Fuzzy control of network control Systems with periodic actuation tasks

Paul E. Méndez-Monroy
Posgrado de Ingenieria-UNAM
Héctor Benitez-Pérez
IIMAS-UNAM
{paul, hector}@uxdea4.iimas.unam.mx

Manel Velasco
Universidad Politécnica de Cataluña
Josep M. Fuertes
Universidad Politécnica de Cataluña
{manel.velasco, josep.m.fuertes}@upc.edu

Abstract— Varying time delays and variable sampling interval degrade the performance of control loops closed over communication networks, (Networked Control System, NCS). To compensate the negative effects of network, in this paper we propose a fuzzy controller for networked control systems with synchronization at the actuation node. Using the one-shot model is obtained some benefits such as consider delays longer than sampling period instead of considering delays less than sampling period and variable sampling interval.

I. INTRODUCTION.

Control loops that are closed over communication networks are known as networked control systems (NCS) (Tipsuwan, 2003). In these systems, the nodes such sensors, controllers and actuators exchange control data through networks. The purpose of the network is to transmit control packets in a reliable, secure, and efficient fashion.

However, a network integrated in the control system also introduces new challenges, caused by the packet-based data exchange between network nodes. Therefore, control algorithms are needed that can handle the communication imperfections and constraints caused by the communication (Hespanha *et. al*, 2007) (Tipsuwan, 2003) (Yang, 2006) (Zhang *et. al*, 2001).

In general, communication imperfections and constraints can be categorized into five types:

(i) Quantization errors in the transmitted signals, due to the finite word length of the transmitted packets.
(ii) Packet loss, due to unreliable transmissions.
(iii) Variable sampling/transmission intervals.
(iv) Variable transmission delays.
(v) Communication constraints, i.e., not all sensor and actuator signals can be transmitted at the same time.

Any of these phenomena can degrade closed-loop performance or, even worse, can harm closed-loop stability of the control system. So, it is important to know how these effects influence the stability properties.

To compensate the negative effects into network, existing research focuses in control approaches with mathematic models with periodic sampling (Zhang *et. al*, 2001) (Liu, 2004).

The key aspect of the design procedure in those approaches describes the behavior of the analog plant at the sampling instants. Moreover, the actuation instants are defined in terms of the sampling instants. Therefore, the synchronism is given by the sampling instants. Once a sample is taken, the control command is computed assuming that the next sample will occur after h time units (i.e., one sampling period), and assuming that the actuation will occur after τ time units (i.e., one time delay). The operation of those models is usually violated in networked control loops because varying time delay insert time unpredictability from sampling to actuation.

A more sophisticated approach is found in (Hu *et. al*, 2007) where in a NCS fully distributed, the controller node computes several control commands that are sent to the actuator node. The latter, applies to the plant the adequate one according to time-stamps logics. This approach however increases network traffic and computation overhead. Alternatively, in (Henzinger *et. al*, 2001) the use of synchronized I/O operations is proposed. Although this solution removes the time variability, in forces artificial input-output latencies within the loop.

Rather than using solutions based in mathematic models with periodic sampling, we propose to use an equivalent model but synchronized at the actuation instants.

The analysis presented next extends previous work (Martí, 2007), which was earlier suggested in (Albertos, 1999), where a mathematical control model with synchronization at the actuation instants was presented. This model establishes new operation rules for control algorithms.

In this paper we propose an approach to design networked control systems where the key aspect is that all the control operations considered periodic actuation rather than assuming periodic sampling.

Therefore, consecutive actuation instants mark the sampling period. The benefit of this approach is that the harmful effects that variable delays and sampling intervals within control loops have in stability/performance are removed if the controller is designed according to these new operation rules. Note also that samples are not required to be strictly periodic. This also facilitates the implementation of control systems where sampling periods are not constant such as in the case of feedback scheduling approaches (Lozoya, 2008), or event based control systems (Aström, 2007).

The requirement of the proposed solution is an operation on absolute time measurements, and therefore requires accurate clock synchronization among the networked nodes. Taking advantage of the IEEE1588 Precision Time Protocol (PTP) standard (IEEE 1588, 2002), clock synchronization is guaranteed.

The rest of this paper is structured as follows. Section II presents the one-shot model to define the tasks execution of

NCS. Section III presents the delay estimation. Section IV presents the fuzzy control for networked control systems with synchronization at the actuation instants. Section V presents a simulation to show its application. Finally, section VI concludes the paper.

II. ONE-SHOT MODEL

A common architecture for a single control loop in a NCS is shown in Fig. 1, where sensor, controller and actuator nodes exchange data via network communication. Sensor node is driven by periodic time, controller node is driven by event and Actuator is driven by strictly periodic time.

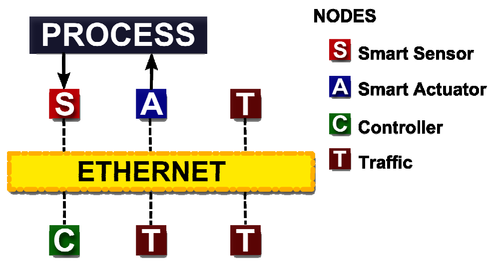

Figure 1. Structure of NCS with traffic nodes

This model estimates the states for the next actuation, so it isn't get a strictly sampling period or estimate the true time delay, just it is necessary an estimated of the time delay.

Consider the traditional mathematical description of a process given by the state-space model of a linear time-invariant discrete-time system with sampling period h (Aström, 1997)

$$\begin{aligned} x_{k+1} &= \Phi_h x_k + \Gamma_h u_k \\ y_k &= C x_k \end{aligned} \quad (1)$$

where x_k is the plant state, u_k and y_k are the inputs and outputs of the plant, matrix $C \in \mathbb{R}^{pxn}$ is the output matrix and matrices Φ_h and Γ_h are obtained using

$$\Phi_h = e^{Ah}, \quad \Gamma_h = \int_0^h e^{As} B ds \quad (2)$$

with $A \in \mathbb{R}^{nxn}$ and $B \in \mathbb{R}^{mxn}$ are the system and input matrices of the continuous-time form

$$\begin{aligned} \dot{x}(t) &= Ax + Bu(t) \\ y(t) &= Cx(t) \end{aligned} \quad (3)$$

For standard closed-loop operation of (2), the control command u_k is given by

$$u_k = Lx_k, \quad L \in \mathbb{R}^{1xn} \quad (4)$$

where L is the state feedback gain obtained using standard control design methods from matrices Φ_h and Γ_h.

The application of (4) to the plant mandates computing the control command with *zero* time. This is physically impossible even for processor-based closed-loop systems because executing the control algorithm takes time.

Model (1) can be augmented to cope with a time delay modeling an I/O latency that appears due to the computation of the control algorithm or due to the insertion of a network within a control loop, as in the case of NCS. The standard model that incorporates a time delay τ, with $\tau < h$, is (Aström, 1997)

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

Model (5) has been often taken as the underlying simple 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.

But if the time delay is variant, Langer than sampling periodic or the sampling interval is variant, this models is useless for NCS.

So, we propose the one-shot model, a mathematic model that synchronizes the operation of each control loop at the actuation instants. Hence, the time elapsed between consecutive actuation instants, named t_{k-1} and t_k, is the sampling period, h. Within this time interval, the system state is sampled, named $x_{s,k} \in (t_{k-1}, t_k)$, and the sampling time recorded, $t_{s,k}$. The difference between this time and the next actuation time

$$\tau_k = t_k - t_{s,k} \quad (6)$$

is used to estimate the state at the actuation instant as

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

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

$$u_k = L\hat{x}_k, \quad L \in \mathbb{R}^{1xn} \quad (8)$$

where L is the controller gain of fuzzy control showed in next section. The control command u_k is held constant within actuation instants.

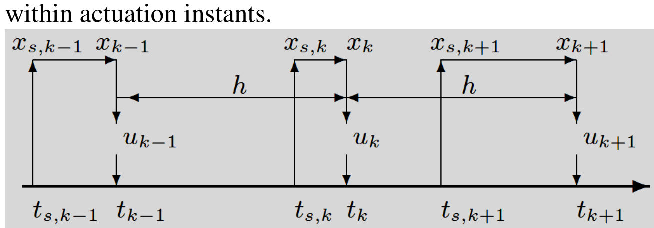

Figure 2. One-shot model

A control strategy using (6)-(8) relies on the time reference given by the actuation instants. In addition, samples are not required to be strictly periodic because τ_k in (6) can vary at each closed-loop operation. The interested reader is referred to (Martí, 2007) for further reading on this control model.

At each actuation, the actuator node, after applying to the plant the control signal, e.g. u_{k-1}, generates the next actuation instant t_k, which is sent to the sensor. The sensor, upon reception of this message or next sampling period, samples the plant $x_{s,k}$ and records the absolute sampling time $t_{s,k}$. The latter, together with t_k is used to compute τ_k. Both $x_{s,k}$ and τ_k are sent to the controller node. Upon reception of this message, the controller node estimates the plant state that will apply at t_k (7) with the next estimated time delay τ_{k+1}, and computes the control command u_k (8). The latter is sent to the actuator that will apply it to the plant at the synchronized actuation instant.

III. VARIABLE TIME DELAY

Time delays and variable sampling interval degrade the performance of a NCS. So, it is important to analyze their behavior focusing to obtain a model that allows a compensation action. However, this model should be simple to be estimated in real time providing a reasonable accuracy.

The NCS architecture (Figure 1) has four types of nodes into an Ethernet network: sensor, controller, actuator and traffic nodes. A control loop considers communication between sensor – controller and controller – actuator nodes. Traffic nodes send periodic or sporadic packets into network, for example, other control loops or monitoring.

All nodes send UDP (web, 1) (User Datagram Protocol) packets to avoid double traffic into network, but it is not possible know if there are packets loss or a maximum time delay because there is no acknowledgement that the send packet had been received.

In this paper, network-induced time delays and variable sampling intervals less than a sampling period are compensated by the one-shot model and the fuzzy model compensates time delays longer than a sampling period.

In the controller is estimated the time delay, when the control node receives a new packet, this contains the state $x_{s,k}$ and the time delay τ_{k-1}. Once time delay τ_{k-1} is recieved the exponential distribution algorithm (Tipsuwan, 2004) is used to estimate the next time delay $\hat{\tau}_k$, this algorithm is widely used to estimate delays in real time, where an offline statistical analysis characterizes mean and standard deviation $q_E = [\eta, \phi]$ of time delays data τ with multiple scenarios of traffic, those are used to form a generalized exponential distribution with a probability density function:

$$P[\tau_k] = \begin{cases} \frac{1}{\phi} e^{-(\tau_k - \eta)/\phi}, & \tau_k \geq \eta \\ 0, & \tau_k < \eta \end{cases} \tag{9}$$

This function $P[\tau_k]$ is calculated for each w previous time delays $\mathbf{T} = \{\tau_{k-w}, \ldots, \tau_{k-2}, \tau_{k-1}\}$, where the expected value of $\hat{\tau}_k$ is $E[\tau_{k|w}] = \eta_{k+1|w} + \phi_{k+1|w}$. First, it is obtained $q_E = [\eta, \phi]$ by statistical analysis, so the new mean $\eta_{k+1|w}$ is the pervious time delay τ_i with the maximum value of the probability function (3) and the new standard deviation $\phi_{k+1|w}$ is the root of the variance's window.

$$\eta_{k+1|w} = \mathrm{T}_i \mid P_{\max_i}[\mathrm{T}] \tag{10}$$

$$\phi_{k+1|w} = \sqrt{\mathrm{var}(\mathrm{T})} \tag{11}$$

so, the next time delay $\hat{\tau}_k$ is estimated as:

$$\hat{\tau}_k = \eta_{k+1|w} + \phi_{k+1|w} \qquad k = 2,3,... \tag{12}$$

Once the next time delay $\hat{\tau}_k$ of system is estimated, it is used to generate a control signal through a fuzzy controller designed in section IV.

IV. FUZZY CONTROL

The fuzzy model (7) is TSK type (Tanaka, 2001) with the time delay $\hat{\tau}_k$ as input of antecedent part and linear discrete models with different sampling periods h_j as consequent part, those are used to model the system dynamics if the estimated time delay is h_j.

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

$$\textit{if} \quad \hat{\tau}_k \quad \textit{is} \quad \alpha_j(\hat{\tau}) \quad \textit{then} \quad \mathbf{x}_j = \mathbf{\Phi}_{h_j} \mathbf{x}_{s,k} + \mathbf{\Gamma}_{h_j} \mathbf{u}_k \tag{13}$$

where $\mathbf{x}(k) \in \mathbb{R}^n$ is state vector of system, $\mathbf{u}(k) \in \mathbb{R}^m$ is input vector of process, $\alpha_j(\hat{\tau})$ is the *j-th* membership function of estimated time delay $\hat{\tau}_k$.

The overall fuzzy model is:

$$\hat{\mathbf{x}}_{k+\hat{\tau}_k} = \sum_{j=1}^{r} \psi_j \left(\mathbf{\Phi}_j \mathbf{x}_{s,k} + \mathbf{\Gamma}_j \mathbf{u}_k \right) \tag{14}$$

where the normalized fire strength ψ_j is:

$$\psi_j = \frac{\alpha_j}{\sum_{s=1}^{r} \alpha_s} \qquad \psi_j \geq 0 \qquad \sum_{j=1}^{r} \psi_j = 1 \tag{15}$$

and $\alpha_j = \exp\left(-\left(\hat{\tau}_k - \rho_j\right)^2 / \sigma_j^2\right)$ is a Gaussian membership function with parameters (ρ_j, σ_j), $(\mathbf{\Phi}_j, \mathbf{\Gamma}_j)$ are the matrices of *j-th* linear discrete model discretized with a sampling period h_j $\quad j = 1 \ldots r$, the discrete local models are:

$$\mathbf{x}_j = e^{(\mathbf{A}h_j)} \mathbf{x}_k + \int_0^{h_j} e^{\mathbf{A}s} ds \mathbf{B} \mathbf{u}_k = \mathbf{\Phi}_j \mathbf{x}_k + \mathbf{\Gamma}_j \mathbf{u}_k \tag{16}$$

so, (h_j, ρ_j, σ_j) for $j = 1, \ldots, r$ are assigned by user according to offline time delay measurement.

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

Once designed the fuzzy model (13) using the estimated time delay $\hat{\tau}_k$ a fuzzy controller is proposed. This is a fuzzy feedback control law like:

$$u_k = -\sum_{j=1}^{r} \psi_j \mathbf{K}_j x_k \qquad j = 1, \ldots, r \tag{17}$$

where $\mathbf{K}_j$ is the feedback matrix of the *j-th* fuzzy rule. This control law is designed like a LQR (Linear Quadratic Regulator) (Zhang, 2007) to minimize a performance index. The control design by LQR for each local model requires the algebraic solution of the Ricatti equation for the $\mathbf{H}_j$ matrix. So, the feedback matrices are calculated like:

$$\mathbf{K}_j = \left(\mathbf{R}_j + \mathbf{\Gamma}_j^T \mathbf{H}_j \mathbf{\Gamma}_j\right)^{-1} \mathbf{\Gamma}_j^T \mathbf{H}_j \mathbf{\Phi}_j \qquad j = 1,\ldots,r \qquad (18)$$

The closed loop system is:

$$x_{k+1} = \sum_{i=1}^{r}\sum_{j=1}^{r} \alpha_i \beta_j \left(\mathbf{\Phi}_i - \mathbf{\Gamma}_i \mathbf{K}_j\right) x_k = \sum_{i=1}^{r}\sum_{j=1}^{r} \alpha_i \beta_j \mathbf{\Lambda}_{ij} x_k \quad (19)$$

with $\mathbf{\Lambda}_{ij} = \mathbf{\Phi}_i - \mathbf{\Gamma}_i \mathbf{K}_j \;\; i = 1,\ldots,r \;\;\; j = 1,\ldots,r$

The following proprieties of the antecedent part (15) are considered for the stability analysis of fuzzy control (17):

$$\psi_i \psi_j \geq 0 \quad \sum_{i=1}^{r}\sum_{j=1}^{r} \psi_i \psi_j = 1 \quad \sum_{i=1}^{r} \psi_i^2 + 2\sum_{i,j}^{i<j} \psi_i \psi_j = 1 \quad (20)$$

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

Lemma 4.1. (Guan, 2004) For any matrices $\mathbf{A}_{ij}, \mathbf{B}_{kg}, \mathbf{P} > 0 \in \mathbb{R}^{nxn}$ for $1 \leq i \leq r$, we have

$$\begin{aligned} &2\sum_{i}^{r}\sum_{j}^{r}\sum_{k}^{r}\sum_{g}^{r} \psi^i \psi^j \psi^k \psi^g \mathbf{A}_{ij}^T \mathbf{P} \mathbf{B}_{kg} \leq \\ &\sum_{i}^{r}\sum_{j}^{r} \psi^i \psi^j \left(\mathbf{A}_{ij}^T \mathbf{P} \mathbf{A}_{ij} + \mathbf{B}_{ij}^T \mathbf{P} \mathbf{B}_{ij}\right) \end{aligned} \qquad (21)$$

where ψ^i has properties (20).

Theorem 4.1. The equilibrium state $x_e = 0$ of closed loop system (19), with control input (17) with two-overlapped fuzzy memberships at most, is asymptotically stable in the large, if there exist μ positive definite matrices $\mathbf{P}_s = \mathbf{P}_s{}^{\mathbf{T}} > 0$ such that:

$$\left(\mathbf{\Lambda}_{ii}^s\right)^{\mathbf{T}} \mathbf{P}_s \mathbf{\Lambda}_{ii}^s - \mathbf{P}_s < 0 \quad i \in S_s \quad s = 1,\ldots,\mu \qquad (22)$$

$$\left(\mathbf{\Lambda}_{ij} + \mathbf{\Lambda}_{ji}\right)^{\mathbf{T}} \mathbf{P}_s \left(\mathbf{\Lambda}_{ij} + \mathbf{\Lambda}_{ji}\right) - 2\mathbf{P}_s < 0 \qquad (23)$$
$$i \in S_s \quad j \in S_s \quad i < j \in S_s$$

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

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

$$\mathbf{V}_k = \sum_{s=1}^{\mu} \lambda_s \left(x_k^T \mathbf{P}_s x_k\right) \qquad (24)$$

where

$$\lambda_s\left(\hat{\tau}_k\right) = \begin{cases} 1 & \hat{\tau}_k \in S_s \\ 0 & \hat{\tau}_k \notin S_s \end{cases} \quad \sum_{s=1}^{\mu} \lambda_s\left(\hat{\tau}\right) = 1 \qquad (25)$$

It can be easily showed that $\mathbf{V}(0) = 0$, $\mathbf{V}_k > 0$ for $x_k \neq 0$, and $\mathbf{V}(\mathbf{x}) \to \infty$ as $\|x_k\| \to \infty$, it is only sufficient shows that $\Delta\mathbf{V}\left(x_k\right) < 0$ to prove that $\mathbf{V}_k$ is a Lyapunov function and the theorem. So, we have:

$$\begin{aligned} &\Delta\mathbf{V}_k = \mathbf{V}_{k+1} - \mathbf{V}_k = \sum_{s=1}^{\mu} \lambda_s \left(x_{k+1}^T \mathbf{P}_s x_{k+1}\right) - \sum_{s=1}^{\mu} \lambda_s \left(x_k^T \mathbf{P}_s x_k\right) \\ &\Delta\mathbf{V}_k = \sum_{s=1}^{\mu} \lambda_s \mathbf{L}_s \quad \mathbf{V}_k^s = x_k^T \mathbf{P}_s x_k \qquad \mathbf{L}_s = \mathbf{V}_{k+1}^s - \mathbf{V}_k^s \end{aligned} \qquad (26)$$

It is enough to show that:

$$\mathbf{L}_s < 0 \quad s = 1,\ldots,\mu$$

Substituting $\mathbf{V}_{k+1}^s$ and $\mathbf{V}_k^s$ in (26) we have:

$$\mathbf{L}_s = \left(\sum_{i \in S_s}\sum_{j \in S_s} \psi_i \psi_j \mathbf{\Lambda}_{ij} x\right)^{\mathbf{T}} \mathbf{P}_s \left(\sum_{i \in S_s}\sum_{j \in S_s} \psi_i \psi_j \mathbf{\Lambda}_{ij} x\right) - x^T \mathbf{P}_s x \; (27)$$

Applying the proprieties (20) to (27)

$$= x^T \left(\sum_{i \in S_s}\sum_{j \in S_s}\sum_{k \in S_s}\sum_{g \in S_s} \psi_i \psi_j \psi_k \psi_g \left(\mathbf{\Lambda}_{ij}^{\mathbf{T}} \mathbf{P}_s \mathbf{\Lambda}_{kg} - \mathbf{P}_s\right)\right) x$$

By lemma 4.1 (21), $\mathbf{L}_s$ is:

$$\mathbf{L}_s \leq x^T \left(\sum_{i \in S_s}\sum_{j \in S_s} \psi_i \psi_j \left(\mathbf{\Lambda}_{ij}^{\mathbf{T}} \mathbf{P}_s \mathbf{\Lambda}_{ij} - \mathbf{P}_s\right)\right) x \qquad (28)$$

$$\leq x^T \left(\begin{aligned} &\sum_{i \in S_s} \psi_i^2 \left(\mathbf{\Lambda}_{ii}^{\mathbf{T}} \mathbf{P}_s \mathbf{\Lambda}_{ii} - \mathbf{P}_s\right) + \\ &\sum_{i \in S_s}\sum_{j \in S_s}^{j<i} \psi_i \psi_j \left(\left(\mathbf{\Lambda}_{ij} + \mathbf{\Lambda}_{ji}\right)^T \mathbf{P}_s \left(\mathbf{\Lambda}_{ij} + \mathbf{\Lambda}_{ji}\right) - 2\mathbf{P}_s\right) \end{aligned}\right) x \quad (29)$$

$$\mathbf{L}_s < 0 \to \Delta\mathbf{V}_k < 0$$

The first term in (29) is negative definite by (22). The second term is negative definite by (23). Thus, the positive definite quadratic function (24) is a Lyapunov function for fuzzy control (17), this implicates asymptotically stability in the large. The proof of theorem is complete.

V. SIMULATION

A simulation setup has been designed in order to evaluate the presented algorithm. Simulations have been carried out with the TrueTime simulator (henriksson, 2002).

The controlled plant used in the simulations is a ball and beam in the form of a double integrator. The plant is controlled by a NCS composed by three nodes: sensor, controller and actuator. The ball and beam state space description is

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t) \qquad (30)$$

where the system output is x_1, the ball position. The control objective is to keep the ball position at zero coordinate, regardless of the perturbations affecting the system.

Control performance is measured during each simulation period (5 *ms*) using a continuous standard quadratic cost function.

For simulation purposes, it is assumed that the plant states are available and therefore there is no need for observers. Control is achieved by a fuzzy controller designed to place the continuous closed loop poles at $s_{1,2} = -2 \pm 5i$ considering an specific actuation period of $h = 300ms$.

The three nodes (sensor, controller and actuator), implemented using a TrueTime kernel, are connected to a TrueTime Ethernet network block (configured at 100 Mbps), in order to be able to exchange messages. Performance is measured from the plant output and represented as a cost function. Also it is important to notice, that in order to provide synchronization at actuation instants, the strictly periodic task to be executed is located in the actuator node.

To show the performance of the method, it is applied to the case study. Firstly getting the behavior of time delay with measurements into an Ethernet network with a range of $[10ms, 700ms]$, once established the behavior of time delay we designed a probability density function with its mean and standard deviation values.

Once the range of variation of the time delay is obtained, we define the partition of the antecedent part to establish how many fuzzy rules are generated; it provides the parameters for membership functions.

For three fuzzy rules the continuous linear model is discretized with a sampling period $h_j \quad j = 1 \ldots r$ which are defining as $\mathbf{T} = [0.3s \quad 0.6s \quad 0.9\,s]^T$.

The antecedent parameters and the maximum bounded are defined by the user using information of offline time delay measurement. So, the parameters ρ_j, σ_j for $j = 1 \ldots r$, with the maximum bounded $\tau^{MAX} = 900ms$ are:

$$\rho_j = [0.3 \quad 0.6 \quad 0.9] \qquad \sigma_j = [0.15 \quad 0.15 \quad 0.15]$$

There are two $\mu = 2$ regions S_s with two overlapped membership functions.

With the fuzzy rules set for the fuzzy model a LQR fuzzy controller (17) is designed for each local model. The feedback law for the first local model is:

$$K_1 = [11.86 \quad 4.8]$$

Once the controller is designed several experiments are conducted to test the performance and robustness to time delays and variable sampling intervals. There are three tests, the first test shows the performance NCS with a normal behavior without external traffic on the network. Second test introduces a node to generate traffic on the network and shows the control performance under variable time delay; finally it presents the performance of the system with a traffic node that generates longer variable time delays.

First test (Figure 3) shows the behavior of NCS without external traffic, each test is compared with a standard control (5) with a constant delay of 20 *ms*, in this test, the standard control has better performance than the fuzzy control with periodic actuation, because the fuzzy control at the beginning of the execution wait until the next period ($t = 300ms$) to execute a first action of control.

The traffic generated by the two controllers is similar due to the absence of network traffic; the only difference is the transmission present in the fuzzy controller from the actuator to the controller.

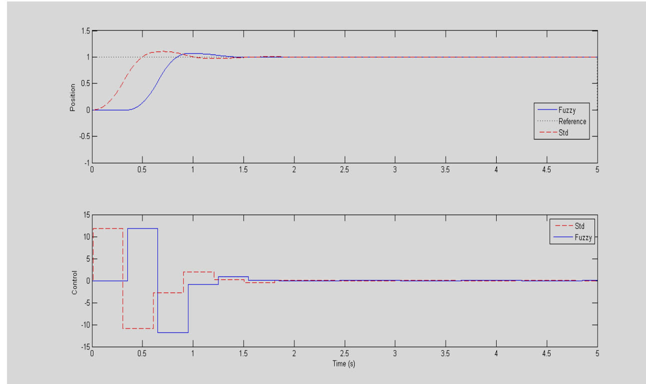

Figure 3. Test 1. Fuzzy and standard control without traffic.

In the second test, the goal is to show the performance of fuzzy control with time delays and variable sampling intervals (Figure 4). It is noted that the fuzzy control is not degraded the system by the presence of network traffic. While the standard control degradation with increased energy presents in the control signal.

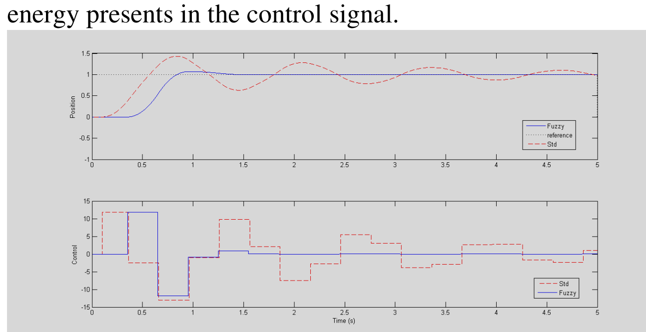

Figure 4. Test 2, Traffic into Ethernet network.

The Figure 5 shows the traffic (50% of bandwidth) supported by the standard controller, although changes in the execution of the work is little noticed these small changes cause a significant impart on the control of the system.

The last test shows the performance of fuzzy controller with delays greater than the sampling period (Figure 7). In the case of standard control with only two delays longer than the sampling period in the transition stage, the system performance is degraded to the point of instability, while the fuzzy control with a greater number of delays longer than the sampling period is kept the stable system and a desired level of performance.

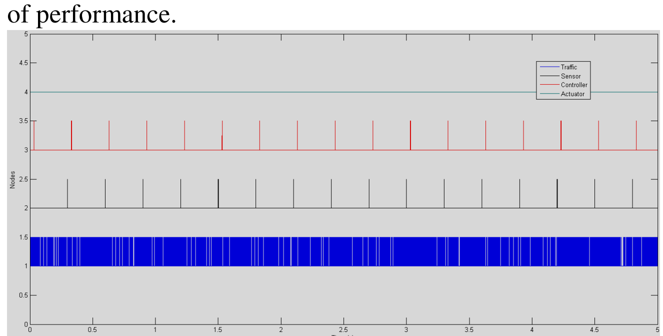

Figure 5. Traffic for standard control with a 50 % bandwidth.

While the fuzzy control input traffic is about 80% of the bandwidth, the performance of the system remains constant with respect to the test without traffic.

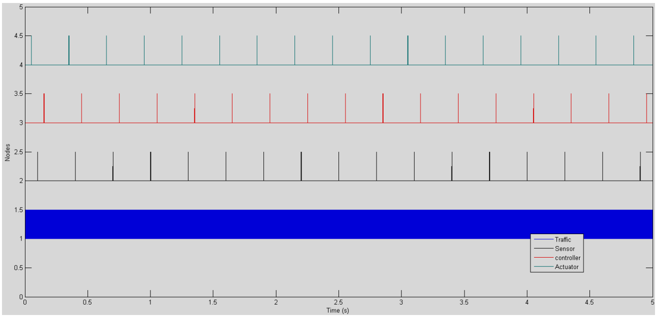

Figure 6. Traffic for fuzzy control with 80% bandwidth

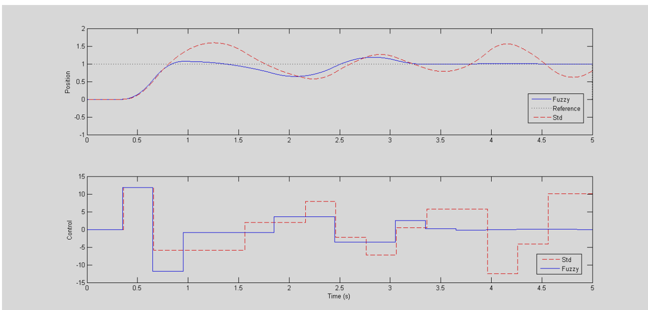

Figure 7. Test 3, Response of NCS with time delays longer.

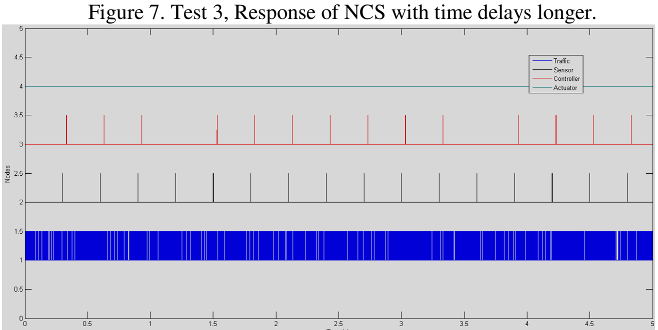

Figure 8. Traffic for standard control.

With a standard controller does not guarantee the system stability to time delays longer a sampling period (Figure 8). While, the fuzzy control shows robustness to time delays longer than sampling period. It generates less energy to apply in the system.

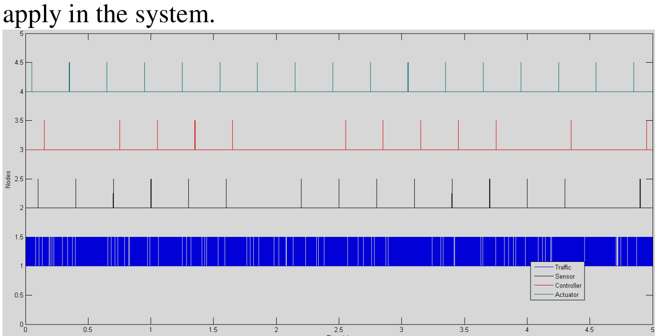

Figure 9. Traffic for fuzzy control

VI. Conclusion

The time delay of a NCS is estimated with a probability density function, the model is computationally simple and light, just is necessary storage *w* data of time delays. It is applied within an Ethernet network with traffic showing with experimental data a good estimation useful for control.

Further, it is presented the design of a fuzzy model from a nonlinear model to compensate the time delays larger than a sampling period synchronized the actuation instants, where this model selects the best discrete model to estimate the next system state as function of the estimated time delay.

Using this fuzzy model, a fuzzy controller is designed to stabilize the NCS, this design is optimal for a static time delay and globally stable with variable time delay longer than a sampling period but bounded. A simulation is used to show the applicability, where the controller was proved for stability and robustness to time delays and variable sampling intervals.

Three tests are presented. Their shows robustness to considerable time delays longer than a sampling period getting a better stability and good performance than standard control that has persistent oscillations and bad performance.

VII. Reference

(Albertos, 1999) P. Albertos and A. Crespo, "Real-time control of non-uniformly sampled systems," *Control Engineering Practice*, Volume 7, Number 4, pp. 445-458, 1999.

(Aström, 1997) K.J. Aström and B. Wittenmark, *Computer-Controlled Systems*, third ed, Prentice-Hall, 1997.

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

(Guan, 2004) Guan, X. and Chen C., 2004: Delay-Dependent Guaranteed Cost Control for T–S Fuzzy Systems with Time Delays, *IEEE Transactions on Fuzzy Systems*, 12-2.

(Henriksson, 2002) D. Henriksson, A. Cervin, and K.-E. °Arz´en, "TrueTime: Simulation of control loops under shared computer resources," *15th IFAC World Congress*, 2002.

(Henzinger, 2001) T.A. Henzinger, B. Horowitz and C.M. Kirsch, "Giotto: a time-triggered language for embedded programming," *First International Workshop on Embedded Software,* LNCS 2211, pp. 166-184, 2001.

(Hespanha, 2007) J. Hespanha, P. Naghshtabrizi, and Y. Xu, "A survey of recent results in networked control systems," Proc. IEEE, pp. 138–162, 2007.

(Hu, 2007) W. Hu, G. Liu, and D. Rees, "Event-driven networked predictive control," *IEEE Transactions on Industrial Electronics,* vol. 54, n.3, 1603- 1613, 2007.

(IEEE 1588m 2002) IEEE 1588TM-2002. Standard for A Precision Clock Synchronization Protocol for Networked Measurement and Control Systems.

(Liu, 2004) X. Liu and A. Goldsmith, "Kalman filtering with partial observation losses," Proc. IEEE Conf. Decision & Control, vol. 4, pp. 4180–4186, 2004.

(Lozoya, 2008) C. Lozoya, P.Mart´ı, M. Velasco, and J.M. Fuertes, "Control Performance Evaluation of Selected Methods of Feedback Scheduling of Real-timeControl Tasks," in 17th IFAC World Congress, July, 2008.

(Martí, 2001) P. Martí, J.M. Fuertes and G. Fohler, "An Integrated Approach to Realtime Distributed Control Systems Over Fieldbuses." *8th IEEE International Conference on Emerging Technologies and Factory Automation,* Antibes Juan-les-pins, France, October 2001.

(Martí, 2007) P. Mart´ı and M. Velasco, "Toward Flexible Scheduling of Real-Time Control Tasks: Reviewing Basic Control Models," *10th International Conference on Hybrid Systems, Computation and Control,* LNCS, 2007.

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

(Tipsuwan, 2003) Y. Tipsuwan and M. Y. Chow, Control methodologies in networked control systems, Control Engineering Practice, Vol. 11, pp. 1099-1111, 2003.

(Tipsuwan, 2004) Tipsuwan Y. and Chow M. Y., 2004: On the Gain Scheduling for Networked PI Controller Over IP Network, *IEEE/ASME Transaction Mechatronics*, 9-3.

(Yang, 2006) T. Yang, "Networked control system: a brief survey," IEE Proc. Control Theory & Applications, vol. 153, pp. 403–412, 2006.

(Zhang, 2001) W. Zhang, M. Branicky, and S. Phillips, "Stability of networked control systems," IEEE Control Syst. Mag., vol. 21, pp. 84–99, 2001.

(Zhang, 2007) Zhang, H., Yang, D. and Chai T., 2007: Guaranteed cost networked control for T–S fuzzy systems with time delays, *IEEE Transactions on Systems Man and Cybernetics part C: Applications and Reviews*, 37-2, pp. 160–172.