\mathscr{L}^2$-induced norm less than $1/\gamma$. The uncertainty block Δ would have $\mathscr{L}^2$-induced norm less than $1/\gamma$ if, for example, $|\delta_i(t)| < \delta_i^+/\gamma$ and $|\dot{\delta}_i(t)| < \tilde{\beta}_i$ and

DOI: 10.1002/rnc

$|\dot{\delta}_i^f(t)| < \tilde{\beta}_i^f$, $i=1,\ldots,n$, where $0<\tilde{\beta}_i^f \leqslant \tilde{\beta}_i < 1$ are such that $(\tilde{\beta}_i+\tilde{\beta}_i^f)/(\sqrt{1-\tilde{\beta}_i}) = (\beta_i+\beta_i^f)/\gamma\sqrt{1-\beta_i}$.

Remark 3

As pointed out by one of the anonymous reviewers, the use of non-causal uncertainty blocks can be avoided by first designing a controller for the case when Δ is replaced by $\Delta_1 := \Delta\Lambda_u$, which is causal and has the same $\mathscr{L}^2$-induced norm as Δ, and then showing that the same controller also stabilizes the original system and achieves the same norm, by using some manipulations and a result from [15]. However, given the result of [25], it is more natural and more straightforward to directly use Δ, which is non-causal, as the uncertainty block.

3. OPTIMAL $\mathscr{H}^\infty$ CONTROLLER DESIGN

To solve the control problem defined in the previous section, we consider a mixed sensitivity minimization problem for the system shown in Figure 4 [20]. Here, $W_2(s)=\frac{1}{s}$, $W_3(s)=\frac{\sigma_1}{s}$, and

$$W_4(s)=\frac{\sigma_2}{s}\begin{bmatrix} \frac{\alpha_2}{\alpha_1} & -1 & 0 & & 0 \\ \frac{\alpha_3}{\alpha_1} & 0 & -1 & & 0 \\ \vdots & \vdots & & \ddots & \vdots \\ \frac{\alpha_n}{\alpha_1} & 0 & 0 & & -1 \end{bmatrix},$$

where $\sigma_1>0$ and $\sigma_2>0$ are design parameters. Furthermore, $d := \dot{q}_d - c$, e_1 is the integral of the error, $y := q_d - q$, and is introduced to achieve tracking (11), and e_2 is introduced to achieve the weighted fairness requirement (12).

Here, the weighting matrix W_1, which was introduced in the previous section, is used to normalize the uncertainty block. Weights W_2 and W_3 are introduced to reject disturbances (in the variations of q_d and c) and achieve the tracking requirement (11). The weighting matrix W_4 is introduced to achieve the weighted fairness requirement (12). Design parameters σ_1 and σ_2, which appear respectively in W_3 and W_4, can be used to assign relative importance to tracking and weighted fairness respectively (see Section 4, Cases 4 and 5).

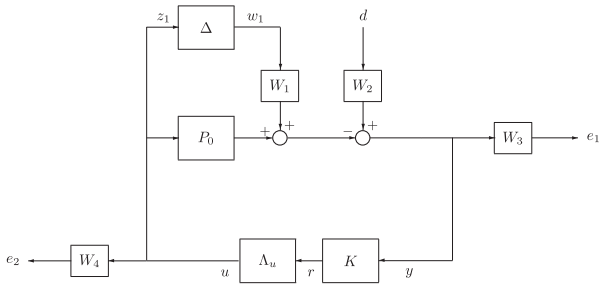

Figure 4. System for the mixed sensitivity minimization problem [20].

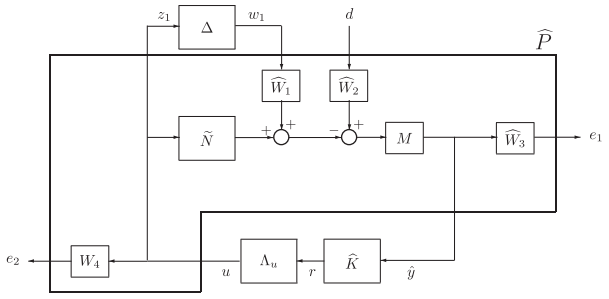

Figure 5. Equivalent system for the mixed sensitivity minimization problem.

Note that the nominal plant, P_o, has a pole at the origin. Furthermore, the integral terms in the weights W_2, W_3, and W_4 forces K to have integral action [26]. Therefore, the sensitivity function of the closed-loop system of Figure 4 has a double zero at the origin, which causes uncontrollable pole-zero cancelations to occur between the weights and the sensitivity. To avoid this problem, we let $P_o(s)=\widetilde{M}^{-1}(s)\widetilde{N}(s)$, where $\widetilde{N}(s)=(1/(s+\varepsilon))\mathbf{1}_n$ and $\widetilde{M}(s)=s/(s+\varepsilon)$, where $\varepsilon>0$ is arbitrary. By using this factorization and making some simple block diagram manipulations, the system in Figure 4 is transformed to the system in Figure 5, where $M(s)=(s+\varepsilon)^2/s^2$, $\widehat{W}_1(s)=\widetilde{M}(s)W_1(s)$, $\widehat{W}_2(s)=1/(s+\varepsilon)$, $\widehat{W}_3(s)=\sigma_1/(s+\varepsilon)$, and

$$\widehat{K}(s)=\frac{s}{s+\varepsilon}K(s). \tag{13}$$

DOI: 10.1002/rnc

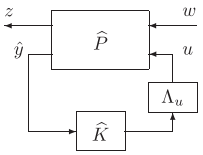

Figure 6. General four-block problem.

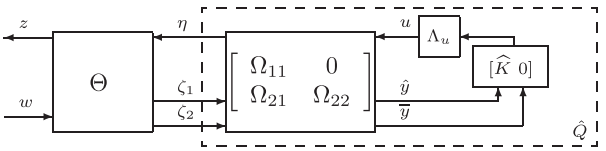

Figure 7. New problem definition under chain-scattering representation.

Therefore, the problem is now transformed into the general four block problem of Figure 6, where the general plant is described as

$$\begin{bmatrix} z \\ \cdots \\ \hat{y} \end{bmatrix} := \begin{bmatrix} z_1 \\ e_1 \\ e_2 \\ \cdots \\ \hat{y} \end{bmatrix} = \begin{bmatrix} 0 & 0 & \vdots & I \\ -\widehat{W}_3 M \widehat{W}_1 & \widehat{W}_3 M \widehat{W}_2 & \vdots & -\widehat{W}_3 M \widetilde{N} \\ 0 & 0 & \vdots & W_4 \\ \cdots\cdots\cdots & \cdots\cdots\cdots & & \cdots\cdots\cdots \\ -M \widehat{W}_1 & M \widehat{W}_2 & \vdots & -M \widetilde{N} \end{bmatrix} \begin{bmatrix} w_1 \\ d \\ \cdots \\ u \end{bmatrix} =: \widehat{P} \begin{bmatrix} w \\ \cdots \\ u \end{bmatrix} \tag{14}$$

and the problem is to design a controller $\widehat{K}$ so that $\|F_l(\widehat{P}, \Lambda_u \widehat{K})\|_\infty < \gamma$, for minimum possible γ, where $F_l(\widehat{P}, \Lambda_u \widehat{K})$ is the closed-loop TFM from w to z. Let us define the *normalized plant*

$$\widehat{P}_\gamma := \begin{bmatrix} \gamma^{-1} I & 0 \\ 0 & I \end{bmatrix} \widehat{P} =: \begin{bmatrix} P_{\gamma_{11}} & P_{\gamma_{12}} \\ P_{\gamma_{21}} & P_{\gamma_{22}} \end{bmatrix},$$

so that $\widehat{K}$ must satisfy $\|F_l(\widehat{P}_\gamma, \Lambda_u \widehat{K})\|_\infty < 1$.

As it was done in [16], we will use chain scattering representations to reduce the above defined 4-block problem to a 1-block problem. It can be shown that $P_{\gamma_{12}}(j\omega)$ has full column rank and $P_{\gamma_{21}}(j\omega)$ has full row rank for all $\omega \in \mathbb{R} \cup \{\infty\}$, which guarantees existence of a solution in the delay-free case (i.e. when $\Lambda_u = I$) for sufficiently large γ [26]. The latter condition also allows us to introduce an output augmentation by defining $\overline{y} := \overline{P}_{\gamma_{21}} w + \overline{P}_{\gamma_{22}} u$, where $\begin{bmatrix} P_{\gamma 21} \\ \overline{P}_{\gamma 21} \end{bmatrix}$ is invertible. Then, the augmented plant

$$\overline{\widehat{P}}_\gamma := \begin{bmatrix} P_{\gamma_{11}} & \vdots & P_{\gamma_{12}} \\ \cdots\cdots & & \cdots\cdots \\ P_{\gamma_{21}} & \vdots & P_{\gamma_{22}} \\ \overline{P}_{\gamma_{21}} & \vdots & \overline{P}_{\gamma_{22}} \end{bmatrix}$$

has a chain-scattering representation $\Psi := \mathrm{CHAIN}(\overline{\widehat{P}}_\gamma)$, which in turn has a (J_{zw}, J_{uw})-lossless factorization

$$\Psi =: \Theta \Omega, \tag{15}$$

as shown in Figure 7, where Θ is (J_{zw}, J_{uw})-lossless and Ω is unimodular [27]. Furthermore, Ω is decomposed as

$$\Omega = \begin{bmatrix} \Omega_{11} & 0 \\ \Omega_{21} & \Omega_{22} \end{bmatrix},$$

where Ω_{11} is $(n_u + n_{\hat{y}}) \times (n_u + n_{\hat{y}})$ dimensional and bistable.

DOI: 10.1002/rnc

From Figure 7, the closed-loop TFM from w to z is $HM(\Theta, \widehat{Q})$, where $\widehat{Q} := HM(\Omega, \Lambda_u[\widehat{K}\ \ 0])$. Since Θ is (J_{zw}, J_{uw})-lossless, $HM(\Theta, \widehat{Q})$ is contractive if and only if $\widehat{Q}$ is contractive [27]. Therefore, the problem of finding a $\widehat{K}$ such that $F_l(\widehat{P}_\gamma, \Lambda_u\widehat{K}) = HM(\Theta, \widehat{Q})$ is contractive is equivalent to finding a $\widehat{K}$ such that $\widehat{Q}$ is contractive. Furthermore, we can write $\widehat{Q} = [Q\ \ 0]$, where $Q := HM(\Omega_{11}, \Lambda_u\widehat{K})$. Therefore, the problem of finding a controller $\widehat{K}$ for the system in Figure 6 is reduced to finding $\widehat{K}$ such that $Q = HM(\Omega_{11}, \Lambda_u\widehat{K})$ is contractive. Since $HM(\Omega_{11}, \Lambda_u\widehat{K}) = HM(\Omega_{11}\Lambda, \widehat{K})$, where $\Lambda = \text{blockdiag}(\Lambda_u, 1)$, the problem is reduced to finding $\widehat{K}$ such that the $\mathscr{H}^\infty$ norm of $Q = HM(\Omega_{11}\Lambda, \widehat{K})$ is less than 1, which is a one block problem (OBP) [16]. Following [16], to obtain a causal controller, we can write $Q = HM(\Omega_{11}\Lambda, \widehat{K}) = HM(\Omega_{11}\Omega_{11_\infty}^{-1}\Lambda, K_\gamma)$, where $\Omega_{11_\infty} := \lim_{s\to\infty}\Omega_{11}(s)$ and $K_\gamma := HM(\Lambda^{-1}\Omega_{11_\infty}\Lambda, \widehat{K})$. In our case, we can choose Ω in (15) such that

$$\Omega_{11_\infty} := \lim_{s\to\infty}\Omega_{11}(s) = \begin{bmatrix} I_n & 0 \\ 0 & \dfrac{\gamma}{\sqrt{D_{21}D_{21}^{\mathrm{T}}}} \end{bmatrix}, \qquad (16)$$

where $D_{21} := \lim_{s\to\infty} P_{\gamma_{21}}(s)$. Then, $\Lambda^{-1}\Omega_{11_\infty}\Lambda = \Omega_{11_\infty}$, and thus we would have

$$K_\gamma := HM(\Lambda^{-1}\Omega_{11_\infty}\Lambda, \widehat{K}) = HM(\Omega_{11_\infty}, \widehat{K}). \qquad (17)$$

Defining $G := \Omega_{11}\Omega_{11_\infty}^{-1}$, the problem can be written as:

OBP(G, Λ): Find a controller K_γ satisfying $\|HM(G\Lambda, K_\gamma)\|_\infty < 1$.

The solution to this problem is found by a sequence of iterations [16]. In each iteration, a problem which is called an *adobe delay problem* is solved. The solution to a generic adobe delay problem will be explained in Section 3.1. Then, the general solution to OBP(G, Λ) will be presented in Section 3.2.

3.1. Solution to a generic adobe delay problem

An adobe delay problem is described as OBP(G_a, Λ_a) where Λ_a, called *adobe delay*, has a special form as:

$$\Lambda_a := \begin{bmatrix} \mathrm{e}^{-h_a s} I_{\mu_a} & 0 \\ 0 & I_{\rho_a} \end{bmatrix},$$

where $\mu_a < n_u + n_{\hat{y}}$ and $\mu_a + \rho_a = n_u + n_{\hat{y}}$ [16]. In this subsection, the solution to this adobe delay problem for a generic bistable

$$G_a = \left[\begin{array}{c|cc} A_a & B_{\mu_a} & B_{\rho_a} \\ \hline C_{\mu_a} & I_{\mu_a} & 0 \\ C_{\rho_a} & 0 & I_{\rho_a} \end{array}\right],$$

where the partitioning is compatible with that of Λ_a, is presented. OBP(G_a, Λ_a) is finding a controller K_a such that $Q_a = HM(G_a\Lambda_a, K_a)$ is contractive. For the delay-free case, i.e. when $\Lambda_a = I$, using (6), K_a is obtained as $K_a = HM(G_a^{-1}, Q_a)$ for any contractive Q_a. However, for the present case, this mapping is not causal in general. Therefore, to find the solution, we proceed as in [16]. We define

$$J_{\mu_a} := [I_{\mu_a}\ \ 0]J_{u\hat{y}}\begin{bmatrix} I_{\mu_a} \\ 0 \end{bmatrix},$$

$$J_{\rho_a} := [0\ \ I_{\rho_a}]J_{u\hat{y}}\begin{bmatrix} 0 \\ I_{\rho_a} \end{bmatrix},$$

$$H_a := \begin{bmatrix} A_a - B_{\rho_a}C_{\rho_a} & -B_{\rho_a}J_{\rho_a}B_{\rho_a}^{\mathrm{T}} \\ -C_{\mu_a}^{\mathrm{T}}J_{\mu_a}C_{\mu_a} & -A_a^{\mathrm{T}} + C_{\rho_a}^{\mathrm{T}}B_{\rho_a}^{\mathrm{T}} \end{bmatrix},$$

$$\Sigma(t) = \begin{bmatrix} \Sigma_{11}(t) & \Sigma_{12}(t) \\ \Sigma_{21}(t) & \Sigma_{22}(t) \end{bmatrix} := \mathrm{e}^{H_a t},$$

and

$$\Sigma_a = \begin{bmatrix} \Sigma_{a_{11}} & \Sigma_{a_{12}} \\ \Sigma_{a_{21}} & \Sigma_{a_{22}} \end{bmatrix} := \Sigma(h_a).$$

Then, the solution to OBP(G_a, Λ_a) exists if and only if $\Sigma_{22}(t)$ is nonsingular for all $t \in [0, h_a]$, and is given by [16]:

$$K_a = HM\left(\begin{bmatrix} I & 0 \\ \Pi_a & I \end{bmatrix}\widetilde{G}_a^{-1}, \widetilde{Q}_a\right), \qquad (18)$$

DOI: 10.1002/rnc

where

$$\widetilde{G}_a := \left[\begin{array}{c|cc} A_a & \Sigma_{a_{22}}^{\mathrm{T}} B_{\mu_a} + \Sigma_{a_{12}}^{\mathrm{T}} C_{\mu_a}^{\mathrm{T}} J_{\mu_a} & B_{\rho_a} \\ \hline C_{\mu_a}\Sigma_{a_{22}}^{-T} & & \\ C_{\rho_a} - J_{\rho_a} B_{\rho_a}^{\mathrm{T}} \Sigma_{a_{22}}^{-1} \Sigma_{a_{21}} & I_{\mu_a+\rho_a} & \end{array}\right],$$

is finite-dimensional and bistable,

$$\Pi_a(s) := \pi_{h_a}\left(\mathrm{e}^{-h_a s}\left[\begin{array}{c|c} H_a & \begin{matrix} B_{\mu_a} \\ -C_{\mu_a}^{\mathrm{T}} J_{\mu_a} \end{matrix} \\ \hline \begin{matrix} C_{\rho_a} & J_{\rho_a} B_{\rho_a}^{\mathrm{T}} \end{matrix} & 0 \end{array}\right]\right)$$

is a finite impulse response (FIR) filter of duration h_a, and $\widetilde{Q}_a$ is contractive, but otherwise arbitrary.

3.2. Solution to the general problem

The general problem, OBP(G, Λ), is solved in N steps, if $\bar{h}_N > 0$, and in $N-1$ steps, if $\bar{h}_N = 0$.

Step 1: Assuming $\bar{h}_N > 0$ (if $\bar{h}_N = 0$, we directly start with step 2, using $\widetilde{\Lambda}_1 := \Lambda$ and $\widetilde{G}_1 := G$), let $\Lambda =: \Lambda_1 \widetilde{\Lambda}_1$, where

$$\Lambda_1(s) := \begin{bmatrix} \mathrm{e}^{-\bar{h}_N s} I_{\mu_1} & 0 \\ 0 & I_{\rho_1} \end{bmatrix},$$

where $\mu_1 = \sum_{i=1}^{N} l_i = n$ and $\rho_1 = n+1-\mu_1 = 1$. Then, using (5), $HM(G\Lambda, K_\gamma) = HM(G\Lambda_1, HM(\widetilde{\Lambda}_1, K_\gamma))$. Letting

$$K_1 := HM(\widetilde{\Lambda}_1, K_\gamma), \tag{19}$$

the problem becomes determining K_1 so that $\|HM(G\Lambda_1, K_1)\|_\infty < 1$, which is the problem discussed in Section 3.1. Therefore, its solution is

$$K_1 = HM\left(\begin{bmatrix} I & 0 \\ \Pi_1 & I \end{bmatrix} \widetilde{G}_1^{-1}, \widetilde{Q}_1\right), \tag{20}$$

where Π_1 and $\widetilde{G}_1$ are respectively determined as Π_a and $\widetilde{G}_a$ in Section 3.1, and $\widetilde{Q}_1$ must be contractive. Note that, by (6),

$$\widetilde{Q}_1 = HM\left(\widetilde{G}_1 \begin{bmatrix} I & 0 \\ -\Pi_1 & I \end{bmatrix}, K_1\right),$$

where K_1 is given by (19). Hence, using (5), we can write $\widetilde{Q}_1 = HM(\widetilde{G}_1 \widetilde{\Lambda}_1, \widetilde{K}_1)$, where

$$\widetilde{K}_1 := HM\left(\widetilde{\Lambda}_1^{-1} \begin{bmatrix} I & 0 \\ -\Pi_1 & I \end{bmatrix} \widetilde{\Lambda}_1, K_\gamma\right). \tag{21}$$

Therefore, the remaining problem is to determine $\widetilde{K}_1$, so that $\|HM(\widetilde{G}_1 \widetilde{\Lambda}_1, \widetilde{K}_1)\|_\infty < 1$, which is considered in the next step.

Step 2: Let $\widetilde{\Lambda}_1 =: \Lambda_2 \widetilde{\Lambda}_2$, where

$$\Lambda_2(s) := \begin{bmatrix} \mathrm{e}^{-(\bar{h}_{N-1} - \bar{h}_N)s} I_{\mu_2} & 0 \\ 0 & I_{\rho_2} \end{bmatrix},$$

where $\mu_2 = \sum_{i=1}^{N-1} l_i = n - l_N$ and $\rho_2 = n+1-\mu_2 = 1 + l_N$. Then, using (5), $HM(\widetilde{G}_1 \widetilde{\Lambda}_1, \widetilde{K}_1) = HM(\widetilde{G}_1 \Lambda_2, HM(\widetilde{\Lambda}_2, \widetilde{K}_1))$. Letting

$$K_2 := HM(\widetilde{\Lambda}_2, \widetilde{K}_1), \tag{22}$$

the problem becomes determining K_2 so that $\|HM(\widetilde{G}_1 \Lambda_2, K_2)\|_\infty < 1$, which is the problem discussed in Section 3.1. Therefore, its solution is

$$K_2 = HM\left(\begin{bmatrix} I & 0 \\ \Pi_2 & I \end{bmatrix} \widetilde{G}_2^{-1}, \widetilde{Q}_2\right), \tag{23}$$

where Π_2 and $\widetilde{G}_2$ are respectively determined as Π_a and $\widetilde{G}_a$ in Section 3.1, and $\widetilde{Q}_2$ must be contractive. Note that, by (6),

$$\widetilde{Q}_2 = HM\left(\widetilde{G}_2 \begin{bmatrix} I & 0 \\ -\Pi_2 & I \end{bmatrix}, K_2\right),$$

where K_2 is given by (22). Hence, using (5), we can write $\widetilde{Q}_2 = HM(\widetilde{G}_2 \widetilde{\Lambda}_2, \widetilde{K}_2)$, where

$$\widetilde{K}_2 := HM\left(\widetilde{\Lambda}_2^{-1} \begin{bmatrix} I & 0 \\ -\Pi_2 & I \end{bmatrix} \widetilde{\Lambda}_2, \widetilde{K}_1\right). \tag{24}$$

DOI: 10.1002/rnc

Therefore, the remaining problem is to determine $\widetilde{K}_2$, so that $\|HM(\widetilde{G}_2\widetilde{\Lambda}_2, \widetilde{K}_2)\|_\infty<1$, which is considered in the next step.

$$\vdots$$

Step N: Let $\widetilde{\Lambda}_{N-1}=:\Lambda_N\widetilde{\Lambda}_N$, where

$$\Lambda_N(s):=\begin{bmatrix} e^{-(\bar{h}_1-\bar{h}_2)s}I_{\mu_N} & 0 \\ 0 & I_{\rho_N}\end{bmatrix},$$

where $\mu_N=\sum_{i=1}^{1} l_i=l_1$ and $\rho_N=n+1-\mu_N=1+\sum_{i=2}^{N} l_i$. Note that, $\widetilde{\Lambda}_N=I$. Then, using (5), $HM(\widetilde{G}_{N-1}\widetilde{\Lambda}_{N-1}, \widetilde{K}_{N-1})=HM(\widetilde{G}_{N-1}\Lambda_N, HM(\widetilde{\Lambda}_N, \widetilde{K}_{N-1}))$. Letting

$$K_N:=HM(\widetilde{\Lambda}_N, \widetilde{K}_{N-1}), \tag{25}$$

the problem becomes determining K_N so that $\|HM(\widetilde{G}_{N-1}\Lambda_N, K_N)\|_\infty<1$, which is the problem discussed in Section 3.1. Therefore, its solution is

$$K_N=HM\left(\begin{bmatrix} I & 0 \\ \Pi_N & I\end{bmatrix}\widetilde{G}_N^{-1}, \widetilde{Q}_N\right), \tag{26}$$

where Π_N and $\widetilde{G}_N$ are respectively determined as Π_a and $\widetilde{G}_a$ in Section 3.1, and $\widetilde{Q}_N$ must be contractive, but otherwise arbitrary. Note that, since $\widetilde{\Lambda}_N=I$, (25) gives $K_N=\widetilde{K}_{N-1}$.

Now, using (6), from (21) we obtain

$$K_\gamma=HM\left(\widetilde{\Lambda}_1^{-1}\begin{bmatrix} I & 0 \\ \Pi_1 & I\end{bmatrix}\widetilde{\Lambda}_1, \widetilde{K}_1\right). \tag{27}$$

Similarly, from (24) we obtain

$$\widetilde{K}_1=HM\left(\widetilde{\Lambda}_2^{-1}\begin{bmatrix} I & 0 \\ \Pi_2 & I\end{bmatrix}\widetilde{\Lambda}_2, \widetilde{K}_2\right). \tag{28}$$

Substituting (28) into (27) and using (5) we obtain

$$K_\gamma=HM\left(\widetilde{\Lambda}_1^{-1}\begin{bmatrix} I & 0 \\ \Pi_1 & I\end{bmatrix}\widetilde{\Lambda}_1\widetilde{\Lambda}_2^{-1}\begin{bmatrix} I & 0 \\ \Pi_2 & I\end{bmatrix}\widetilde{\Lambda}_2, \widetilde{K}_2\right). \tag{29}$$

Proceeding like this, through the first $N-1$ steps and using the fact that $\widetilde{K}_{N-1}=K_N$, which is given by (26), we obtain

$$K_\gamma=HM\left(\widetilde{\Lambda}_1^{-1}\begin{bmatrix} I & 0 \\ \Pi_1 & I\end{bmatrix}\widetilde{\Lambda}_1\cdots\widetilde{\Lambda}_{N-1}^{-1} \times\begin{bmatrix} I & 0 \\ \Pi_{N-1} & I\end{bmatrix}\widetilde{\Lambda}_{N-1}\begin{bmatrix} I & 0 \\ \Pi_N & I\end{bmatrix}\widetilde{G}_N^{-1}, \widetilde{Q}_N\right). \tag{30}$$

On noting that $\widetilde{\Lambda}_1^{-1}=\Lambda^{-1}\Lambda_1$, $\widetilde{\Lambda}_1\widetilde{\Lambda}_2^{-1}=\Lambda_2, \ldots, \widetilde{\Lambda}_{N-2}\widetilde{\Lambda}_{N-1}^{-1}=\Lambda_{N-1}$, and $\widetilde{\Lambda}_{N-1}=\Lambda_N$, we can rewrite (30) as

$$K_\gamma=HM(\Pi_\Lambda G_\Lambda^{-1}, Q_\Lambda), \tag{31}$$

where

$$\Pi_\Lambda:=\Lambda^{-1}\prod_{i=1}^{N}\Lambda_i\begin{bmatrix} I & 0 \\ \Pi_i & I\end{bmatrix}$$

is a system which involves delays and FIR filters (note that time-advances introduced by Λ^{-1} are all cancelled by Λ_i's; i.e. Π_Λ is causal), $G_\Lambda:=\widetilde{G}_N$ is a finite-dimensional and bistable system, and $Q_\Lambda:=\widetilde{Q}_N$ is such that $\|Q_\Lambda\|_\infty<1$, but otherwise arbitrary.

Once K_γ is found as in (31), using (5) and (6), $\widehat{K}$ is found by inverting (17) and the desired controller K is found from (13) as

$$K(s)=\frac{s+\varepsilon}{s}HM(\Omega_{11_\infty}^{-1}\Pi_\Lambda(s)G_\Lambda^{-1}(s), Q_\Lambda(s)). \tag{32}$$

By decomposing Π_k's as

$$\Pi_1=:[\Pi_{11}^1\ \ \Pi_{12}^1\ \ \cdots\ \ \Pi_{1N}^1],$$

where Π_{1j}^1 is $1\times l_j$ dimensional,

$$\Pi_2=:\begin{bmatrix}\Pi_{11}^2 & \Pi_{12}^2 & \cdots & \Pi_{1(N-1)}^2 \\ \Pi_{21}^2 & \Pi_{22}^2 & \cdots & \Pi_{2(N-1)}^2\end{bmatrix},$$

DOI: 10.1002/rnc

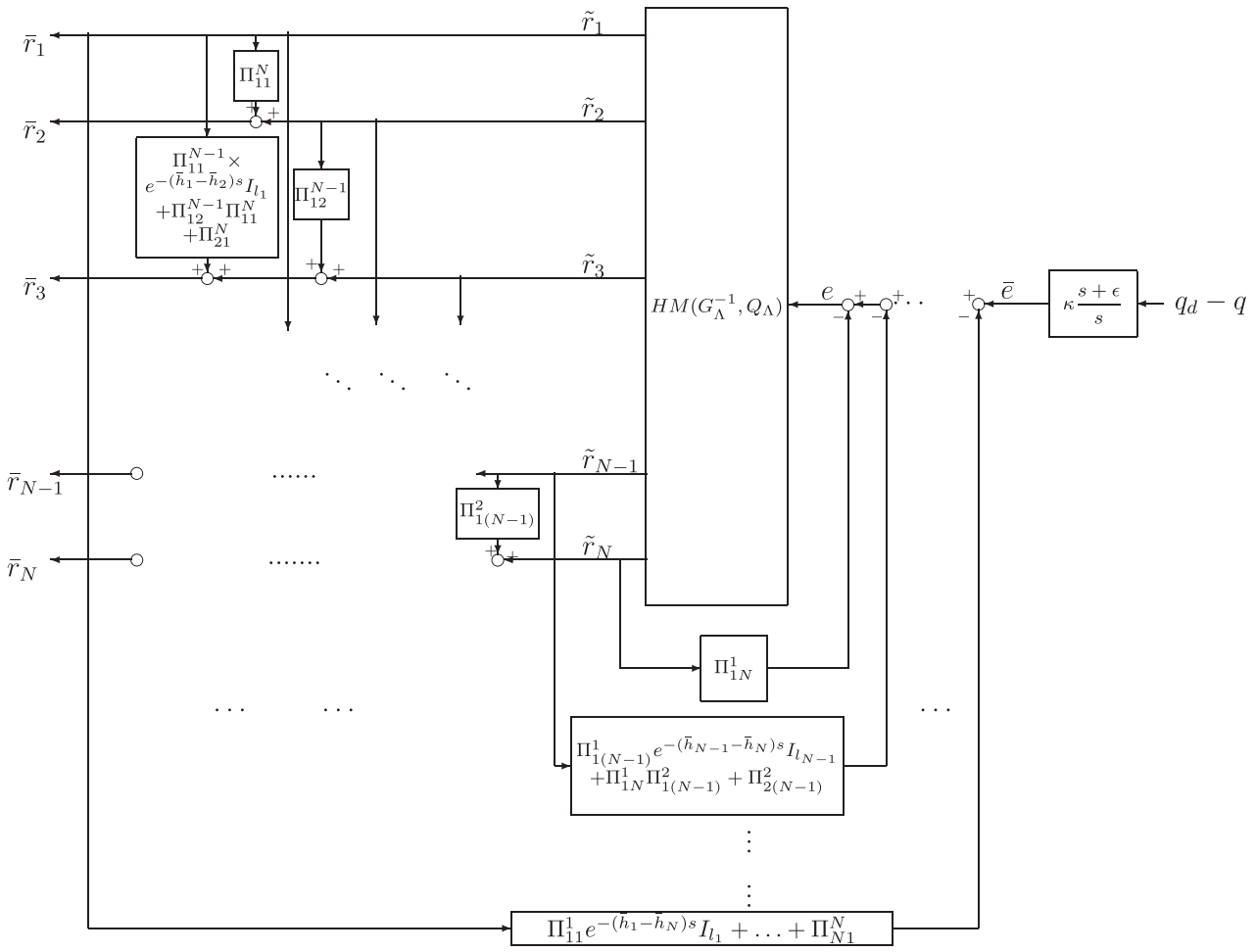

Figure 8. The implementation of the controller K.

where Π_{1j}^2 is $l_N\times l_j$ and Π_{2j}^2 is $1\times l_j$ dimensional, ..., and

$$\Pi_N =: \begin{bmatrix} \Pi_{11}^N \\ \Pi_{21}^N \\ \vdots \\ \Pi_{N1}^N \end{bmatrix},$$

where Π_{j1}^N is $l_{j+1}\times l_1$ $(j=1,\ldots,N-1)$ and Π_{N1}^N is $1\times l_1$ dimensional, the controller K can be implemented as shown in Figure 8. Here, $\kappa(s+\varepsilon)/(s)$ is a proportional-integral term, where

$$\kappa := \frac{\gamma}{\sqrt{D_{21}D_{21}^{\mathrm{T}}}} = \frac{\gamma}{2\sqrt{2\sum_{i=1}^n(\delta_i^+)^2}},$$

$HM(G_\Lambda^{-1},Q_\Lambda)$ is a finite-dimensional system (assuming Q_Λ is finite-dimensional) parameterized by Q_Λ, which must be contractive, and each Π_{ij}^k is an FIR filter.

DOI: 10.1002/rnc

Furthermore,

$$\bar{r}_1 := \begin{bmatrix} r_1 \\ \vdots \\ r_{l_1} \end{bmatrix} \quad \bar{r}_2 := \begin{bmatrix} r_{l_1+1} \\ \vdots \\ r_{l_1+l_2} \end{bmatrix}, \ldots, \bar{r}_N := \begin{bmatrix} r_{\sum_{i=1}^{N-1} l_i+1} \\ \vdots \\ r_n \end{bmatrix}.$$

In the above, we assumed that $\gamma>0$ is such that there exists a solution to the adobe delay problem at each step. To find minimum such γ and the corresponding controller, i.e. to determine the *optimal* controller $K_{\text{opt}}(s)=((s+\varepsilon)/s)\widehat{K}_{\text{opt}}(s)$, where $\widehat{K}_{\text{opt}}$ solves

$$\inf_{\widehat{K}} \| F_l(\widehat{P}, \Lambda_u \widehat{K}) \|_\infty =: \gamma^{\text{opt}}, \tag{33}$$

we first find the minimum γ, call it γ^0, for which there exists a (J_{zw}, J_{uw})-lossless factorization (15). If step 1 also has a solution for this γ, we let $\gamma^1=\gamma^0$. Otherwise, we increase γ and determine the minimum γ, call it γ^1, for which there exists a solution to the adobe delay problem of step 1. After solving step k ($k=1,\ldots,N-1$), and thus determining γ^k, if step $k+1$ also has a solution for this γ, we let $\gamma^{k+1}=\gamma^k$. Otherwise, we increase γ and determine the minimum γ, call it γ^{k+1}, for which there exists a solution to the adobe delay problem of step $k+1$ (of course, we resolve all the previous steps for this new γ). In this way, γ^{opt} in (33) is determined as γ^N at the end of step N. The controller given by (32) for $\gamma=\gamma^{\text{opt}}$ is the optimal controller.

Examining Figure 8, the controller to be implemented involves a proportional-integral term (the right-most block in Figure 8), which can simply be realized as

$$\dot{x}(t) = \kappa\varepsilon(q_d(t)-q(t))$$
$$\bar{e}(t) = x(t)+\kappa(q_d(t)-q(t))$$

where x is the scalar state variable. This block is followed by a LTI block with TFM $HM(G_\Lambda^{-1}, Q_\Lambda)$ put in a feedback loop with N FIR filters. FIR filters are also connected from the kth output of this block to $(k+1)$th, ..., Nth output ($k=1,\ldots,N-1$). The state-space dimension of the LTI block with TFM $HM(G_\Lambda^{-1}, Q_\Lambda)$ is equal to the state-space dimension of G_Λ^{-1} plus the state-space dimension of Q_Λ. By tracking back the design steps given above, it is seen that the state-space dimension of G_Λ^{-1} is the same as the state-space dimension of $G:=\Omega_{11}\Omega_{11_\infty}^{-1}$, which is same as the state-space dimension of Ω_{11}, since $\Omega_{11_\infty}^{-1}$ is a constant matrix. The state-space dimension of Ω_{11}, finally, is equal to the state-space dimension of the general plant in (14), which is $n+1$ (the second and fourth block rows can be realized commonly as a second order system, additional $n-1$ states are needed to realize the third block row). Therefore, if Q_Λ is chosen as constant (it can simply be chosen as zero), the state-space dimension of the LTI block with TFM $HM(G_\Lambda^{-1}, Q_\Lambda)$ is simply equal to $n+1$. With $Q_\Lambda=0$, a state-space realization of $HM(G_\Lambda^{-1}, Q_\Lambda)$ can be written as

$$\dot{x}(t) = (A_a - \Sigma_{a_{22}}^{\mathrm{T}} B_{\mu_a} C_{\mu_a} \Sigma_{a_{22}}^{-\mathrm{T}} - \Sigma_{a_{12}}^{\mathrm{T}} C_{\mu_a}^{\mathrm{T}} J_{\mu_a} C_{\mu_a} \Sigma_{a_{22}}^{-\mathrm{T}}) x(t) + B_{\rho_a} e(t)$$
$$\widetilde{r}(t) = -C_{\mu_a} \Sigma_{a_{22}}^{-\mathrm{T}} x(t),$$

where $\widetilde{r} := [\widetilde{r}_1^{\mathrm{T}} \ \widetilde{r}_2^{\mathrm{T}} \ \cdots \ \widetilde{r}_{N-1}^{\mathrm{T}} \ \widetilde{r}_N^{\mathrm{T}}]^{\mathrm{T}}$, $x(t)$ is the $n+1$ dimensional state vector, and the appearing matrices are as defined in Section 3.1, corresponding to Step N. Furthermore, each FIR filter, whose impulse response is in the form of (1), can easily be realized in discrete-time using $\frac{h}{\tau}$ delay elements, where h is the length of the impulse response and τ is the sampling period. Therefore, the implementation of the overall controller is relatively simple.

4. SIMULATION STUDIES

In this section, we consider a number of example cases to illustrate the time-domain performance of the proposed controller. We also compare the present results with the results obtained by using the controller design approach of [19]. Simulations are done using MATLAB Simulink, where non-linear effects (hard constraints) are also taken into account. Furthermore, rather than using the fluid-flow network model used for controller design, we use a discrete model for all the simulations. We assume that data flow consists of discrete packets of size 1 Mbits each. All the

DOI: 10.1002/rnc

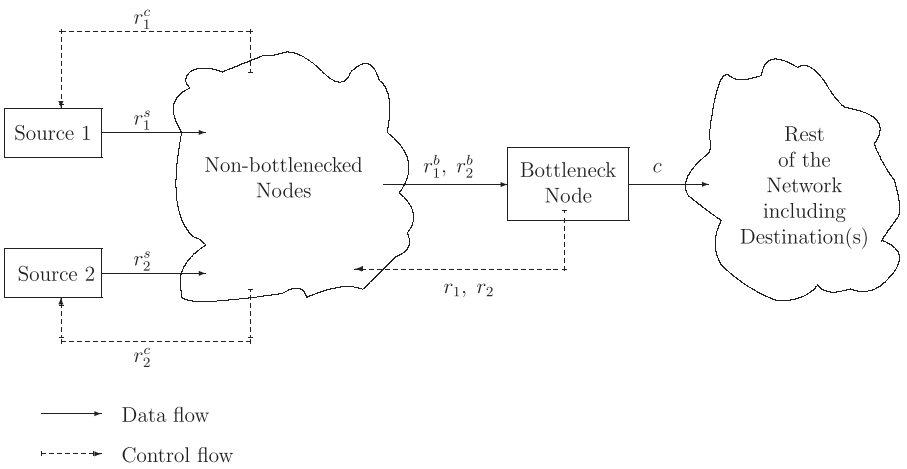

Figure 9. Topology of the example network.

Table I. Design parameters.

Case	h_1	h_2	δ_1^+	δ_2^+	β_1	β_2	β_1^f	β_2^f	α_1	α_2	σ_1	σ_2	γ^{opt}
1,2,6,7	3	1	0.5	1	0.6	0.5	0.3	0.2	$\frac{2}{3}$	$\frac{1}{3}$	0.25	0.25	5.7098
3	3	1	2	1	0.6	0.5	0.3	0.2	$\frac{2}{3}$	$\frac{1}{3}$	0.25	0.25	7.6280
4	3	1	0.5	1	0.6	0.5	0.3	0.2	$\frac{2}{3}$	$\frac{1}{3}$	1	0.25	10.7857
5	3	1	0.5	1	0.6	0.5	0.3	0.2	$\frac{2}{3}$	$\frac{1}{3}$	0.25	1	7.2040
8	1	1	0.5	1	0.6	0.5	0.3	0.2	$\frac{2}{3}$	$\frac{1}{3}$	0.25	0.25	4.1868

links are assumed to have a physical capacity of 100 Mbits/second. Therefore, each data packet is modeled as a pulse of width 10 milliseconds. Control packets, which carry rate information from the bottleneck node to the sources, on the other hand, have much smaller sizes. The output of the controller is assumed to be sampled at a rate of 0.2 kHz. That is, the controller at the bottleneck node sends a control packet to each source at every 5 ms. Each source updates its data sending rate as soon as a new control packet arrives (if the current rate is r packets/second, then a packet of size 1 Mbits is send every $\frac{1}{r}$ seconds). Note that, due to the presence of time-varying backward time-delays, control packets are not necessarily received, and hence, data sending rates are not necessarily updated at equal intervals. A constant simulation step size of 1 millisecond is used for all simulations.

We consider a network with two sources as shown in Figure 9. The nominal time-delays (in seconds), controller design parameters, and the resulting optimal sensitivity level, γ^{opt}, for each case are shown in Table I. In all cases, we take $Q_\Lambda = 0$ and $h_i^f = h_i^b = \frac{1}{2}h_i$, $i = 1, 2$. In all cases, the buffer size (maximum queue length) is taken as 60 packets and the desired queue length, q_d, is taken as half of this value, 30 packets. The rate limits of the sources are taken as 100 packets/second in all cases except Case 6. The capacity of the outgoing link is taken as 90 packets/second in all cases except Case 7. The uncertain part of the actual time-delays (in seconds) are shown in Table II. The results are shown in Figures 10–17, where q is the queue length, $q(t)$ (whose scale is shown on the right-hand-side of each graph), and r_i^s, for $i = 1, 2$, is the actual rate, $r_i^s(t) := \min(\max(r_i^c(t), 0), d_i)$, of data sent from source i at

DOI: 10.1002/rnc

Table II. The uncertain part of the actual time-delays.

Case	i	$\delta_i^b(t)$	$\delta_i^f(t)$
1, 4–8	1	$0.2+0.3\sin\left(\frac{2\pi}{30}t\right)$	$0.1+0.2\sin\left(\frac{2\pi}{70}t\right)$
	2	$0.4+0.3\sin\left(\frac{2\pi}{50}t\right)$	$0.1+0.1\sin\left(\frac{2\pi}{100}t\right)$
2, 3	1	$1.2+0.3\sin\left(\frac{2\pi}{30}t\right)$	$0.1+0.2\sin\left(\frac{2\pi}{70}t\right)$
	2	$0.4+0.3\sin\left(\frac{2\pi}{50}t\right)$	$0.1+0.1\sin\left(\frac{2\pi}{100}t\right)$

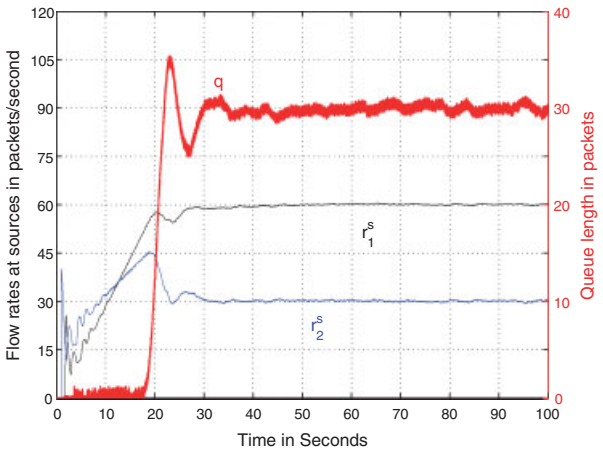

Figure 10. Simulation results for Case 1.

time t, where d_i is the rate limit of source i and $r_i^c(t) = r_i(t-\tau_i^b(t))$ is the rate command received at source i at time t.

Case 1: This is the central case, which we will compare all other simulation results. As shown in Figure 10, the queue length remains almost zero for a duration of about 18 s. This is the time required for the incoming rates to reach the capacity of the outgoing link. After a transient, which includes a small overshoot, an oscillatory steady-state is reached at about 40 s. The high-frequency oscillations in the queue length are due to discrete arrival/departure of packets (those oscillations would not be seen if a fluid-flow model was used). Besides those oscillations, the existence of time-varying forward time-delays also cause oscillations, especially at the steady-state. As shown in Figure 10, at steady-state, the queue length

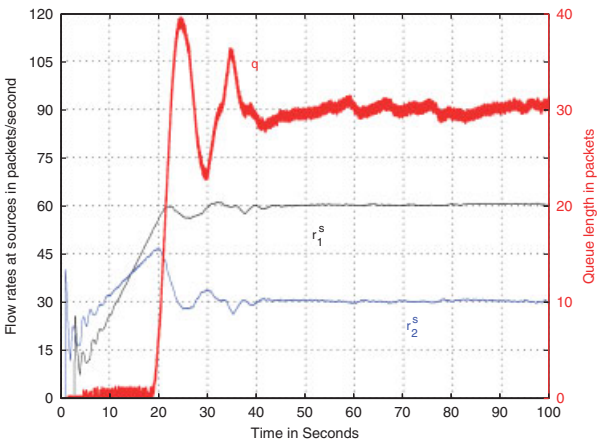

Figure 11. Simulation results for Case 2.

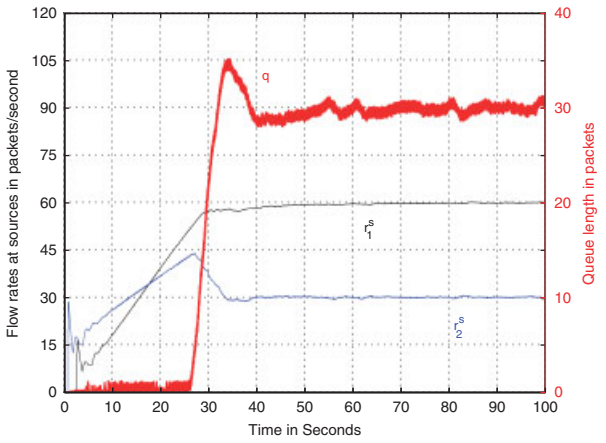

Figure 12. Simulation results for Case 3.

oscillates around its desired value, q_d, and the flow rates oscillate around the values given by (12). Also note that, the controller is more conservative on rate 1, than it is on rate 2. The reason for this is that the nominal delay of channel 1 is higher than that of channel 2.

Case 2: We have the same controller as in Case 1, but the actual delay in channel 1 is now increased. As shown in Figure 11, this results in a longer transient response and more overshoot.

Case 3: We increased the value of the design parameter δ_1^+ four times as shown in Table I. This makes the

DOI: 10.1002/rnc

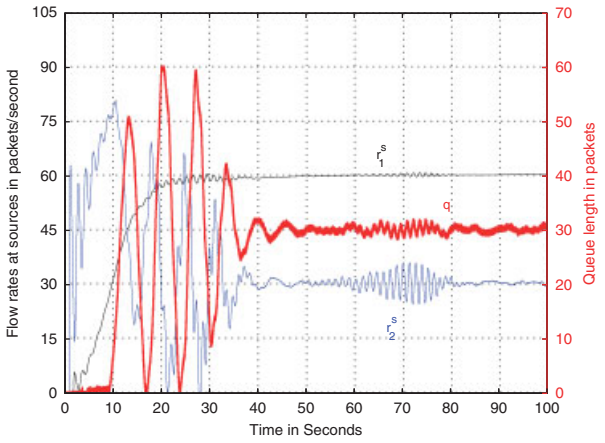

Figure 13. Simulation results for Case 4.

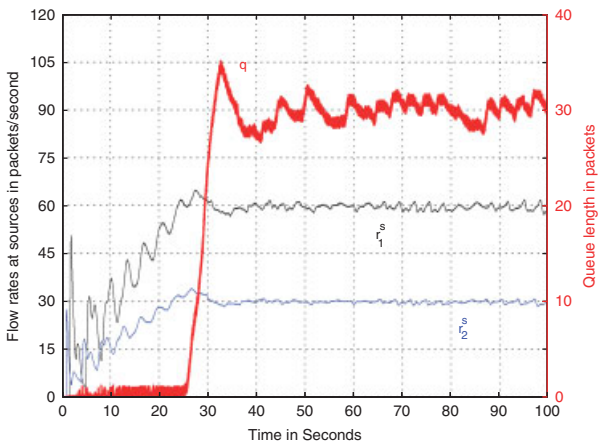

Figure 14. Simulation results for Case 5.

resulting controller more robust, but more conservative. As shown in Figure 12, when we apply the same actual delays as in Case 2, it takes a longer time for the queue to respond, but the overshoot is smaller.

Case 4: To show the effect of the design parameter σ_1, we increased its value four times as shown in Table I. This makes the response faster but more oscillatory as shown in Figure 13.

Case 5: To show the effect of the design parameter σ_2, we increased its value four times as shown in Table I. This makes the the fairness condition (12) to

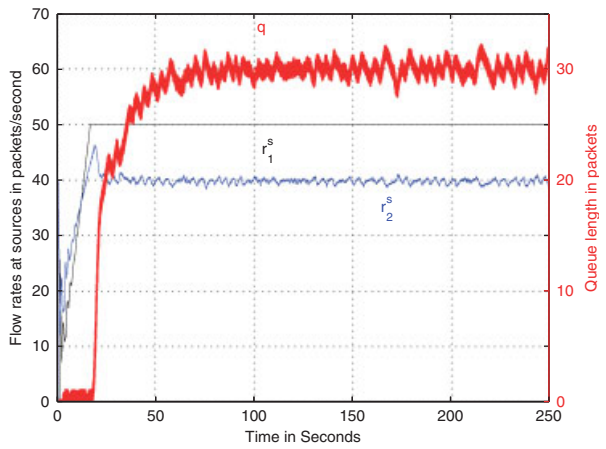

Figure 15. Simulation results for Case 6.

be satisfied even during the transient response, but now the response is slower as shown in Figure 14.

Case 6: The rate limits of the sources are decreased to 50 packets/second. This causes the rate of the first source to saturate as shown in Figure 15. The controller, however, increases the rate of the second source to compensate. Because of this extra compensation, however, the response here is slower compared to the central case (note the difference in the time-scale of this graph compared to the previous ones).

Case 7: To simulate the effects of cross and reverse traffic, in this case we consider the changes in the capacity, c, of the outgoing link. We assume that c changes as a square wave as shown in Figure 16. The response now undergoes a transient at every change of the capacity as shown in the same figure. The same steady-state as in Case 1, however, is reached before the next change. In this case, after each sudden drop of the capacity, the queue length reaches the buffer limit for a short duration of time and a small amount of data is lost (which must be re-transmitted). This could be avoided by reducing the desired queue length, q_d. This would, however, increase the under utilization of the outgoing capacity (indicated by a zero queue length) following each sudden increase.

Case 8: We take equal nominal delays in two channels (hence only one adobe delay problem is solved to design the controller). As shown in Figure 17, the

DOI: 10.1002/rnc

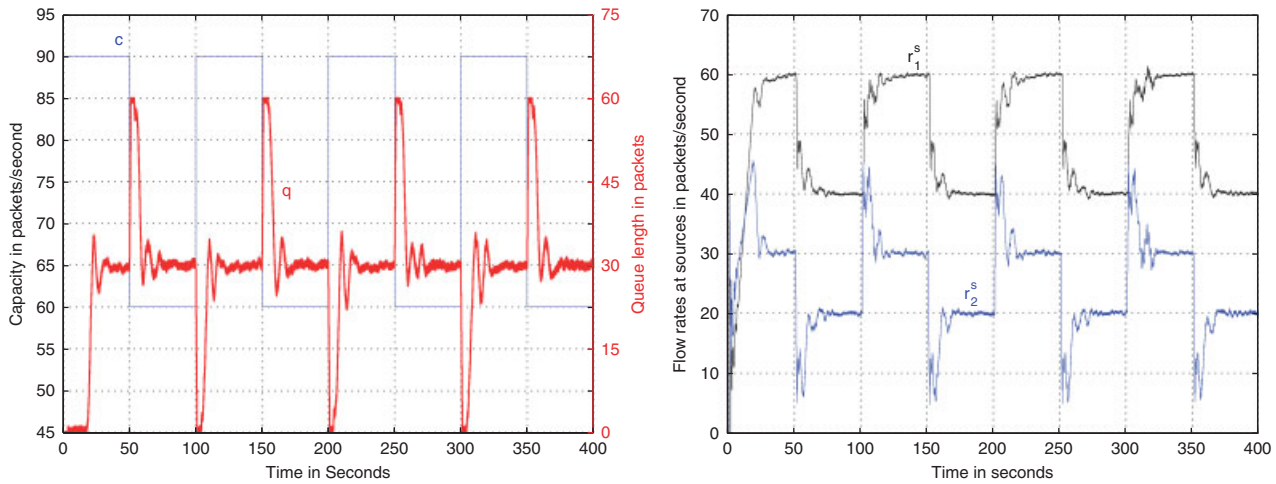

Figure 16. Outgoing link capacity and simulation results for Case 7.

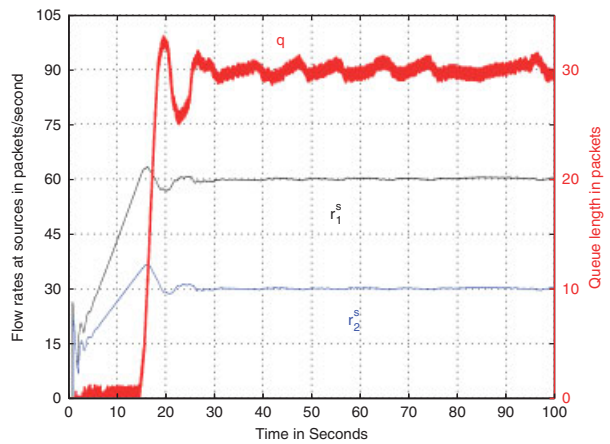

Figure 17. Simulation results for Case 8.

response is faster compared to Case 1. This is due to smaller nominal delay in channel 1. The rate response of the controller is the same in both channels (apart from the ratio α_1/α_2) since the nominal delays are equal.

To compare our controller to the controller proposed in [19], we also design a controller using the approach of [19] using the design parameters (except σ_1 and σ_2, which are not used in the approach of [19], where tracking and robustness are achieved by solving a two-block problem and fairness is achieved by including the fairness weights in the controller derivation) shown in Table I for Case 1. When we take the uncertain part of the actual delays as shown in Table II for Case 1, we obtain the response shown in Figure 18. The queue response, when compared to the response shown in Figure 10, is slower but has less overshoot. The two rates, in the case of the controller of [19], are also closer to each other, apart from a ratio given by the fairness weights. This is due to the fact that in [19] fairness is achieved by including the fairness weights in the controller derivation. The present approach has more design flexibility since relative weights of robustness, tracking, and fairness can be defined using parameters σ_1 and σ_2.

When the actual time-delay in channel 1 is increased as shown in Case 2 of Table II, the controller designed by the approach of [19] produces an unstable response as shown in Figure 19. This shows that the controller designed by the approach proposed here has better robustness properties than the controller of [19], especially when there is an imbalance among the uncertain parts of the actual delays in different channels. Even the controller of [19] is redesigned using a larger δ_1^+ as shown in Case 3 of Table I, the response is still unstable as shown in Figure 20.

DOI: 10.1002/rnc

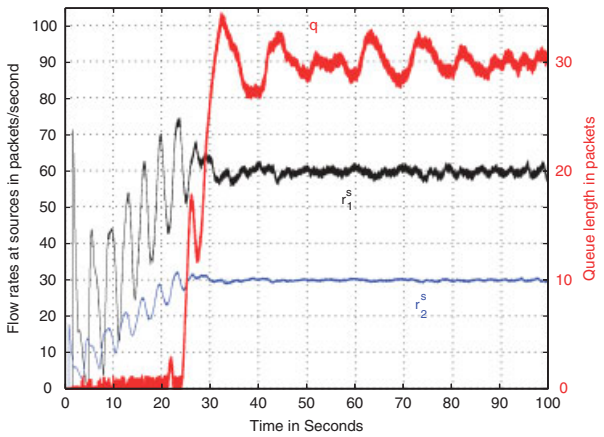

Figure 18. Results of [19] for Case 1.

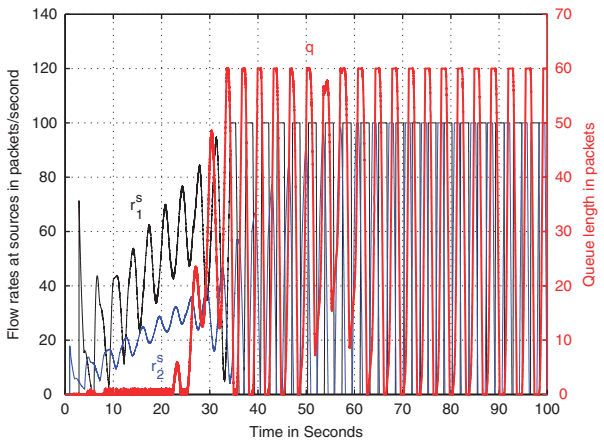

Figure 19. Results of [19] for Case 2.

5. CONCLUSIONS

In this work, robust controller design has been considered for the flow control problem in data-communication networks as defined in [19]. A controller, which is robust against uncertain time-varying multiple time-delays and which satisfies tracking and fairness requirements, is designed by solving an $\mathscr{H}^\infty$ optimization problem using the method of [16]. Unlike [19], where only a suboptimal solution could be found, the present approach allows to design an optimal controller. The present approach also provides more design flexibility, since relative weights of robustness, tracking, and fairness can be defined using parameters σ_1 and σ_2.

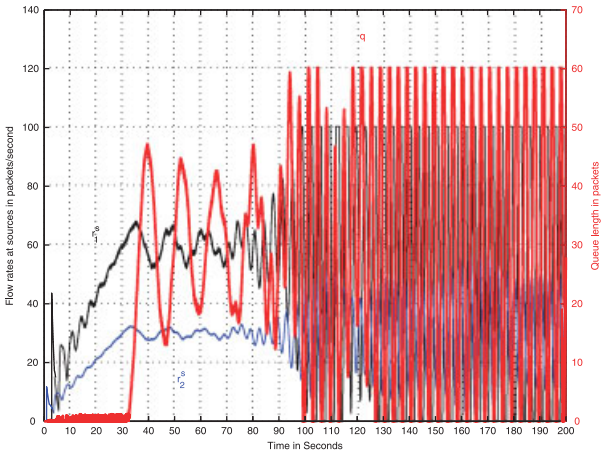

Figure 20. Results of [19] for Case 3.

As opposed to our earlier work [21], here, using [25], we allowed the uncertain part of the time-delays to be negative. The improvement in the time-domain results obtained by this relaxation can be observed from the simulation results given in [25].

The mathematical model represented by (7) and (9) may also appear in flow control problems in areas such as transportation networks, material transport systems, manufacturing systems, and process control [8]. Therefore, the controller design approach presented here may also be applied to flow control problems in areas other than data-communication networks. In fact, model (7)–(9) is simply a multi-input integrating system with different uncertain time-varying time-delays with a jitter effect in its input channels. Note that, the present approach may be extended to a case when there are multiple output channels with similar time-delays. Therefore, our approach may be extended to the control of any integrating system with multiple uncertain time-varying time-delays in its input and/or output channels.

In this work, we considered a network with only a single-bottleneck node. The present approach may be extended to the case of multiple bottleneck nodes along the lines of [37].

DOI: 10.1002/rnc

REFERENCES

1. Altman E, Başar T, Srikant R. Multi-user rate-based flow control with action delays: a team-theoretic approach. *Proceedings of ACM/SIGCOMM*, San Diego, CA, U.S.A., 1997; 2387–2392.
2. BenMohamed L, Meerkov S. Feedback control of congestion in store-and-forward datagram networks: the case of a single congested node. *IEEE/ACM Transactions on Networking* 1993; **1**:693–708.
3. Bonomi F, Fendick KW. The rate-based flow control framework for the available bit rate ATM service. *IEEE Network* 1995; **9**(2):25–39.
4. Floyd S. End-to-end congestion control schemes: utility functions, random losses and ECN marks. *ACM Computer Communication Review* 1994; **24**:10–23.
5. Kunniyur S, Srikant R. TCP and explicit congestion notification. *Proceedings of the INFOCOM*, Tel Aviv, Israel, March 2000; 1323–1332.
6. Laberteaux KP, Rohrs CE, Antsaklis PJ. A practical controller for explicit rate congestion control. *IEEE Transactions on Automatic Control* 2002; **47**:960–978.
7. Mascolo S. Smith's principle for congestion control in high-speed data networks. *IEEE Transactions on Automatic Control* 2000; **45**:358–364.
8. Niculescu SI. *Delay Effects on Stability*: *A Robust Control Approach*. Lecture Notes in Computer and Information Science, vol. 269. Springer: Berlin, 2001.
9. Toker O, Özbay H. $\mathscr{H}^\infty$ optimal and suboptimal controllers for infinite dimensional SISO plants. *IEEE Transactions on Automatic Control* 1995; **40**:751–755.
10. Curtain RF, Zwart H. *An Introduction to Infinite-Dimensional Linear Systems Theory*. Springer: New York, 1995.
11. Foias C, Özbay H, Tannenbaum A. *Robust Control of Infinite Dimensional Systems*: *Frequency Domain Methods*. Lecture Notes in Computer and Information Science, vol. 209. Springer: New York, 1996.
12. Nagpal KM, Ravi R. $\mathscr{H}_\infty$ control and estimation problems with delayed measurements: state-space solutions. *SIAM Journal on Control and Optimization* 1997; **35**: 1217–1243.
13. Tadmor G. The standard $\mathscr{H}_\infty$ problem in systems with a single input delay. *IEEE Transactions on Automatic Control* 2000; **45**:382–397.
14. Meinsma G, Zwart H. On $\mathscr{H}^\infty$ control for dead-time systems. *IEEE Transactions on Automatic Control* 2000; **45**:272–285.
15. Mirkin L, Raskin N. Every stabilizing dead-time controller has an observer-predictor-based structure. *Automatica* 2003; **39**:1747–1754.
16. Meinsma G, Mirkin L. $\mathscr{H}^\infty$ control of systems with multiple I/O delays via decomposition to adobe problems. *IEEE Transactions on Automatic Control* 2005; **50**:199–211.
17. Özbay H, Kalyanaraman S, İftar A. On rate-based congestion control in high-speed networks: design of an $\mathscr{H}^\infty$ based flow controller for a single bottleneck. *Proceedings of the American Control Conference*, Philadelpia, PA, U.S.A., June 1998; 2376–2380.
18. Özbay H, Kang T, Kalyanaraman S, İftar A. Performance and robustness analysis of an $\mathscr{H}^\infty$ based flow controller. *Proceedings of the IEEE Conference on Decision and Control*, Phoenix, AZ, U.S.A., December 1999; 2691–2696.
19. Quet P-F, Ataşlar B, İftar A, Özbay H, Kalyanaraman S, Kang T. Rate-based flow controllers for communication networks in the presence of uncertain time-varying multiple time-delays. *Automatica* 2002; **38**:917–928.
20. Ataşlar B. Robust flow control for data communication networks. *Ph.D. Dissertation*, Anadolu University, Eskişehir, Turkey, September 2004 (in Turkish).
21. Ünal HU, Ataşlar-Ayyıldız B, İftar A, Özbay H. Robust controller design for multiple time-delay systems: the case of data-communication networks. *Proceedings of the 17th International Symposium on Mathematical Theory of Networks and Systems*, Kyoto, Japan, July 2006.
22. Zames G. On the input-output stability of time-varying nonlinear feeedback systems Part I: conditions derived using concepts of loop gain, conicity, and positivity. *IEEE Transactions on Automatic Control* 1966; **AC-11**:228–238.
23. Sandberg IW. On the L_2-boundedness of solutions of nonlinear functional equations. *Bell System Technical Journal* 1964; **43**:1581–1599.
24. Ünal HU, İftar A. A small gain theorem for systems with non-causal subsystems. *Automatica* 2008; **44**:2950–2953.
25. Ünal HU, İftar A. Utilization of non-causal uncertainty blocks in the robust flow controller design problem for data-communication networks. *Proceedings of the 17th IEEE International Conference on Control Applications*, San Antonio, TX, U.S.A., September 2008; 61–66.
26. Zhou K, Doyle JC, Glover K. *Robust and Optimal Control*. Prentice-Hall: Englewood Cliffs, NJ, 1996.
27. Kimura H. *Chain-Scattering Approach to H_∞ Control*. Birkhauser: Boston, 1996.
28. Srikant R. *The Mathematics of Internet Congestion Control*. Birkhäuser: Basel, 2004.
29. Chen S, Nahrstedt K. An overview of Quality-of-Service routing for the next generation high-speed networks: problems and solutions. *IEEE Network* 1998; **12**(6):64–79.
30. Jiang H, Dovrolis C. Passive estimation of TCP round-trip times. *ACM SIGCOMM Computer Communications Review* 2002; **32**(3):75–88.
31. Lim H, Xu K, Gerla M. TCP performance over multipath routing in mobile ad hoc networks. *Proceedings of the IEEE International Conference on Communications*, Anchhorage, AK, U.S.A., 2003; 1064–1068.
32. Niculescu SI. On delay robustness analysis of a simple control algorithm in high-speed networks. *Automatica* 2002; **38**: 885–889.
33. Quet P-F, Özbay H. On the design of AQM supporting TCP flows using robust control theory. *IEEE Transactions on Automatic Control* 2004; **49**:1031–1036.

DOI: 10.1002/rnc

34. Yan P, Gao Y, Özbay H. A variable structure control approach to active queue management for TCP with ECN. *IEEE Transactions on Control Systems Technology* 2005; **13**: 203–215.
35. Manfredi S, di Bernardo M, Garofalo F. Reduction-based robust active queue management control. *Control Engineering Practice* 2007; **15**:177–186.
36. Lin C-L, Chen C-H, Huang H-C. Stabilizing control of networks with uncertain time varying communication delays. *Control Engineering Practice* 2008; **16**:56–66.
37. Munyas İ. Yelbaşı Ö, Biberović E, İftar A, Özbay H. Decentralised robust flow controller design for networks with multiple bottlenecks. *International Journal of Control* 2009; **82**:95–116.

DOI: 10.1002/rnc