INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 13(2), 251-267, April 2018.

Control-Scheduling Codesign for NCS based Fuzzy Systems

P.E. Mendez-Monroy, I. Sanchez Dominguez, A. Bassam, O. May Tzuc

Paul Erick Mendez-Monroy*, Israel Sanchez Dominguez
IIMAS-Merida Universidad Nacional Autonoma de Mexico,
Parque Cientifico y Tecnologico de Yucatan,
5.5Km Carretera Sierra Papacal - Chuburna,
C.P. 97302 Sierra Papacal, Yucatan, Mexico
*Corresponding author: erick.mendez@iimas.unam.mx
israel.sanchez@iimas.unam.mx

Ali Bassam, Oscar May Tzuc
Facultad de Ingenieria, Universidad Autonoma de Yucatan,
Av. Industrias no Contaminantes,
Apdo. Postal 150 Merida, Yucatan, Mexico
baali@correo.uady.mx, maytzuc@gmail.com

Abstract: In the present paper, a fuzzy codesign approach is proposed to deal with the controller and scheduler design for a networked control system which is physically distributed with a shared communication network. The proposed fuzzy controller is applied to generate the control with different sampling-actuation periods, the configuration supposes a strict actuation period disappears the jitter. The proposed fuzzy scheduling is designed to select the sampling-actuation period. So, the fuzzy codesign reduces the rate of transmission when the system is stable through the scheduler while the controller adjusts the control signal. The fuzzy codesign guarantees the stability of all the system if the network uncertainties do not exceed an upper bound and is a low computational cost method implemented with an embedded system. An unstable, nonlinear system is used to evaluate the proposed approach and compared to a hybrid control, the results show greater robustness to multiple lost packets and time delays much larger than the sampling period. This paper is an extension of [20].[a]
Keywords: codesign, dynamic scheduling, fuzzy control, networked control system.

[a]Reprinted (partial) and extended, with permission based on License Number 4275590998661 IEEE, from "Electrical Engineering, Computing Science and Automatic Control, 2017 14th International Conference on".

1 Introduction

This paper is an extension of [20]. The system model with sampling-actuation periods in [20] is modified using the one-step control input, the network imperfection estimation is improved, the feedback matrices are calculated in a simpler way. Finally, the analysis of stability and the analytical codesign are also presented.

Networked control systems (NCSs) are composed of physically distributed agents that can sense the environment, act on it, and communicate with one through a communication network to achieve some common goals. These characteristics have made them a topic of current interest in the control area.

By including a communication network within the control loop, considerations are presented for its design. Among these considerations the most important are time delays [13] [27] [11] [12], packet losses [13] [27] [11] [12], signal quantization [14] [21] and scheduling [6] [28]. These have been investigated with results reported in the literature. In addition, because of the advantages of reduced system wiring, simple installation, increased system flexibility and resources sharing, NCSs have been finding applications in DC motors [5] [9], robot control [28], vehicle robot [25] and ball maglev system [11], among others.

Time delays in NCSs are the major cause of system performance deterioration and potential system instability. Time delays have been modelled by using various formulations such as constant delay [22], independently random delay [1] and random delay governed by Markov chain [32]. Sometimes, the time delays are included as time-varying input delays of the system [30]. Yi, An and Choi performed a control system over a wireless network with communications only in the control channel. They employ a full-order observer to estimate the system states without time delays. The control-actuator time delay is measured at the plant and the observed states are used as a predictor to generate the system state with the expected time delay. Finally, they employ a LQR control to generate the control signal based on the next estimated states.

Therefore, the analysis and synthesis of NCSs with both time delays and packet losses is a persistent problem in the challenging but practical problem. In the literature, some important methodologies, such as stochastic control [11], predictive control [5] [10] [30], robust control [12], and state feedback control [31], are proposed to compensate time delays and/or packet losses have been proposed.

At the last years, fuzzy logic control has received great attention from academic and industrial communities. More recently, the fuzzy control has developed strategies for NCS, In the work of Peng and Yang [24], a delay distribution-dependent design method for NCS Takagi Sugeno fuzzy systems [26] was proposed taking into consideration of the probabilistic interval distribution of the communication delay. Tong, Qian and Lui [29] used a fuzzy predictive controller to counteract time delays in the feedback channel. Where the fuzzy controller estimates the variations of the control signal based on the differences between the reference error and the control error applied. Chai et al. [4] investigated the state feedback and dynamic output feedback controller design for membership functions and time delays in premise variables into the controller design. The resulting conditions were expressed in terms of SOS-based inequalities.

In the case of lost packets, commonly the effect has been modelled by a Bernoulli process and strategies in static/dynamic output feedback and model predictive control problems for discrete-time T-S systems with lost packets were investigated in [7] [34]. In [15], Li, Wu, and Feng used a fuzzy model to describe a nonlinear plant with an output feedback controller $H\infty$ and modelling the lost packets as a Bernoulli random binary distribution.

It is noticed that most of the existing control methodologies for NCSs adopt a sampling period regardless of network Quality-of-Services (QoS) variations. In practical circumstances, the network QoS always fluctuates due to changes in the traffic load and available network resources.

In regard to QoS, Tipsuwan and Chow [28] proposed a gain scheduling controller for NCSs, where the control parameters were adjusted on-line based on network QoS variations and Chow [6] optimized the control parameters for gain scheduling controller to improve the NCSs performance. However, these works only focused on the controller design. More recently, BenĂtez et al. [3] presented a frequency control of multiple network control systems, this takes into account information from the network transmissions, where asymptotic stability of the systems is ensured when the time delay is bounded.

This paper shows a fuzzy NCS codesign controller-scheduler to adapt simultaneously the control signal and sampling period with estimated network imperfections. This introduces a neural model to estimate the time delay and lost packets as a compound time. A fuzzy control with the estimated compound time as the antecedent part is used to minimize the network effects. Finally, a fuzzy scheduler is designed to modify the sampling-actuation period based on the system performance and network utilization.

The paper is organized as follows: section 2 introduces a dynamic model based on actuation periods and shows a recurrent neural network to estimate some network imperfections. In section 3 the fuzzy model with different actuation periods and the stability analysis is presented. Section 4 summarizes the design of a fuzzy scheduler to modify the actuation period. Section 5 presents an analytical codesign and section 6 presents the experimental case to evaluate the codesign performance, it is compared with a hybrid controller. Finally, the conclusions are provided in section 7.

2 Preliminaries

2.1 Periodic Actuation Model

NCS is defined as: A feedback control system closed via a communication channel, it may be shared with other control loops or nodes outside the control system. A spatially distributed system for a single control loop is shown in Figure 1, where the sensor, controller, and actuator nodes exchange information via a communication network.

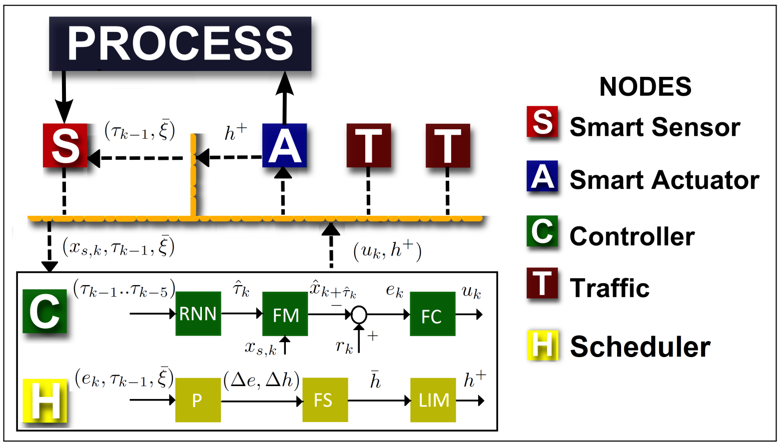

Figure 1: Configuration of NCS with traffic nodes

The most general continuous state-space representation of a linear system with m inputs, p outputs and n state variables is written in the following form

$$\begin{aligned} \dot{x}(t) &= Ax + Bu(t) \\ y(t) &= Cx(t) \end{aligned} \tag{1}$$

where $A \in \mathbb{R}^{nxn}$, $B \in \mathbb{R}^{nxm}$ and $C \in \mathbb{R}^{pxn}$ are the system, input and output matrices of the continuous-time state space respectively. $x_k \in \mathbb{R}^n$ is the process state vector, $u_k \in \mathbb{R}^m$ and $y_k \in \mathbb{R}^p$ are the inputs and outputs of the process.

The continuous linear system Eq. 1 can be discretized assuming a zero-order hold [2] for the input vector with a sampling period h to

$$\begin{aligned} x_{k+1} &= \Phi_h x_k + \Gamma_h u_k \\ y_k &= C x_k \end{aligned} \tag{2}$$

where the matrices $\Phi_h \in \mathbb{R}^{nxn}$ and $\Gamma_h \in \mathbb{R}^{nxm}$ are obtained by

$$\Phi_h = e^{Ah}, \quad \Gamma_h = \int_0^h e^{As} B ds \tag{3}$$

For standard closed-loop operation of the discrete system (Eq. 2), a controller can be designed using feedback control as follows

$$u_k = Kx_k, \quad K \in \mathbb{R}^{mxn} \tag{4}$$

where K is the state feedback matrix obtained using standard control design methods.

The application of the control signal (Eq. 4) to the process forces to computing it with *zero* time. Nevertheless, this is physically impossible even for processor-based systems due to algorithm computational time.

Taking into account this limitation. The discrete model (Eq. 2) can be augmented to cope with a time delay due to the insertion of a network/processor within a control loop, as in the case of NCS [2]. The standard model that incorporates a time delay τ less than one sampling period ($\tau < h$), is

$$x_{k+1} = \Phi_h x_k + \Phi_{h-\tau}\Gamma_\tau u_{k-1} + \Gamma_{h-\tau} u_k \tag{5}$$

The equation 5 has been often taken as the essential control model for design and analysis of NCS. This model assumes a time reference given by the sampling instants with a fixed time delay from sampling to actuation. However, this model is useless if the time delay is variable and/or greater than one sampling period or the sampling interval is variable. [17].

The task execution model proposed is shown in figure 2. It aims is to use strict periodic sampling and actuation [18] into the space state model decreasing the variability in the time delays and the sampling intervals. This model estimates the states as a function of the actuation periods [16], making only necessary to estimate the compound time in a set of multiples of the actuation period.

The model synchronizes the operation of each control loop at the actuation instants. Hence, t_k is the actuation instant, the actuation interval is the time elapsed between consecutive actuation instants, named t_{k-1} and t_k, h is the actuation period. Within this actuation interval, the system state is sampled, named $x_{s,k}(t_{s.k}) \in [t_{k-1}, t_k]$ where $t_{s,k}$ is the sampling time recorded. Eq. 6 represents a time delay τ_k used to estimate the state at the actuation instant and the Eq. 7 represents the discrete system with periodic actuation.

$$\tau_k = t_k - t_{s,k} \tag{6}$$

$$\hat{x}_k = \Phi_{\tau_k} x_{s,k} + \Gamma_{\tau_k} u_{k-1} \tag{7}$$

Finally, making use of $\hat{x}_k$, the control command is computed as

$$u_k = K\hat{x}_k, \quad K \in^{mxn} \tag{8}$$

where K is the feedback matrix that is designed in next section.

The control command u_k is held constant within actuation period with a zero-order Hold (ZOH).

At each control cycle the information flow in the NCS, the sensor node begins sampling the process $x_{s,k}$ in time $t_{s,k}$, the time delay and lost packets are estimated. It sends the data to the codesign node where is used to generate u_k (Eq. 6)-(Eq. 8), also the scheduler generates the next actuation period if it is necessary. The control and period are sent to the actuator node applying the control to the process. Finally, the actuator node sends t_k and the period to the sensor node to apply the sampling period and calculate the time delay and lost packets. The cycle starts again.

With the strict periodic sampling and actuation h, the time delay τ is restricted to multiples of the actuation period and the sampling intervals can be used to control the network bandwidth consumption. The model has several properties for controllers, it is compatible with standard

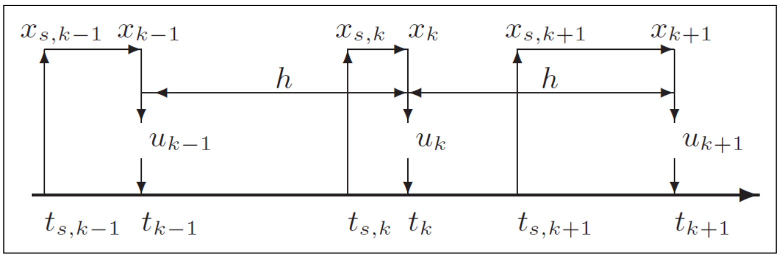

Figure 2: Periodic Actuation Model

scheduling because does not demand any specific timing constraints. The scheduling jitter is absorbed with a priori knew of time reference t_k and the sampling jitter disappears using the periodic actuation model (Eq. 7) absorbs the irregular sampling and variable actuation intervals.

The next section presents an approach to estimate the time delay and lost packets with a recurrent neural network.

2.2 Time delay and lost packets estimation

One major challenge for NCS design is the effect of time delays and lost packets in a control loop. The time delays occur when the system components exchange data across the network. It can degrade the performance or even destabilize the system. The time delay τ_k assumes lower and upper bounds. In case of time delays is

$$0 < \tau_{min} \leq \tau_k \leq \tau_{max} \tag{9}$$

On the other hand, lost packets can be the consequence of a link failure, generated purposefully to avoid congestion or guarantee the most recent data to be sent. Normally, feedback controllers can tolerate a certain amount of lost packets. However, consecutive lost packets have an impact of degradation on the overall system performance. Hence, the next actuation period is a compound time υ between time delays and lost packets formed as follows

$$\upsilon \equiv t_{k+1} - t_k = (\bar{\xi} + 1)h + \tau_{k+1} - \tau_k \tag{10}$$

where h denotes the actuation period, t_k the actuation instant and $\bar{\xi}$ the estimated lost packets.

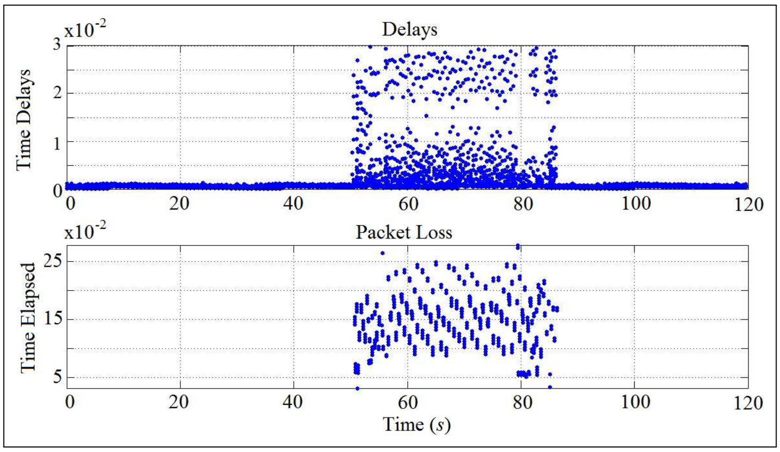

Figure 3: Time delays and time elapsed between lost packets with traffic (50 - 80 s).

Eq. 10 defines a compound time with the analytic bounds for time delay and lost packets, the aim is to design a recurrent neural network (RNN) using compound time values as input, and time delay forecast as output. The collection of time delay data (τ_k) is analyzed and modelled to achieve the lowest discrepancies between the observed delay and the predicted delay (Fig. 3). Figure 3 illustrates the time delays and time elapsed between consecutive lost packets. Heavy traffic is generated into 50-80 s showing a maximum compound time of 300 ms.

The RNN is formed with three layers (Fig. 4), the input layer with 6 delayed inputs, one feedback input and bias, the hidden layer with 10 tansig nonlinear neurons, and the output layer with one linear discriminatory purelin neuron and bias.

$$\begin{array}{ll} I = \{\tau_{k-i}, \hat{O}_{k-1}, b_1\} & i = 1-6 \\ H_j = tansig\left[\sum_{i=1}^{7} IW_{ij} I_i\right] & \\ \hat{\tau}_{k+1} = O = purelin\left[\sum_{j=1}^{10} OW_j H_j + b_2\right] & \end{array} \tag{11}$$

The Levenberg-Marquardt algorithm was used for training. The number of hidden nodes selected was the RNN with the best validation performance.

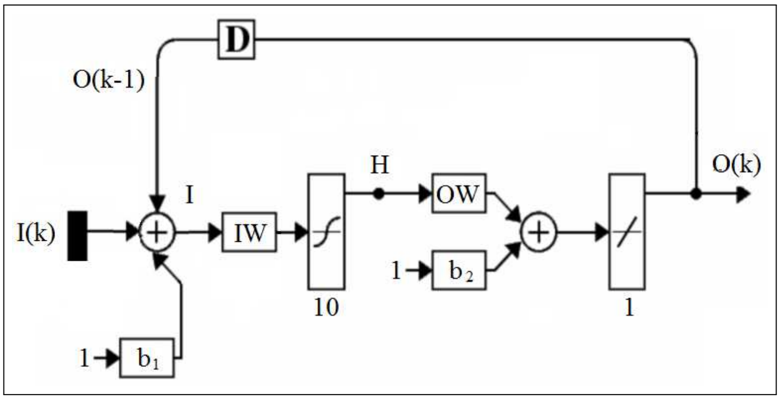

Figure 4: Structure of recurrent neural Network for time delay estimation

3 Fuzzy control

A new modelling method for nonlinear NCS with compound time is presented in this section. The process under consideration is a nonlinear discrete-time system represented by the TSK fuzzy model [26], It has the compound time $\hat{\upsilon}$ as antecedent input and linear discrete models (Eq. 7) with different sampling periods h_j as consequent output. By modelling the system dynamics in function of the compound time.

So, defining r fuzzy rules, the j-th rule has form

$$if\ \hat{\upsilon}\ is\ \alpha_j\ then\ x_j = \Phi_j x_{s,k} + \Gamma_j u_{k-1} \tag{12}$$

where $x_k \in \mathbb{R}^n$ is state vector, $u_k \in \mathbb{R}^m$ is input vector, α_j is the j-th membership function.

The overall fuzzy model is:

$$\hat{x}_k = \sum_{j=1}^{r} \psi_j \left[\Phi_j x_{s,k} + \Gamma_j u_{k-1}\right] \tag{13}$$

with the normalized fire strength ψ_j as

$$\sum_{j=1}^{r} \psi_j = 1 \quad \psi_j \geq 0 \quad \psi_j = \frac{\alpha_j}{\sum_{s=1}^{r} \alpha_s} \tag{14}$$

and

$$\alpha_j = \exp\left[-\frac{[\hat{\upsilon}-\rho_j]^2}{\sigma_j^2}\right] \tag{15}$$

α_j is a Gaussian membership function with parameters $[\rho_j, \sigma_j]$. On the other hand, $[\Phi_j, \Gamma_j]$ are the matrices of *j-th* linear discrete model discretized with a sampling period $[h_j,\ j = 1 \ldots r]$, the discrete local models are:

$$x_j = e^{[Ah_j]}x_{s,k} + \int_0^{h_j} e^{As}dsBu_k = \Phi_j x_{s,k} + \Gamma_j u_{k-1} \tag{16}$$

So, $[h_j, \rho_j, \sigma_j]$ for $j = 1, \ldots, r$ are assigned by user according to range of the compound time.

With this fuzzy model, the estimated system state is obtained by compensating the time delays, variable sampling intervals and lost packets. The action is to smoothly switch between discrete models to generate the best estimate of the state according to the estimated compound time $\hat{\upsilon}$.

Once designed the fuzzy model $\hat{x}_k$ using the estimated compound time $\hat{\upsilon}$ a fuzzy controller is proposed. This is a fuzzy feedback control law like:

$$u_k = -\sum_{j=1}^{r} \psi_j K_j \hat{x}_k \quad j = 1, \ldots, r \tag{17}$$

where K_j is the feedback matrix of the *j-th* fuzzy rule. This control law is designed like a LQR (Linear Quadratic Regulator) [33] to minimize a performance index. The control design by LQR for each local model requires the algebraic solution of the Ricatti equation for the H_j matrix. So, the feedback matrices are calculated like:

$$K_j = R_j^{-1}\Gamma_j^T H_j \quad j = 1, ..., r \tag{18}$$

The closed loop system is:

$$x_{k+1} \;=\; \sum_{i=1}^{r}\sum_{j=1}^{r} \alpha_i\beta_j \left[\Psi_i - \Gamma_i K_j\right] x_k = \sum_{i=1}^{r}\sum_{j=1}^{r} \alpha_i\beta_j \Lambda_{ij} x_k \tag{19}$$

with $\Lambda_{ij} = \Phi_i - \Gamma_i K_j \quad i = 1, \ldots, r \quad j = 1, \ldots, r$

The properties of the antecedent part (Eq. 14) are considered for the stability analysis of fuzzy control u_k, the Eq. 20 are complementary properties of fuzzy sets.

$$\begin{array}{l} \psi_i\psi_j \geq 0 \\ \sum_{i=1}^{r}\sum_{j=1}^{r} \psi_i\psi_j = 1 \\ \sum_{i=1}^{r} \psi_i^2 + 2\sum_{i,j}^{i<j} \psi_i\psi_j = 1 \end{array} \tag{20}$$

3.1 Stability analysis

Based on the properties of fuzzy control and assume that two-overlapped fuzzy memberships at most, a stability analysis of closed loop fuzzy control is presented. First, it is necessary to define the following lemma to prove stability analysis.

Lemma 1. *[8] For any real matrices A_i, B_i for $1 \leq i \leq r$, $P > 0 \in \mathbb{R}^{nxn}$, we have*

$$2\sum_{i}^{r}\sum_{j}^{r} \psi^i\psi^j A_i^T P B_j \;\;\leq\;\; \sum_{i}^{r}\sum_{j}^{r} \psi^i \left[A_i^T P A_i + B_i^T P B_i\right] \tag{21}$$

where he normalized fire strength ψ^i has the properties showed in Eq. 20 and Eq. 14.

So, the stability of the fuzzy model (13) and the fuzzy control (17) can be guaranteed if the following Theorem is fulfilled.

Theorem 1. The equilibrium state $x_e = 0$ of closed loop system with control input (19 with two-overlapped fuzzy memberships at most, is asymptotically stable in the large, if there exist μ positive definite matrices $P_s = P_s^T > 0$ such that:

$$[\Lambda_{ij} + \Lambda_{ji}]^T P_s [\Lambda_{ij} + \Lambda_{ji}] - 2P_s < 0 \quad i \in S_s \quad s = 1, \ldots, \mu \tag{22}$$

$$\Lambda_{ii}^T P_s \Lambda_{ii} - P_s < 0 \quad i \in S_s \quad j \in S_s \quad i < j \in S_s \tag{23}$$

$\Lambda_{ij} = \Phi^i - \Gamma^i K^j$ where $S = \{S_1, S_2, \ldots, S_\mu\}$ are μ regions where two fuzzy rules are fired at most (two overlapped fuzzy memberships), where S_s contains the indexes of fired membership functions in **s** region.

Proof: We suppose that there exist μ matrices $P_s = P_s^T > 0$ so (22) and (23) are satisfied. Considering a candidate Lyapunov function like:

$$V_k = \sum_{s=1}^{\mu} \lambda_s \left[x_k^T P_s x_k\right] \tag{24}$$

where

$$\lambda_s [\hat{\upsilon}] = \begin{cases} 1 & \hat{\upsilon} \in S_s \\ 0 & \hat{\upsilon} \notin S_s \end{cases} \quad \sum_{s=1}^{\mu} \lambda_s [\hat{\tau}] = 1 \tag{25}$$

It can be easily showed that $V[0] = 0$, $V_k > 0$ for $x_k \neq 0$, and $V[x] \to \infty$ as $\|x_k\| \to \infty$, it is only sufficient shows that $\Delta V[x_k] < 0$ to prove that V_k is a Lyapunov function and the theorem is fulfilled. □

So, we have:

$$\Delta V_k = V_{k+1} - V_k = \sum_{s=1}^{\mu} \lambda_s \left[x_{k+1}^T P_s x_{k+1}\right] - \sum_{s=1}^{\mu} \lambda_s \left[x_k^T P_s x_k\right]$$

By reordering and set the matrix $V_k^s = x_k^T P_s x_k$

$$\begin{aligned} \Delta V_k &= \sum_{s=1}^{\mu} \lambda_s L_s \\ L_s &= V_{k+1}^s - V_k^s \end{aligned} \tag{26}$$

It is enough to show that:

$$L_s < 0 \quad s = 1, ..., \mu \tag{27}$$

Substituting V_{k+1}^s and V_k^s in (26) we have:

$$L_s = \left[\sum_{i \in S_s} \sum_{j \in S_s} \psi_i \psi_j \Lambda_{ij} x\right]^T P_s \left[\sum_{i \in S_s} \sum_{j \in S_s} \psi_i \psi_j \Lambda_{ij} x\right] - x^T P_s x \tag{28}$$

$$L_s \leq x^T \left[\sum_{i \in S_s} \sum_{j \in S_s} \psi_i \psi_j \left[\Lambda_{ij}^T P_s \Lambda_{ji} - P_s\right]\right] x \tag{29}$$

$$\leq x^T \left[\begin{array}{l} \sum_{i \in S_s} \psi_i^2 \left[[\Lambda_{ii} + \Lambda_{ii}]^T P_s [\Lambda_{ii} + \Lambda_{ii}] - P_s\right] + \\ \sum_{i \in S_s} \sum_{j \in S_s}^{j<i} \psi_i \psi_j \left[\frac{1}{2} [\Lambda_{ij} + \Lambda_{ji}]^T P_s [\Lambda_{ij} + \Lambda_{ji}] - 2P_s\right] \end{array} \right] x \tag{30}$$

$$L_s < 0 \to \Delta V_k < 0$$

The first term in (30) is negative definite by (22). The second term is negative definite by (23). Thus, the definite positive quadratic function (24) is a Lyapunov function for the fuzzy control (17), this implicates asymptotically stability in the large. The proof of the theorem is complete.

4 Scheduling

In this section, the scheduling theory is used to design an online feedback scheduler for the NCS based on the performance of the control system and the load conditions of the communication network. The scheduler adjusts the sampling/actuation period in function of the QoC and the QoS, the next period is selected like a multiple of the period base (h) between the lower and upper bounds obtained by an analysis of the process. The idea is to keep the deadline rate and the system performance at the reference level by adjusting the transmission period. It is proposed a local fuzzy scheduler for each sensor node present in the communication network, which based on external traffic, perform a dynamic scheduling also called feedback scheduling.

The codesign controller/scheduling has as aim to minimize the effects of the time delays and lost packets using the fuzzy controller, while the fuzzy scheduling minimizes the transmission messages without degrading the process performance, modifying the sampling/actuation period into the range that the fuzzy controller guaranteed the stability.

The proposed dynamic scheduling has the configuration shown in Figure 1, its behaviour is as follows: the sensor node sends packets with the system information and its execution period. The controller node adds the error and control signal to the packet and sends it to the actuator node. The actuator computes the system performance and the deadline rate each scheduling period and sends it to the sensor node. Finally, the sensor node modifies the period h^+ based on the information of the deadline rate and the system performance. A deadline occurs when a packet is upgraded after a time limit h_{max} or when the packet is lost.

In terms of control, the manipulated variable is the actuation period and the controlled variables are the deadline rate and system performance.

The actuator node module calculates the system operation Δe through the mean absolute error (MAE) of n received packets with a scheduling period δ. While the deadline rate Δh is calculated with the m packets received that have exceeded their deadline h_{max} plus the lost packets ξ during the scheduling period.

$$\begin{aligned} \Delta e &= \frac{\sum_{k\in n}|e_k|}{n} \\ m &= \{\forall k\in n|\tau_k > h_{max}\}+\xi \\ \Delta h &= \frac{m}{\delta}\sum_{k\in n} h_k \end{aligned} \tag{31}$$

The deadline rate as a controlled variable is a common metric for the quality of service (QoS). from a real-time point of view is also an important factor that degrades the quality of control (QoC). The QoS controls the number of lost deadlines at an acceptable low level.

In addition, The Δh as a controlled variable can simultaneously address the problems of variable time delays and lost packets. When Δh is kept at a low level, the delays of most packets are less than the deadline and the number of packets lost is limited. As a consequence, the impact of the delay and the lost packets in the QoC is minimized.

The sampling period affects the lost deadline rate, with short sampling periods increasing network utilization, which inherently causes that the network imperfections increase and vice versa. With heavy traffic load, the probability of collisions between nodes is greater, which potentially increases both the time delay and the lost packets and at the same time the lost deadline rate. On the other hand, a largely lost deadline rate can generally be reduced by increasing the sampling period, particularly when the network is overloaded.

According to the control theory of sampled data, short sampling periods generate better QoC. In this context, QoC can be improved by increasing the utilization efficiency by adjusting the sampling period. This justifies the choice of the sampling period as a manipulated variable that is adjusted with respect to the network conditions, where variations in the unpredictable and dynamic traffic load in the NCS can be effectively compensated.

With the metric of the QoC (Δe) and QoS (Δh), the fuzzy scheduler is designed as a control problem, where the metrics form the antecedent part and the consequent part are parallel feedback matrices. The i-th fuzzy rule has the form of:

$$if\ \Delta e\ is\ \beta_{1j}\ and\ \Delta h\ is\ \beta_{2j}\ then\ \bar{h} = Fz \tag{32}$$

where β are the membership functions and $z \in \mathbb{R}^2$ with $z = [\Delta e\ \Delta h]$, F is the feedback scheduling matrix. the overall fuzzy scheduler is:

$$\bar{h} = \sum_{i=1}^{M} \prod_{j=1}^{2} \beta_{ij} F_i z \tag{33}$$

where M is the number of fuzzy rules. The new sampling period is assigned in the interval:

$$h^+ = \{h_{min} \le \bar{h} \le h_{max}\} \tag{34}$$

Given the absence of a mathematical model that describes the relationship between the lost deadline rate, the EAM and the sampling period, the bounds and the fuzzy scheduler are determined based on experimentation.

4.1 Analytical codesign

If the fuzzy model is locally controllable, i.e. (Φ_i, Γ_i), $i = 1, ..., l$, are controllable pairs, the feedback control matrices K_i, $(i = 1, 2, ..., l)$ can be calculated using eq. (18), with matrices R_i and Q_i to get a desired performance.

The procedure of the fuzzy codesign is as follows:

Step 1: Define the intervals of the compound time based on the measurements of the network to define the bounds of the compound time.

Step 2: For the plant (eq. 1), define the h_i actuation periods for each F_i matrix in the fuzzy scheduler. Define the membership functions for the control error Δe and deadline rate Δh.

Step 3: Discretize the model (eq. 1) for each h_i to set Γ_i and Φ_i in the fuzzy controller.

Step 3: Verify that all local discrete systems are controllable. That is, $rank(\Gamma_i, \Phi_i\Gamma_i, ..., \Phi_i^{n-1}\Gamma_i) = n, \quad i = 1, ..., l.$

Step 4: Calculate the feedback matrices K_i of each local system via LQR assigning the matrices R_i and Q_i according to the desired performance (Eq. 18).

Step 5: Find all the μ regions with almost two overlap rules and apply Theorem 4.1 to check the stability of the closed-loop. If the system is not stable, go back to step 4 to reassign the matrices R_i and Q_i.

5 Experiments and results

The fuzzy codesign approach is tested in a nonlinear MIMO system with an Ethernet network, the experiment generated variable traffic into the network with external nodes. The performance is compared with a Hybrid controller [11].

5.1 Case study

The case study is a MIMO nonlinear, open-loop unstable and time-varying system. It is a 2-DOF helicopter system integrated to an Ethernet network (Figure 5), detailed information can be found in [19]. The sensor and actuator nodes are Pentium 4 with 1028 Mb RAM with an INTEL 10/100 Mb Ethernet card, each has an XPC target operative systems and are connected through a switch, the controller is an embedded system with a microcontroller board based on the ATmega32u4 and the Atheros AR9331. The Atheros processor uses Linino OS a Linux distribution. The board has built-in Ethernet slot, a 16 MHz crystal oscillator. The sensor node has an A/D 10 bits resolution and the actuator has a D/A 8 bits resolution. The tasks for the NCS are set as follows, *5 ms* as minimum sampling period of sensor and actuator nodes, the controller node is driven by event.

The experiment consists of a 2D helicopter simulator mounted on a fixed base with two propellers that are driven by DC motors as is shown in Figure 5. The front propeller controls the elevation of the helicopter nose about the pitch axis and the back propeller controls the motion about the yaw axis. The pitch and yaw angles are measured using high-resolution encoders.

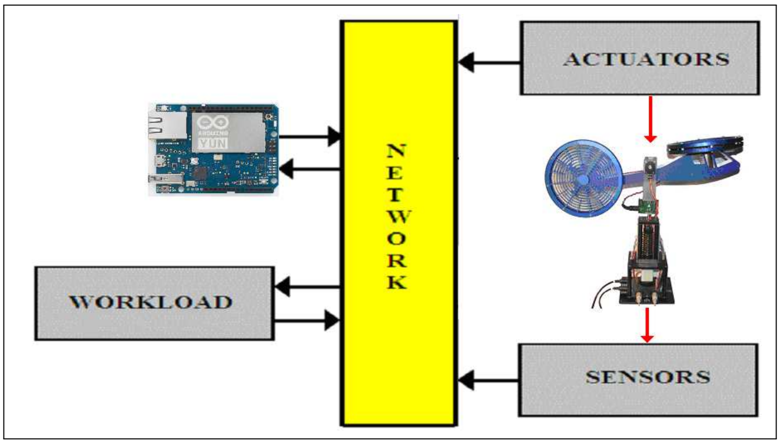

Figure 5: Networked Control System with workload

The Euler-Lagrange method is used to derive the nonlinear equations describing the motions of the helicopter [19]. From its nonlinear equations of motions, the linear continuous state space models with $x = [\theta \ \psi \ \dot{\theta} \ \dot{\psi}]$ and $u = [V_p \ V_y]$ are

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \end{aligned} \tag{35}$$

$$
\begin{aligned}
A &= \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{B_p}{J_p+ml^2} & 0 \\ 0 & 0 & 0 & -\frac{B_y}{J_y+ml^2} \end{bmatrix} \\
B &= \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \frac{K_{pp}}{J_p+ml^2} & \frac{K_{py}}{J_p+ml^2} \\ \frac{K_{yp}}{J_y+ml^2} & \frac{K_{yy}}{J_y+ml^2} \end{bmatrix} \\
C &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}
\end{aligned} \tag{36}
$$

where K_{pp} is the thrust torque acting on pitch axis from pitch motor, K_{py} the thrust torque acting on yaw axis from pitch motor, K_{yp} the thrust torque acting on pitch axis from yaw motor, K_{yy} the thrust torque acting on yaw axis from yaw motor, B_p the viscous damping coefficient about pitch axis, B_p the viscous damping coefficient about yaw axis, l the center of mass length along helicopter bod, g the gravity constant, m the total moving mass of the helicopter, J_p the total moment of inertia about pitch pivot, J_y the total moment of inertia about yaw pivot, V_p the voltage of pitch motor, V_y voltage of yaw motor, θ the angle about pitch axis and ψ the angle about yaw axis.

5.2 Results

The fuzzy model is obtained discretizing the helicopter model (Eqs. (35)-(36)) with different sampling-actuation periods $h_j = [5,\ 10,\ 15,\ 20]\ ms$, the compound time is previously analyzed generating the bounds $\upsilon = [5 - 300]\ ms$. The fuzzy model has four rules with the antecedent parameters

$$\rho = [5,\ 10,\ 15,\ 20]x10^{-3}$$
$$\sigma = [12,\ 24,\ 30,\ 30]x10^{-4}$$

The fuzzy controller is designed by LQR method and the Lyapunov conditions are fulfilled (22)-(23).

$$K = \begin{bmatrix} 15.4 & 1.53 & 4.91 & 0.677 & 12.2 & 0.718 \\ -1.97 & 17.3 & -0.241 & 6.2 & -1.24 & 7.03 \end{bmatrix}$$

The fuzzy scheduler is designed with the sampling-actuation periods $h_j = [5,\ 10,\ 15,\ 20]\ ms$. The range of the lost deadlines is $\Delta e = [0,\ 1]$, the MAE is $\Delta h = [0,\ 1.4]$ and the scheduling period $\delta = 1\ s$.

The network imperfections and the system performance were measured to show the codesign approach with a square reference. The overall system performance is compared with a hybrid control designed to be stable in the compound time range.

The network behaviour with medium traffic generates a peculiar behaviour in the delay and the loss of deadlines (6). The time delay increases at the start of traffic with a maximum of 31 ms and average of 5 ms, but as the traffic increases, delays become deadlines with a 5.8% and average of 132 ms between missed deadlines. The upper graph shows the time delay where the average traffic starts in 50 s, however, the effect starts seconds later and decreases around the second 55, giving rise to an increase in the lost deadlines (80 s), but its effect extends to the 85s because the switch continues to empty the queue.

In this case, the fuzzy scheduler change the assignment policy for the sampling-actuation period (7) (inf.), When medium traffic appears, the EAM (sup.) had variations greater than 0.1,

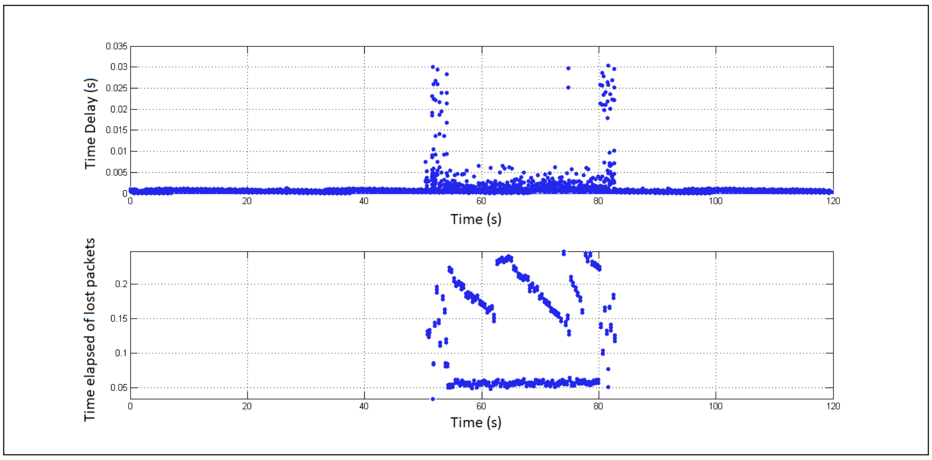

Figure 6: Time delays and time elapsed between lost packets with medium traffic (50 - 80 s).

which represents an error in steady state greater than 2% , and added to the presence of 5.8 of lost deadlines (med.), the fuzzy scheduler decides to decrease the sampling period slowly but without overloading the network and gradually improve the performance of the system.

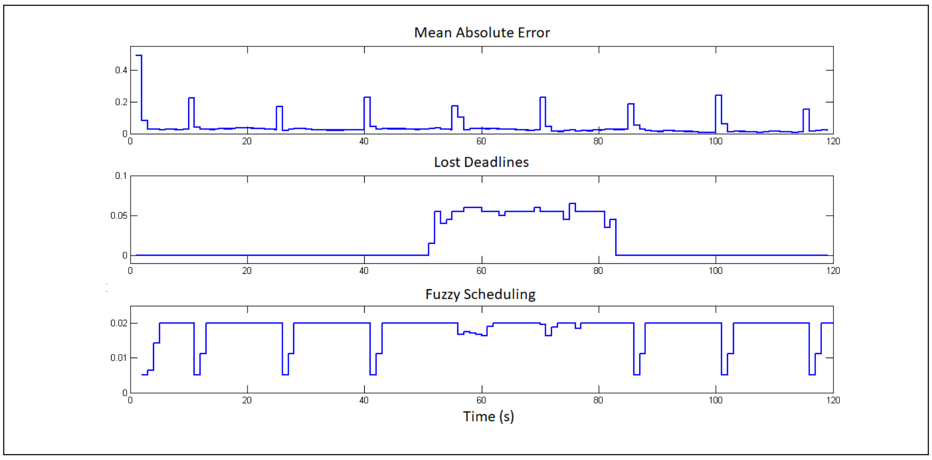

Figure 7: MAE, lost deadlines and the sampling-actuation period with medium traffic.

With medium traffic, the fuzzy codesign and the hybrid control remain stable (Fig. 8), but the fuzzy codesign (sup.) is the one that fulfils the control criteria. The codesign (sup.) with medium traffic has an overshoot of $\zeta = 15°$, with a stable steady error $e_s s = 0.5$ and an setting time $T_s = 4$ s.; while the hybrid control (inf.) degrades its performance against average traffic with an overshoot of $\zeta = 34°$, with a stable steady error $e_s s = 6$ and with oscillations. At this level of traffic, the hybrid control ensures system stability but with poor performance.

In the case of heavy traffic (Figure 3), the time delay and the lost deadlines were increased, the maximum delay was 30 ms, with an average of 12 ms, and deadlines were 7.3% and the average time of 122 ms between lost deadlines. The heavy traffic started at 50 s, however, its effect was visible until the 85 s even when the traffic was finished in the 80 s.

Figure 9 shows the pitch position for both the fuzzy codesign and the hybrid controller. In case of the heavy traffic measured (Figure 3), the hybrid controller has an erratic behavior with an overshoot $\zeta = 36°$, a steady stable error $e_s s = 10$ and setting time $T_s > 9$ s. While, the fuzzy codesign curves indicate a stable behavior with an overshoot $\zeta = 30°$, a steady stable error

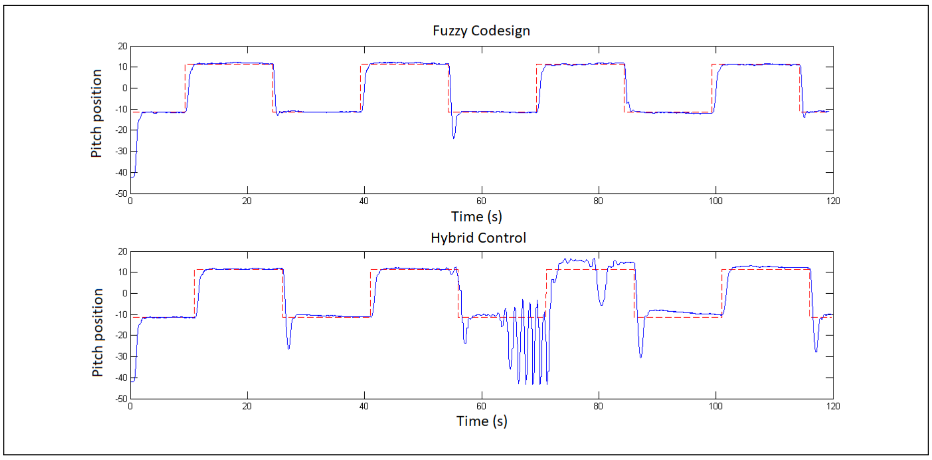

Figure 8: Comparison between the codesign and an hybrid controller with medium traffic.

$e_s s = 0.8$ and setting time $T_s = 2$ s.

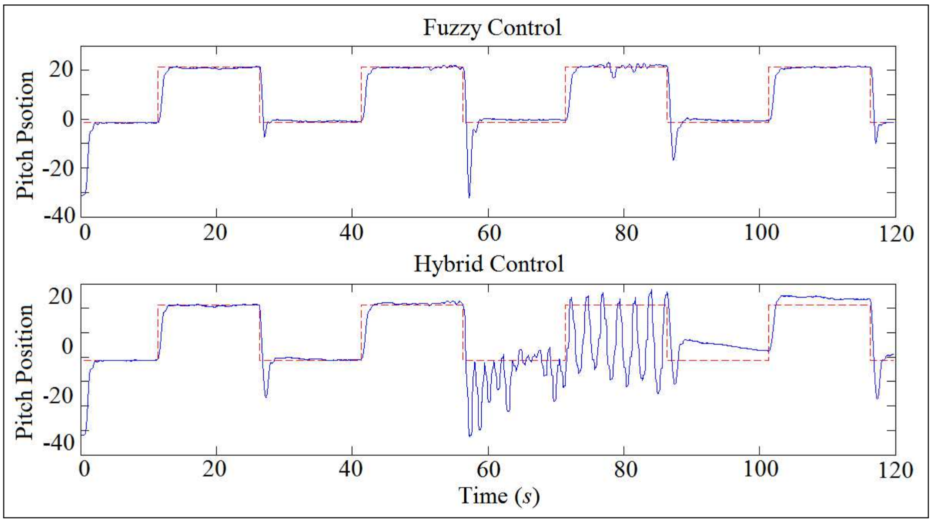

Figure 9: Comparison between the codesign and an hybrid controller with heavy traffic.

The fuzzy scheduler changes the sampling period (Figure 10), Because the MAE (sup.) has variations greater than 0.25 representing a steady-state error greater than 5%, And added to the presence of 7.8% of lost deadlines (med.), The fuzzy scheduler decides to slow down the sampling period in order to correct the error even when traffic on the network increases. This demonstrates that the fuzzy codesign can apply a dynamic control dependent on the network behaviour with a stable design in all the range of υ.

6 Conclusion

A fuzzy codesign approach was presented to minimize the effects of the network-induced imperfections. The approach was designed with a controller and scheduler together in function of the network imperfection measurements and the continuous model of the system. The fuzzy controller is designed to select the control signal depending on the compound time estimation, with multiple discrete model that represents the process in function of the actuation periods dis-

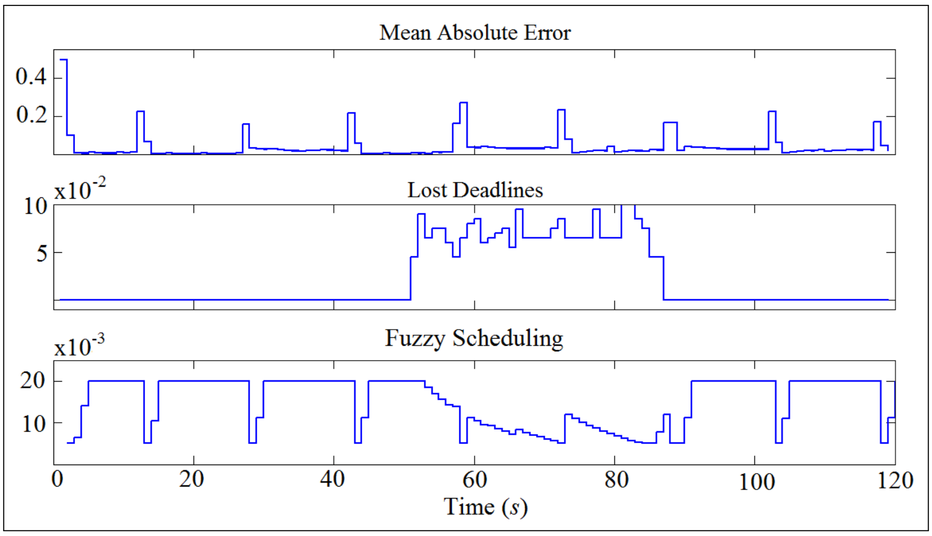

Figure 10: MAE, lost deadlines and the sampling-actuation period with heavy traffic.

appeared the sampling and communication jitter. The fuzzy scheduler was designed to control the sampling-actuation period in function of the system performance and the behaviour of the network. The codesign approach is applied to a nonlinear, time-varying MIMO system interconnected with an Ethernet network and employing an embedded system as the controller-scheduler. The codesign performance was compared with a hybrid control designed to counteract the effects of delay in the same range as codesign. The fuzzy codesign had the best performance within the entire range of compound time.

Acknowledgement

The authors would like to thank DCC-IIMAS-UNAM for the support and for their technical support to Sandra Sauza Escoto and Joaquin Morales Rosales. This work was supported by UNAM-PAPIIT IA104218.

Bibliography

[1] Albertos, P.; Crespo, A. (1999); Real-time control of non-uniformly sampled systems, *Control Engineering Practice*, 7(4), 445-458, 1999.

[2] Aström, K.J. (2007); *Event Based Control, In Analysis and Design of Nonlinear Control Systems: In Honor of Alberto Isidori*, Springer Verlag, 2007.

[3] Benitez-Perez, H. et al. (2016); A Fuzzy Networked Control System Following Frequency Transmission Strategy, *International Journal of Computers Communications & Control*, 11(1), 11-25, 2016.

[4] Chae, S.; Nguang, S. K. (2014); SOS based robust H∞ fuzzy dynamic output feedback control of nonlinear networked control systems, *IEEE Trans. Cybern.*, 44(7), 1204-1213, 2014.

[5] Chai, S.; Liu, G.P.; Rees, D.; Xia Y. (2008); Design and practical implementation of internet-based predictive control of a servo system, *IEEE Trans. on Control Systems Technology*, 16(1), 158-168, 2008.

[6] Chow, M.-Y.; Tipsuwan, Y. (2003); Gain adaptation of networked DC motor controllers based on QoS variations, *IEEE Trans. on Industrial Electronics*, 50(5), 936-943, 2003.

[7] Du, D. (2012); Reliable H∞ control for Takagi-Sugeno fuzzy systems with intermittent measurements, *Nonlinear Analysis: Hybrid Systems*, 6(4), 930-941, 2012.

[8] Guan, X.P.; Chen, C. L. (2004); Delay-dependent guaranteed cost control for TS fuzzy systems with time delays, *IEEE transactions on fuzzy systems*, 12(2), 236-249, 2004.

[9] Hespanha, J.; Naghshtabrizi, P.; Xu, Y. (2007); A survey of recent results in networked control systems, *Proc. IEEE*, 138-162, 2007.

[10] Hu, H.; Liu, G.; Rees, D. (2007); Event-driven networked predictive control, *IEEE Transactions on Industrial Electronics*, 54(3), 1603- 1613, 2007.

[11] Ji, K.; Kim, W.J. (2007); Stochastic optimal control and network co-design for networked control systems, *International Journal of Control, Automation and Systems*, 5(5), 515-525, 2007.

[12] Ji, K.; Kim, W.J. (2007); Robust control for networked control systems with admissible parameter uncertainties, *International Journal of Control, Automation and Systems*, 5(4), 372-378, 2007.

[13] Li, H.; Sun, Z.; Liu, H.; Chow, M.Y. (2008); Predictive observer-based control for networked control systems with network-induced delay and packet dropout, *Asian Journal of Control*, 10(6), 1-3, 2008.

[14] Li, K.; Baillieul, J. (2004); Robust quantization for digital finite communication bandwidth (DFCB) control, *IEEE Trans. on Automatic Control*, 49(9), 1573-1584, 2004.

[15] Li, H.; Wu, C.; Feng, Z. (2015); Fuzzy dynamic output-feedback control of non-linear networked discrete-time system with missing measurements, *IET Control Theory and Applications*, 9(3), 327-335, 2015.

[16] Lozoya, C.; Marti, P.; Velasco, M.; Fuertes, J.M. (2008); Control Performance Evaluation of Selected Methods of Feedback Scheduling of Real-time Control Tasks, *17th IFAC World Congress*, July, 2008.

[17] Marti, P.; Velasco, M. (2007); Toward Flexible Scheduling of Real-Time Control Tasks: Reviewing Basic Control Models, *10th International Conference on Hybrid Systems, Computation and Control, LNCS*, 2007.

[18] Mendez-Monroy, P.E.; Velasco, M.; Fuertes, J.M.; Benitez-Perez, H. (2012); Fuzzy observer based Fault Detection for Network Control Systems with Periodic Actuation Tasks, *8th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes (SafeProcess 2012)*, 2012.

[19] Mendez-Monroy, P. E.; Benitez-Perez, H. (2011); Fuzzy control with estimated variable sampling period for non-linear networked control systems: 2-DOF helicopter as case study, *Transactions of the Institute of Measurement and Control*, 34(7), 802-814, 2011.

[20] Mendez-Monroy, P. E. (2017); Fuzzy codesign for networked control systems, *14th International Conference on Electrical Engineering, Computing Science and Automatic Control*, DOI: 10.1109/ICEEE.2017.8108887, 2017.

[21] Montestruque, L.A.; Antsaklis, P.K. (2007); Static and dynamic quantization in model-based networked control systems, *International Journal of Control*, 80(1), 87-101, 2007.

[22] Nilsson, J.; Bernhardsson, B.; Wittenmark, B. (1998); Stochastic analysis and control of real-time systems with random time delays, *Automatica*, 34(5), 57-64, 1998.

[23] Park, J.; Kim, J.; Park, D. (2001); LMI-based design of stabilizing fuzzy controllers for nonlinear systems described by Takagi-Sugeno fuzzy model, *Fuzzy Sets and Systems*, 122, 73-82, 2001.

[24] Peng, C.; Yang, T.C. (2010); Communication-delay-distribution-dependent networked control for a class of T-S fuzzy systems, *IEEE Transaction on Fuzzy Systems*, 18(2), 326-335, 2010.

[25] Seiler, P.; Sengupta, R. (2005); An H∞ approach to networked control, *IEEE Transaction on Automatic Control*, 50(3), 356-364, 2005.

[26] Tanaka, K.; Wang, H. O. (2001); *Fuzzy Control Systems Design and Analysis: A Linear Matrix Inequality Approach*, Wiley & Sons, Inc., 2001.

[27] Tipsuwan, Y.; Chow, M.Y. (2003); Control methodologies in networked control systems, *Control Engineering Practice*, 11, 1099-1111, 2003.

[28] Tipsuwan, Y.; Chow, M. Y. (2004); On the Gain Scheduling for Networked PI Controller Over IP Network, *IEEE/ASME Transaction Mechatronics*, 9(3), 491-498, 2004.

[29] Tong, S.W.; Qian, D.W.; Liu, G.P. (2014); Networked Predictive Fuzzy Control of Systems with Forward Channel Delays Based on a Linear Model Predictor, *International Journal of Computers Comnunications & Control*, 9(4), 471-481, 2014.

[30] Yi, H.C.; An C.J.; Choi J.Y. (2017); Compensation of Time-Varying Delay in Networked Control System over Wi-Fi Network, *International Journal of Computers Comnunications & Control*, 12(3), 415-428, 2017.

[31] Yu, M.; Wang, L.; Chu, T.G., Xie, G.M. (2004); Stabilization of networked control systems with data packet dropout and network delays via switching system approach, *Proc. of the 43th IEEE Conference on Decision and Control*, 3539-3544, 2004.

[32] Zhang, W.; Branicky, M.S.; Phillips, S.M. (2001); Stability of networked control systems, *IEEE Cont. Sys. Mag.*, 21(1), 84-99, 2001.

[33] Zhang, H.; Duan, G.; Xie, L. (2007); Linear quadratic regulation for linear time-varying systems with multiple input delays, *Automatica*, 42(9), 1465-1476, 2007.

[34] Zhao, Y.; Gao, H.; Chen, T. (2010); Fuzzy constrained predictive control of non-linear systems with packet dropouts, *IET control theory and applications*, 4(9), 1665-1677, 2010.