Scheduling Strategy Using Frequency Transition for a Helicopter Simulation as a Network Control System Approximation

Oscar ESQUIVEL-FLORES[1]**, Héctor BENÍTEZ-PÉREZ**[2]**, Paul MÉNDEZ-MONRROY**[3]

[1] Posgrado en Ciencia e Ingeniería de la Computación,
U.N.A.M., México,
oaefmcc@hotmail.com.

[2] I.I.M.A.S., D.I.S.C.A., U.N.A.M.
Apdo. Postal 20-706, Del. A. Obregón, C.P. 01000, México,
hector@uxdea4.iimas.unam.mx.

[3] Posgrado en Ingeniería, U.N.A.M.,
México,
paul@uxdea4.iimas.unam.mx

Abstract. This paper describes a scheduling strategy for a real time distributed system based on modifying the transmission frequency of nodes in a communication network. Scheduling is critical as it impacts the system performance due to limited computing resources. This work presents a linear time invariant model based on the frequency transition of nodes in distributed system. In this model a Linear Quadratic Regulator (LQR) controller is designed to schedule the transmission frequency of nodes. The controller assigns the transmission frequency into a region with minimum and maximum bounds intended to satisfy the network's utility. A 2DOF helicopter simulation shows the effectiveness of this scheduling strategy.

Keywords. Distributed Systems, Real-Time, Control, Frequency transmission.

1. Introduction

In recent years Real Time Distributed Systems (RTDS) have become widely used in industry and in research, such as mission-critical and long-running applications; thus they need to maintain consistency or recover from errors without suspending execution. Characteristics of RTDS include the ability to complete within time restrictions, and to provide coherence, adaptability, and stability. Recent applications of RTDS with time restrictions are implementations of Networked Control Systems (NCS), which consist of several nodes which participate in the control process and sensor/actuator activities. In order to achieve the overall objectives of all tasks in a global and distributed manner, it is necessary that each node exchanges their own information properly through communication media in a real time environment [7]. In these applications, the time requirements of a NCS must be scheduled. In general there are two types of tasks in a NCS.

The first is a periodic task that is time-triggered, in which tasks τ_i have a transmission time c_i a constant period of execution p_i, and a deadline d_i. The second type of task is aperiodic. Thus, the sum of the transmission times of n nodes' tasks, divided into their periods p_i for a fixed priority scheduler [11], is feasible if:

$$U = \sum_{i=1}^{n} \frac{c_i}{p_i} \le n\left(2^{\frac{1}{n}} - 1\right) \tag{1}$$

Due to optimum fixed priority scheduler possesses an upper bound to processor utilization [6] is necessary to consider (1) for each node involved in the distributed system. The network scheduler is critical in a NCS, since if there is no scheduling between nodes, data transmissions may occur simultaneously leading to collisions or bandwidth violations. This behaviour results in transmission with time delays, leading to failure in complying with deadlines, data loss, and subsequent decrease in system performance. A good scheduling control algorithm minimizes the decrease in system performance [1]; nevertheless, there are no global schedulers that guarantee optimal system performance [8]. Some strategies include methods for nodes to generate proper control actions in order to optimally utilize bandwidth [4,5]. In the digital control case, the performance only depends on the sampling frequency without uncertainties. In the digital control case, the performance only depends on the sampling frequency without uncertainties. For networked control frequency transmission (FT) is a significant factor. The minimum FT f_b is necessary to guarantee good system performance without decreasing the network performance. As the FT increases the system performance improves; however, the load on the network also increases until a maximum FT f_c is reached, then the system performance decreases because the network performance is overloaded. It is very important to modify the FT to obtain better system performance within a bounded region that is particularly defined for the current system needs. This paper presents a scheduling strategy for modifying the FT of nodes in a distributed system by controlling the transmission frequency relations. We propose a linear model in which the coefficients of the state matrix are the relations between the transmission frequencies of each node. The model uses an LQR feedback controller to modify the FT in a region located between the maximum and minimum bounds to ensure system schedulability. This network system is modelled as linear since the scheduling transitions are within a context of normal and periodic responses, although this is not always a normal condition in these types of systems. This idea is reinforced through a 2DOF helicopter simulation benchmark. This case provides a good approximation of a system response in which the main results are perform under a typical fault scenario for demonstration proposes. The FTs are discrete; thus, they change the observed phenomenon over a specific time. The transition period corresponds to a systematic observation of the phenomenon, and the minimum period is that of the possible tasks related to this modification. The phenomenon represents non-linear situations with respect to sudden changes in states, failure situations, situations, or saturation in the channel or traffic, among others. However; it is possible to propose a linear model in the context of proper use of the network, thereby deferring the modelling of nonlinearity in these systems until future work. The aim of this work is to tune the frequency for task communications based on real-time constraints and scheduling. A goal is to maintain the schedulability and thus the viability of the communication. The synchronization issue is outside the scope of this paper. The rest of this paper is organized as follows. Section 2 describes a FT model and proposes the matrix coefficients of the model. Section 3 presents a particular NCS as a study case, and Section 4 describes the numerical simulations of the presented model and the performance of the LQR controller. Brief conclusions are presented at the end.

2. Frequency Transition Model

A set real-time tasks $\Phi = \{\tau_1, \tau_2, ..., \tau_n\}$, where $\tau_i = \{c_i, p_i\}$, $i = \{1, 2, ..., n\}$. Here Φ is the set of tasks, τ_i is the *i-th* task, n is the number of tasks which are periodic, independent and preemptive, c_i denotes the execution time of τ_i, p_i denotes the period of task τ_i. This approach drives the (FT) based on three parameters: minimum frequency f_m, maximum frequency f_h and real frequency f_r. FT dynamics can be modelled as a linear time-invariant subsystem which state variables are transmission frequencies of the sensor nodes involved on the system. Note that for each task of a sensor S_i $i = \{1, 2, ..., n\}$ frequency can be expressed as $f_i = 1/p_i$. We assume there is a relationship between frequencies which define the internal behaviour of the network, and the desired FT and external input frequencies, which serve as coefficients of the linear system:

$$\begin{aligned} x\left(k+1\right) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) \end{aligned} \tag{2}$$

$A \in \Re^{nxn}$ is the relationships matrix between transmission frequencies of the sensor nodes, $B \in \Re^{nxn}$ is the scale frequencies matrix, $C \in \Re^{nxn}$ is the matrix with frequencies ordered, $x(k+1)$ is a real frequencies vector, and $y(k)$ is the vector of output frequencies. The input $u(k)$ is input vector of FT of the nodes into RTDS system. It is important to note that the relations between the frequencies of the n nodes leading to the system (2) are schedulable with respect to the use of the network transmission frequencies, equation (1) The relationship matrix is dependent on each case study, but is mainly related to the fractions between each node transmission. Therefore, it is possible to control the FT through the input vector u such that the outputs y are in a region L that is non-linear where the system is schedulable. During the time evolution of the system (2), the output frequencies could be stabilized by a controller within the schedulability region L [3]. Figure 1 shows the dynamics of the frequency system and the desired effect when it is controlled. It also defines a common region L for a set of frequencies. Each task of the node *i* of the system starts with a frequency f_i, and the controller modifies the period p_i in order to converge in a region where the system performance is close to optimal.

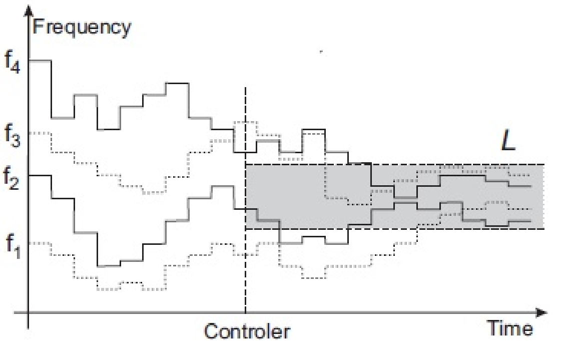

Figure 1. Frequencies of transmission bounded by a schedulability region

The objective in modifying the frequency is to achieve coordination between the nodes to obtain the desired transmission frequencies.

Relationship Matrix

Let $a_{ij} \in A$ be given by a function of minimal frequencies f_m of node *i*, and $b_{ij} \in B$ be given by a function of maximal frequencies f_m:

$$a_{ij} = \chi\left(f_m^1, f_m^2, ..., f_m^n\right), \; b_{ij} = \omega\left(f_x^1, f_x^2, ..., f_x^n\right)$$

In this particular case, the matrix is built in terms of the local relations between the greatest common divisor as shown in next section; however, this is not the only procedure to be followed. The LQR controller is used to obtain the feedback matrix *K*. The input is given by a function of the minimal frequencies and the real frequencies of the *i* nodes in the distributed system, hence $u = K\left(\overline{f}_m - \overline{f}_r\right)$ where

$$\overline{f}_m = \left[f_m^1, f_m^2, ..., f_m^n\right]^{\mathrm{T}}, \; \overline{f}_r = \left[f_r^1, f_r^2, ..., f_r^n\right]^{\mathrm{T}}.$$

Then the system (2) can be written as:

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ x(k+1) &= A\overline{f}_r + B\left(K\left(\overline{f}_m - \overline{f}_r\right)\right) \end{aligned} \qquad (3)$$

where *K* is the control gain defined as the basics of a controller algorithm.

3. Case of Study

The RTDS used for implementation purposes of this paper is a 2-DOF helicopter prototype [10]. Following section briefly introduces and describes this 2-DOF helicopter prototype and its controller design, as well as an experimental approach to express this prototype as a NCS.

Helicopter Dynamic Model and Its Control Design.

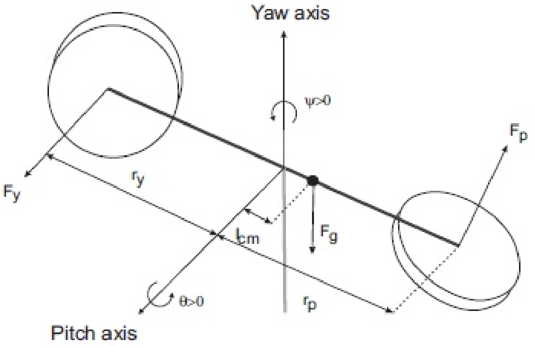

Figure 2. Dynamics of the 2-DOF helicopter

The case of study is a prototype of a helicopter system integrated to a CanBus network with two propellers that are driven by DC motors. The front propeller controls the elevation of the helicopter nose about the pitch axis (θ) and the back propeller controls the side to side motions of the helicopter about the yaw axis (ψ). The pitch and yaw angles are measured using high-resolution encoders. A brief description of the helicopter model is presented, however detailed information can be found in [10]. The dynamics of the helicopter is developed based on kinetic and potential energy, this model is used to design a position controller. The helicopter centre of mass is described in *xyz* cartesian coordinates with respect the pitch and yaw angles, see Figure 2. The Euler-Lagrange equations are used to obtain nonlinear equations of motion for the 2 DOF Helicopter, which are used to derive the linear state model, and subsequently, to design the position controller. As the helicopter represents a non-linear system, it is required to perform a linearization around the point $\left[\theta_0 = 0, \psi_0 = 0, \theta_0^{'} = 0, \psi_0^{'} = 0\right]$. From this, the linearization of the motion equation is obtained as follows:

$$\begin{aligned} \left(J_{eq,p} + m_{heli} l_{cm}^2\right)\theta^{''} = \; & K_{pp}V_{m,p} + K_{py}V_{m,y} - B_p\theta^{'} \\ & - m_{heli} g l_{cm} \end{aligned} \qquad (4)$$

$$\begin{aligned} \left(J_{eq,y} + m_{heli} l_{cm}^2\right)\psi^{''} = \; & K_{pp}V_{m,y} + K_{py}V_{m,p} - B_p\psi^{'} \\ & - 2m_{heli} l_{cm}^2 \theta\psi\dot{\theta} \end{aligned} \qquad (5)$$

Substituting is $x=\left[\theta,\psi,\theta'\psi'\right]'$ in (4) and (5) and solving for the x' the following linear model of state space is obtained:

$$x' = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\dfrac{B_p}{J_{eq,p}+m_{heli}l_{cm}^2} & 0 \\ 0 & 0 & 0 & -\dfrac{B_y}{J_{eq,y}+m_{heli}l_{cm}^2} \end{bmatrix} x$$

$$+\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \dfrac{K_{pp}}{J_{eq,p}+m_{heli}l_{cm}^2} & \dfrac{K_{py}}{J_{eq,p}+m_{heli}l_{cm}^2} \\ \dfrac{K_{yp}}{J_{eq,y}+m_{heli}l_{cm}^2} & \dfrac{K_{yy}}{J_{eq,y}+m_{heli}l_{cm}^2} \end{bmatrix}$$

where $K_{pp}, K_{yy}, K_{py}, K_{yp}$ are the torque-constants used to obtain coupled torques acting on the pitch and yaw axes; for the state space model the input u and output y vectors are $u=\left[V_{m,p},V_{m,y}\right]'$ and $y=\left[x_1,x_2,x_3,x_4\right]'$, V_{mp} is the input pitch motor voltage and V_{my} is the input yaw motor voltage. Notice that, since the output matrix is the identity matrix, all states are measurable.

The model makes use of several Simulink and Matlab programs to develop the helicopter basic dynamics, by running a simulation of the closed-loop response, using the position controller. Regarding control issues, a FF+LQR+I controller is designed. This controler regulates the pitch axis of the helicopter, using feed-forward (FF) and proportional-velocity (PV) compensators, while the yaw axis only makes use of a PV control. The FF+LQR+I controller uses an integrator in the feedback loop to reduce the steady-state error, by a feed-forward and proportional-integral-velocity (PIV) algorithms to regulate the pitch, and only a PIV to control the yaw angle. The FF+LQR+I control converges $\left(\theta,\psi,\theta'\psi'\right)\rightarrow\left(\theta_d,\psi_d,\theta_d',\psi_d'\right)$ where θ_d is the desired pitch angles and ψ_d is the desired yaw angle, such that:

$$\begin{bmatrix} u_p \\ u_y \end{bmatrix} = \begin{bmatrix} K_{ff}\dfrac{m_{heli}gl_{cm}\cos\theta_d}{K_{pp}} \\ 0 \end{bmatrix}$$

The addition of an integrator requires to introduce the states $x_5'=\theta$ and $x_6'=\psi$, so the linear state-space model is augmented as:

$$x' = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -\dfrac{B_p}{J_{eq,p}+m_{heli}l_{cm}^2} & 0 & 0 & 0 \\ 0 & 0 & 0 & -\dfrac{B_y}{J_{eq,y}+m_{heli}l_{cm}^2} & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} x$$

$$+\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \dfrac{K_{pp}}{J_{eq,p}+m_{heli}l_{cm}^2} & \dfrac{K_{py}}{J_{eq,p}+m_{heli}l_{cm}^2} \\ -\dfrac{K_{yp}}{J_{eq,y}+m_{heli}l_{cm}^2} & -\dfrac{K_{yy}}{J_{eq,y}+m_{heli}l_{cm}^2} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} x$$

Using the adequate Q and R weighting matrices, the control gain is as follows:

$$k = \begin{bmatrix} 18.9 & 1.98 & 7.48 & 1.53 & 7.03 & 0.77 \\ -2.22 & 19.4 & -0.45 & 11.9 & -0.77 & 7.03 \end{bmatrix}$$

Thus, the FF+LQR+I controller is:

$$\begin{bmatrix} u_p \\ u_y \end{bmatrix} = \begin{bmatrix} K_{ff}\dfrac{m_{heli}gl_{cm}\cos\theta_d}{K_{pp}} \\ 0 \end{bmatrix} - \begin{bmatrix} k_{11} & k_{12} & k_{13} & k_{11} \\ k_{21} & k_{22} & k_{23} & k_{24} \end{bmatrix}$$

- $$\begin{bmatrix} \theta-\theta_d \\ \psi-\psi_d \\ \theta' \\ \psi' \end{bmatrix} - \begin{bmatrix} k_{15}\left(\theta-\theta_d\right)+\int k_{16}\left(\psi-\psi_d\right) \\ k_{25}\left(\theta-\theta_d\right)+\int k_{26}\left(\psi-\psi_d\right) \end{bmatrix}$$

Experimental Approach

In order to study the impact of network utilization on closed control loop, the 2-DOF

Helicopter control model is built as a RTDS. Several nodes are connected through a common communication network. The experiment focus on network scheduling, and the main objective is to balance the amount of data sent through the network, in order to avoid latency and under sampling. We consider the special case of a RTDS in which the structure is based on sensor, controller, actuator, and master scheduler nodes. Figure 3 shows the Networked Control System consisting of eight nodes with a real-time kernel, connected through a network type CSM/ AMP (CAN). The rate of sending data is 10 Mb/s which is not likely to data loss. The real time simulation tool was Truetime [9] which is based on Matlab/Simulink.

Four sensor nodes execute periodic tasks to sense control signals, as well as other additional periodic tasks. Each task has a period p_i and time consumption c_i as shown in Figure 3.

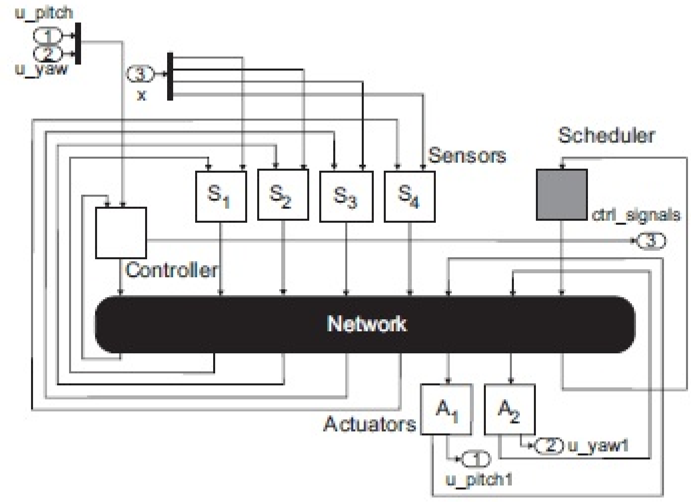

Figure 3. Networked Control System to analyze frequency transmission through the network.

The sensed control signals are $x = (\theta, \psi, \theta' \psi')$. This model has a controller node, depicted on the left side as shown in Figure 3. This controller takes the control law from the FF+LQR module by means of a task, which activates by event. The time consumption of the controller task is the maximum average time it takes to compute the control law. The controller node uses the values from sensors, and sends control outputs u_p and u_y, that correspond to the pitch and yaw voltages. Two actuator nodes, located on the bottom right corner shown in Figure 3, receive signals from the controller node. Finally, the scheduler node, located on the top right corner shown in Figure 3, organizes the activity of the other seven nodes, and it is responsible for periodic allocation bandwidth, used by these nodes. Each node initializes, specifying the number of inputs and outputs of the respective True Time kernel block, defining a scheduling policy, and creating periodic tasks for the simulation. These tasks involve parameters about the periodic times and the consumption times. The task periodic times define the time interval between tasks, whereas the consumption times referto the execution time of the task. Previous system was included into feedback control loop of 2-DOF helicopter model.

Changes on the real-time task parameters of the RTDS commonly impact on network utilization, and therefore, on the control performance [4, 5]. The problem to tackle, thus, is to find a proper way to schedule the common communication network of the RTDS, based on managing an accurate sampling period, capable of keeping both, the network load and required integrated performance.

4. Network Scheduling and Related Work

This section mentions a previous work related to the scheduling transmission rate of data in a real-time distributed systems based on the modification of the sampling periods and develops the frequency transition model described in section 2. Menéndez and Benítez [8] design a global scheduling strategy based on the analysis of NCS, they show that the performance of the system depends not only on the sampling periods of its individual components, but also on the time dispersions amongst these periods. The scheduling strategy consist of a global operating period for the whole system called base period ρ, in addition there is a common period-range for every node participating in the NCS, this period-range is obtained by moving away from the base period within certain percentage called dispersion factor λ. Base period and dispersion factor are used to obtain the actual operational period for

each node *i* of the system: $n_i = \rho(1 \pm \lambda)$.

Some tests are presented in [8] to quantify the NCS quality performance under a particular scheduling strategy and evaluate the performance of a NCS. When all nodes are competing to obtain the network and get bandwidth according to the network access control algorithm, the system trend to be unstable. As part of the scheduling strategy, the scheduler node allocates a bandwidth share to every node by means of assigning a time-window when to transmit, independently form the network protocol used, which must be considered solely as the network access controller. To have a performance criterion, and thus, quantify the system's quality performance, the integral of the absolute value of the error IAE is used:

$$IAE = \int_{t_0}^{t_f} |e(t)| dt \approx \sum_{k=k_0}^{k_f} |r(kh) - y(kh)|$$

where $r(kh)$ is reference signal, $y(kh)$ system output signal, $t_0(k_0)$ and $t_f(k_f)$ are the minimum and final and initial simulation times.

The 2-DOF Helicopter system presents a stable behaviour using a base period into the interval [0.005, 0.017] seconds and dispersion fixed in 5% for all nodes; however when the task period is out of this interval and/or scheduler does not assign a proper bandwidth the system become unstable easily, see Figure 4.

It is important to emphasize that successful network management is a key point to achieve system schedulability and thus obtain a good performance. A disadvantage of previous proposal consist on it is a static scheduling model, this means that accurate Real-Time parameters are computed out line and there is not mechanism to modify these parameters on line, which is very useful under fault scenarios. In this work, we focus on the sensor nodes, with the objective of controlling the data FT through system (1).

Each node has a transmission frequency f_r, and sets the minimum frequencies f_m and maximum f_x between each node. The region *L* should meet the restriction in terms of the FT representation and the utilization of system performance. The elements of the matrices for system (2) are defined as follows:

$$\bar{a}_{ij} = \begin{cases} \dfrac{\Lambda(f_m^1, f_m^2, f_m^3, f_m^4)}{f_m^i} & i=j \\[2ex] \dfrac{f_m^j}{f_m^i} & i \neq j \end{cases}$$

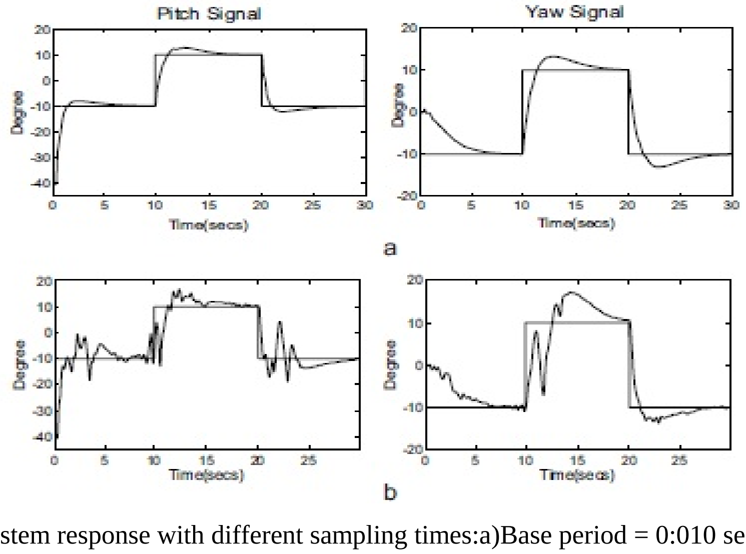

Figure 4. System response with different sampling times:a)Base period = 0:010 secs., b)Base period=0:22secs.

$$\bar{b}_{ij} = \begin{cases} f_x^i & i = j \\ 0 & i \neq j \end{cases}$$

$$\bar{c}_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$

$\Lambda\left(f_m^1, f_m^2, f_m^3, f_m^4\right)$ is the planning cycle in terms of frequencies of the tasks; here, we refer it as only Λ.

It is very important to consider the computational time for the tasks of each node *i* as an additional state in order to represent (2). Using (3) we can rewrite the states of the system (2) as:

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \\ x_4(k+1) \\ x_5(k+1) \end{bmatrix} = \begin{bmatrix} \frac{\Lambda}{f_m^1} & \frac{f_m^2}{f_m^1} & \frac{f_m^3}{f_m^1} & \frac{f_m^4}{f_m^1} & 0 \\ \frac{f_m^1}{f_m^2} & \frac{\Lambda}{f_m^2} & \frac{f_m^3}{f_m^2} & \frac{f_m^4}{f_m^2} & 0 \\ \frac{f_m^1}{f_m^3} & \frac{f_m^2}{f_m^3} & \frac{\overline{\Lambda}}{f_m^3} & \frac{f_m^4}{f_m^3} & 0 \\ \frac{f_m^1}{f_m^4} & \frac{f_m^2}{f_m^4} & \frac{f_m^3}{f_m^4} & \frac{\Lambda}{f_m^4} & 0 \\ c_1 & c_2 & c_3 & c_4 & 1 \end{bmatrix} \begin{bmatrix} f_r^1 \\ f_r^2 \\ f_r^3 \\ f_r^4 \\ x_c \end{bmatrix} +$$

$$+ \begin{bmatrix} f_x^1 & 0 & 0 & 0 & 0 \\ 0 & f_x^2 & 0 & 0 & 0 \\ 0 & 0 & f_x^3 & 0 & 0 \\ 0 & 0 & 0 & f_x^4 & 0 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} \bullet$$

$$\begin{bmatrix} k_1^s & 0 & 0 & 0 & 0 \\ 0 & k_{21}^s & 0 & 0 & 0 \\ 0 & 0 & k_3^s & 0 & 0 \\ 0 & 0 & 0 & k_4^s & 0 \\ 0 & 0 & 0 & 0 & k_c^s \end{bmatrix} \left(\begin{bmatrix} f_m^1 \\ f_m^2 \\ f_m^3 \\ f_m^4 \\ x_c^r \end{bmatrix} - \begin{bmatrix} f_r^1 \\ f_r^2 \\ f_r^3 \\ f_r^4 \\ x_c \end{bmatrix} \right)$$

where *c₁, c₂, c₃, c₄* are the consumed times from each process considering the periodic tasks within their respective periods, x_c are the real consumptions, and x_c^r are the reference consumptions and k_i^s is the control gains. Thus,

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \\ x_4(k+1) \\ x_5(k+1) \end{bmatrix} = \begin{bmatrix} \frac{\Lambda}{f_m^1} & \frac{f_m^2}{f_m^1} & \frac{f_m^3}{f_m^1} & \frac{f_m^4}{f_m^1} & 0 \\ \frac{f_m^1}{f_m^2} & \frac{\Lambda}{f_m^2} & \frac{f_m^3}{f_m^2} & \frac{f_m^4}{f_m^2} & 0 \\ \frac{f_m^1}{f_m^3} & \frac{f_m^2}{f_m^3} & \frac{\Lambda}{f_m^3} & \frac{f_m^4}{f_m^3} & 0 \\ \frac{f_m^1}{f_m^4} & \frac{f_m^2}{f_m^4} & \frac{f_m^3}{f_m^4} & \frac{\Lambda}{f_m^4} & 0 \\ c_1 & c_2 & c_3 & c_4 & 1 \end{bmatrix} \begin{bmatrix} f_r^1 \\ f_r^2 \\ f_r^3 \\ f_r^4 \\ x_c \end{bmatrix} +$$

$$\begin{bmatrix} k_1^s f_x^1\left(f_m^1 - f_r^1\right) \\ k_2^s f_x^2\left(f_m^2 - f_r^2\right) \\ k_3^s f_x^3\left(f_m^3 - f_r^3\right) \\ k_4^s f_x^4\left(f_m^4 - f_r^4\right) \\ k_1^s\left(f_m^1 - f_r^1\right) + k_2^s\left(f_m^2 - f_r^2\right) + k_3^s\left(f_m^3 - f_r^3\right) + \\ k_4^s\left(f_m^4 - f_r^4\right) + k_c^s\left(x_c^r - x_c\right) \end{bmatrix}$$

and the output vector is:

$$\begin{bmatrix} y_1(k+1) \\ y_2(k+1) \\ y_3(k+1) \\ y_4(k+1) \\ y_c(k+1) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f_r^1 \\ f_r^2 \\ f_r^3 \\ f_r^4 \\ x_c \end{bmatrix}$$

Since the data, FT of the network nodes using an NCS can increase without limit due to uncertainties during sensing, the system may become unstable; similarly, a low data FT results in under sampling which leads to control performance weakness. Therefore, it is important to manage the sensor nodes dynamically to ensure that they are restricted to the schedulability region where the system is stable. It is possible to realize a network planning outline; however, uncertainties inherent to the network such as traffic or delays require dynamic management of the transmission frequencies. Although in practice this control accepts frequencies outside of the stability region, it is possible to balance the use of the network and maintain the system at acceptable performance levels.

5. Numerical Simulations

Numerical simulations of the system 2 were performed both without a scheduler and with the proposed scheduler for values of the maximum, minimum, and real frequencies. Accounting for the computational time, the values used for the sensor nodes are shown in the Table 1.

Table 1. Frequencies and consumed time of the sensors in NCS

node	Max. Freq.	Min. Freq.	Real Freq.	Consume
1	60	310	40	0.001
2	50	270	250	0.001
3	50	270	100	0.001
4	45	300	50	0.001

The respective eigenvalues of matrix A system using the values in table 1 are:

$$\lambda_1 = 1.0000, \quad \lambda_2 = 3.3308, \quad \lambda_3 = -0.8556,$$
$$\lambda_4 = -0.6835, \quad \lambda_5 = -0.5000$$

Thus, the dynamic system is unstable.

LQR control

The LQR method has been chosen because it is necessary to use a bounded time response to guarantee stability, considering the nonlinearity conditions as those mentioned above. Furthermore, this strategy is not focused on convergence of the states; the focus is on planning the elements from the perspective of dynamic systems in real time [2]. Now the system is invariant over time, given a bound due to the minimum possible period for changing scenarios. In this sense, this is a restriction seen as a future case study, and in its current form the modification of the FT is only a parametric change which is adapted from a gain regulator. The latter is bounded given the context of the problem observed. The weight matrices *Q* and *R* were chosen as diagonal matrix, where the elements main diagonal of Q are 10 and elements in main diagonal of R are 100. The gain matrix K is and new states matrix are:

$$K = \begin{bmatrix} 0.317 & 0.004 & 0.004 & 0.004 & -0.059 \\ 0.004 & 0.317 & 0.004 & 0.004 & -0.035 \\ 0.004 & 0.004 & 0.317 & 0.004 & -0.035 \\ 0.004 & 0.004 & 0.004 & 0.317 & 0.032 \\ -0.007 & -0.008 & -0.008 & -0.007 & 2.046 \end{bmatrix}$$

The eigenvalues of new matrix A_c=*(A-BK)* are:

$$\lambda_1 = -1.049, \quad \lambda_2 = -98.007, \quad \lambda_3 = -94.915$$
$$\lambda_4 = -85.423, \quad \lambda_5 = -85.382$$

Figure 5 shows the dynamics of the FT model controlled by an LQR controller. In this case no fault is presented and a transition should be performed in terms of the initial time and steady state time, meaning that the system response is not feasible during this interval. The LQR controller modifies the FT rate into the schedulability region in which the system performance is suitable. Large sampling periods result in insufficient data to compute the control law in the controller node, this degrade quality of control. Nevertheless, small sampling periods increases network bandwidth demand. A typical fault scenario where some transmission frequency values of the sensor nodes are outside of schedulability region becomes the system to experiment under sampling or overload, thus the IAE considerably increases. Figure 6 shows different values of the IAE of pitch angle signal and yaw angle signal using sampling periods between 1 millisecond and 20 milliseconds, it means 1000 Hz and 50 Hz. Thus sampling periods which produces small values of IAE bound the internval of maximum and minimum FT.

Case study response

The typical response of the pitch value is presented for a common distribution of the computer network, using the True Time toolbox [2, 9]. Further review of this case study is presented in [8]. The case study is performed using the True Time tool-box where the actual computer network performance helps to approximate its response. Figure 7 shows the computer network as well.

As shown in Figure 8 in the case of a small failure in one of the elements, the computer network system tries to communicate whenever it loses tracking due to a fault presence. Although the frequency transition is not aggressive, this example presents how FT control strategy is suitable for fault accommodation in terms of fault presence and the related modification in the communication. Frequency transmission manage through second 12 to second 30 and second 50 to second 60 restore data transmission rate over the network avoiding under sampling this result in stability respect to IAE.

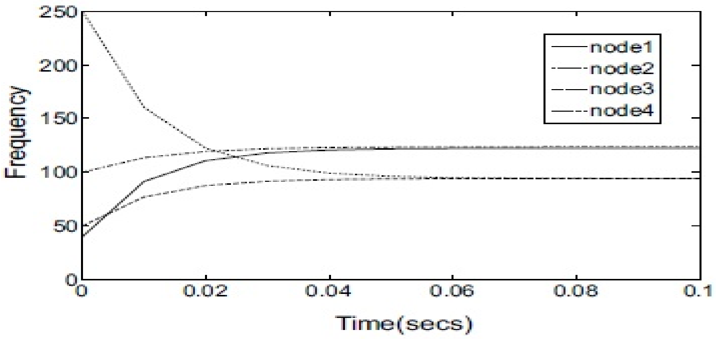

Figure 5. Frequency response controlled by a LQR controller.

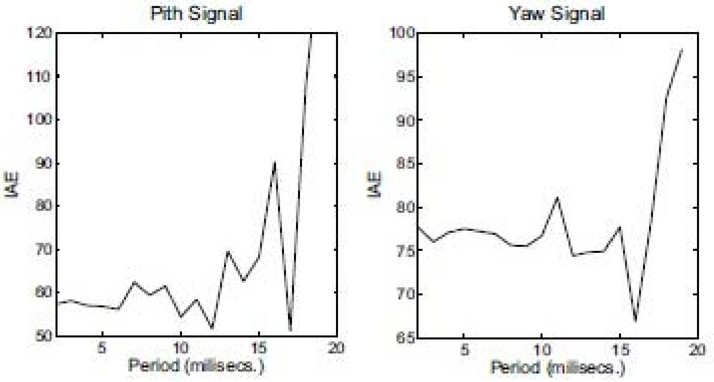

Figure 6. Values of IAE of pitch and yaw signals for different sampling periods.

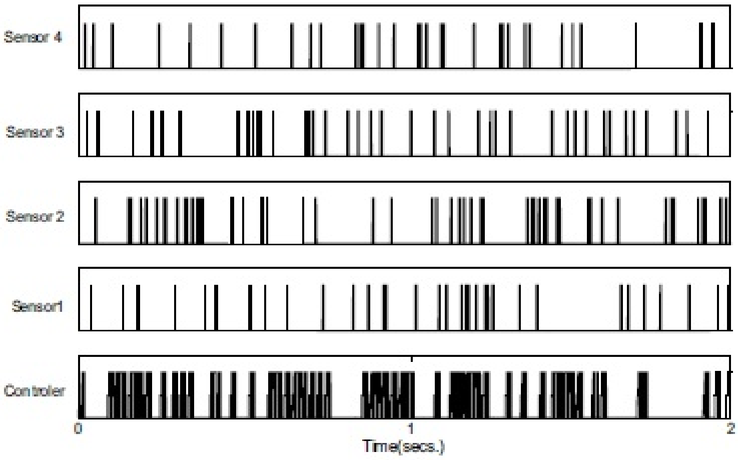

Figure 7. Sensing activity and the related use of the computer network using the TrueTime toolbox.

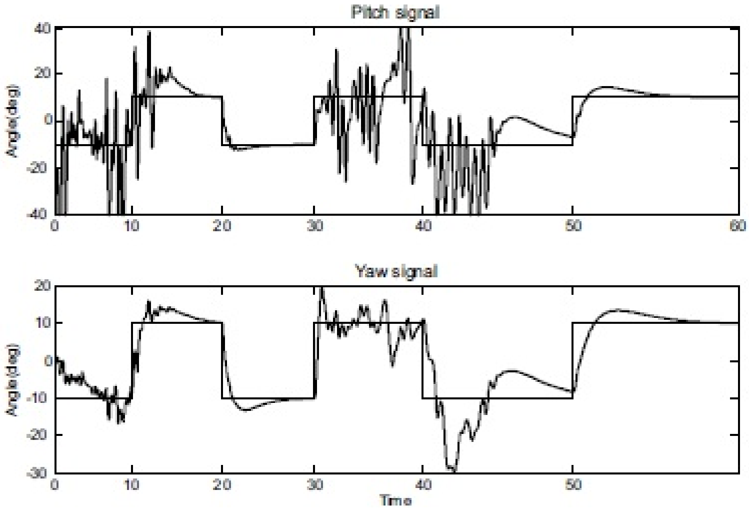

Figure 8. Sensing pitch and yaw angle considering fault scenario and modifying frequency transmission.

6. Conclusions

The present work deals with the available transitions for achieving schedulability. Saturated conditions are not considered. We have presented a linear time invariant model of nodes and their FT as involved in a distributed system. The significance of controlling the frequencies stems from the system schedulability. The key feature of the LQR control approach is a simple design with good robustness and performance capabilities that allow the frequencies to be easily modified. We have shown via numerical simulations the performance of the proposed control scheme. In both cases, the case study and the network controller, it is possible to show the actual feasibility of this co-design strategy since a complex problem of non-coupled dynamics is presented. Further study is needed in order to propose a holistic integration of both dynamics that can be developed through high order observers. This work would be of special interest.

Acknowledgements

Grants PAPIIT 103310, PICCO-ICYTDF 10-53, and CONACYT (PhD Grant) are specially appreciated and Mr. Adrian Duran for his valuable comments.

REFERENCES

1. BRANICKY, M., V. LIBERATORE, S. PHILLIPS, **Networked Control System Co-simulation for Co-design**, Proceedings of the American Control Conference, vol. 4, June 2003, pp. 3341-3346.
2. CERVIN, A., D. HENRIKSSON, B. LINCOLN, J. EKER, K. ARZEN, **How Does Control Timing Affect Performance?**, Control Systems Magazine, vol. 23, no. 3, June 2003, pp. 16-30.
3. ESQUIVEL-FLORES, O., H. BENÍTEZ-PÉREZ, P. MÉNDEZ-MONROY, A. MENÉNDEZ, **Frequency Transition for Scheduling Management using Dynamic System Approximation for a Kind of NCS**, ICIC Express Letters, part B: Applications, vol. 1, no. 1, September 2010, pp. 93-98.
4. LIAN, F., J. MOYNE, D. TILBURY, **Time Delay Modelling and Sample Time Selection for Networked Control Systems**, Proceedings of ASME-DSC, vol. XX, 2001.
5. LIAN, F., J. MOYNE, D. TILBURY, **Network Design Considerations for Distributed Networked for Distributed Control Systems**, IEEE Transactions on Control Systems Technology, vol. 10, no. 2, March 2002, pp. 297- 307.
6. LIU, C., J. LAYLAND, **Scheduling Algorithms for Multiprogramming in a Hard-Real-Time Environment**, Journal of the Association for Computing Machinery, vol. 20, no. 1, January 1973, pp. 46-61.
7. LIAN, F., J. YOOK, D. TILBURY, J. MOYNE, **Network Architecture and Communication Modules for Guaranteeing Acceptable Control and Communication Performance for Networked Multi-agent Systems**, IEEE Transactions on Industrial Informatics, vol. 2, no. 1, Feb 2006, pp. 12-24.
8. MENÉNDEZ, A., H. BENÍTEZ-PÉREZ, **Scheduling Strategy for Real-Time Distributed Systems**, Journal of Applied Research and Technology, vol. 8, no. 2, August 2010, pp. 177-185.
9. OHLIN, M., D. HENRIKSSON, A. CERVIN, **TrueTime 1.5 Reference Manual**, Department of Automatic Control, Lund University, 2007.
10. QUANSER, **Quanser 2 DOF Helicopter. User and Control Manual**, Quanser Innovate-Educate, 2006.
11. SHA, L., T. ABDELZAHER, K. ARZEN, A. CERVIN, T. BAKER, A. BURNS, G. BUTAZZO, M. CACCAMO, J. LEHOCZKY, A. MOK, **Real Time Scheduling Theory: A Historical Perspective**, Real time systems, Kluwer Academic Publishers, vol. 23, no. 2-3, Nov 2004, pp. 101-155.