Networked control with delay measurement and estimation

Nikolai Vatanski, Jean-Philippe Georges, Christophe Aubrun, Eric Rondeau,
Sirkka-Liisa Jämsä Jounela

HAL Id: hal-00347467
https://hal.archives-ouvertes.fr/hal-00347467

NETWORKED CONTROL WITH DELAY MEASUREMENT AND ESTIMATION

Nikolai Vatanski[1], Jean-Philippe Georges[2], Christophe Aubrun[2], Eric Rondeau[2] and Sirkka-Liisa Jämsä-Jounela[1]

1*) Laboratory of Process Control and Automation, Helsinki University of Technology, P.O.Box 6100, FIN-02150 TKK, Finland, Phone: +358-9-451 2669 Fax: +358-9-451 3854

Nikolai.Vatanski@tkk.fi

2) Research Centre for Automatic Control (CRAN UMR 7039), Henri Poincaré University, CRAN-CNRS UMR 7039, Faculté des Sciences et Techniques, BP 239, 54506 Vandoeuvre les Nancy Cedex, France

Abstract: In research on networked control systems (NCS), the performance of the control system is assumed to be affected by network-induced parameters such as delays, jitters, packet losses and link failures. In this paper, two delay compensation methods are studied, adaptive Smith Predictor and robust control, in the framework of Networked Control Systems while considering the protocol specifications. First, a clock synchronization protocol is implemented in each device of the network, thereby providing the possibility of end-to-end delay measurement, and a compensation strategy based on this measurement and the adaptive Smith Predictor is applied. Secondly, the end-to-end upper bound delay estimate is calculated on the basis of the network architecture and the traffic estimation, and then subsequently used in a robust control based compensation strategy. The switched Ethernet network is used as an example of the communication network of the NCS. *Copyright © 2007 IFAC*

Keywords: networked control systems, QoS estimation, delay compensation, real time systems, network calculus

1. INTRODUCTION

Future automation systems, and even those currently in use today, will to an increasing extent consist of a large number of intelligent devices and control systems connected by local or global communication networks. In these Networked Control Systems (NCSs), communication between controllers, sensors and actuators is performed through the networks. In most cases, the network delay in the feedback loop does not significantly affect the performance of the control system. However, for some time-constrained, dynamically fast systems care should be taken when implementing a NCS. For such processes, the communication network in the feedback control loop introduces a significant additional delay, either constant or time varying, that makes the analysis and control design more complex.

Excellent introductory papers to the topic of NCS have been published by Tipsuwan and Chow (2003), Huo, Fang and Ma (2004), Li, Zhi, and Youxia (2002), and for the latest extensive review paper, see Yang (2006). Two main research directions can be distinguished depending on whether the NCS is studied from the control point of view or from the communication point of view.

In the first case, the controller adapts to network perturbations, and this corresponds to the approach usually called "control over network". In this context, the network is a passive resource providing constraints expressed in terms of QoS. The control theory of delayed systems is applied to compensate the effects of communication support in order to guarantee the Quality of Performances (QoP) of the application. In most of the studies carried out in this area, the network is a black box that is modelled in a simplified way. The network structure is considered to be fixed or is assumed to behave in a pre-specified manner. More generally, the network is associated to a delayed system with time varying properties, and the network delay is assumed to be smaller than the sampling period. The other common approach is to approximate the performance of the network on the basis of a model. The objective is to design a control law that is able to compensate the effect of the network such that the application exhibits acceptable performance even under worst-case fluctuations of QoS. Several compensation strategies have been proposed based on the information available from the network or on assumptions about the application. The proposed control strategies vary from the traditional PID gain scheduling (Tipsuwan and Chow, 2002) and the state feedback controller (Yue, Han, and Peng, 2004), to more sophisticated controllers such as H_∞ controllers (Yue, Han and Lam 2005), Generalized Predictive Controllers (GPC) (Tang and de Silva, 2006), LQR controllers (Zhang and Hua-Jing 2006), LMI based control, (Yu, Wang, Chu and

Hao 2003), stochastic optimal controllers (Nilsson, 1998) when delays are shorter than the sampling period, and (Hu and Zhu, 2003) in the reverse case. Since the field of the NCS is, from the control point of view, closely related to the more traditional field of control of the time delay systems (TDS), these results can also been utilized in many cases. Recent developments in time delay systems analysis can be found in Richard (2003).

In the second case, the network is an active resource and can be controlled. In NCS this approach is called "control of network". The network adaptation can be either defined in a best effort mode in continually optimising the network performances in an endogenous way or in adjusting the QoS relatively to the QoP knowledge (exogenous way). The effects of the network on the control of the whole system have been studied in Juanole and Mouney (2006) and in Yook, Tilbury and Soparkar (2001). A theoretical framework is then presented, which allows the impacts of time delays on the performance of the system to be precisely modelled, and these models are then used to determine the optimal network configuration for the given control system. While Juanole and Mouney (2006) associate the framework's priorities to the phase margin, Yook et al. (2001) present a performance degradation function for a different architecture. Walsh, Ye and Bushnell (2002) consider scheduling strategies to guarantee both stability and controller performance. They propose and validate algorithms in which message priorities are dynamically computed. The results are merged in a new protocol called try-once-discard (TOD), for which Walsh, Belfiman and Bushnell (2002) provide an analytical proof of the global exponential stability for both the new protocol and the more commonly used (statically scheduled) access methods. In Kim, Lee, Kwon and Park (2003), a new scheduling strategy has been proposed based on a time division protocol. The objective is to obtain a maximum allowable delay bound for the scheduling of an NCS. The proposed method is formulated in terms of linear matrix inequalities, and addresses the tightness of the delay bound. Finally, this issue is generalized in Nesic and Teel (2004), who propose a unifying framework to generate new scheduling protocols that preserve the stability properties of the system. It relies on specific modelling of the scheduling protocol that is used to compare the Round Robin and the TOD strategies.

In this paper, two delay compensation methods are studied, adaptive Smith Predictor and robust control, in the framework of Networked Control Systems while considering the protocol specifications. Switched Ethernet is used as the communication protocol to evaluate the performance of the methods for networked control.

In Chapter 2, the coupling between QoS parameters and the controller design is based on delay measurements and the adaptive Smith Predictor. The delay measurement is obtained via a synchronisation step of all the computers' clocks by using the IEEE 1588 standard which is commonly implemented in industrial Ethernet switches. The clock synchronisation facilitates the delay measurement and allows the delay variations to be dynamically obtained considering the intrinsic properties and activities of the network. This information can be used by the Smith Predictor controller. Finally, numerical experiments demonstrate that the proposed control scheme is able to ensure stability and performance robustness of a network controlled system.

In Chapter 3, robust control theory is combined with network calculus theory. A network model is proposed to describe both the internal network properties (bandwidth, latency, buffer performance, topology,…) and the traffic, represented as minimal service curves and maximal traffic curves, respectively. These curves are used in the network calculus theory to estimate bounded end-to-end delays which can be translated into uncertainties in the robust controller design. Simulations based on the same scenarios as those presented in Chapter 2 are used to evaluate the performance of the second approach.

2. IMPROVED CONTROL OVER A NETWORK USING DELAY MEASUREMENTS

2.1 Network delay measurement

The possibility of delay observation enables evaluation of the online QoS performance and use of the obtained performance information in control. This would eliminate the need for a network model, which is often difficult to obtain. Therefore, there is a clear need for new instruments in the network to measure the network delay.

Usually, the delay estimate relies on the round trip time (RTT) measurement due to its easy implementation: no clock synchronisation is required since the computations are running on the same device. However, for the NCS, RTT may not be adequate and instrumentation for measuring the end-to-end delay should be developed. The main difficulty in measuring an end-to-end transmission delay is due to timing issues like non-synchronized clocks and scheduling policies of the operating system stack. In the switched Ethernet network, the prediction problem is further amplified due to time variance of the

transmission delay, resulting from the load of traffic variation or even the considerable changes in the network topology, especially when the network is shared with applications other than process control.

The techniques for measuring end-to-end delay often rely on the synchronisation of the clocks of the two end-systems, and sharing the same clock reference for timing events like frame sending and reception. Initially, Mills (1991) proposed a Network Time Protocol (NTP) for synchronizing the clocks of computer systems over packet-switched data networks. However, the accuracy of the NTP protocol may not be adequate for distributed systems with hard time constraints. In order to cope with the inaccuracy, a new clock synchronisation procedure protocol, IEEE 1588, has recently been defined, enabling the development of one-way delay measurement methodologies for distributed systems with real time constraints.

In the IEEE 1588 standard two kinds of clocks are defined: boundary and ordinary. The boundary clocks are used by network devices (like switches) and the ordinary clocks are implemented in the devices with a single network interface card, like sensors, actuators etc. Each network interface card with a boundary clock can act either as a master or as an ordinary clock in one network segment. The synchronisation is driven by master/slaves exchanges. The time reference for this, the grandmaster clock, is chosen according to its stability and accuracy. This clock is unique for the whole architecture. The PI control algorithm is applied in reducing the error between the master clock and the slaves.

The method proposed here for the end-to-end delay measurement consists of the synchronisation procedure of the network devices with the IEEE 1588 protocol, followed by frame time stamping. After the device synchronisation, the frames are time stamped at the sender side and the difference between the time stamp included in the frame and the reception time is computed at the receiver side. As the frame's Ethernet header does not contain a field to add this time stamp, the information should be defined in the protocol used at the application layer. In the switched Ethernet architectures, the PTP protocol, needed for the IEEE 1588 synchronisation, can easily be implemented by considering the switches as master clock references. Specific improvements in this case have been proposed by Jasperneite, Shebab and Weber (2004), and Gaderer, Loschmidt and Sauter (2005). The originality of the proposed approach is related to the accuracy of this estimate, especially compared to RTT measurements. In fact, this accuracy is emphasized in the context of switched topology.

Since the obtained delay measurement corresponds to the delay experienced by the last message for a given flow, this methodology only enables an estimate to be made for the latest delay. As this is based on past observations, it gives information about the trends and time variance of the delay. As a consequence of the procedure, the measurement of the delay will only be available at the receiver side. In networked control systems, this implies that the transmission delay of the last control information is stored on the actuator, even though this knowledge is more relevant to the controller. However, the information about the sensor path delay is available at the controller side, and thus can be used directly for control compensation.

2.2 Adaptive Smith predictor for delay compensation

A unified approach for control over the network for networked control systems requires that the controller has the ability to react to the evolution of QoS parameters (and, in particular, the delay). The system should be robust and capable of compensating the delay induced by the network. Prior to presenting the compensation approach, the following assumptions about the process and the network are made:

1. The transmission delay induced by the communication network is long compared to the effective time constant τ of the plant.

2. The sampling period necessary to capture all relevant process dynamics is significantly smaller than the network induced delay.

3. After each clock cycle, the sender attempts to send a packet containing continuously updated information

4. After each clock cycle, the receiver examines its input queue for newly received information. If no new packet is received then the previous information is used.

5. All control and measurement information in the network is sent in a single packet.

6. No packet losses occur in the communication network.

Consider a SISO, open loop, stable linear process with transfer function $P(z)$. The adaptive Smith predictor configuration scheme which integrates the measurement information is shown in Figure 1. In the figure, the minor compensation loop consisting of sensor path delay measurement $\delta_s^*(n)$, plant model $P^*(z)$ and actuator path delay measurement $\delta_a^*(n)$, is introduced around the nominal controller

6

$C(z)$. Assuming that the plant model is known, then the measurement for the sensor path delay is available and equals the actual sensor delay value $\delta_s(n)$, and the measurement for the actuator path delay is available and equals the actual actuator delay value $\delta_a(n)$ the delayed process measurement and the delayed process output estimate become equivalent:

$$\delta_s(n) \cdot P(z) \cdot \delta_a(n) = \delta_s^*(n) \cdot P^*(z) \cdot \delta_a^*(n) \tag{1}$$

Observe that the signal reaching the controller, designated as $\varepsilon_c(n)$ in the diagram, is a corrected error signal given by:

$$\varepsilon_c(n) = r(n) - (\delta_s^*(n) P^*(z) \delta_a^*(n) - \delta_s(n) P(z) \delta_a(n)) u(n) - P^*(z) u(n) \tag{2}$$

or

$$\varepsilon_c(n) = r(n) - P^*(z) u(n)$$

This implies, in the case where the plant and the time variant delays $\delta_a(n)$ and $\delta_s(n)$ are commutative, that the error signal reaching the controller is calculated on the basis of an undelayed estimate of the process output. The net result of introducing the minor loop is therefore compensation of the time delay factor caused by the network delays from the feedback loop, where it causes stability problems and moving it outside the loop where it has no deteriorating effect on closed-loop stability. Notice, however, that in the general NCS case, the commutativity assumption for the linear system and a time variant delay does not hold. In other words, $\delta_a P(z) \neq P(z)\delta_a$, which implies that grouping the time-variant delays and grouping the controller and linear time-invariant system together is not possible, and therefore only the sensor path delays can be eliminated (Bauer et al 2001). This elimination is possible since the sensor time delay can be determined in real time at the controller side using the delay measurement methodology. Even though the elimination of the actuator delay is not possible, this effect can also be reduced by an adaptive mechanism, for example by gain scheduling. This is illustrated as the "Adaptation mechanism" block in the scheme in Figure 1.

The main benefit of the Smith predictor scheme compared to other time delay compensators (TDC), is the online use of the delay value. In the NCS case this is beneficial, and the delay measurement can be integrated directly and modified online as the delay value changes. In several other TDC schemes the controller is obtained as a result of optimisation, which may have time constraints to be applied in NCS.

The main drawback of the Smith predictor technique scheme is its sensitivity to model mismatch, either in the delay or in the rational plant part. In the face of even small delay measurement errors, which are inevitable in the NCS, the Smith predictor may give poor performance as the error between the real value is amplified by the high controller gain. For this reason, it is thus imperative in applications of the Smith predictor compensation for networked control to apply an adaptive procedure. One commonly used approach is to update the model parameters as the process parameters vary. This may, however, lead to closed loop instability as a result of the model parameter mismatch. Furthermore, the adaptation procedure may not be suitable for the NCS environment due to timing constraints.

In this paper a different adaptation procedure is proposed which is expected to fit better to the NCS environment. In order to ensure the robust stability and performance of the system, the adaptation will be achieved by changing the nominal controller parameters according to the changes in the delay value error. In the case where the delay uncertainty increases significantly, the controller gain is reduced temporarily in order to satisfy the robust stability and performance conditions. Several studies have been presented, e.g. Morari and Zafiriou (1989), in which the necessary and sufficient conditions for systems, including the Smith predictor, to remain robustly stable in the presence of a small mismatch in delay are given.

2.3. Simulations for improved control over network using delay measurements

Next, the effectiveness of the proposed control over the network configuration using the measured network delay will be illustrated through an example via MATLAB simulations, and also on the experimental platform developed for behavioural analysis of the NCS.

Simulations of the Smith predictor approach in the Matlab environment

Suppose that the sensors, actuators and controller of the plant are spatially distributed and connected over the communication network. In continuous time, the process model of the real-time process and a nominal controller (time in ms) are described as follows:

$$P(s) = \frac{2}{(s+5)(s+0.2)} \tag{3}$$

$$C(s) = \frac{K_P s + K_I}{s}, \quad K_P = 0.5508, \quad K_I = 0.4529 \tag{4}$$

The controller parameters for the nominal controller were obtained by minimizing the integral of the square errors (ISE) for the system with a constant network delay of 1 ms in the actuator and sensor paths.

The transmission delays over the network in the sensor and on the actuator side were assumed to vary randomly between zero and the upper delay value estimate. Four cases were simulated with different upper bound delay values, starting from the shortest delays with the upper bound of 0.5ms, and increasing the upper bound delay to the value of 3.5ms. Figure 2 shows the simulated network delays in both directions, in the sensor and actuator paths.

The simulation results for the nominal controller are illustrated in Figure 3. According to the figure, the nominal controller is unable to maintain the performance of the system. Even though the performance is acceptable at the lowest delay value, the performance significantly decreases as the network delay values increase and, finally, even becomes unstable at the largest delay values. Therefore, it can be concluded that a compensation procedure is required.

Figure 4 presents the case in which the Smith predictor based compensation strategy is applied with the same nominal controller. In the simulations it was assumed that the model of the rational part of the process is known and that the only uncertainty arises from the delay uncertainty caused by the data transmission in the communication network. This uncertainty was simulated by introducing the additional dead time to the network delay estimates in the Smith predictor minor loop. This addition corresponded to the fact that the delay measurement on the sensor side, as well as on the actuator side, arrives at the controller later than the time when the delay occurred. From Figure 4 it can be concluded that the

performance of the system is improved significantly and the control becomes less sensitive to the delay effect produced by the network.

Testing the Smith predictor approach in the experimental platform

A platform for the behavioural analysis of the networked control systems was developed for studying the effects of the communication network on the performance of the control system. Physically the platform consists of the communication network and computers dedicated to simulate the real time process and the controller. The Ethernet switches are augmented with additional embedded intelligence in order to cope with real-time and fault tolerance requirements. In order to cope with real time requirements, the sampled versions of the process model and the controller were directly programmed using C- language and, to further reduce the calculation period, real time programming was necessary and performed through the RealTime Application Interface (RTAI), which enables writing applications with strict timing constraints for the Linux operating system, www.rtai.org.

In the simulation of the process and a controller, three different kinds of threads were defined. The first thread type is responsible for computing a new control action (or the output of the process). Since computation of the control action has the hardest real time constraints, management of the communication network cannot be achieved by this thread only. Two other thread types have been developed to handle this problem. One thread type to copy and to send the information delivered by the first thread type to the second computer, and a second thread type to retrieve the information from the packet sent by the second computer and to forward it to the first thread. The information exchanged between the threads on one side is managed by a shared memory, where every buffer is monitored and locked by the thread that accesses it. Maximum priority is given to all these three threads compared to the other tasks on the computer in order to respect the hard time constraints. The FIFO scheduling policy was selected on the system as presented in the POSIX1 standard.

The application protocol defined for the communication is based on the network stack Ethernet/IP/UDP (losses are not managed in this case by the transport protocol). Packets sent contain an identifier of the communication, a packet number, the value exchanged by the computers, a time stamp and a value of the last delay. These last two fields are necessary for measurement of the one way delay.

In order to achieve the clock synchronisation of the two PCs, a PTP daemon is implemented on each computer. The synchronisation is performed online in parallel with the control computation and packet transmission. As was presented in the Section 2.2, the controller is only aware of the delay from the sensors to the controller, but not of the delay from the controller to the process, which is only stored on the process side. For this reason, a field that corresponds to the last delay measurement stored in the process was added at the application protocol level, i.e. the controller will receive this information and will be able to use it for the Smith predictor.

The platform was used to test the performance of the Smith predictor when the delay estimate was modified online using the result from the IEEE-based, one-way delay algorithm. In the simulations, the continuous transfer functions for the process and for the controller were discretized with a sampling time of 0.2 ms. The sampling time was determined according to the rule of thumb presented in Åström and Wittenmark, (1984). The discrete controllers and the process were implemented in C-code and simulated on the platform. In the experimental prototype the dynamics were slowed down by a factor of ten due to computational issues. In other words, the time interval at which the control actions (and the process) were calculated was 2 ms. However, because the scaling has been performed also in the network part, the performance of the platform well reflected the actual situation.

The simulation result that corresponds to the fourth case obtained with the Matlab simulation presented in Figures 2 and 4 is illustrated in Figure 5. The round trip time that occurred in the network is shown on the left side of Figure 5 and the corresponding process response on the right side. According to Figure 5, the round trip time experienced by the controller is around 70 ms, which corresponds to the upper bound of 35 ms for the sensor and actuator path delays, or to the 3.5 ms that was used in the Matlab simulation.

From the process response (right panel of Figure 5), we can conclude that the combined Smith predictor and the delay measurement algorithm is able to maintain the performance level of the system. As can be seen in the figure, the response is not always identical to the response in Figure 4 for every step. The difference was due to the network delay difference between the Matlab simulations and the simulations on the experimental prototype. In actual fact, the form of the responses becomes more similar as the delay variation in both cases is reduced, and almost identical if the delay is set to constant. (The small difference in the overshoot was caused by the discretisation effect).

11

3. IMPROVED CONTROL OVER THE NETWORK USING DELAY ESTIMATIONS

3.1. Upper bound delay estimation

The upper bound delay estimation algorithm applies ideas from network calculus theory, see Cruz, (1991), Le Boudec and Thiran, (2001). The approach consists of modelling switches as a combination of basic components: multiplexers, demultiplexers and FIFO queues, as shown in Figure 6. The first step in the switch modelling involves determination of an upper bound delay for the crossing of each of the basic components. The upper bound delay over the switch is then the sum of the upper bound delays over the basic components:

$$\overline{D_{switch}} = \overline{D_{mux}} + \overline{D_{queue}} + \overline{D_{output}} \tag{5}$$

where the notation $\overline{D}$ will be used to represent the upper bound value of the delays.

Maximum delay for crossing the Ethernet switch

In the mathematical analysis, the traffic arriving at the switch, both periodic and aperiodic is modelled as a 'leaky bucket controller'. Data will arrive at the leaky rate only if the level of the bucket is less than the maximum bucket size. In network calculus theory the traffic models are represented as arrival curves and, with the assumption that the traffic follows the leaky bucket model and that the incoming rate is limited by the port capacity, these curves are affine and have the form:

$$b(t) = \min(C_{in}t \quad \sigma + \rho t) \tag{6}$$

where σ is the maximum amount of data that can arrive in a burst, ρ is an upper bound of the average rate of the traffic flow, and C_{in} is the capacity of the input port. In the same way, service curves are used to represent the minimal data processing activity of the components. Typical arrival and service curves are shown in Figure 7.

The approach used in analysing the upper bound delay for crossing a two-inputs multiplexer is shown next as introduced in the work presented in Georges, Divoux, & Rondeau (2005). The approach is based on the evolution of a specific parameter, the backlog. The backlog is the number of bits waiting in the component, and it is a measure of congestion over the component. For the arrival curves in Figure 7, the upper bound backlog occurs at time t where the following line is a maximum:

$$b_1(t) + b_2(t + L/C_2) - C_{out}t \tag{7}$$

where b_1 and b_2 are the arrival curves of stream 1 and 2 at time t, L is the maximum length of the frames, C_2 is the capacity of the import port 2, and C_{out} is the capacity of the output link.

When the upper bound backlog over the component is known, the upper bound delay over the component is then obtained by dividing the maximum backlog value by the capacity of the output link of the multiplexer.

In a FIFO m-inputs multiplexer, the delay for any incoming bit from the stream i is upper-bounded by:

$$\overline{D_{mux,i}} = \frac{1}{C_{out}} \min_k \overline{B_{mux,k}} \tag{8}$$

where $\overline{B_{mux,k}}$ is an upper-bound of the backlog in the bursty periods u_k, such that $1 \le k \le m$.

For $k = i$, the bursty period is defined by $u_i = \sigma_i/(C_i - \rho_i)$ and the backlog is upper-bounded by :

$$\overline{B_{mux,i}} = \sum_{z=1;z\neq i}^{m} \left(\sigma_z + \rho_z \left(u_i + \frac{L_z}{C_z} \right) \right) + u_i(C_i - C_{out}) \tag{9}$$

where σ_i is the burstiness of the stream i, ρ_i is the average rate of arrival of the data of stream i, L_i is the maximum length of the frames of stream i, and C_i is the capacity of the import port i. For $k \neq i$ such that $1 \le k \le m$, we have $u_k = \sigma_k/(C_k - \rho_k) - L_k/C_k$ and

$$\overline{B_{mux,i}} = \sum_{z=1; z \neq k}^{m} \left(\sigma_z + \rho_z \left(u_k + \frac{L_z}{C_z} \right) \right) + u_k (C_k - C_{out}) - \rho_i \frac{L_i}{C_i} + L_k \qquad (10)$$

For the FIFO queue the delay of any byte is upper-bounded by:

$$\overline{D_{queue}} = \frac{1}{C_{out}} \frac{(C_{in} - C_{out})}{C_{in} - \rho_{in}} \sigma_{in} \qquad (11)$$

For the demultiplexer it is assumed that the time required to route the output port is relatively negligible compared to the other delays, i.e. the demultiplexer does not generate delays.

Maximum end-to-end delays for crossing a switched Ethernet network

Computation of the upper bound end-to-end delays requires that special attention is paid to the input parameters of Equations 8-11. The maximum delay value $\overline{D}$ depends on the leaky bucket parameters: the maximum amount of traffic σ that can arrive in a burst, and the upper bound of the average rate of the traffic flow ρ. In order to calculate the maximum delay over the network, it is hence necessary that the envelope (σ, ρ) is known at every point in the network. However, as shown in Figure 8, only the initial arrival curve values (σ^0, ρ^0) are usually known, and the values for other arrival curves have to be determined. To calculate all the arrival curve values the following equations can be used:

$$\sigma_{out} = \sigma_{in} + \rho_{in} D$$
$$\rho_{out} = \rho_{in} \qquad (12)$$

For example, for the arrival curve (σ^1, ρ^1) in Figure 8 the envelope after the first switch is:

$$(\sigma^1, \rho^1) = (\sigma^0 + \rho^0 \overline{D_{switch}}, \rho^0) \qquad (13)$$

The last part of the method used to obtain the upper-bounded delay estimate is the resolution of the burstiness characteristic of each flow at each point in the network. First, the burstiness values are determined by solving the equation system:

$$
\begin{bmatrix}
a_{11} & a_{12} & \ldots & a_{1n} \\
a_{21} & a_{22} & \ldots & a_{2n} \\
\vdots & & & \vdots \\
a_{n2} & a_{n2} & \ldots & a_{nn}
\end{bmatrix}
\cdot
\begin{bmatrix}
\sigma_1 \\
\sigma_2 \\
\vdots \\
\sigma_n
\end{bmatrix}
=
\begin{bmatrix}
b_1 \\
b_2 \\
\vdots \\
b_n
\end{bmatrix}
\tag{14}
$$

and, after solving the above equation, the upper bound end-to-end delays are obtained from

$$
\overline{D}_i = \frac{\sigma_i^h - \sigma_i^o}{\rho_i}
\tag{15}
$$

where h is the number of crossed switches. For a complete discussion about the algorithm interested, readers may refer to Georges, et al. (2005).

3.2 Robust control design for delay tolerance

The information about the bounded delay obtained with network calculus is used to obtain a control law that satisfies the design specifications, and maintains the system performance even in situations during which the delay equals the upper-bound (worst-case uncertainty). The network delays are quantified in terms of an uncertainty around the nominal plant, and a mixed sensitivity based robust controller design approach is implemented.

The system that is controlled by the networked controller, the system G_p, is considered to be a combination of the nominal plant (assumed to be fixed and certain) and uncertain (unknown, but bounded) dynamical effects of the network E :

$$
G_p = G(1 + E)
\tag{16}
$$

15

It is assumed that the dynamic behaviour of G_P is described not only by a single linear time-invariant model but also by an uncertainty set Π of possible linear time-invariant models. The norm-bounded uncertainty description set Π is generated by allowing H_∞ norm-bounded stable perturbations caused by the network to the nominal plant. This will correspond to a continuous description of the model uncertainty with an infinite number of possible plants G_P in the set Π. With the weighting functions, w_I, and normalized perturbations, Δ_I, the expression for the networked controlled plant G_p becomes:

$$G_P(s) = G(s)(1 + w_I(s)\Delta_I(s)); \underbrace{|\Delta_I(j\omega)| \le 1 \forall \omega}_{\|\Delta_I\|_\infty \le 1} \tag{17}$$

The information about the upper bound network delay is utilized in creating the weighting functions w_I. The weight determination procedure consists of selecting the transfer function of a nominal plant $G(s)$, and determining at each frequency the smallest radius $l_I(\omega)$ which includes all possible plant and network combinations Π:

$$l_I(\omega) = \max_{G_p \in \Pi} \left| \frac{G_p(j\omega) - G(j\omega)}{G(j\omega)} \right| \tag{18}$$

and choosing the weight w_I such that:

$$|w_I(j\omega)| \ge l_I(\omega), \forall \omega \tag{19}$$

For the SISO system, the smallest radius $l_I(\omega)$ for all possible plant and network combinations where the actuator path delay and the sensor path delay are combined (lumped) into a single multiplicative complex uncertainty weight can be presented as:

$$l_I(\omega) = \max_{G_P \in \Pi} \left| \frac{e^{-j\omega(\delta_a + \delta_s)}G(j\omega) - G(j\omega)}{G(j\omega)} \right| = \left| e^{-j\omega(UBD_1 + UBD_2)} - 1 \right| \tag{20}$$

16

where UBD_1 is the upper bound delay for the packet transmission in the actuator path, and UBD_2 the upper bound delay in the sensor path. Thus, for the frequencies below $\pi/(UBD_1 + UBD_2)$ this relative error is upper bounded by $\left| e^{-j\omega(UBD_1 + UBD_2)} - 1 \right|$ and, from the frequency $\pi/(UBD_1 + UBD_2)$ using the Euler identity ($e^{-j\omega\pi} = -1$), we obtain 2 as the maximum of $l_I(\omega)$:

$$l_I(\omega) = \begin{cases} \left| e^{-j\omega(UBD_1 + UBD_2)} - 1 \right|; \omega < \pi/(UBD_1 + UBD_2) \\ 2; \omega > \pi/(UBD_1 + UBD_2) \end{cases} \tag{21}$$

Rational transfer functions are usually preferred for $w_I(j\omega)$. The following weight is considered to be a good approximation of the delay for the H_∞ synthesis (Wang, Lundström and Skogestad, 1994):

$$w_I(s) = \frac{(UBD_1 + UBD_2) \cdot s}{1 + (UBD_1 + UBD_2) \cdot s / 3.465} \tag{22}$$

This weight was obtained experimentally from the complex Pade approximant by increasing its values at higher frequency without changing the value at low frequencies.

After the weight selection for a SISO case, the controller synthesis problem can be solved in a straightforward manner since, in the case with one complex multiplicative perturbation, the robust performance (RP) problem

$$RP \Leftrightarrow \max_\omega(\left|w_P S\right| + \left|w_I T\right|) < 1 \tag{23}$$

can be approximated as a weighted mixed sensitivity problem, where the robust performance condition is slightly strengthened (Skogestad and Postlethwaite, 2005):

$$\left\| \begin{matrix} w_P S \\ w_I T \end{matrix} \right\|_\infty = \max_\omega(\sqrt{\left|w_P S\right|^2 + \left|w_I T\right|^2}) < \frac{1}{\sqrt{2}} \tag{24}$$

where w_P is a weight for the sensitivity function S (usually an approximate of an integrator), and T is the complementary sensitivity function. As the obtained controller is of relatively low order, there is little need for model reduction techniques. For MIMO systems the situation is different, and more advanced controller synthesis techniques such as μ synthesis are required, in addition to the model reduction techniques.

3.3. Simulations for improved control over network using delay estimates

Simulations of the robust control design in the Matlab environment

In the simulations of the NCS, the network of a real time process, a controller, and two overload traffic stations connected over a full duplex Ethernet switch were used. The overload traffic over the duplex Ethernet switch was simulated as an additional transport delay. The structure of the system is shown in Figure 9.

To calculate the upper bound delay, the initial leaky bucket values of each stream were first identified. Six messages are sent periodically. The traffic sent from the process to the controller is given by $b_1^0(t)$, and the traffic from the controller to the process by $b_2^0(t)$. The upper-bounds for these traffics are computed in order to obtain the upper bounds, UBD_1 and UBD_2. We also consider background traffic $(b_3^0(t), b_4^0(t), b_5^0(t), b_6^0(t))$ from the stations to the process and to the controller in order to overload the network:

$$
\begin{aligned}
b_1^0(t) = b_2^0(t) = \sigma_1^0 + \rho_1 t = 72 + 7200t \\
b_3^0 = b_4^0(t) = b_5^0(t) = b_6^0(t) = \sigma_3^0 + \rho_3 t = 1526 + 305200t
\end{aligned}
\tag{25}
$$

Next, the route of each stream was identified and the output burstiness equations were formulated. After solving the burstiness values, the end-to-end upper bound delay for streams 1 and 2 are:

$$
UBD_1 = UBD_2 = \frac{\sigma_1^1 - \sigma_1^0}{\rho_1} \approx 3.5\,ms
\tag{26}
$$

In the simulation of the network, the transmission delays over the network in the sensor and on the actuator side were assumed to vary randomly between zero and the upper delay value estimate. The same network delay variation as was used for evaluating the Smith predictor based compensation approach was implemented (Figure 2).

The robust controller was obtained by solving the mixed sensitivity problem presented in Equation 23. The weight presented in Equation 22 was chosen as a weighting function w_I for the complementary sensitivity function T. The frequency response of the obtained weight compared to the ideal case illustrated by Equation 20 for the case when the upper bound delay for the actuator and sensor paths, is 3.5 is shown in Figure 10.

The following approximation of an integrator was implemented as a weighting function w_P for the sensitivity function S:

$$w_P(s) = \frac{s/M + \omega_B}{s + \omega_B A} \tag{27}$$

where ω_B is the bandwidth where control is effective, M is the desired maximum peak of ω_B, and A is an arbitrary small number used to avoid numerical problems. The H_∞ optimal controller for this mixed sensitivity problem was found using the Matlab[TM] Robust control toolbox. The following controller was obtained:

$$C_r(s) = \frac{17.01 \cdot s^3 + 96.88 \cdot s^2 + 60.8 \cdot s^2 + 8.421}{s^4 + 24.53 \cdot s^3 + 125.3 \cdot s^2 + 113.6 \cdot s^2 + 0.015}$$

The performance of the system under the robust controller is illustrated in Figure 11. We can conclude from the figure that the performance criteria are achieved when the delay is increased; in fact the response of the process under the robust controller is almost completely insensitive to the increase in the delay. However, this approach provides relatively conservative results in terms of performance compared to the Smith predictor based approach, especially when the overall network delay is short.

The simulation result that reflects the second case obtained with the Matlab simulation presented in Figures 2 and 11 is illustrated in Figure 12. The round trip time that occurred in the network is shown on the left side of Figure 12, and the corresponding process response on the right side. According to Figure 12, the round trip time experienced by the controller is around 30 ms, which corresponds to the delay of 15 ms for the sensor and actuator paths or the 1.5 ms delay used in the Matlab simulations. We can conclude from the response curve of the process on the right side that the controller based on the robust control design is able to maintain the performance level of the system. The response time of the process is equal to 200 ms, and corresponds closely to the result obtained with the Matlab simulations (20 ms).

4. CONCLUSIONS

In this paper two delay compensation methods were studied, the adaptive Smith Predictor and robust control, in the framework of Networked Control Systems while considering the protocol specifications. The switched Ethernet was used as the communication protocol to evaluate the performance of the methods for networked control.

In the first case, the coupling between QoS parameters and the controller design was based on delay measurements and the adaptive Smith Predictor. The delay measurement was obtained via a synchronisation step of all the computers' clocks by the IEEE 1588 standard. The delay variations were continuously monitored, and the information was used by the Smith Predictor.

In the second case, robust control theory was combined with network calculus theory. A network model was proposed to describe both the internal network properties and the network traffic. The traffic over the switch was represented as minimal service curves and maximal traffic curves, respectively. These curves were used in the network calculus theory to estimate bounded end-to-end delays, which were then translated into uncertainties in the robust controller design.

Finally, numerical experiments with Matlab simulations and runs on an experimental prototype were presented. The results showed that the proposed control schemes were able to ensure robustness of stability and performance of the network controlled system.

REFERENCES

Boudec, Le, J.-Y., & Thiran P. (2001). Network *Calculus, A Theory of Deterministic Queueing Systems for the Internet*. In: Lecture Notes in Computer Science. NJ: Springer Verlag.

Bauer, P. H., Sichitiu, M. L. & Premaratne, K., (2001). On the nature of the time-variant communication delays. In *IASTED Conference on Modeling, Identification and Control, Innsbruck*, Austria, pp. 792–797.

Cruz, R. (1991). A Calculus for Network Delay, Part I: Network Elements in Isolation. *IEEE Transactions on Information Theory*, 37, 114–131.

Gaderer G., Loschmidt, P., & Sauter, T. (2005). IEEE 1588 Real-time networks with hybrid master group enhancements. In *4th International Workshop on Real.Time Networks (RTN'05)*, Palma de Mallorca, Spain.

Georges, J.-P., Divoux, T., & Rondeau, E. (2005). Confronting the performances of a switched Ethernet network with industrial constraints by using the Network Calculus. *International Journal of Communication Systems*, 18(9), 877-903.

Hu, Z.S., & Zhu, Q.X. (2003). Stochastic optimal control and analysis of stability of networked control systems with long delay. *Automatica*, 39, 1877–1884.

Huo, Z., Fang, H. & Ma, C. (2004). Networked control system: state of the art, In *Fifth World Congress on Intelligent Control and Automation, WCICA2004,* 2, 1319 – 1322.

Jasperneite, J., Shebab, K., & K. Weber (2004). Enhancement to the time synchronisation standard IEEE-1588 for a system of cascaded bridges. In *5th IEEE International Workshop on Factory Communication Systems (WFCS'04)*, Vienne, Austria, 39-44.

Juanole G. & M. Mouney. Real time distributed systems: QoS and impact on the performance of process control applications. In *17th International Symposium on Mathematical Theory of Networks and Systems (MTNS'06),* July 2006, Kyoto, Japan, 1739-1746.

21

Kim, D.S., Lee, Y.S., Kwon, W.H., & Park, H.S. (2003). Maximum allowable delay bounds of networked control systems. *Control Engineering Practice*, 11, 1301–1313.

Li, S., Zhi, W., & Youxian S. (2002). Fundamental problems of networked control system from the view of control and scheduling, In *28th Annual Conference of the Industrial Electronics Society Conference,(IECON02),* 3, pages 2503- 2508.

Mills, D.M. (1991). Internet Time Synchronisation: the network time protocol. *IEEE Transactions on Automatic control*, 49(10), 1650-1667.

Morari, M. & Zafiriou, E. (1989). *Robust Process Control*, NJ: Prentice Hall.

Nilsson, J., (1998). *Real-time control systems with delays.* Ph.D. dissertation, Lund Institute of Technology.

Nesic, D., & Teel, A.R. (2004). Input-to-state stability of networked control systems. *Automatica*, 40(12), 2121-2128.

Richard, J.-P. (2003). Time-delay systems: an overview of some recent advances and open problems. *Automatica*, 39, 1667 – 1694.

RTAI, the RealTime Application Interface for Linux, rtai.org.

Skogestad, S. & Postlethwaite, I. (2005). *Multivariable feedback control - Analysis and design,* 2nd Edition, West Sussex: Wiley.

Tang, P. L., & de Silva, C.W. (2006). Compensation for transmission delays in an Ethernet-based control network using variable-horizon predictive control. *IEEE Transactions on Control Systems Technology*, 14(4), 707 – 718.

Tipsuwan, Y., & Chow, M.-Y. (2003). Control methodologies in networked control systems. *Control Engineering Practice*, 11, 1099–1111.

Tipsuwan, Y., & Chow, M.-Y., (2002). Gain adaptation of networked mobile robot to compensate QoS deterioration. In *28th annual conference of the IEEE industrial electronics society (IECON 02)*, Sevilla, Spain, 3146–3151.

Wang, Z.-Q., Lundstrom, P., & Skogestad, S., (1994). Representation of uncertain time delays in the H infinity framework. *International Journal of Control*, 59, 627-638.

Walsh, G.C., Belfiman, O., & Bushnell, L.G. (2002). Error encoding algorithms for networked control. *Automatica*, 38(2), 261-267.

Walsh, G.C., Ye, H., & Bushnell, L.G. (2002) Stability analysis of networked control systems. *IEEE Transactions on Control Systems Technology*, 10(3), 438-446.

Yang, T.C. (2006) Networked control system: a brief survey. In *IEE Proceedings of Control Theory and Applications*, 153, July 2006, 403-412.

Yue, D., Han Q.-L., & Lam, J. (2005). Network-based robust H∞ control of systems with uncertainty. *Automatica,* 41(6), 999-1007.

Yu, M., Wang, L. Chu, T, & Hao, F. (2004). An LMI approach to networked control systems with data packet dropout and transmission delays. In *43rd IEEE Conference on Decision and Control, CDC2004,* 4. December 2004, 3545 – 3550.

Yue, D., Han, Q.L., & Peng, C. (2004). State feedback controller design of networked control systems. *IEEE Transactions on Circuits Systems,* 51, 640–644.

Yook, J.K., Tilbury, D.M., & Soparkar, N.R. (2001). A design methodology for distributed control systems to optimize performance in the presence of time delays. *International Journal of Control*, 74(1), 58-76.

Zhang, L., & Hua-Jing, F. (2006). A novel controller design and evaluation for networked control systems with time-variant delays. *Journal of the Franklin Institute*, 343(2), 161-167.

Astrom K. J. & Wittenmark, B. (1984). *Computer Controlled Systems-Theory and Design*. Prentice-Hall, NJ: Englewood Cliffs.
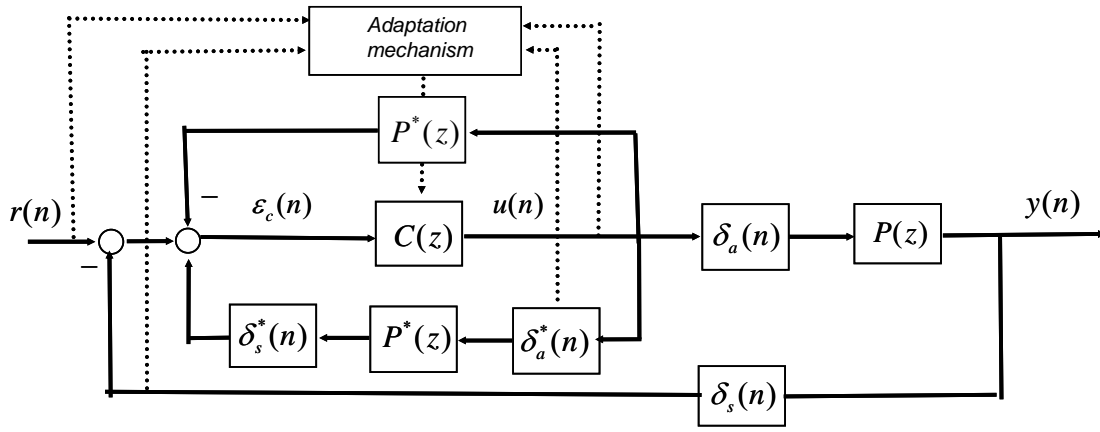
Figure captions

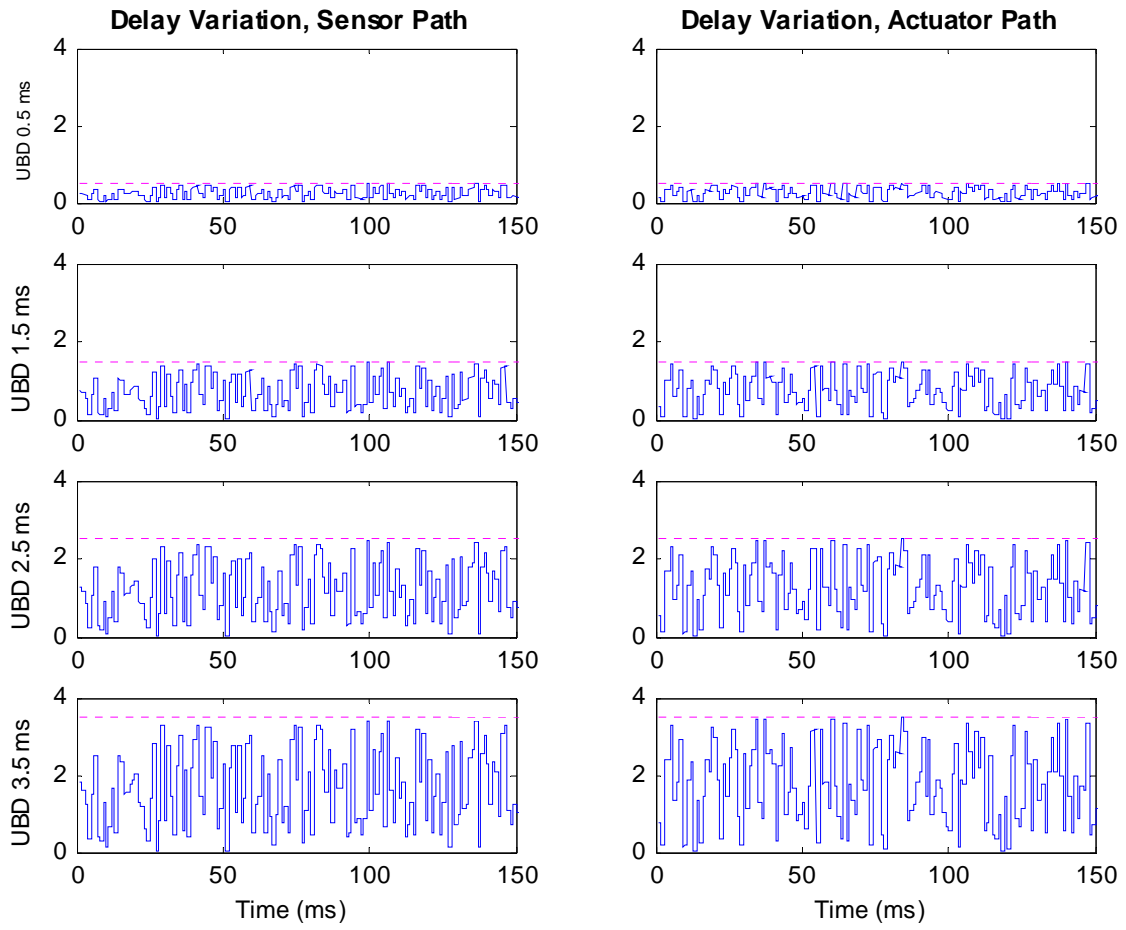Fig. 1. An adaptive Smith predictor for compensation of the sensor path network induced delay.

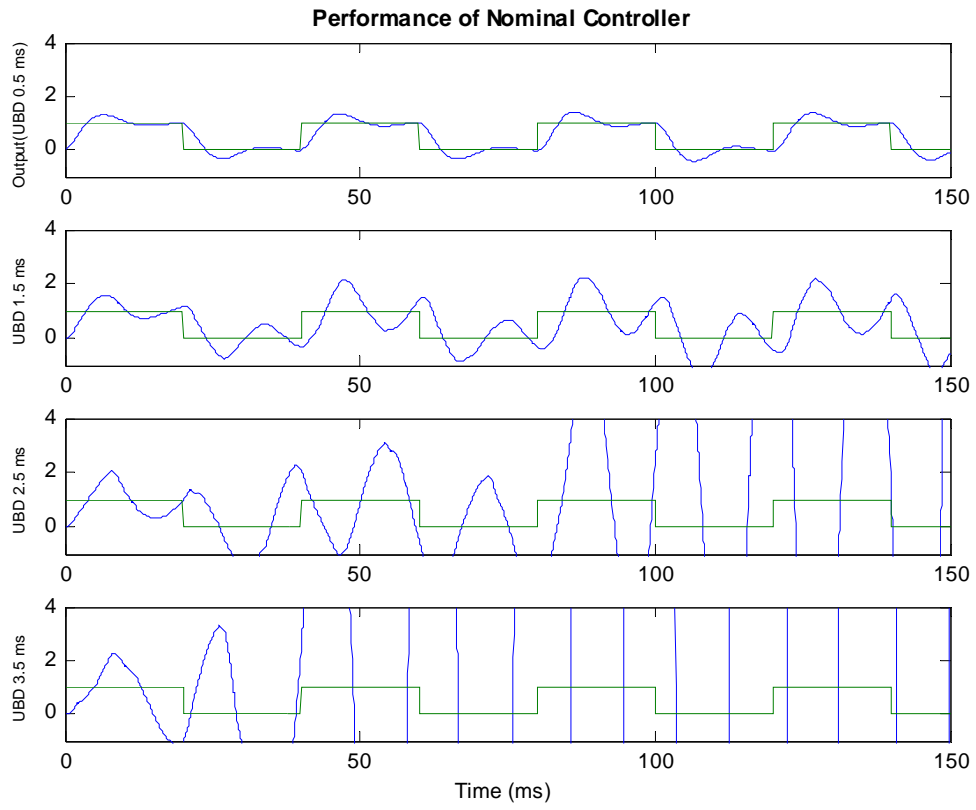Fig. 2. Evolution of the sensor and actuator network delays.

Fig. 3. Performance of the nominal (IAE optimised) PI controller for the nominal case, and for the cases where the delay increases.
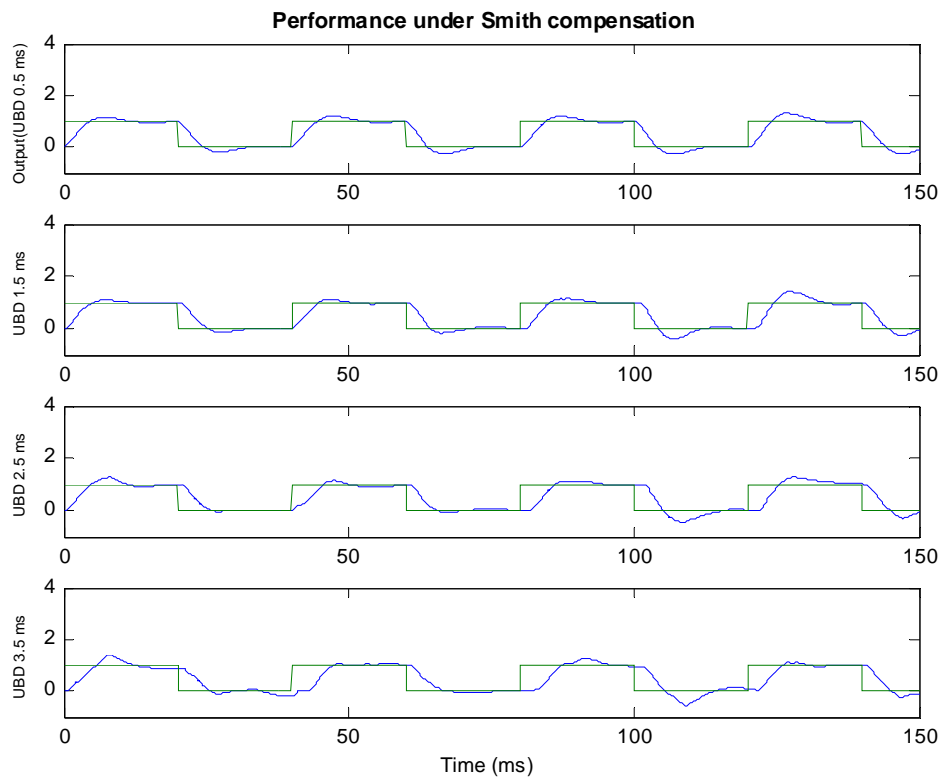
Fig. 4. The performance of the Smith predictor compensation scheme when the delay varies between zero and the upper bound delay.
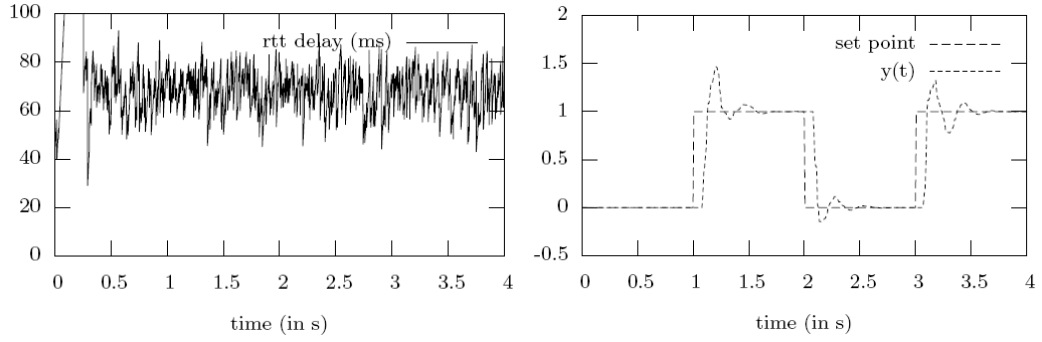
Fig. 5. Performance of the Smith predictor with the online delay parameter under a varying network delay.
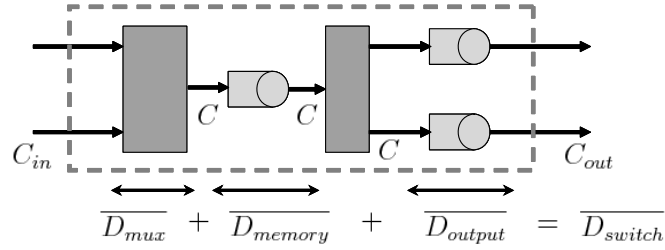
Fig. 6. Model of a 2-port switch in a full duplex mode based on shared memory and a cut-through management.
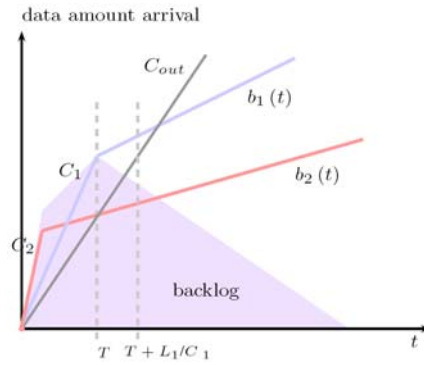
Fig. 7. Arrival and service curves and backlog evolution inside the two-input FIFO multiplexer.
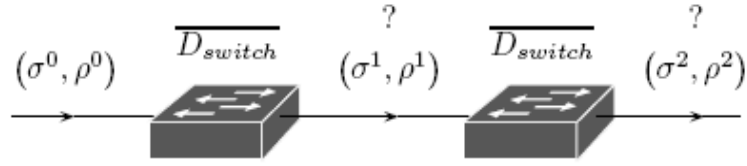
Fig. 8. Burstiness along a switched Ethernet network.

Fig. 9. Structure of the network.

Fig. 10. Weight selection for the network induced delay ($UBD_1 = 3.5$ $UBD_2 = 3.5$).

Fig. 11. Performance of the robust controller during the increased delay.

Fig. 12. Performance of the robust controller under a varying network delay.

Fig. 1. An adaptive Smith predictor for compensation of the sensor path network induced delay.

Fig. 2. Evolution of the sensor and actuator network delays.

Fig. 3. Performance of the nominal (IAE optimised) PI controller for the nominal case, and for the cases where the delay increases.

Fig. 4. The performance of the Smith predictor compensation scheme when the delay varies between zero and the upper bound delay.

Fig. 5. Performance of the Smith predictor with the online delay parameter under a varying network delay.

$$\overline{D_{mux}} \quad + \quad \overline{D_{memory}} \quad + \quad \overline{D_{output}} \quad = \quad \overline{D_{switch}}$$

Fig. 6. Model of a 2-port switch in a full duplex mode based on shared memory and a cut-through management.

Fig. 7. Arrival and service curves and backlog evolution inside the two-input FIFO multiplexer.

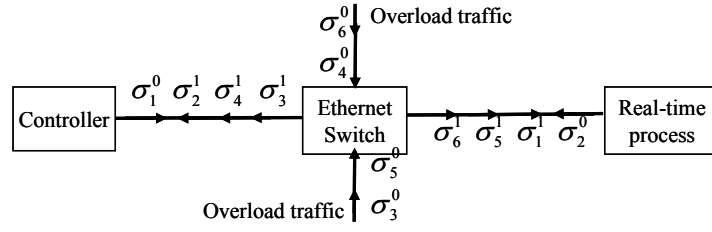Fig. 8. Burstiness along a switched Ethernet network.
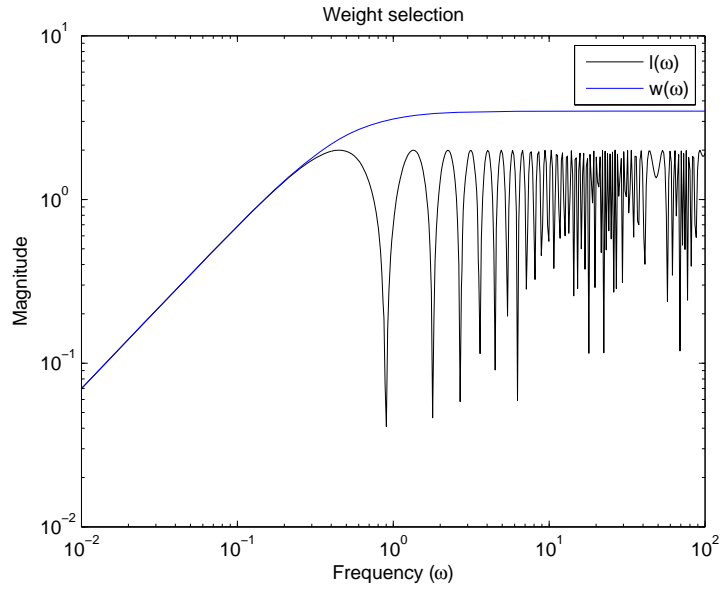

Fig. 9. Structure of the network.

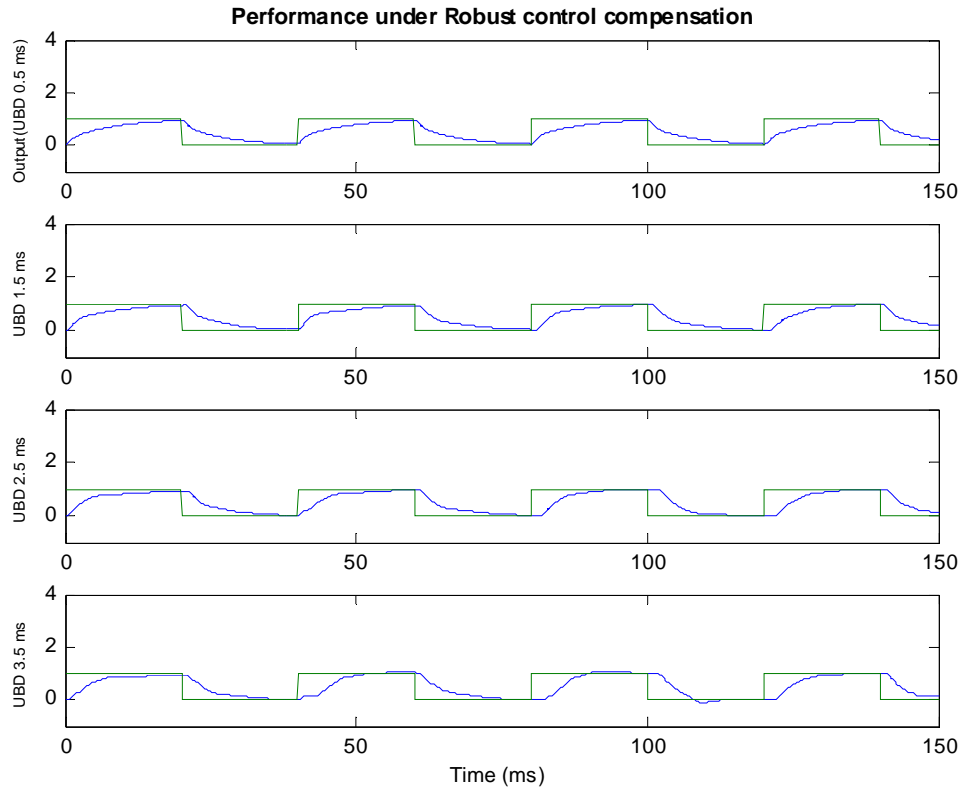Fig. 10. Weight selection for the network induced delay ($UBD_1 = 3.5$ $UBD_2 = 3.5$).

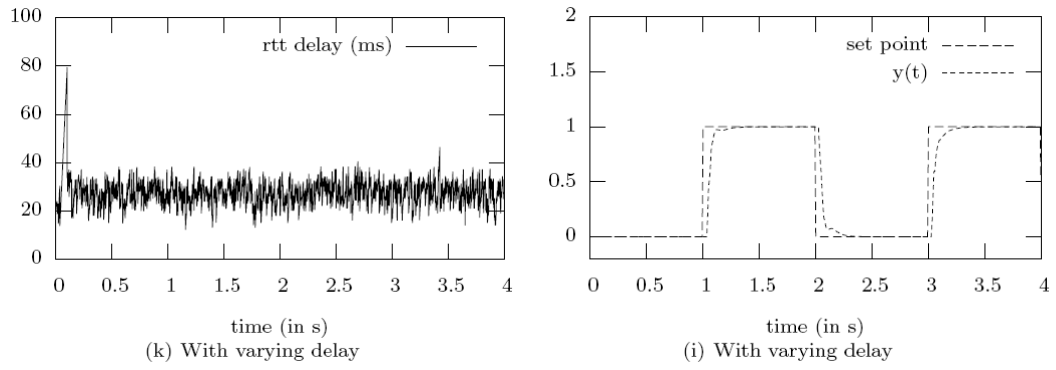Fig. 11. Performance of the robust controller during the increased delay.

Fig. 12. Performance of the robust controller under a varying network delay.