Sensors **2012**, *12*, 16591-16613; doi:10.3390/s121216591

OPEN ACCESS

sensors

ISSN 1424-8220

www.mdpi.com/journal/sensors

Article

Network Adaptive Deadband: NCS Data Flow Control for Shared Networks

Miguel Díaz-Cacho [1,*], Emma Delgado [2], José A. G. Prieto [2] and Joaquín López [3]

[1] School of Computer Engineering, Computer Network Engineering, University of Vigo, 32004 Ourense, Spain

[2] School of Industrial Engineering, Control Engineering, University of Vigo, 36310 Vigo, Spain; E-Mails: emmad@uvigo.es (E.D.); jagprieto@uvigo.es (J.A.G.P.)

[3] School of Industrial Engineering, Robotics, University of Vigo, 36310 Vigo, Spain; E-Mail: joaquin@uvigo.es

* Author to whom correspondence should be addressed; E-Mail: mcacho@uvigo.es; Tel.: +34-986-812-222; Fax: +34-986-814-014.

Received: 6 October 2012; in revised form: 23 November 2012 / Accepted: 23 November 2012 / Published: 3 December 2012

Abstract: This paper proposes a new middleware solution called Network Adaptive Deadband (NAD) for long time operation of Networked Control Systems (NCS) through the Internet or any shared network based on IP technology. The proposed middleware takes into account the network status and the NCS status, to improve the global system performance and to share more effectively the network by several NCS and sensor/actuator data flows. Relationship between network status and NCS status is solved with a TCP-friendly transport flow control protocol and the deadband concept, relating deadband value and transmission throughput. This creates a deadband-based flow control solution. Simulation and experiments in shared networks show that the implemented network adaptive deadband has better performance than an optimal constant deadband solution in the same circumstances.

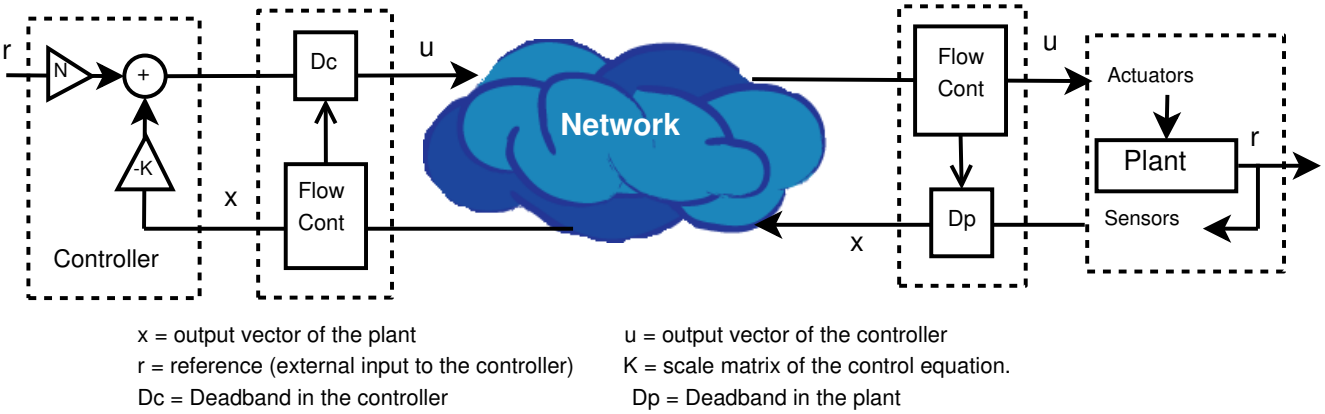
Keywords: deadband control; NCS; tele-operation; network protocols; sensors/actuators

---

1. Introduction

Control systems are basically composed by a controller and a plant, where the controller receives the information about the state of the plant, computes the new action and sends it to the plant. The controller uses the required hardware and software to obtain the control law. The plant uses actuators to apply the received actions from the controller, and uses sensors to measure the state of the plant (Figure 1). Sensors and actuators were traditionally connected to the control system through a dedicated network. If the control loop between controller and plant is closed through a shared digital communication network such as Internet, stability cannot be guaranteed because of variable communication delays and data packet loss. But the use of Internet as intermediate network has some advantages, such as the high availability, testability, simple installation as well as low cost. Networked Control Systems (NCS) cover tele-operation, tele-robotics and network-based control. The applications are in traffic control, robotics, domotics, tele-operation, in general Internet of things and can be extended to serve as remote back-up controllers for some critical plants in case of a disaster.

Figure 1. NAD basic scheme.



The adoption of TCP/IP protocol stack is mandatory to send packets across the Internet, and researchers should work with this restriction. Major contributions of the researches in NCS over Internet are in the controller-design, in designing new transport protocols (in layer 4 of the TCP/IP stack) or in the local segment of the extremes of the NCS. The Internet constraints for NCS (such as delay and packet losses) are basically caused by the queue size of the intermediate routers and the capacity of the network link between both extremes of the NCS. Variable delay and variable packet losses are mainly due to the unpredictable network occupation of the background traffic and the transmission rate of the NCS [1]. The transmission rate is closely related with the sampling period of the NCS and the network constraints [2]. A big effort is made to calculate the limits of the sampling periods that maintain the NCS stability, such as the *Maximum allowable sampling period* (or MASP) of an NCS (see the work of Nesić *et al*. [3] and references cited therein).

In our previous study [4], we conclude that an adequate transport scheme can solve the transmission rate problem in Internet-based NCSs. The transport scheme should have flow-control, best-effort and fairness features, as well as an NCS data priority system. Some transport protocols meet the first three

specifications, but compete with control data scheduling strategies implemented in the control design, to solve the variable delay and packet losses problem.

In this work, a general scheme to combine control and network status is presented. The general scheme is implemented in a middleware solution that uses a transport protocol that satisfies the aforementioned features, in combination with the deadband concept that works as the NCS data priority system (see Figure 1 for a basic Linear Quadratic Regulator LQR control system). However, a more generic adaptive deadband solution may be implemented with other adaption law, mainly to optimize the power consumption, highly recommended in sensor networks.

The starting point to implement the proposed middleware is a properly designed NCS in the expected network scenario, taken into account delays and packet losses. Several works have been made to design such an NCS, using different strategies that will be described in Section 2.

The concept of deadband applied to NCS was initially studied by Otanez *et al.* [5]. NCS deadband focuses on the relative value of NCS data to be transmitted. In deadband sampling solution, a data is sent only if its difference with the last data sent is bigger than a defined value. This value is called *deadband*.

$$DFi_{send} = \begin{cases} 1 \text{ if } \|x_{ls} - x\| \geq \delta \\ 0 \text{ if } \|x_{ls} - x\| < \delta \end{cases} \tag{1}$$

A deadband transmission filter can be formulated as in Equation (1), where δ is the deadband value, x_{ls} is the last data sent and x is the candidate data to be sent. In Figure 1, D_c and D_p are the deadband filters for the controller and plant sides respectively.

Constant deadband solution attempts to minimize the throughput while maintaining acceptable performance for an NCS. However, a constant deadband that minimizes the throughput may not be suitable for networks with variable congestion scenarios, basically due to two remarks:

Remark 1 *If the network is congested, a minimum transmission throughput can result in a reception throughput below the minimum required to maintain stability, due to packet loss.*

Remark 2 *If the network is not congested, increasing the transmission rate improves the performance of the NCS, but the rate should stay below the available capacity on the network, because the delay increases due to saturation of the queues of the intermediate routers [4].*

The packet selection process that allows the deadband solution is suitable as a prioritization system for NCS data on shared networks, but does not take into account the state of the network to maximize the performance of NCS. This paper proposes a deadband modulation system taking into account the previous mentioned remarks. To module the deadband, a TCP-friendly flow control algorithm is used as transport protocol, and the complete middleware solution is called TCP-friendly NAD.

A flow is TCP-friendly if its arrival rate is approximately equal or slower than the rate of a conformant TCP connection under the same network conditions [6].

$$\frac{1}{IPG} \equiv th|_{pkt/sec} = \sqrt{3}/(RTT\sqrt{2p}) \tag{2}$$

The conformant TCP throughput without retransmission follows Equation (2) for fixed packet size. In Equation (2), p represents the probability of packet losses and RTT the round trip time. IPG is

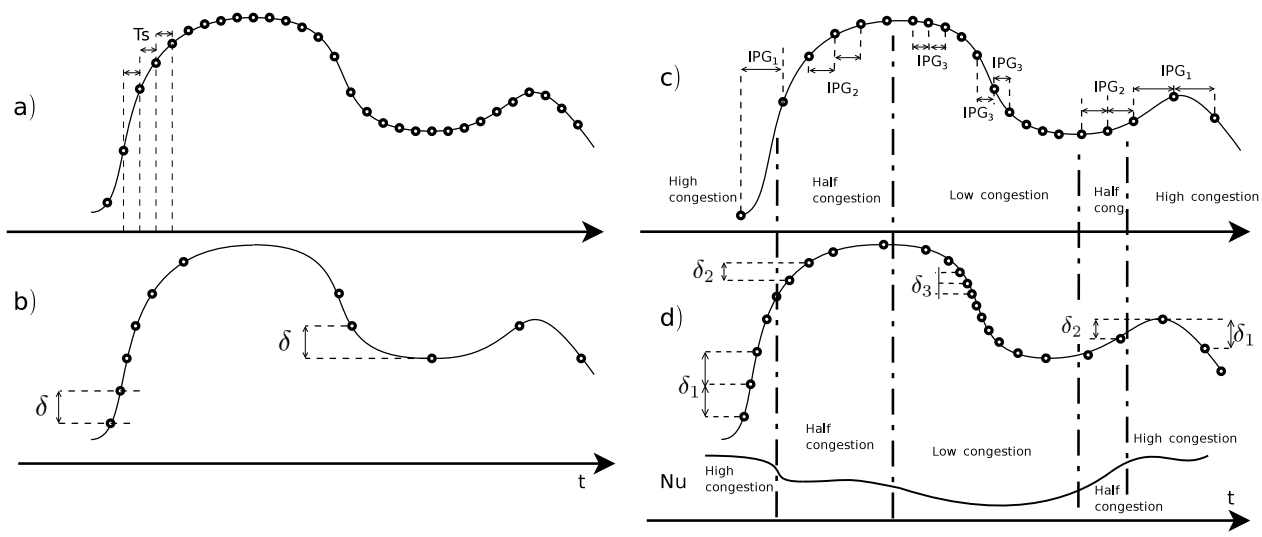
an acronym for Inter-Packet Gap, which represents the time between packets sent, and th stands for throughput, in packets per second.

TCP-friendly requirements can be mandatory for new Internet protocols in order to share the Internet with good results. This feature can be exploited by other shared networks, and could be necessary for multi-agent systems.

The benefits of the proposed transport scheme can be seen in Figure 2. This figure shows a comparison between commonly used transmission schedulers and the NAD, where the points represent a transmission event.

- Figure 2(a) represents a transmission scheduler for a constant sampling rate with period T_s, but without transport flow control or deadband.
- Figure 2(b) represents a transmission scheduler for a constant deadband sampling with a deadband value of δ according to Equation (1).
- Figure 2(c) represents a transmission scheduler based on a TCP-friendly transport protocol, with different network congestion scenarios, where N_u represents the network use. In this figure, the IPG depends on the transport flow control algorithm, that generally depends on the network usage: with high congestion situations, the packet send period or IPG is high (IPG_1), and it decreases in low congestion scenarios ($IPG_1 > IPG_2 > IPG_3$).
- Figure 2(d) represents the proposed network adaptive deadband solution, a deadband scheduler based on the network usage N_u. In this subfigure the value of the deadband depends on N_u. It will have a larger deadband (δ_1) in a high congestion network environment, and a smaller deadband (δ_3) in a low congestion situation. Note that $\delta_1 > \delta_2 > \delta_3$.

Figure 2. Points representing data transmission events for diverse transmission schedulers. (**a**) Fixed sampling rate with period T_s; (**b**) constant deadband with value δ; (**c**) variable transmission rate with the Inter-Packet Gap (IPG) value depending on the transport flow control algorithm; (**d**) network adaptive deadband (NAD) tracking network usage N_u.



The main reason for using this deadband solution rather than the constant deadband, amplitude relative deadband [7] or pure transport protocol is that the reduction of data sent caused by the constant

deadband is not always required on shared networks (see remark 2). Besides, this reduction is at the expense of an increment in the NCS error or a decrement in the psychophysical perception of the tele-operated system. At the same time, packet loss in the case of constant deadband can cause instabilities if the transmission scheduling rule is in the low limit of acceptable throughput (see remark 1). A pure transport protocol such as the trinomial algorithm, adapted to the network scenario, can share the network with fairness, but it does not guarantee any of the NCS requirements to maintain stability.

On the other hand, a schema that combines the two aforementioned solutions can share the network with fairness requirements, and at same time, they can maintain the best available performance in a variable network scenario.

The paper is organized as follows. Motivation, originality and introduction to preliminary concepts such as TCP-friendly and deadband are advanced in this section. An overview of the related work is given in Section 2. Section 3 shows a general scope of Network Adaptive Deadband and defines the different elements involved in the scheme. The proposed TCP-friendly based deadband control is presented in Section 4 and its application to a tele-operated gantry-crane is showed in Section 5. Simulation and experimental results that compare constant deadband and adaptive deadband in a shared network are presented in Sections 6 and 7 respectively. Finally, the conclusions and future work are described in Section 8.

2. Related Work

Network and control system relationship in an Internet based NCS or in tele-operation systems is usually limited to adapt the controller design to several network variables (such as the delay constraint, the delay variation and the number of consecutive losses).The common solutions for the control system use the network as a passive element, and in the case of Internet, solve the transport problem using the UDP transport protocol. On the other hand, common network solutions for NCS investigate the creation of new protocols and data flow control algorithms. In this case, control systems are used as applications without considering the contents of the data to be transmitted. Modern techniques have emerged with the generic name of co-design that try to model and integrate the network in the control system. Some authors refer generically to this topic as Network Based Control (NBC). There are different solutions to this problem that we classify in the following four approaches.

2.1. Packet and Deadband Based Control

This line scans the contents and structure of *NCS data packets* to optimize the network performance and efficiency [8,9]. These methods allow to work on predictive control, because the controller (or the plant) can send actual, past and future *NCS data*, in the same packet. Thus, if the data is lost or not sent, the other extreme has predictive information received in previous packets [1,10].

Deadband control applied to NCS focuses initially on modeling Networked Control Systems [5] analyzing the performance effects of limiting packet transmission to reduce the network traffic. This makes the communication medium more efficient, and time delays are minimized.

In [7] a passivity approach for deadband control is made for haptic tele-operation, and later, perception Weber-inspired deadband approaches were made [11]. In [2] a study about the relationship

between the sampling rates of a control system, and the transmission rates of a local communication network, is made. A similar solution to the one proposed in this paper has been presented in [12]. However, the work in [12] was made specifically for wireless sensor networks; in [13], the predictive deadband allows to study the influence of different factors, such as the CPU load, on the efficiency of deadband sampling. These works focus the deadband study to improve the performance of sensor networks, optimizing the network and/or the power consumption of the whole system.

2.2. Codesign of Control and Transmission Rate

This line of work studies protocols, middleware systems and control theory. The work of Nesic and Teel [3,14] focuses on the transmission rate (or sampling) problem of nonlinear systems.

In event-triggered control, the control input is updated when a certain condition of the state of the plant is met. For example the work of P. Taboada and M. Mazo in [15,16] and the references therein present very interesting results in modeling sampling rate (or transmission rate) and control system. In self-triggering control [17], the controller calculates the next sampling-time as a function of the state of the plant. The target of the triggering control is to minimize the energy requirements, without compromise the performance of the control system.

2.3. Codesign of Control and Gain

In [18] a middleware system that modifies control gains in PID controllers is studied. The system takes into consideration the network status. The controller output algorithm depends on the remote plant and the controller global configuration. The state of the network is obtained by sending probe packets. More recently, in [19] a network delay-dependent gain scheduling law is used on an Ethernet shared network.

2.4. NCS Specific Transport Protocols

In the transport layer, there are some solutions related to the UDP protocol, developing flow-control and TCP-friendly features, such as TFRC [20], DCCP [21] or the trinomial algorithm [22]; Recently, Wirz *et al.* have worked in transport protocols for robotics, such as the Bidirectional Transport Protocol (BTP) [23]. BTP is an end-to-end flow control protocol that minimizes the round trip time while maximizing the transmission rate. The Tele-operation Data Transport Protocol (TDTP) [9] focuses on the bidirectionality of the NCS data flow to improve the efficiency of the transmission rate. Bidirectionality of the NCS systems in transport protocols is used to determine the network status in the packet header of the reverse flow.

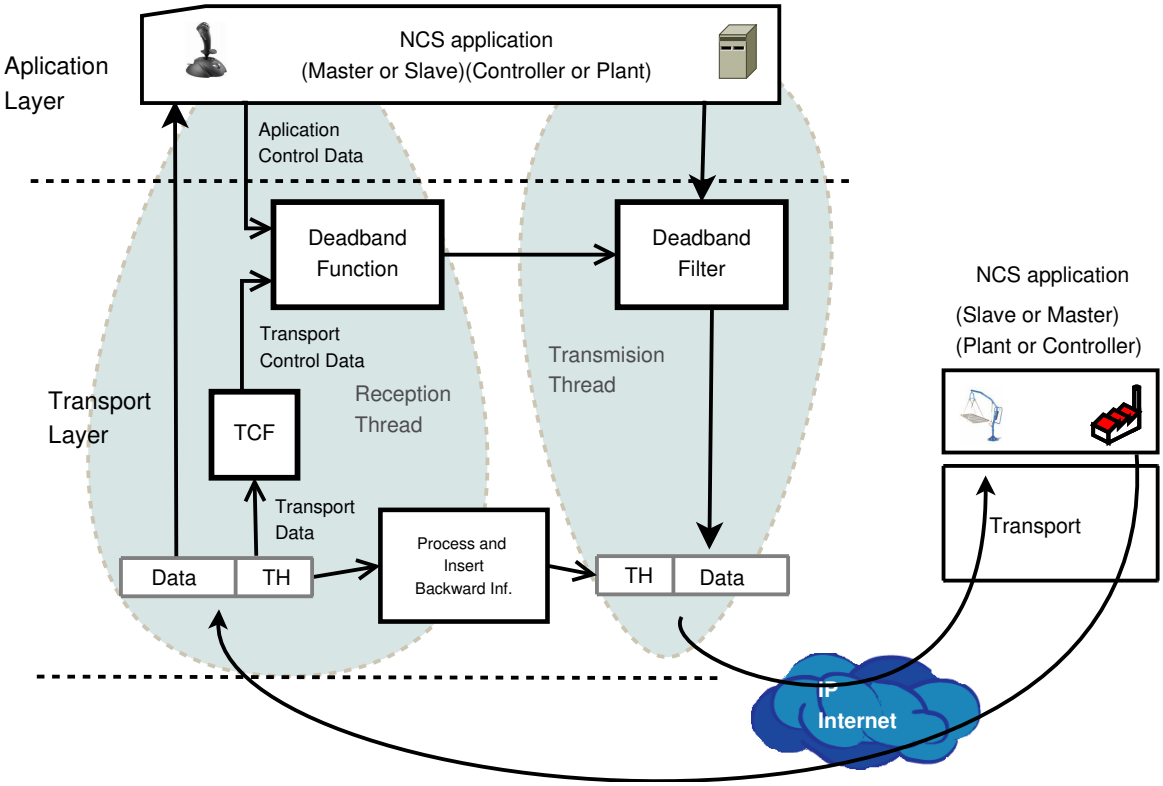
In the Link Layer of the OSI model, CAN, HART or EtherCat, and the technologies like Ethernet, Wifi 802.11, bluetooth or ZigBee are suitable as the local segment of the Internet NCS connection with an appropriate gateway to the Internet link.

To our knowledge, no work has been done to relate deadband sampling and transport flow control algorithms.

3. Network Adaptive Deadband Proposed Scheme

The solution presented here is a middleware that combines transport and application layer functionality. The application layer will deliver NCS status data to the transport layer and the transport layer looks to the application data fields. Figure 3 shows a detailed box diagram of the functionality and data flow between the layers. Network status is calculated using a bidirectional transport architecture, where the network information is inserted in the transport header.

Figure 3. Network Adaptive Deadband general middleware.



The middleware uses two threads. The *Reception thread* delivers application data directly to the NCS. The *Transport Data* (TD) of the *Transport Header* (TH) are processed in the transport layer by the *Transport Control Function* (TCF), where the flow control algorithm is implemented. The TCF generates *Transport Control Data* (TCD) that includes processed information regarding the network status. This TCD is provided to the *Deadband Function* (DF), which relates throughput and deadband. Optionally, application control information such as error, *Average Error* (AE) or predictive data, can be used as DF input. These data are called *Application Control Data* (ACD).

The *Transmission thread* filters NCS data with the *Deadband Filter* (DFi), which uses the deadband value obtained dynamically by the DF in the reception thread. Deadband filter is implemented using the sending event function defined in Equation (1). The proposed adaptive deadband formulates the deadband function as in Equation (3), where th, RTT, p are the throughput, round-trip time and packet

losses respectively ; these variables are used as TCD data. Average error (AE) or instantaneous error ($error$) are proposed ACD data.

$$\begin{aligned} \delta &= f(TCD, ACD) \\ \delta &= f((th, RTT, p), (AE, error)) \end{aligned} \tag{3}$$

4. TCP-Friendly Adaptive Deadband

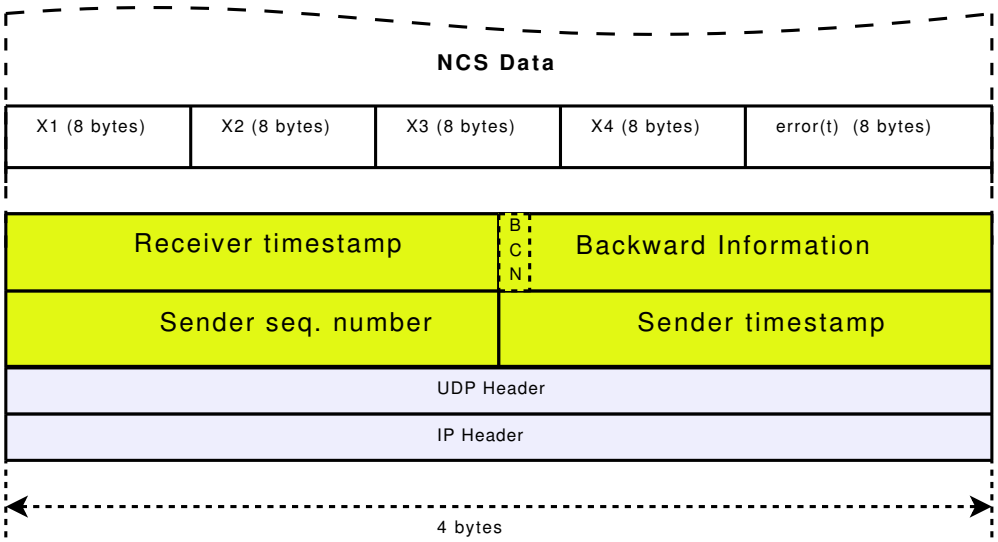
In this section, the transport control scheme and the deadband function for the TCP-friendly Adaptive Deadband are described.

4.1. Transport Control Scheme

The implemented transport scheme focuses on the flow control algorithm and on the use of bidirectionality in NCS data flows.

The transport control function (TCF) implemented is the trinomial flow control algorithm [22]. The trinomial algorithm calculates the IPG and differentiates increment and decrement transmission rate intervals. To compute the transmission rate, the algorithm uses three parameters (α, β, γ), and the RTT. The RTT is measured using timestamps. The sender selects an increment or decrement interval depending on a backward packet loss tag. This tag, called Backward Congestion Notification (BCN), is inserted in the reverse flow by the receiver and received by the sender. The details of this tag and the way it is included in a bidirectional data flow, are explained in our previous work [9,24].

Figure 4. Bidirectional transport header for trinomial algorithm and data structure for tele-operated gantry crane.



In the direct flow between sender and receiver, the receiver detects the loss of packets by computing a sequence number and reports it to the sender setting the BCN tag to one ($BCN = 1$). If there are not packet losses, the BCN tag is set to zero ($BCN = 0$). Figure 4 shows the transport header format

of the implemented trinomial algorithm, with the BCN tag, the sender sequence number and sender and receiver timestamps.

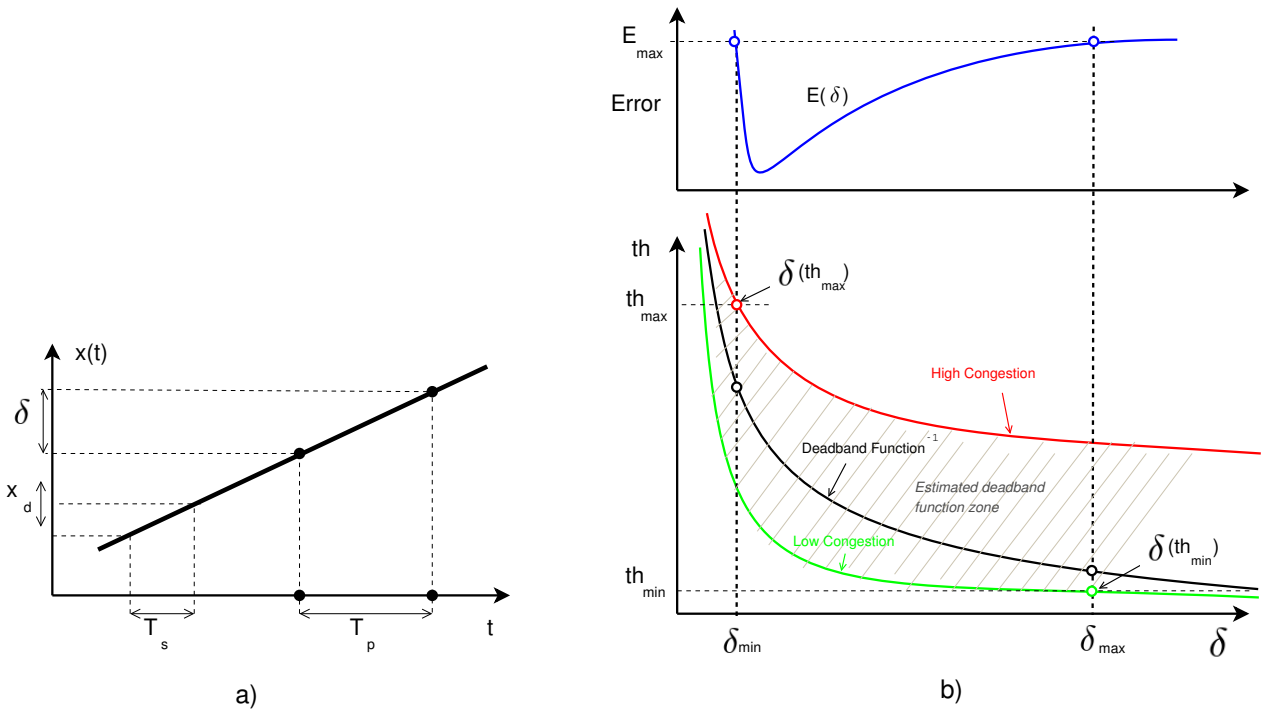
$$IPG_{i+1} = \begin{cases} \dfrac{IPG_i \cdot W}{IPG_i + W} \text{ if } BCN = 1 \\ \\ \dfrac{IPG_i}{\beta} \text{ if } BCN = 0 \end{cases} \tag{4}$$

Equation (4) shows the way the transmission rate is decremented (with $BCN = 1$) or incremented (with $BCN = 0$), where $W > 0$ is a variable depending on γ, α and RTT. Parameter $\beta < 1$ is an initial parameter. To know how this algorithm is determined and the details of how it works, see the work of Liu *et al*. [22].

4.2. Deadband Function

In this work, we have chosen as deadband function the relationship between transmission rate (or throughput th) and deadband ($\delta = f(th) \equiv \delta_{th}$). The implementation of the trinomial algorithm presented in Section 4.1 meets the TCP-friendly requirements, and gives a way to calculate the throughput th by calculating the inter packet gap $IPG = 1/th$.

Figure 5. (**a**) Linear approach of $th(\delta)$; (**b**) Throughput and error versus deadband.



Obtaining an analytical solution to the relationship between th and deadband ($\delta = f(th) \equiv \delta_{th}$) is complex and can be attempted using rate-distortion theory [25] or a simple linear approach. From the rate-distortion theory, the throughput-deadband relationship is a continuous, monotonically decreasing convex function, and tends to have very similar forms in real implementations. Figure 5(b) shows these

relationships. Experiments, simulations and related literature [1,2,5,7,25–27] show that this relationship always has that form.

The purpose of this work is not to determine an exact representation of δ_{th} but a deadband function easy to implement, maintaining the lower and upper bound of the transmission rate to achieve the performance of the control system. A similar simulated method for Wireless Sensor Networks, was used in the work of Xia *et al.* [12]. Lower and upper bounds of the throughput can be obtained by diverse methods or analysis, such as the ones presented by Tatikonda *et al.* [28]. From [4], in Internet with TCP congestion, the upper limit of the transmission rate should be smaller than without congestion (th_{max}); with UDP congestion, the lower limit should be bigger than without congestion (th_{min}). However, the DF has the form in Figure 5(b), where the bounds are also represented.

The linear approach of the DF can be obtained with a basic analysis for a linear and uniform output signal ($x(t)$) in one extreme of the NCS with high sampling rate (sr), as seen in Figure 5(a), where $T_p \equiv IPG$ is the transmission period, $T_s \equiv 1/sr$ is the sampling period, δ is the fixed deadband value and x_d is the difference between consecutive sampled data. In this case, these parameters satisfy Equation (5).

$$\frac{1}{T_p} = th = \frac{x_d}{\delta} \cdot \frac{1}{T_s} \tag{5}$$

However, in packet based NCS, the output is a discrete function. Therefore, in order to satisfy $\lim_{\delta \to 0} th = sr$, the previous equation has to be changed to:

$$th = \frac{x_d}{\delta + x_d} \cdot sr \tag{6}$$

In dynamic environments, the output signals are nonlinear and discontinuous, and the DF δ_{th} changes dynamically with the environmental states. To reflect this non-linearity and discontinuities in output signals, the transmission rate is estimated as an average of the different linear functions adding a new parameter n_1 in Equation (6). The transmission rate ($th(\delta)$) is then characterized as a function of the system and controller dynamics, reference trajectory, and the chosen deadband [2] in a simple way.

$$th(\delta) = \frac{c}{\delta^{n_1} + c} \cdot sr \tag{7}$$

In Equation (7), the parameter c is directly related with the average of the modulus of the output signal variation (x_d), and should always have a positive value. The parameter n_1 is obtained by simulation and is in the range of $0 < n_1 < 1$.

The function δ_{th} should be have a maximum deadband δ_{max} and a minimum deadband δ_{min} to maintain acceptable performance. The maximum deadband depends on the minimum acceptable throughput $th_{min} = 1/IPG_{max}$ and a measure of maximum allowed error of the NCS (E_{max}), as seen in Equation (8).

$$\delta_{max} = min\{\delta_{th_{min}}, \delta_{E_{max}}\} \tag{8}$$

The minimum deadband depends on the maximum acceptable throughput th_{max}, depending on the queue size of the intermediate routers of the network [4].

These values are important to limit the range of the deadband function and to guarantee the stability of the NCS.

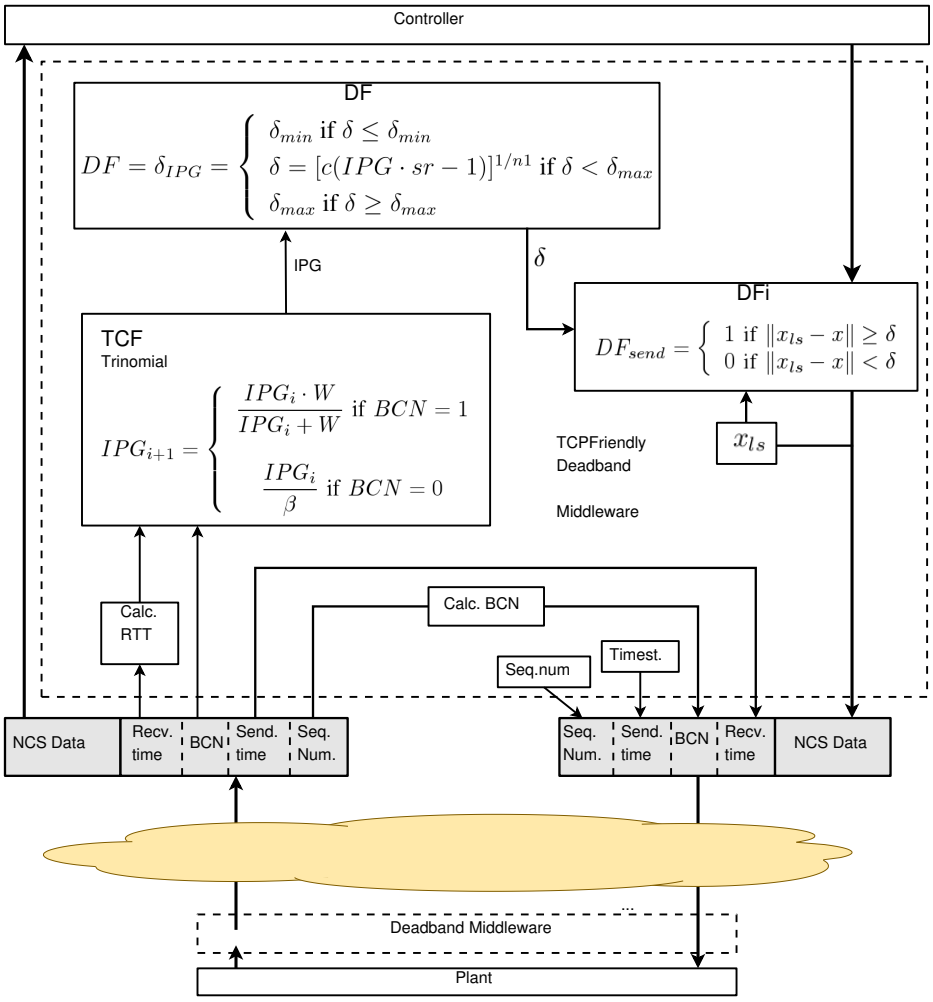
Figure 5(b) represents the typical relationship between deadband and throughput ($th(\delta)$) or error ($E(\delta)$), where deadband limits are also reflected. The $th(\delta)$ function varies with the traffic load of the shared network, as seen in the figure. The error *vs.* deadband curve is represented as $E(\delta)$ and his formulation is out of the scope of this paper. Equation (7) corresponds to the DF^{-1} throughput curve of the Figure 5(b).

For fixed packet size, th of Equation (7) can be replaced by $1/IPG$, and the deadband function DF can be represented as:

$$DF = \delta_{IPG} = \begin{cases} \delta_{min} \text{ if } \delta \leq \delta_{min} \\ \delta = [c(IPG \cdot sr - 1)]^{1/n1} \text{ if } \delta < \delta_{max} \\ \delta_{max} \text{ if } \delta \geq \delta_{max} \end{cases} \tag{9}$$

Figure 6 represents the TCP-friendly Adaptive Deadband scheme on the controller side, but it is similar on the plant side.

Figure 6. TCP-friendly adaptive deadband.



4.2.1. Determining the Optimal Constant Deadband

These approaches can be used to determine the constant deadband filter, by using the procedure presented in [5], and summarized below:

$$\begin{aligned} P(\delta) &= K_1 \cdot th_n(\delta) + K_2 \cdot E_n(\delta) \\ P(\delta_c) &= min_\delta\{K_1 \cdot th_n(\delta) + K_2 \cdot E_n(\delta)\} \end{aligned} \tag{10}$$

Constant deadband selection should follow the criteria in Equation (10), where $P(\delta)$ is a deadband dependent global performance function and δ_c is the optimal deadband value. Throughput and error should be normalized ($th_n = th/sr$ and $E_n = E/E_{max}$). K_1 and K_2 are weighting and scale factors.

The results obtained in simulation show that optimal deadband for an NCS varies with network conditions and reference signals.

4.2.2. Stability Issues

There are some stability analyses for NCS using deadband. In [5] the deadband is modeled as a vanishing perturbation. Then, if the original system is exponentially stable, the existence of a Lyapunov function is guaranteed in the perturbed system. Using the BIBO (Bounded Input Bounded Output) concept for NCS with LQR controllers, we can use the principle that any stable and linear system with bounded inputs generates bounded outputs. In a similar way, Lian *et al.* in [2] demonstrate stability for networked multi-agent systems that use deadband in the communication modules. For adaptive but bounded deadband, a similar principle can be used.

For tele-operated systems, the passivity approach based on scattering transformation [7] may be used to demonstrate stability using a bounded deadband.

5. TCP-Friendly Adaptive Deadband for a Tele-Operated Gantry Crane

The proposed schema has been tested in some control systems in our laboratory, such as a Mass-Spring and a gantry crane with a PID controller.

To present a practical implementation of the proposed solution, we use an application where the tele-operated slave is a laboratory crane, which is a scale model of a typical industrial gantry crane, and the master is a human tele-operator. The crane is equipped with actuated motors and sensors. Although the true dynamics of the crane are nonlinear and coupled, it will be assumed that the crane movements are smooth and slow, and that the swing angles of the load are small enough. Under these assumptions, the linearization of the crane is adequate to model the system. Furthermore, in this situation, the x, y, z dynamics become uncoupled. The vertical (or z) dynamic is controlled by the motor that lifts the rope. The x and y dynamics are formally equivalent and contain the load balance or swing. We will concentrate our attention on the x (or y) movement of a cart, provided by a motor, hanging a swinging load, not actuated, with a rope of fixed length.

The relation between the cart position $x_s(t)$, the swing angle $\theta_s(t)$ and the force $F_s(t)$ exerted by the cart motor is given by:

$$\begin{aligned} F_s(s) &= M_s\lfloor(s^2 + B_g s)\cdot x_s(s) - M_\mu \cdot g \cdot \theta_s(s)\rfloor \\ F_s(s) &= (-L_0 M_s)(s^2 + B_0 s + \omega_0^2)\cdot \theta_s(s) \end{aligned} \tag{11}$$

where M_s is the mass of the cart or trolley, $\omega_0 = \sqrt{(1+M_\mu)\frac{g}{L_0}}$, with L_0 the rope length, $M_\mu = \frac{M_{load}}{M_s}$ with M_{load} the load mass and B_g, B_0 the damping coefficients. We have used the subscript s to denote that this is the slave system in the tele-operation scheme. A state-spaced model can be defined considering the state vector $z(t) \in \mathbb{R}^n$ as:

$$z = \begin{pmatrix} z_s(t) & v_s(t) & \theta_s(t) & \omega_s(t) \end{pmatrix}^T \tag{12}$$

where x_s, v_s, θ_s and ω_s are the slave position, velocity, swing angle and angular velocity. In state-space the system is described with the following expressions:

$$\begin{aligned} \begin{pmatrix} \dot{z}_1 \\ \dot{z}_2 \\ \dot{z}_3 \\ \dot{z}_4 \end{pmatrix} &= \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & -B_s & M_\mu \cdot g & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\omega_0^2 & -b_0 \end{pmatrix} \cdot \begin{pmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{1}{M} \\ 0 \\ -\frac{1}{L_0 M_s} \end{pmatrix} F_s(t) \\ \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} &= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{pmatrix} \end{aligned} \tag{13}$$

Or in compact form:

$$\begin{aligned} \dot{z}(t) &= Az(t) + BF_s(t) \\ y(t) &= Cz(t) \end{aligned} \tag{14}$$

For the implemented dynamic model of the 3D crane Inteco Ltd. the numerical values are:

$$\dot{z}(t) = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & -18.3 & 0.959 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -9.5 & -0.003 \end{pmatrix} z(t) + \begin{pmatrix} 0 \\ 7.2 \\ 0 \\ -0.95 \end{pmatrix} \cdot F_s(t) \tag{15}$$

The controller at the master side receives the state information from the slave side and implements a linear quadratic regulator LQR for a steady-state optimal control. The selection of weighting matrices Q_1 and Q_2 is made so that the cost function is:

$$\begin{aligned} J &= \textstyle\int_0^\infty (z^T Q_1 z + F_s^T Q_2 F_s)dt \\ Q_1 &= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 25 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \; Q_2 = 2.0408 \end{aligned} \tag{16}$$

Moreover we introduce the Pincer procedure, a modification to the performance criterion that allows a degree of direct control over the settling time:

$$\begin{aligned} J_\alpha &= \int_0^\infty (z^T Q_1 z + u^T Q_2 F_s) e^{2\alpha t} dt = \int_0^\infty (w^T Q_1 w + u^T Q_2) dt \\ w(t) &= e^{\alpha t} z(t),\ u(t) = e^{\alpha t} F_s(t) \end{aligned} \tag{17}$$

From the definitions in Equation (17) the revised system for use in a LQR computation for the feedback gain matrix is:

$$\begin{aligned} \dot{w}(t) &= (\alpha I_{4x4} + A) w(t) + B u(t) \\ u(t) &= N \cdot r(t) - K \cdot w(t) \end{aligned} \tag{18}$$

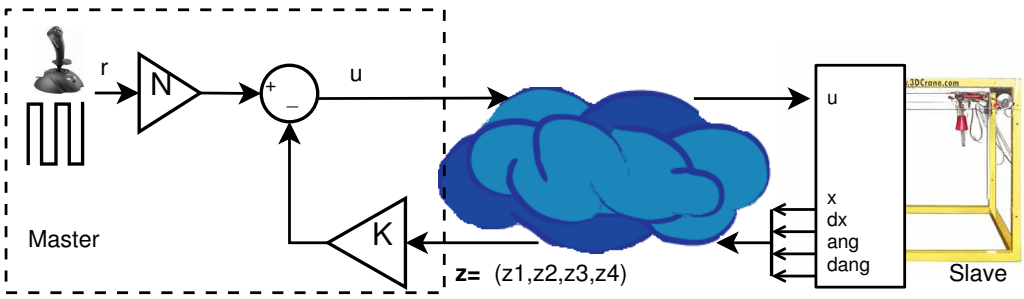
where $r(t)$ is the reference for the controller. If the designer selects the settling time $t_s = 3$, within which all states should settle to less than 2%, it is obtained:

$$\begin{aligned} z(t) &= z(0) e^{-\alpha t} \\ z(t_s) &= z(0) e^{-\alpha t_s} \leq 0.02 \cdot z(0) \Rightarrow \alpha > \frac{L_n 50}{t_s} \end{aligned} \tag{19}$$

The slave and the LQR controller by the master are using now the equations in Equation (11). After some trials, the requirements of the system are accomplished with a control law given by $K = (12.1661, 0.6646, -22.8596, -3.4107)$ and $N = 12.1661$.

Figure 7 shows a representation of the tele-operated gantry crane, simulated in this work.

Figure 7. Tele-operated gantry crane.

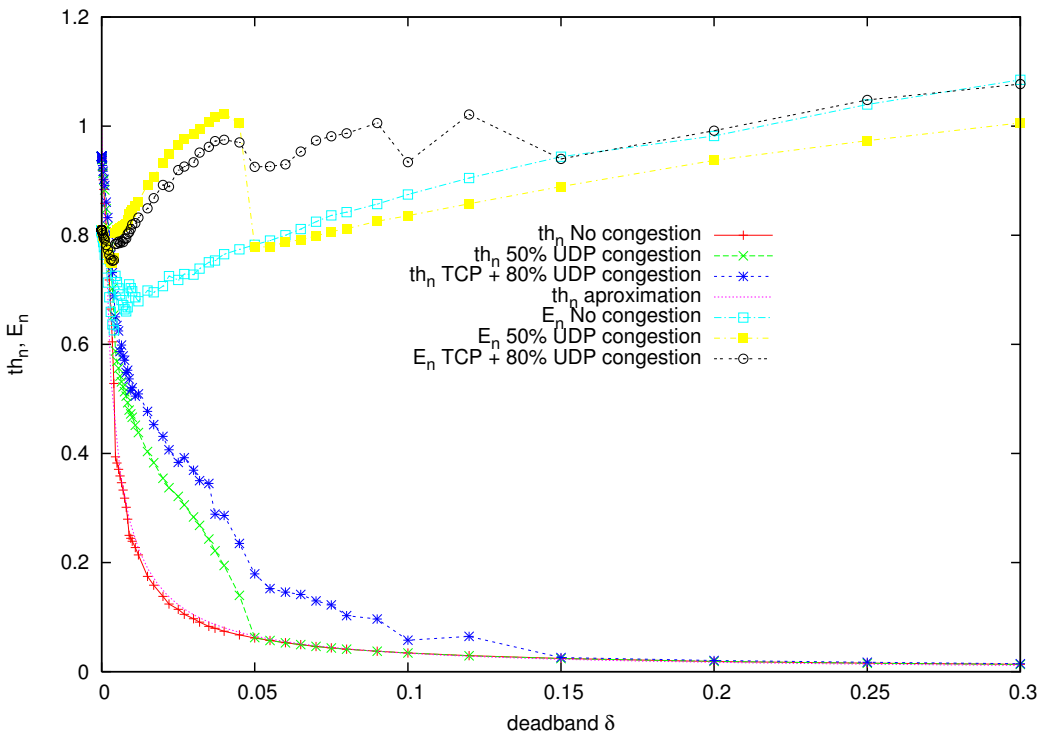


5.1. Adaptive Deadband Function

The deadband function should be determined for the master and the slave sides, but as illustrative example, only the process for the master side is presented. In this process, the slave side remains with a constant deadband of value 0.

Figure 8 represents the experimental functions δ_{th} and δ_E for the network conditions of the experiments, using the gantry crane with sinusoid reference signal of amplitude 10 and period 15 s. The approximate functions given in Equation (7) are also represented.

Figure 8. Throughput (th_n) and average error (E_n) response *vs.* deadband (δ) with different network status for a tele-operated gantry crane. Reference signal is $10sin(2\pi\frac{1}{15}t)$.



The function $th(\delta)$ parameters are approximated as $c = 0.0035$ and $n_1 = 1$. For higher reference periods, the deadband function parameter c should be decreased, and parameter n_1 varies with more abrupt signals, such as a square signal.

The normalized function is:

$$th_n(\delta) = \frac{0.0035}{0.0035 + \delta} \tag{20}$$

Therefore, in Figure 8 for $E_{max} = 2.5$ ($E_n = 1$), the deadband has an approximate value of $\delta_{max} = 0.25$, and $\delta_{min} = 0$, for the maximum throughput th_{max} that equals the sampling rate of $1,000$ pkt/s.

The deadband function DF for the studied tele-operated gantry crane is:

$$\delta_{IPG} = \begin{cases} \delta = 0.0035 \cdot (IPG \cdot 1000 - 1) \text{ if } \delta < 0.25 \\ 0.25 \text{ if } \delta > 0.25 \end{cases} \tag{21}$$

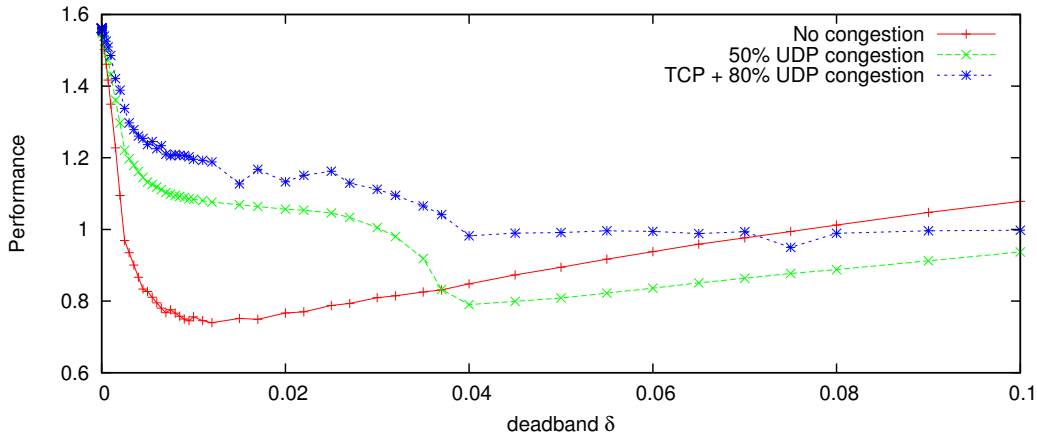
On the slave side, using the same approximation methodology, the parameters c and n_1 are the same than on the master side, but the function $E(\delta)$ remains approximately constant. Therefore, a value of $\delta_{max} = 10$ can be adopted.

5.2. Constant Deadband for a Gantry Crane

The same reference signals are used to obtain the constant deadband. Results obtained in simulation and approximated minimum are shown in Figure 9 for some network congestion situations. The

minimum is within the range of $0.01 < \delta_c < 0.05$. In the Figure 9, the optimal constant deadband value is approximately $\delta_c = 0.02145$ for a sinusoid reference signal with period 15 s.

Figure 9. Constant deadband, experimental and approximated for a gantry-crane. Reference signal is $10sin(2\pi\frac{1}{15}t)$.



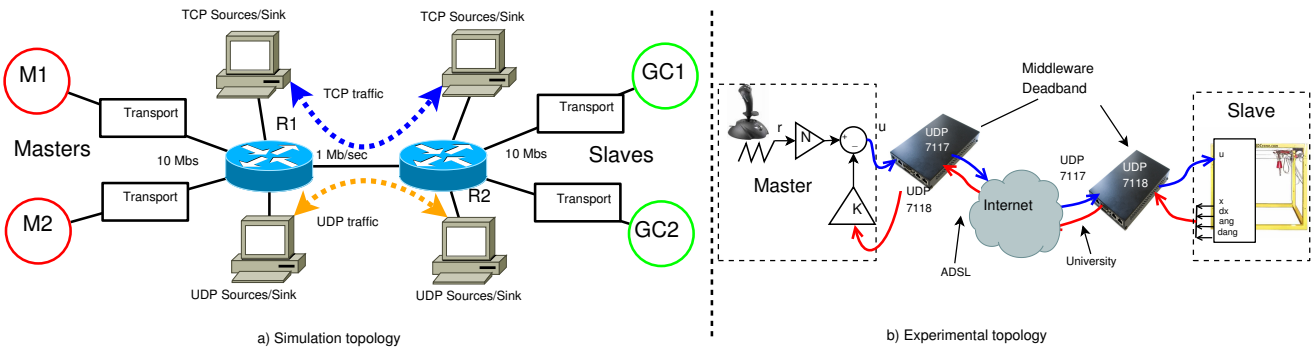
6. Simulated Example: Sharing the Network

The simulations in this section compare constant deadband (CD) with network adaptive deadband (NAD) in different network scenarios. The reason to make this comparison is to show the effectiveness of the NAD solution with respect to the CD solution in scenarios with high congestion when more than one NCS is used.

Simulations were made with two tele-operated gantry cranes. The two systems use different reference signals. The gantry crane 1 (gc1) uses a sinusoid reference signal with period 15 s and amplitude 10, and gantry crane 2 (gc2) uses a triangle reference signal with period 18 s and amplitude 10. Both reference signals are continuous and slow and have similar deadband parameters.

We use the co-design framework in [4] based on the simulation suite NS2. The network topology implemented is represented in Figure 10. The transport boxes in the figure can use a configured NAD transport schema, a constant deadband transport schema or a pure bidirectional transport protocol.

Figure 10. Simulation (**a**) and Experimental (**b**) topologies.



To simulate the Internet, we use a scenario with TCP and UDP traffic from diverse sources. The network topology uses a bottleneck trunk link between two routers with 1 Mbps capacity. The access

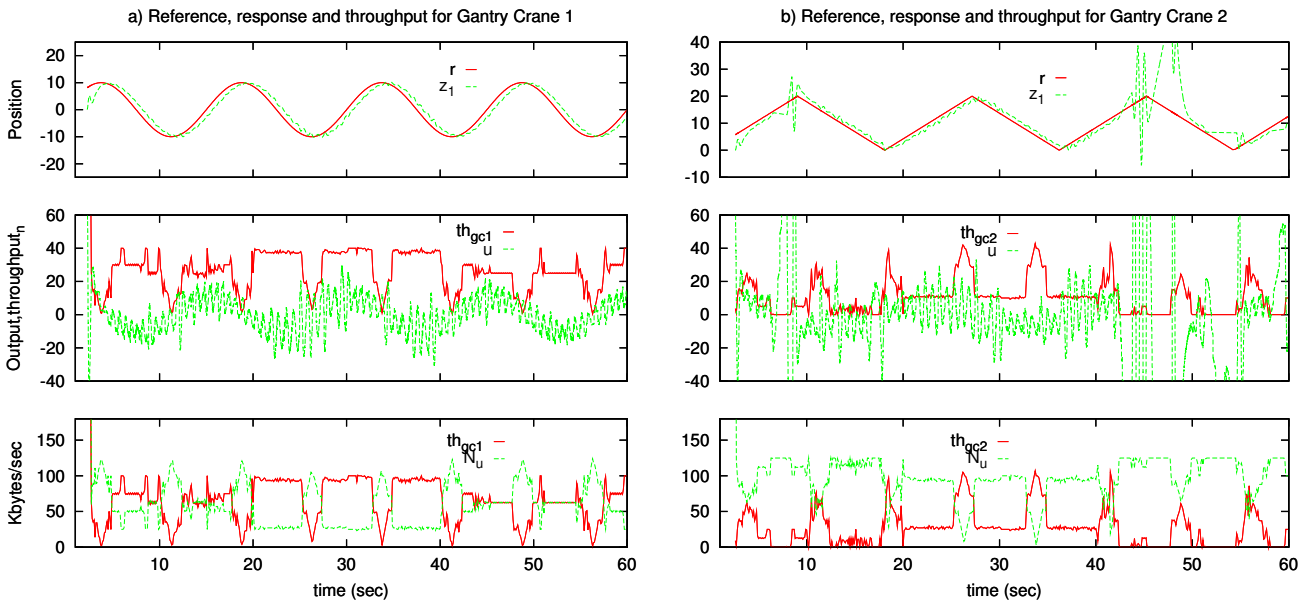
links capacity to the routers is 10 Mbps. All links are symmetric. The two routers implement a Drop-Tail queue with a queue limit of 15 packets. Simulations were performed over a total of 100 s with hard bottleneck congestion of 80% due to UDP and TCP traffic, but no UDP congestion is simulated between 20 s and 40 s in both directions to show Best Effort characteristics.

A comparison between the optimal constant deadband functions and the adaptive deadband functions is made, with deadband function parameters deduced in Section 5, as shown in Table 1.

Table 1. Deadband function for a Gantry Crane.

$ref_{gc1} = 10 \cdot sin(2\pi/15 \cdot t)$		
$ref_{gc2} = 10 \cdot triang(T = 18.1)$		
	master to slave (ms)	slave to master (sm)
$th_n(\delta)$	$\frac{0.0035}{0.0035+\delta}$	$\frac{0.0035}{0.0035+\delta}$
$(\delta_{min}, \delta_{max})$	$(0, 0.25)$	$(0, 10)$
δ_c	0.02145	1

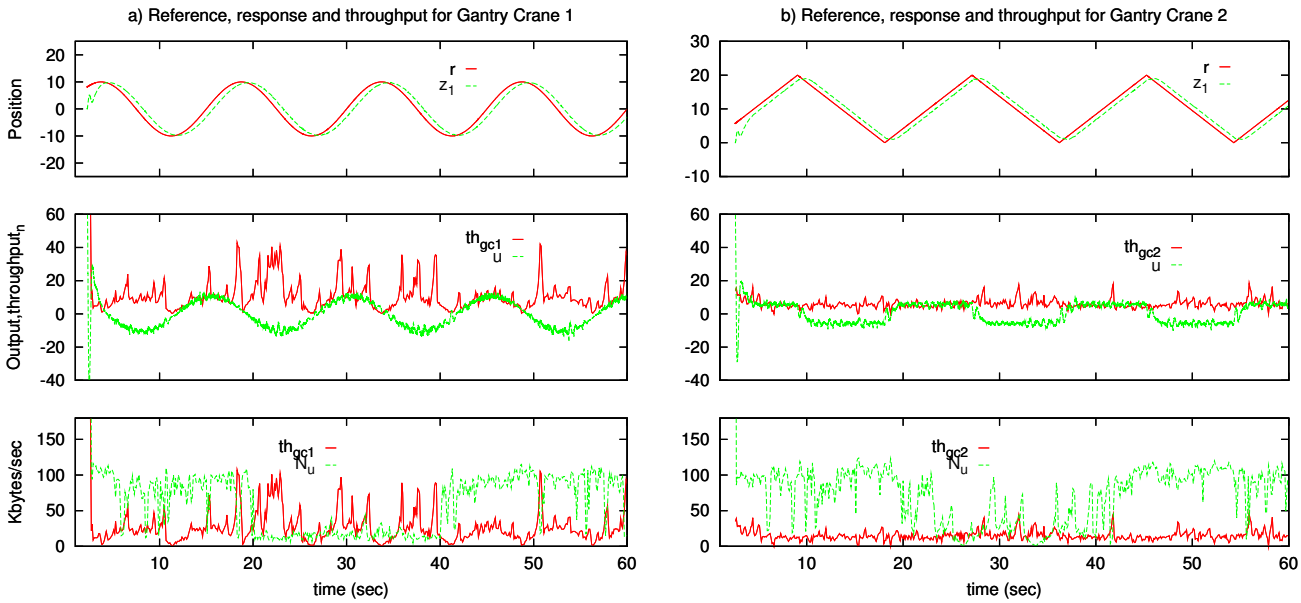
Figure 11. Simulation results for constant deadband.



The results are presented in Figure 11 for constant deadband and in Figure 12 for adaptive deadband. In the figures, the top graphs represent reference position of the master (r), and output of the slave (x_1). The middle graphs represent the output of the master (u) compared with a suitable scaled throughput from the master to the slave, for clarity purposes. The bottom graphs represent the throughput of the tele-operated system from the master to the slave (th_{ms}), compared with the remaining use of the network N_u. Figure 11 in the constant deadband case shows that the two data flows compete to gain

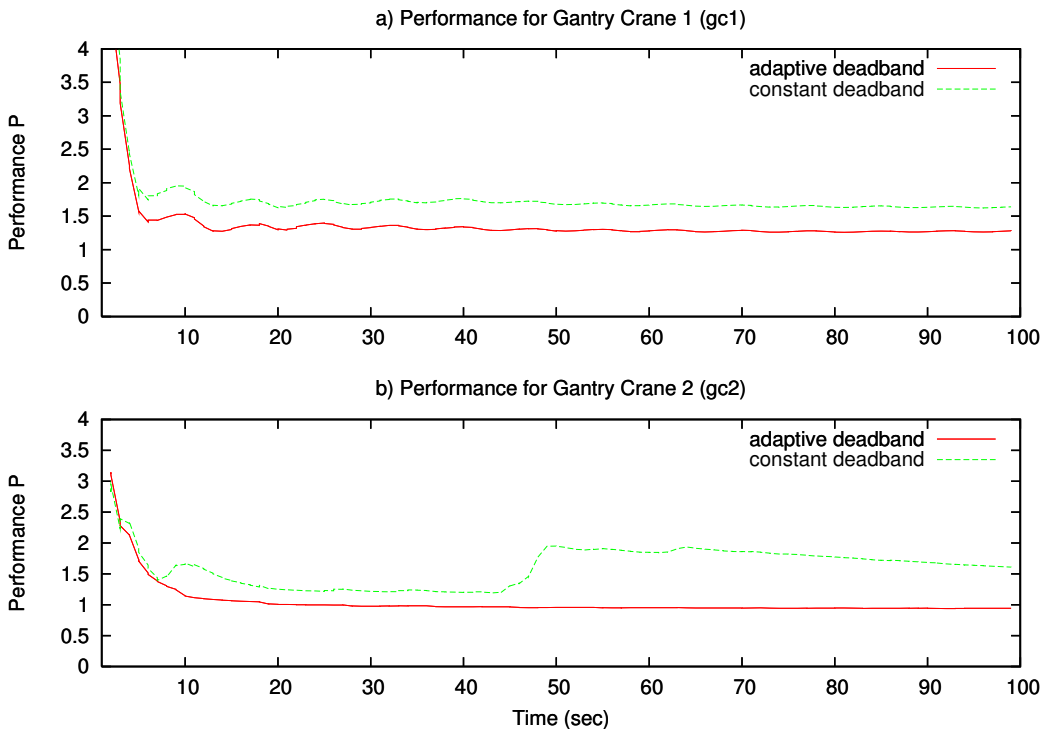
the network resources, and one of them (or the two) can be unstable. Using the sinusoid reference signal, the throughput is low on the top and bottom parts of the signal because of slow variation, and the gantry crane 2 has problems to maintain stability.

Figure 12. Simulation results for Network Adaptive Deadband.



In Figure 12, with high variations of the master output, the bandwidth consumption of the system is high, until the output stabilizes, but remains fair with the rest of network traffic.

Figure 13. Performance comparison between optimal constant deadband and network adaptive deadband in the simulation.



Finally, Figure 13 shows a performance (Equation (10) with $K_1 = K_2 = 1$) comparison between TCP-friendly adaptive deadband and optimal constant deadband for the two gantry cranes. The results show that the adaptive deadband has better performance than the constant deadband for the long time operation.

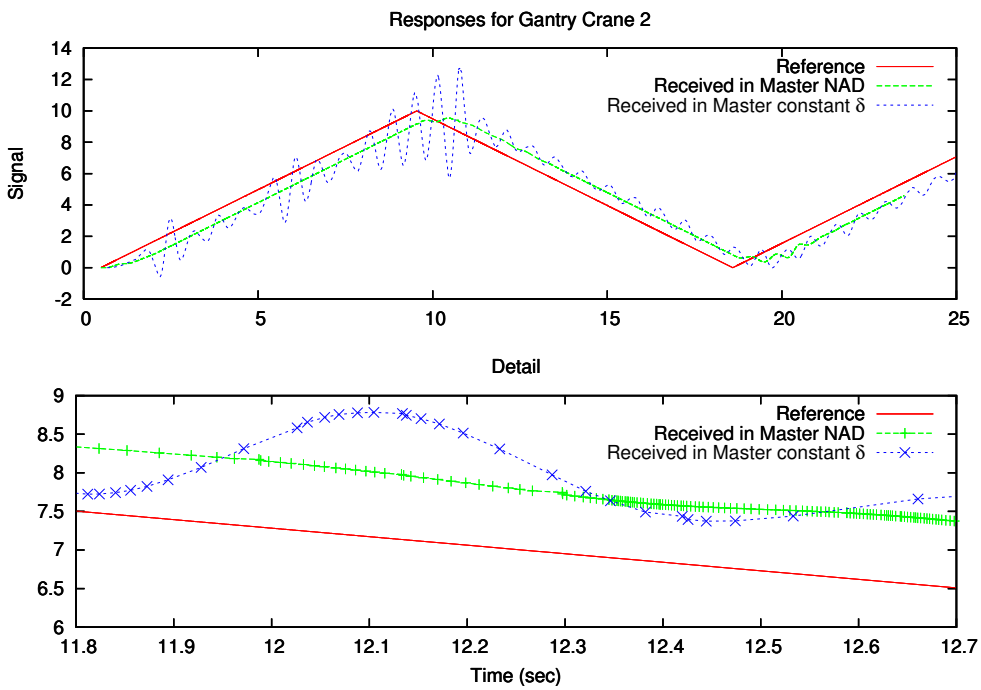
7. Experimental Results

To validate the effectiveness of the proposed solution, NAD middleware was implemented in a real Internet scenario between an emulated Master through a normal ADSL Internet access and an emulated Slave through a high capacity Internet access in a University laboratory. The transport protocol used is presented in [9], based on a bidirectional implementation of the trinomial algorithm. The transport packet header is over the UDP protocol as shown in Figure 4, where the data structure is also depicted.

Experimental platform is presented in Figure 10(b), where the plant and controller are the emulated gantry crane and the LQR controller described in Section 5. The emulated system uses a triangle reference signal with period 18 s and amplitude 10. Similar transport parameters as gc2 in the simulated scenario of Section 6 were used. The parameters are represented in Table 1. Measured network conditions between both extremes have an average RTT of $43msec$ and 20% packet losses at a transmission rate of $200pkts/s$ with $100bytes$ packet size.

The experiment compares the same tele-operation system with different transport solutions, in particular with an only TCP-friendly transport protocol, with the constant deadband solution and our NAD solution. Results are not presented graphically for the TCP-friendly transport protocol because the tele-operated system becomes unstable. Graphical results for the other cases are presented in Figure 14.

Figure 14. Compared results between NAD and constant deadband in the real Internet link for the gantry crane and the LQR controller.



The results show that the NAD has better performance than constant deadband. In some circumstances, with high congestion, constant deadband can become unstable while NAD maintains stability.

8. Conclusions and Future Works

This paper proposes a new middleware solution to improve long-time operation of an NCS through the Internet or any shared network based on IP technology. This solution is useful for sensor networks where the network use or the power consumption is of special interest. The proposed middleware takes into account the network status and the NCS status to decide whether to send the output of the system. In that way, the network can be shared more effectively between several NCS or other bidirectional real-time flows, and the middleware can be used as a complement to a well-designed NCS on an unpredictable shared network. In this case, simulation and experimental results show that the proposed solution has better performance than a TCP-friendly transport protocol or a constant deadband solution.

The proposed scheme, called Network Adaptive Deadband or NAD, is a solution for network and control integration, using a simple relationship between deadband and TCP-friendly transport throughput. An experimental procedure is presented to estimate this relationship. Simulation and experimental results show that the middleware presents good performance in any network conditions, with a TCP-friendly transport control protocol applied to an estimated deadband function.

The results are of interest to maintain NCS stability and TCP-friendly flows, through IP-based networks. TCP-friendly protocols and deadband sampling do not ensure by itself best network and control system performance, but together they may give enough reasons to improve the use of the network and to maintain the control system stability. The utility of the network adaptive deadband is not restricted only to IP network applications, but in fact is also applicable to collaborative robotics in local networks, sensor networks, MMMS tele-operation or plants with several data flows.

The simulation and experimental results show that the middleware presents good performance in any network conditions, with a TCP-friendly transport control protocol applied to an estimated deadband function. The results evidence that network adaptive deadband is a good solution to share a public network as Internet for several NCS. The optimal constant deadband criterion can be adequate when there are no other NCS data flows that can collapse the network, but can be inefficient and generate instability caused by non-regulated flow control. Adaptive deadband can regulate the data flow depending on network status, and permits other NCS data flows to share the network. The utility of the network adaptive deadband involves not only IP network applications but also collaborative robotics in local networks, MMMS tele-operation or plants with several data flows.

The solution presented has a practical nature, but as a future line, a more theoretical depth is required to refine the deadband function and to analyze and model the network-dependent deadband. To do this, control system equations and transport models should be mixed. The research lines of event-triggering and the attempts to relate protocols and control design are interesting start points.

Acknowledgments

The authors would like to thank all the reviewers and editors for the valuable and constructive comments. They have been taken into account in reviewing this manuscript.

This work has been partially supported by Spanish Science and Technology Ministry in the Project DPI2010-20466-C02-01.

References

1. Zhao, Y.B.; Liu, G.P.; Rees, D. Packet-based deadband control for internet-based networked control systems. *IEEE Trans. Control Syst. Technol.* **2010**, *18*, 1057–1067.
2. Lian, F.L.; Yook, J.K.; Tilbury, D.M.; Moyne, J. Network architecture and communication modules for guaranteeing acceptable control and communication performance for networked multi-agent systems. *IEEE Trans. Ind. Inf.* **2006**, *2*, 12–24.
3. Nesic, D.; Teel, A.; Carnevale, D. Explicit computation of the sampling period in emulation of controllers for nonlinear sampled-data systems. *IEEE Trans. Autom. Control* **2009**, *54*, 619–624.
4. Díaz-Cacho, M.; Delgado, E.; Barreiro, A. Control/network codesign basics for ip-based shared networks. *Math. Probl. Eng.* **2011**, *2011*, 23.
5. Otanez, P.; Moyne, J.; Tilbury, D. Using Deadbands to Reduce Communication in Networked Control Systems. In *Proceedings of the American Control Conference*, Anchorage, AK, USA, 8–10 May 2002; Volume 4, pp. 3015–3020.
6. Floyd, S.; Fall, K. Promoting the use of end-to-end congestion control in the Internet. *IEEE/ACM Trans. Netw.* **1999**, *7*, 458–472.
7. Hirche, S.; Hinterseer, P.; Steinbach, E.; Steinbach, E.; Buss, M. Network Traffic Reduction in Haptic Telepresence Systems by Deadband Control. In *Proceedings of the 16th IFAC World Congress*, Prague, Czech Republic, 4–8 July 2005.
8. Zhao, Y.B.; Liu, G.P.; Rees, D. Design of a packet-based control framework for networked control systems. *IEEE Trans. Control Syst. Technol.* **2009**, *17*, 859–865.
9. Diaz-Cacho, M.; Barreiro, A.; Rivera, M. Bidireccionalidad y eficiencia en el transporte de datos de tele-operación a través de redes IP. *Revista Iberoamericana de Automática e Informática Industrial (RIAI)* **2010**, *7*, 99–110.
10. Zhao, Y.; Liu, G.; Rees, D. Integrated predictive control and scheduling co-design for networked control systems. *Control Theory Appl. IET* **2008**, *2*, 7–15.
11. Vittorias, I.; Kammerl, J.; Hirche, S.; Steinbach, E. Perceptual Coding of Haptic Data in Time-Delayed Tele-operation. In *Proceedings of World Haptics 2009, 3rd Joint Eurohaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, Salt Lake City, UT, USA, 18–20 March 2009; pp. 208–213.
12. Xia, F.; Ma, L.; Peng, C.; Sun, Y.; Dong, J. Cross-layer adaptive feedback scheduling of wireless control systems. *Sensors* **2008**, *8*, 4265–4281.
13. Vasyutynskyy, V.; Kabitzsch, K. Towards Comparison of Deadband Sampling Types. In *Proceedings of IEEE International Symposium on the Industrial Electronics, ISIE 2007*, Vigo, Spain, 4–7 June 2007; pp. 2899–2904.

14. Nesić, D.; Teel, A. Sampled-data Control of Nonlinear Systems: An Overview of Recent Results. In *Perspectives in Robust Control*; Moheimani, S., Ed.; Springer: London, UK, 2001; Volume 268, pp. 221–239.
15. Mazo, J.M.; Tabuada, P. Decentralized event-triggered control over wireless sensor/actuator networks. *IEEE Trans. Autom. Control* **2011**, *56*, 2456-2461.
16. Mazo, M.; Cao, M. Decentralized Event-triggered Control with Asynchronous Updates. In *Proceedings of 2011 50th IEEE Conference on the Decision and Control and European Control Conference, (CDC-ECC)*, Orlando, FL, USA, 12–15 December 2011; pp. 2547–2552.
17. Camacho, A.; Martí, P.; Velasco, M.; Lozoya, C.; Villà, R.; Fuertes, J.; Griful, E. Self-triggered Networked Control Systems: An Experimental Case Study. In *Proceedings of 2010 IEEE International Conference on the Industrial Technology (ICIT)*, Viña del Mar, Chile, 14–17 March 2010; pp. 123–128.
18. Tipsuwan, Y.; Chow, M.Y. Gain scheduler middleware: A methodology to enable existing controllers for networked control and tele-operation–part I: Networked control. *IEEE Trans. Ind. Electron.* **2004**, *51*, 1218–1227.
19. Cuenca, A.; Salt, J.; Sala, A.; Piza, R. A delay-dependent dual-rate pid controller over an ethernet network. *IEEE Trans. Ind. Inf.* **2011**, *7*, 18–29.
20. Floyd, S.; Handley, M.; Padhye, J.; Widmer, J. Equation-based congestion control for unicast applications. *SIGCOMM Comput. Commun. Rev.* **2000**, *30*, 43–56.
21. Kohler, E.; Handley, M.; Floyd, S. Designing DCCP: Congestion Control Without Reliability. In *Proceedings of the 2006 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, SIGCOMM '06*, Pisa, Italy, 11–15 September 2006; pp. 27–38.
22. Liu, P.; Meng, M.H.; Liu, P.; Yang, S. An end-to-end transmission architecture for the remote control of robots over IP networks. *IEEE/ASME Trans. Mechatron.* **2005**, *10*, 560–570.
23. Wirz, R.; Marin, R.; Ferre, M.; Barrio, J.; Claver, J.; Ortego, J. Bidirectional transport protocol for tele-operated robots. *IEEE Trans. Ind. Electron.* **2009**, *56*, 3772–3781.
24. Diaz-Cacho, M.; Barreiro, A.; Rivera, M. Bidirectionality in IP based Tele-operation. In *Proceedings of the 4th International Scientific Conference on Physics and Control, PHYSCON 2009*, Catania, Italy, 1–4 September 2009.
25. young Lee, J.; Payandeh, S. Signal Processing Techniques for Haptic Data Compression in Tele-operation Systems. In *Proceedings of the Haptics Symposium (HAPTICS)*, Vancouver, Canada, 4–7 March 2012; pp. 371–376.
26. Hirche, S.; Hinterseer, P.; Steinbach, E.; Buss, M. Transparent data reduction in networked telepresence and teleaction systems. Part I: Communication without time delay. *Presence: Tele-operators Virtual Environ.* **2007**, *16*, 523–531.
27. Gao, Z.; Xie, R.; Chen, Q.; Hu, W. A Deadband-Control and Prediction Co-Design Approach for Networked Robotic Systems. In *Proceedings of 2009 IEEE International Conference on the Robotics and Biomimetics (ROBIO)*, Guilin, China, 19–23 December 2009; pp. 2215–2220.

28. Tatikonda, S.; Mitter, S. Control under communication constraints. *IEEE Trans. Autom. Control* **2004**, *49*, 1056–1068.