ON THE NATURE OF THE TIME-VARIANT COMMUNICATION DELAYS

PETER H. BAUER AND MIHAIL L. SICHITIU

Dept. of EE
University of Notre Dame
Notre Dame, IN 46556, USA

KAMAL PREMARATNE

Dept. of ECE
University of Miami
Coral Gables, FL 33124,USA

ABSTRACT

This paper establishes two major classes of system theoretic models that allow descriptions of interconnections of discrete time systems over communications channels with time-variant delays. Unlike previous followed approaches, the introduced models allows to separate the time-variant delay from the system interface. Applications of these models include teleoperation systems, congestion control in data networks, integrated communications and control systems, etc.

Keywords: modeling, time-variant delays, control through data networks.

1. INTRODUCTION

Control applications, where a plant and a controller will exchange sensor and control information through a network, are currently receiving a lot of attention in the literature, see for example [1]-[13]. The most efficient networks for data exchange proved to be the packet switched networks with their different incarnations: Internet, ATM, etc. Unfortunately the efficiency of the packet switched networks is traded off against the quality of service provided to the end users. We are specifically interested in the delays encountered by a data packet that is transmitted between two linear systems, i.e. a controller and a plant. These delays are usually not only non-negligible, but are also time-variant and they tend to vary in unpredictable ways (usually as a result of network congestion). While there are a number of other parameters that characterize a communication link (like bandwidth and error rate for example) in this paper we will focus on modeling the time-variant delays (TVD) and the required linear system interfaces associated with the use of communication links in feedback systems. The derived models will cover a large spectrum of "real world" applications: control over the Internet, dedicated ICCS (integrated communication and control systems), congestion control in the ABR option of ATM networks, etc. Closely related to the work introduced here are the papers by Chan and Özgüner [1],[2] and also [4], since this work also uses a discrete time model with variable delays. The scope of this paper is sufficiently general to include modeling time-variant delays for TCP/IP and ABR traffic in ATM networks, which so far has not been done. Furthermore the results presented in this paper separate the time-variant delay from the linear system interface. This has many advantages for the system analysis and has not been done before. Current work on this issue can be divided into two categories: the first category describes conventional controller-plant feedback systems where the interconnections are made via a data network [1]-[7], whereas the second category is congestion control of the network itself as in the ABR option in ATM networks [8]-[11]. The work in [12],[13] is applicable to both situations. In both cases the occurring delays are often highly time-variant in nature.

This work was supported by NSF grants ANI 9726253, ANI 9726247

Consider the discrete time system presented in Figure 1 where by $P(z)$ we denote the plant, by $C(z)$ the controller and by $\tau_1(n)$ and $\tau_2(n)$ the two delays associated with the two communication links between the plant and the controller. A system of this type was modeled and analyzed in [15] by treating the time-variant delay and the interface jointly.

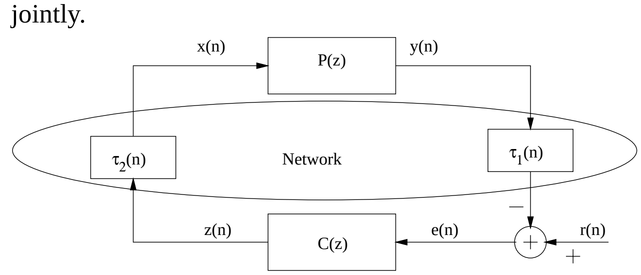

Figure 1: Feedback through data networks

We can distinguish two distinct components of the time-variant delay [14]: the first one is the delay introduced by the communication link while the second one is the interface we use to connect the network to the receiving/sending system. Let us analyze the two components separately. This interface arises as a system theoretic necessity rather than being motivated by hardware.

2. THE TIME-VARIANT DISCRETE TIME DELAY

In Figure 2 the first component of a time-variant delay communication link is shown : the delay $d(n)$. In what follows we will assume for simplicity that at the network ends we have single input single output (SISO) systems, i.e. the signals at the the network ends are scalar. The input to the time-variant delay $d(n)$ is a sequence of input sets $u(n)$ whereas the output is given by a sequence of output sets $v(n)$. The sets $u(n)$ and $v(n)$ describe all input/output signals (scalar or otherwise) that are entering or leaving the block $d(n)$ within a sampling period, i.e., between time instant $n-1$ and time instant n. The output signal is a sequence of sets satisfying the following relation:

$$v(m) = \bigcup_{n+d(n)=m} u(n) \tag{1}$$

i.e. the set $v(m)$ is the union of all sets $u(n)$ which satisfy $n + d(n) = m$ with $n \geq 0,\ d(n) \geq 0$.

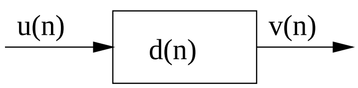

Figure 2: A time-variant delay

We will use the following notations:

- $M(n)$ is the cardinality of the input $u(n)$ at time n.
- $N(n)$ is the cardinality of the output $v(n)$ at time instant n.
- $d(n)$ is a function $d : \mathcal{N} \rightarrow \mathcal{N}$ (where $\mathcal{N}$ is the set of positive integers) satisfying the inequality:

$$0 \leq d_{\min} \leq d(n) \leq d_{\max} < \infty. \tag{2}$$

$d(n)$ represents the delay the communication link introduces at time instant n for the input $u(n)$.

The properties of the delay are completely described by the function $d(n)$. The function $N(n)$ can be computed from the values of $d(n)$ with the following expression:

$$N(n) = \sum_{i \in U(n)} card(M(i)) \tag{3}$$

with U(n) defined as:

$$U(n) = \{i \mid i + d(i) = n,\ i \geq 0\} \tag{4}$$

where by $card(A)$ we denote the cardinality of the set A. $U(n)$ is the set of time instances that produce outputs at time instant n.

Notice that at some time instances the set $v(n)$ may contain more than one value while at other time instances the set may be empty. This is due to the discrete nature of time in our system. Intuitively, between two ticks of the system clock there might be one packet arriving from the communication link, more than one packet or none.

Equations (1-4) represent the most general description of a time-variant discrete time delay. By nature these delay inputs/outputs must be multi-valued (due to the finite and discrete time resolution). This model does not make any assumption on how the delay is interfaced with the linear system (input/output buffer, register, FIFO queues, etc.) and describes solely the nature of the delay itself.

For the case of multi-input multi-output systems we can have two possible implementations:

- if the components of a multi-output signal are individually sent in different packets, then we will can treat each component as a scalar signal and use the model presented above.
- if all the components of a multi-output signal are packed and sent together then we can use the same approach with the difference that instead being sets of scalars $u(n)$ and $v(n)$ will be sets of vectors.

3. THE INTERFACE

As we saw in the previous section the output signal set of a time-variant delay is of varying cardinality thus making it impossible to represent it as a scalar time function. In order to "connect" the communication link to a linear time-invariant (LTI) system we must define an interface that would make the output signal proper, that is of constant dimension. For obvious reasons we choose scalar signals, since SISO systems are assumed. Depending on the type of application different interfaces must be used.

3.1. The Hold Freshest Sample (HFS) Interface

In the case of a regular plant/controller pair which exchanges samples through a packet switched network, a common solution for the case when a new sample did not arrive on time is to use the "hold freshest sample" interface (see Figure 3). This interface holds the most recently received sample both at the plant and controller input for as long as necessary, i.e. until a more recent sample arrives. If more than one sample arrives in any one sampling interval, the most recent sample is chosen. If an old sample arrives after a newer one was received it is simply discarded, being considered outdated. The latter might happen if the packets can arrive out of order (for example on a network using UDP/IP with dynamical routing). Of course, any implementation of this interface relies on the existence of a time-stamp that will indicate the "age" of the samples. (If the communication channel is a FIFO structure, then a time stamp is not necessary to implement an HFS interface.)

We will provide here a formal description of the HFS interface:

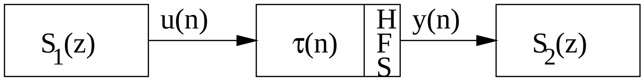

Figure 3: A time-variant delay with a hold freshest sample interface

At any time instant n, using the notations defined in equations (3),(4) we can have one of the following three situations:

(a) $N(n) = 0$

that is there is no signal coming at the output of the time-variant delay block. In this case we simply hold the previous sample, thus we have:

$$y(n) = y(n-1)$$

(b) $\max(U(n)) < f(n-1)$

that is all the signals that arrived at time n are older than the freshest sample at time $n-1$ where $f(n)$ is the time-stamp of the freshest sample received by time instant n. Also in this situation we discard the newly arrived (but older) signals and we hold the previous sample:

$$y(n) = y(n-1)$$

(This situation cannot occur in FIFO structures.)

(c) $\max(U(n)) > f(n-1)$

that is at least one more recent sample arrived at time instant n and we choose the most recent one:

$$y(n) = u(f(n))$$

where we update $f(n)$ by:

$$f(n) = \max(U(n))$$

We can combine the three cases presented above by writing:

$$\begin{aligned} y(n) &= u(f(n)) \quad \text{where} \\ f(n) &= \max\{U(n) \cup \{f(n-1)\}\}. \end{aligned}$$

Thus we obtained a scalar output from a scalar input, i.e. we avoided output signal sets at the delay block. Furthermore we can write:

$$y(n) = u(n - \tau(n)) \tag{5}$$

with

$$\tau(n) = n - f(n). \tag{6}$$

i.e. $y(n)$ is now defined for all n. Observe that without defining a proper interface between the time-variant delay and the linear system, equation 5 does not describe the time-variant delay, since there might be time instances n for which no $y(n)$ exists!

With the help of equations (5,6), one can now analyze the resulting system [12, 13] using the available linear systems tools.

Notice that if $d(n)$ is bounded as in equation (2) then $\tau(n)$ is also bounded:

$$0 \le \tau_{\min} \le \tau(n) \le \tau_{\max} < \infty \tag{7}$$

Using a notation similar with the one in [2], we can write equation (5) as:

$$y(n) = \alpha_{\tau_{\min}}(n)u(n-\tau_{\min}) + \ldots + \alpha_{\tau_{\max}}(n)u(n-\tau_{\max})$$

where $\alpha_i(n) \in \{0, 1\},\ i = \tau_{\min}, \ldots, \tau_{\max}, \quad \forall n \ge 0$ and $\sum_{i=\tau_{\min}}^{\tau_{\max}} \alpha_i(n) = 1 \ \ \forall n \ge 0$

The corresponding state space model is given by the system matrices:

$$A = \underbrace{\begin{pmatrix} 0 & 1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \ldots & 1 \\ 0 & 0 & 0 & \ldots & 0 \end{pmatrix}}_{\tau_{\max}} \quad B = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix} \tag{8}$$

$$C(n) = (\alpha_{\tau_{\max}}(n) \ldots \alpha_1(n))\,; \quad D(n) = (\alpha_0(n))\,;$$

where if $\tau_{\min} > 0$ we have $\alpha_j(n) = 0$ $\forall j = 0 \ldots \tau_{\min} - 1 \ \ \forall n \ge 0$.

Notice that by HFS definition the coefficients $\alpha_i(n)$ can not vary arbitrarily from one time instant to another: the delay $\tau(n)$ is restricted by $\tau(n+1) \le \tau(n) + 1$, and hence we have:

$$\alpha_i(n) = 1 \Rightarrow \alpha_j(n+1) = 0 \ \ \forall j > i+1$$

In other words the used sample from the time-variant delay output ages with time, but not faster. Depending on the particular applications (CANs and ICCS, control over ATM networks, control over Internet, etc.) additional constraints on $\tau(n)$ can be formulated.

The delays encountered on the communication link $d(n)$ are in general not equal to the delays perceived at the output of the HFS interface $\tau(n)$. Given the values of $d(n)$ one can compute the values of $\tau(n)$ but not the other way around. The bounds on the values of $d(n)$ and $\tau(n)$ are however equal: considering equations (2) and (7), we have that $d_{\min} = \tau_{\min}$ and $d_{\max} = \tau_{\max}$.

3.2. The Variable Bit-Rate (VBR) Interface

Modeling a variable bit rate channel requires to quantify the number of bits $u(n)$ sent into the channel at any given time n. Hence, the input $u(n)$ signifies the amount of data sent per discrete time unit rather than being the actual number symbol that is transmitted like in other applications. (A specific example of such a variable bit rate delay interface would be in congestion control, where buffer occupancy levels need to be controlled by varying the source rates.)

Let us focus on the link between the controller and the plant as it is shown in Figure 4.

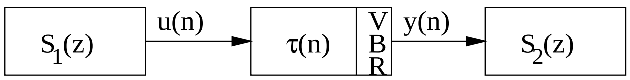

Figure 4: A time-variant delay with a VBR interface. $u(n)$ denotes the amount of data entering the delay during a clock cycle. $y(n)$ is the amount of data exiting the delay

For now, we assume that there are no packet/cell losses on the communication channel, and we do not require the FIFO property. (If necessary, such a constraint could be easily incorporated in the model.) Also, $S_2(z)$ is a buffer or a queue which can be modeled as a digital integrator.

At any time instant n, using the notations defined in equations (3),(4) we can have one of the following two situations:

(a) $N(n) = 0$

that is there is no signal at the output of the time-variant delay (i.e. no packet was received). In this case:

$$y(n) = 0$$

(b) $N(n) > 0$

that is at least one signal is exiting the time-variant delay (i.e. at least one packet was received). In this case all the values that arrived during this time period are added, (i.e. representing the total number of received bits) since the communication link cannot lose data:

$$y(n) = \sum_{i \in U(n)} u(i)$$

The corresponding state space model is:

$$A = \underbrace{\begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ 0 & 0 & 0 & \dots & 0 \end{pmatrix}}_{\tau_{\max}} \quad B(n) = \begin{pmatrix} \beta_1(n) \\ \beta_2(n) \\ \beta_3(n) \\ \vdots \\ \beta_{\tau_{\max}}(n) \end{pmatrix} \tag{9}$$

$$C = \begin{pmatrix} 1 & 0 & \dots & 0 & 0 \end{pmatrix} \quad D(n) = (\beta_0(n)).$$

If the communication link is lossless (i.e. all data that enters the communication link arrives at its output) and at each time instant we transmit exactly one data packet, the coefficients $\beta_i(n)$ are given by:

$$\beta_i(n) = \begin{cases} 1 & \text{if } d(n) = i \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

with $i = d_{\min}, \dots, d_{\max}$.

In this case we have $\sum_{i=d_{\min}}^{d_{\max}} \beta_i(n) = 1$, with $\beta_i(n) \in \{0, 1\}$ $i = d_{\min}, \dots, d_{\max}$.

Consider the case of a lossy communication link (i.e. some packets may be dropped while transmitted through the channel) and assume that at each time instant all the data is shipped via a single packet (this packet may be dropped with a probability depending on the error rate). In this case we have: $\sum_{i=d_{\min}}^{d_{\max}} \beta_i(n) \in \{0, 1\}$, with $\beta_i(n) \in \{0, 1\}$ $i = d_{\min}, \dots, d_{\max}$.

Consider the most general case with a lossy link and allow the data transmitted at each time instant to be shipped via multiple packets that may arrive at destination at different time instants. Then $\sum_{i=d_{\min}}^{d_{\max}} \beta_i(n) \le 1$, with $\beta_i(n) \in [0, 1]$ $i = d_{\min}, \dots, d_{\max}$.

Using equation (9) we can also write for the VBR interface:

$$y(n) = \sum_{i=\tau_{\min}}^{\tau_{\max}} \beta_i(n-i)u(n-i) \tag{11}$$

In case the communication channel is a FIFO structure, additional constraints similar to those presented in section 3.1 must be imposed.

3.3. The Maximum Delay (MD) Interface

Of course it is well known, that it is always possible to reduce the problem of time-variant delays to a time-invariant problem if we introduce additional delays such that the sum of the delays experienced by a signal on the communication links and the artificial delays is always constant (i.e. time-invariant). Thus if the delays encountered on the communication links are bounded as in equation (2) we can make the total delay equal to $d_{\max}$. We will call this way of handling the delays on the communication links the "maximum delay interface". Similarly to the case of the hold freshest sample interface the implementation of the maximum delay interface is dependent on the existence of a time-stamp on the communication packets.

4. PROPERTIES OF TIME-VARIANT DELAYS

4.1. Commutativity of TVDs

The question asked in this section is whether or not we can interchange two different time-variant delays and obtain the same equivalent system in both cases.

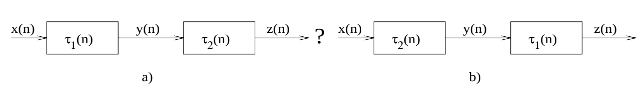

Figure 5: Commutativity of two time-variant delays

With the notations in Figure 5 we have for case (a):

$$y(k) = \bigcup_{i \in U(k)} x(i) = \bigcup_{i+\tau_1(i)=k} x(i)$$

$$z(n) = \bigcup_{k+\tau_2(k)=n} y(k) = \bigcup_{i+\tau_1(i)+\tau_2(i+\tau_1(i))=n} x(i) \quad (12)$$

Using a similar deduction we obtain for case (b):

$$z(n) = \bigcup_{i+\tau_2(i)+\tau_1(i+\tau_2(i))=n} x(i) \quad (13)$$

Comparing equations (12) and (13) we can state that in general commutativity does not hold. One notable exception is the case when the two delays are constant. The addition of a hold freshest sample interface or of a variable bit-rate interface does not change the situation, i.e. the two time-variant systems remain non-commutative.

4.2. Commutativity of a TVD with a LTI System

The question asked in this section is whether or not we can interchange a time-variant delay and a linear time-invariant system and obtain the same equivalent system in both cases.

A positive answer to this question would enable us to simplify a closed loop system involving communication links (see for example Figure 1) by allowing the grouping of the time-variant delays (which can be further concatenated as we will see in section 4.3) and by grouping the controller and the plant into one linear time-invariant system.

Unfortunately it can be shown that both for the HFS interface and for the VBR interface the commutativity with an LTI system does not hold in general. The proof would be beyond the scope of this paper.

4.3. Reducing Two TVDs to One Equivalent TVD

The question answered in this section is: can we reduce two time-variant delays to one equivalent time-variant delay?

An affirmative answer will enable us to model multiple communication links with time-variant delays with only one equivalent communication link with time-variant delays.

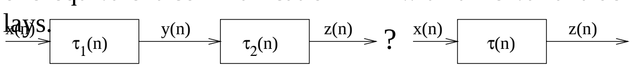

Figure 6: Reducing two delays to one equivalent delay

Following the deduction presented in section 4.1 we obtain:

$$z(n) = \bigcup_{i} x(i) \quad \text{such that } i + \tau_1(i) + \tau_2(i + \tau_1(i)) = n$$

Thus given the values of $\tau_1(i)$ and $\tau_2(i)$ we can construct $\tau(i)$ as:

$$\tau(i) = \tau_1(i) + \tau_2(i + \tau_1(i)) \quad (14)$$

Thus we can reduce the two time-variant delays to one equivalent time-variant delay. Moreover, if we have $\tau_{1\min} \le \tau_1(n) \le \tau_{1\max}$ and $\tau_{2\min} \le \tau_2(n) \le \tau_{2\max}$ then we have:

$$\tau_{1\min} + \tau_{2\min} \le \tau(n) \le \tau_{1\max} + \tau_{2\max}.$$

Notice that reducing the two pairs of time-variant delays in section 4.1 yields two different time-variant delays having the same bounds on the delays.

4.4. Bounds of the variation of the queuing delays

The delays encountered in communication networks have, in general, four components: packet delays, transmission delays, processing delays, queueing delays and access delays. The first three categories are fairly constant, but last two categories are the major source of time-variance.

For the queueing delays, the maximum and minimum delay ($\tau_{\max}$ and $\tau_{\min}$ respectively) are simple to compute: the minimum delay is zero, corresponding to an empty tandem queue (all queues from the source to the destination are empty), while the maximum delay occurs when all the queues from the switch to the source are at maximum occupancy and can be computed given the queues length and their depletion rates.

Since the variation of the delay is due to the variation in the queue length in the switches between the source and the destination we may be able to derive better bounds on the variation of the delays. This requires knowledge of at least bounds on the input/output rate of the buffers in between source and destination. If we can bound the input rate $g(n)$ into a single queue by

$$0 < \gamma_1 g_0 \le g(n) \le \gamma_2 g_0 < \infty \quad (15)$$

where g_0 is the fixed rate at which the queue is depleted, then we can find bounds for the variation of the HFS and VBR delays.

4.4.1. Bounds of the variation of HFS queuing delays

If the delays are generated by a queue satisfying equation (15) and if the delay outputs are processed using an HFS interface, then the delay variation is bounded by:

$$1 - \frac{1}{\gamma_1} \le \tau(n+1) - \tau(n) \le 1 - \frac{1}{\gamma_2}. \quad (16)$$

In general the bounds in equation (16) are not integers and they should be interpreted as a long term delay slope: if $1 - \frac{1}{\gamma_2} \le \frac{q}{p}$ then $\tau(n+p) - \tau(n) \le q$ where p and q are positive integers.

Of course the delay can not decrease with more that the difference between the maximum and the minimum delay:

$$\max\{1 - \frac{1}{\gamma_1}, -(\tau_{\max} - \tau_{\min})\} \le \tau(n+1) - \tau(n) \le 1 - \frac{1}{\gamma_2} < 1 \quad (17)$$

For an excellent and comprehensive treatise of this subject, see the work of Cruz [16], [17].

4.4.2. Bounds of the variation of VBR queuing delays

If the delays are generated by a queue satisfying equation (15) and if the delay outputs are processed using an HFS interface, then the delays encountered by the packets of any input source will have a delay variance bounded by:

$$-1 < \gamma_1 - 1 \le \tau(n+1) - \tau(n) \le \gamma_2 - 1. \qquad (18)$$

Similar to the HFS case, the delay bounds in general are not integers and should be interpreted as a long term delay slope.

5. CONCLUSION

This paper introduced a model for time-variant delays, that describes the nature of the delay itself regardless of the type of system the communication channel needs to interface with. In a second step three classes of delay-linear system interface are introduced:

(1) The HFS interface, applicable to plant-controller type configurations with communication links. The recently considered problems of Integrated Communication Control Systems, teleoperation systems, etc. fall into this category.

(2) The VBR interface, applicable to time-variant delay channels that carry bit streams of variable rates. Congestion control problems fall into this category.

(3) The Maximum Delay Interface. Current Video/Audio services over the Internet fall into this category.

Finally, some basic properties of these models are studied.

6. REFERENCES

[1] H. Chan, Ü. Özgüner "Closed loop control of systems over a communications network with queues", International Journal of Control, vol. 62, no. 3, pp. 493-510, 1995.

[2] H. Chan, Ü. Özgüner. "Optimal control of systems over a communication network with queues via a jump system approach" *Proceedings of the 1995 IEEE Conference on Control Applications*, pp. 1148-1153, 1995.

[3] H. Chan, Ü. Özgüner "Control of Interconnected Systems over a Communication Network with Queues", *Proceedings of the* 33^{rd} *Conference on Decision and Control*, pp.4104-4109, 1994.

[4] R. Krtolica, Ü. Özgüner , H. Chan, H. Goktas J. Winkelman and M. Liubakka "Stability of linear feedback systems with random communication delays", *International Journal of Control* vol. 59, no. 4, pp. 925-953, April 1994.

[5] Y. Halevi and A. Ray, "Integrated Communication and Control Systems: Part I - Analysis" *Journal of Dynamic Systems, Measurement and Control*, vol. 110, pp. 367-373, Dec. 1988.

[6] Shin K. G. and X. Cui, "Computing time delay and its effects on real-time control systems" *IEEE Transaction on Control Systems Technology* vol. 3,no. 2, pp. 218-224, Jun 1995.

[7] A. Bemporad, "Predictive Control of Teleoperated Constrained Systems with Unbounded Communication Delays" *Proceedings of the* 37^{th} *IEEE Conference on Decision and Control*, Tampa, Florida, pp. 2133-2138, December 1988.

[8] C. E. Rohrs, R. A. Berry and S. J. O'Halek "Control Engineer's look at ATM congestion avoidance" *Computer Communication* vol. 19 no. 3, pp. 226-234, Mar. 1999.

[9] Charles E. Rohrs and Randall A. Berry "A Linear Control Approach to Explicit Rate Feedback in ATM Networks" *IEEE Infocom* 1997 Kobe, Japan vol. 3, pp.277-282, 1997.

[10] L. Benmohamed and S. M. Meerkov "Feedback Control of Congestion in Packet Switching Networks: The Case of a Single Congested Node" *IEEE/ACM Transactions on Networking*, vol. 1, no. 6, Dec. 1993.

[11] L. Benmohamed and S. M. Meerkov " Feedback Control of Congestion in Packet Switching Networks: The Case of Multiple Congested Nodes" *International Journal of Communication Systems* vol. 10, no. 5 pp. 227-246, Sep-Oct 1997.

[12] P. Bauer, M. Sichitiu and K. Premaratne "Controlling an Integrator Through Data Networks: Stability in the Presence of Unknown Time-Variant Delays", *Proceedings of the 1999 IEEE International Symposium on Circuits and Systems*, Orlando, Florida, vol. V, pp. 491-494, May 30-Jun. 2 1999.

[13] P. H. Bauer, M. L. Sichitiu and K. Premaratne "Closing the Loop through Communication Networks: The Case of an Integrator Plant and Multiple Controllers" *Proceedings of the* 38^{th} *IEEE Conference on Decision and Control*, Phoenix, Arizona, pp. 2180-2185, Dec. 7-10, 1999.

[14] Cédric Lorand "Stabilitätsuntersuchung von diskreten Systemen mit zeitvariablen Totzeiten" *Diplomarbeit - Technische Universität München* Nov. 1998.

[15] M. M. Ekanayake, *Robust Stability of Discrete Time Nonlinear Systems*. PhD thesis, University of Miami, 1999.

[16] R. L. Cruz, "A calculus for network delay, part I: Network elements in isolation," *IEEE Transactions on Information Theory*, vol. 37, pp. 114–131, Jan. 1991.

[17] R. L. Cruz, "A calculus for network delay, part II: Network analysis," *IEEE Transactions on Information Theory*, vol. 37, pp. 132–141, Jan. 1991.