

## Impact of a (m,k)-firm Data Dropouts Policy on the Quality of Control

Flavia Felicioni, Ning Jia, Ye-Qiong Song, Françoise Simonot-Lion

► **To cite this version:**

Flavia Felicioni, Ning Jia, Ye-Qiong Song, Françoise Simonot-Lion. Impact of a (m,k)-firm Data Dropouts Policy on the Quality of Control. 6th IEEE International Workshop on Factory Communication Systems, Jun 2006, Torino, Italy, IEEE, pp.353-359, 2006, Factory Communication Systems, 2006 IEEE International Workshop on. <inria-00113827>

**HAL Id: inria-00113827**

**<https://hal.inria.fr/inria-00113827>**

Submitted on 14 Nov 2006

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Impact of a $(m,k)$ -firm Data Dropouts Policy on the Quality of Control

Felicioni Flavia  
FCEIA – UNR  
Universidad Nationale de Rosario  
Rosario - Argentina  
flaviaf@fceia.unr.edu.ar

Jia Ning, Song YeQiong, Simonot-Lion Françoise  
LORIA – INPL  
Campus Scientifique – BP 239  
54506 – Vandoeuvre lès Nancy France  
{Ning.Jia,song,simonot}@loria.fr

## Abstract

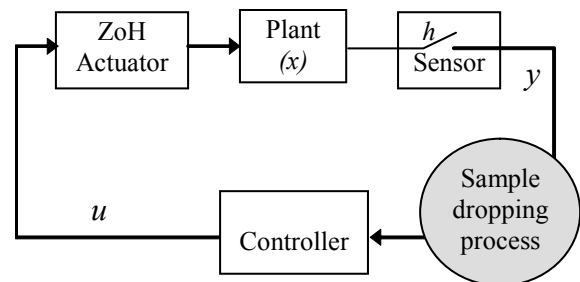
*In this paper, we propose a method for the design of an operational architecture of a Networked Control System (NCS). We consider a control system whose main goal is to control the position of a cart moving along a rail. The implementation of the controller is done through a distributed architecture in which a shared network supports the transmission of the samples between the sensor and the controller. For efficiently handling network congestion, we propose to apply a selective sample drop policy according to a  $(m,k)$ -pattern in order to decrease the network bandwidth required by this application during network overload periods. The paper shows how to determine the values of the parameter  $k$  that preserve the stability of the system and then how to identify the value of  $m$  and the  $(m,k)$ -pattern in order to optimize the system performance.*

## 1. Introduction

In this paper, we consider a multi-variable plant and its computer-based control system. The input delivered to the plant is always the last reference computed by the controller (thanks to a zero-order-hold actuator). The output of the controlled system (the plant) is sampled (period  $h$ ) and each sample is transmitted to the controller through a network. This will introduce a delay between the sampling instant and the control law completion. If this delay is constant, the value of this delay can be easily taken into account for the design of the control law. Nevertheless, the constant-delay assumption cannot be guaranteed if the network is a resource shared by several applications especially due to network overload period.

For handling the network overload while observing the quality of control (QoC), two strategies can be applied:

- The first one is based on the augmentation of the sampling period in order to reduce the required bandwidth. However, changing the sampling period of a controlled system alters its dynamics and needs a complex analysis of the system and its controller.



**Figure 1. Closed loop control system with sample dropping process**

- The second approach is to selectively drop some instances of system outputs in order to resolve the network congestion.

This last solution is discussed in this paper and we propose, for a given kind of controlled system, a technique for the definition of the sample drop policy.

More precisely, the proposed solution is based on the  $(m,k)$ -firm model [1], [2]. In this model, a recurrent activity is said to be under the  $(m,k)$ -firm constraint if at least  $m$  among  $k$  consecutive instances have to be processed before their deadline;  $m$  and  $k$  are integers and  $0 < m \leq k$ . Figure 1 shows that the effect of this policy is that some measures on the system are not transmitted to the controller. Consequently, this effect can potentially modify the Quality of the Control.

Notice that most control systems can tolerate some misses of samples while preserving an acceptable level of QoC. So, this comment justifies identifying regular pattern of possible sample drops that can be cyclically repeated. Behind this idea, we focus our study on a  $(m,k)$ -firm approach; the problem is therefore to specify

the controller and the  $(m,k)$ -firm parameters that ensure a required level of QoC.

In order to determine the  $(m,k)$ -firm parameters, we develop a systematic study of the impact of a  $(m,k)$ -pattern (the distribution of the  $m$  over  $k$  consecutive samples) on the QoC. Precisely, we have to evaluate the values of  $m$  and  $k$  and the distribution of the possibly rejected samples that guarantee the required QoC. There has been relatively little further work investigating the impact of sample loss on QoC. The main approaches, proposed, for example, in [3, 4, 5], modelled the sample loss as a random process; nevertheless, random process may be suitable to describe the drops due to transient transmission errors but it does not fit to the selective drop process. For this approach, only some statistical results have been provided.

An important contribution has been provided by Ramanathan [2]. In this work, the author proposed a scheduling technique based on the  $(m,k)$ -firm policy that discards selectively the task instances to handle the overload in processor, and he proposed a methodology for modifying the control law to reduce the degradation of QoC due to sample loss. However, the issue that how to formally choose an acceptable  $(m,k)$ -firm constraint that guarantees control system stability wasn't addressed in this work; furthermore, the proposed algorithm discards the task instances without taking into account explicitly the QoC, therefore, the resulting distribution of sample drops may not be optimal from point of view of QoC.

In [6], we have presented a formal analysis that derived, for a one-dimensional control system, the values of  $m$  and  $k$  and the distribution of the sample drops that guarantee the stability and minimize the degradation of QoC. In this paper, we extend this study to multiple dimensional control systems. We first identify the stability region of control system and determine the values of  $k$  that correspond to this stability region. The parameter  $m$  and the form of the  $(m,k)$ -pattern are found thanks to a systematic Matlab/Simulink simulation sequence. In this paper, we detail the proposed technique on a case study (the control of the position of a cart guided on a rail).

The paper is organized as follows. In section 2, we formalize the system under study and give the preliminary notions (the cart and its controller). Section 3 presents how to identify the values of parameter  $k$  under which the system remains stable. Section 4 concerns the definition of  $m$  and of the  $(m,k)$ -pattern. Finally, in section 5, we conclude on the proposed approach and present the perspective.

## 2. System Description

### 2.1. Model of the controlled system

The system under study is a cart whose movement is guided along a rail (see Figure 2). The objective is to control its position that is the reference. In this paper, the controller parameters, and the stability and performance analysis of the system were obtained through an analysis of a simplified model of the cart assuming no friction on the rail.

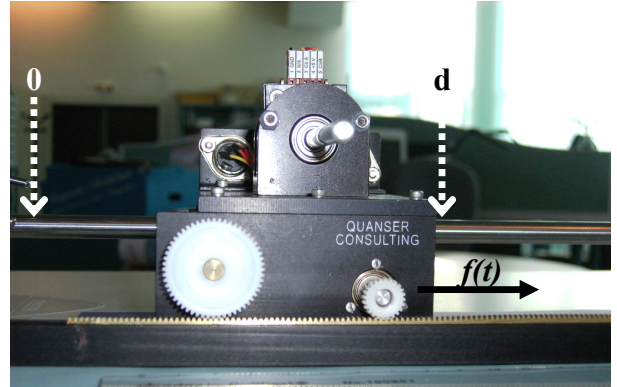


Figure 2. Cart as the controlled system

The state variables of this system are  $d$ , the position of the cart along the rail measured from a reference point and  $\dot{d}$ , its speed ( $x^T = [d, \dot{d}]$ ). The simplified model of the system can be described by the following differential equation where  $u$  is the input and  $k_1, k_2$  two parameters:

$$\ddot{d} = -k_1\dot{d} + k_2u \quad (1)$$

We identified the system parameters,  $k_1, k_2$ , thanks to some open-loop experiments and obtained  $k_1 = 12.6559$  and  $k_2 = 1.9243$ .

The continuous model of the system based on the state space representation is given by:

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx + Du \end{cases} \quad (2)$$

where  $A = \begin{bmatrix} 0 & 1 \\ 0 & -k_1 \end{bmatrix}$ ,  $B = \begin{bmatrix} 0 \\ k_2 \end{bmatrix}$ ,

For this paper, we considered that the output of the system corresponds to its state variables. So:

$$C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \text{ and } D = 0.$$

Finally, the discrete-time open loop model is given by the equation (3).  $h$  is the sampling period.

$$\begin{cases} x_{t+1} = \Phi(h)x_t + \Gamma(h)u_t \\ y_t = x_t \end{cases} \quad (3)$$

with:

$$\Phi(h) = \begin{bmatrix} 1 & \frac{1-e^{-k_1 h}}{k_1} \\ 0 & e^{-k_1 h} \end{bmatrix}, \Gamma(h) = \frac{k_2}{k_1} \begin{bmatrix} h - \frac{1-e^{-k_1 h}}{k_1} \\ 1 - e^{-k_1 h} \end{bmatrix}$$

## 2.2. Specification of the controller

In order to control the cart position, we used a *State-Feedback Controller* defined by

$$u_t = L(x_{ref} - x_t) \quad (4)$$

where  $x_{ref} = [d_{ref} \ 0]^T$  (the reference is only the position,  $d_{ref}$ , of the cart) and  $L = [k_c \ k_d]$ . In this example, the parameters,  $k_c$  and  $k_d$ , were evaluated by applying the LQR method and, for a sampling period  $h = 0.01s$ , we obtained  $k_{c,0.01} = 121$  and  $k_{d,0.01} = 6.5$ .

## 3. Study of the system stability

In this section, we propose an approach for the evaluation of the values of  $k$  ensuring the stability of the controlled system under  $(m,k)$ -firm policy. We recall that, for this example, this strategy implements the reject of system outputs,  $y_t$ , according to a  $(m,k)$ -pattern.

Intuitively, we can say that, if the system is stable for a  $(1,k)$ -pattern, it will remain stable, for each  $(m,k)$ -pattern where  $1 \leq m \leq k$ . Taking into account a dropping process according to a  $(1,k)$ -pattern means to increase the sampling period ( $2h, 3h$ , etc.). So, we propose in a first step to study the stability of the system for a sampling period  $h_c > h$ .

Let us introduce the matrix  $\Psi(h_c) = \Phi(h_c) - \Gamma(h_c)L$  that is:

$$\Psi(h_c) = \begin{bmatrix} 1 - k_c \frac{k_2}{k_1} \left( h_c - \frac{X}{k_1} \right) & \frac{1 - e^{-k_1 h_c}}{k_1} - k_d \frac{k_2}{k_1} \left( h_c - \frac{X}{k_1} \right) \\ k_c \frac{k_2}{k_1} (X) & e^{-k_1 h_c} - k_d \frac{k_2}{k_1} (X) \end{bmatrix}$$

with  $X = 1 - e^{-k_1 h}$ .

According to the Jury's criterion, a sufficient and necessary condition for the stability of the system is to satisfy the following three conditions:

- 1)  $a_2 < 1$
- 2)  $a_2 > a_1 - 1$
- 3)  $a_2 > -a_1 - 1$

$$\text{where: } \begin{cases} a_1 = \Psi_{1,1} \Psi_{2,2} - \Psi_{1,2} \Psi_{2,1} \\ a_2 = -\Psi_{1,1} - \Psi_{2,2} \end{cases}$$

$\Psi_{i,j}$  is the element of the  $i^{\text{th}}$  line and  $j^{\text{th}}$  column of matrix  $\Psi$ .

$a_1$  and  $a_2$  are expressed with respect to the controller parameters ( $k_c, k_d$ ) and the sampling period ( $h_c$ ).

In order to determine the longest sampling period that preserves the system stability, we fixed the parameter  $k_d$  to the value evaluated for the initial sampling period  $h = 0.01$  ( $k_d = k_{d,0.01}$ ). Thanks to *Mathematica*, we obtain the region  $R_{stability}$  in the plan  $(h_c, k_c)$  where the three Jury's conditions are satisfied. This region is hachured in figure 2. It is bounded by the functions:

- $h_c = h$ ,
- $k_{cMin} = 0$ ,
- and  $k_{cMax}(hc)$  defined as:

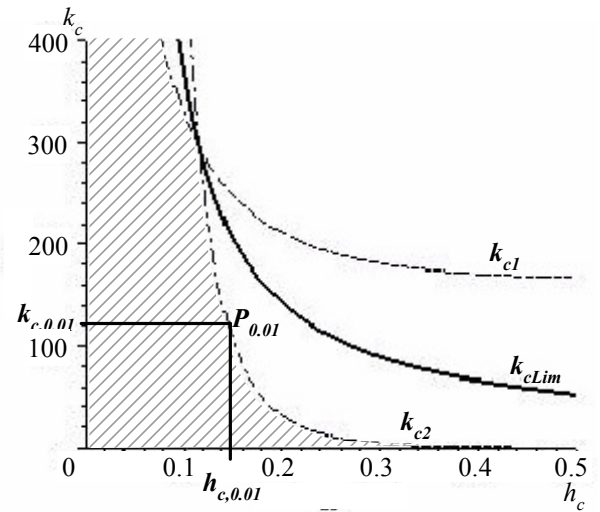
$$k_{cMax}(hc) = \begin{cases} k_{c1}(hc) & \text{if } k_{c1}(hc) < k_{cLim}(hc) \\ k_{c2}(hc) & \text{if } k_{c1}(hc) > k_{cLim}(hc) \end{cases}$$

where:

$$k_{cLim}(hc) = \frac{4k_1 e^{k_1 h_c}}{h_c k_2 (e^{k_1 h_c} - 1)}$$

$$k_{c1}(hc) = \frac{2k_1(k_1 + k_1 e^{k_1 h_c} + k_2 k_{d,0.01} - k_2 k_{d,0.01} e^{k_1 h_c})}{k_2(2 - 2e^{k_1 h_c} + h_c k_1 + h_c k_1 e^{k_1 h_c})}$$

$$k_{c2}(hc) = \frac{k_1(k_1 + k_2 k_{d,0.01})(e^{k_1 h_c} - 1)}{k_2(-1 - h_c k_1 + e^{k_1 h_c})}$$



**Figure 2. Stability domain in the space  $(k_c, h_c)$  evaluated for  $k_{d,0.01}=6.5$**

Let us, now, consider the value of the controller parameter  $k_{c,0.01}$  as determined in section 2.2, the point  $(h_{c,0.01}, k_{c,0.01})$  defined as the intersection of  $k_{cMax}(hc)$

$$\text{and } k_c = k_{c,0.01}, \text{ and } h_{MAX} = \left\lfloor \frac{h_{c,0.01}}{h} \right\rfloor h$$

- if  $(h_{MAX}, k_{c,0.01})$  does not belong to the stability region  $R_{stability}$ , we have to reconsider the controller in order to identify another admissible 3-uple  $(h, k_c, k_d)$  and to apply the same approach,

- if  $(h_{MAX}, k_{c,0.01})$  belongs to the stability region  $R_{stability}$ , the maximum length of the  $(m, k)$ -pattern is  $k_{worst} = \left\lfloor \frac{h_{MAX}}{h} \right\rfloor$ ; that is the case in the presented case study where  $k_{worst} = 14$ .

The different results determined in section 2.1, 2.2 and 3 are summarized in Table 1.

**Table 1. Parameters of the system, the controller and the  $(m, k)$ -pattern policy**

Cart parameters $\ddot{d} = -k_1 \dot{d} + k_2 u$	$k_1 = 12.6559$ $k_2 = 1.9243$ .
Controller parameters $u_i = [k_c \quad k_d](x_{ref} - x_i)$	$h = 0.01$ $k_{c,0.01} = 121$ $k_{d,0.01} = 6.5$
Possible length of the $(m, k)$ -pattern	$1 < k \leq 14$

#### 4. $(m, k)$ -pattern identification

In section 3, we have identified the values of  $k$  that preserves the stability of the system under a  $(l, k)$ -firm sample dropping policy. We assume that in this case, the pattern is  $\{\tau_1, \tau_2, \dots, \tau_k\}$  with  $\tau_1 = 1$  and  $\tau_i = 0$  for  $i \neq 1$ .

For the identification of  $m$  and of the distribution of  $m$  over  $k$ , we assume that  $\tau_1 = 1$ . We study the response of the system:

- to a given reference applied to the cart: the cart has to shift two centimeters from an initial position and with an initial zero speed,
- and under a given  $(m, k)$ -pattern where  $k$  ensures the stability of the system.

Intuitively, greater is  $m$ , greater is the quality of control. Therefore, this second part of the method can be viewed as an optimization problem. It consists in maximizing the performance while ensuring the stability of the system.

In this paper, we consider that the quadratic LQR cost is a performance indicator and we choose it as the criterion to minimize. So, we simulate with

Matlab/Simulink, the models of the system, of its controller and of the  $(m, k)$ -firm sample dropping policy and measure the LQR cost:

- for each possible couple  $(m, k)$  (in this case study,  $k = 2, \dots, 14$  and  $m = 1..k$ ),
- and for each  $(m, k)$ -pattern  $\{\tau_1, \tau_2, \dots, \tau_k\}$  where  $\tau_1 = 1$ .

The LQR cost  $J$  is given by:

$$J = \frac{1}{Nh} \sum_{i=1}^N ((d - d_{ref})^2 + Ru_i^2) \quad (5)$$

where  $Nh$  is the time horizon (i.e.  $N$  times of sampling period),  $R$  is the weight for taking into account the voltage limitation.

In the next sections, we only present some results obtained for  $k=10$ .

##### 4.1. Ideal configuration

In the next sections, we will compare each obtained LQR cost with the one obtained for an ideal configuration whose parameters are:

- $h = 0.01$
- $(k, k)$ -pattern

For this ideal configuration, taking  $R=0.00006$ ,  $N=30000$ ,  $k_{c,0.01} = 121$  and  $k_{d,0.01} = 6.5$ , according to equation 5, the LQR cost  $J_{ref} = 4.624 \cdot 10^{-3}$ .

The performance degradation,  $Q$ , will be evaluated as:

$$Q = \frac{J_{(m,k),\{\cdot\}} - J_{ref}}{J_{ref}}$$

where  $J_{(m,k),\{\cdot\}}$  is the quadratic LQR cost obtained for the given  $(m, k)$ -pattern.

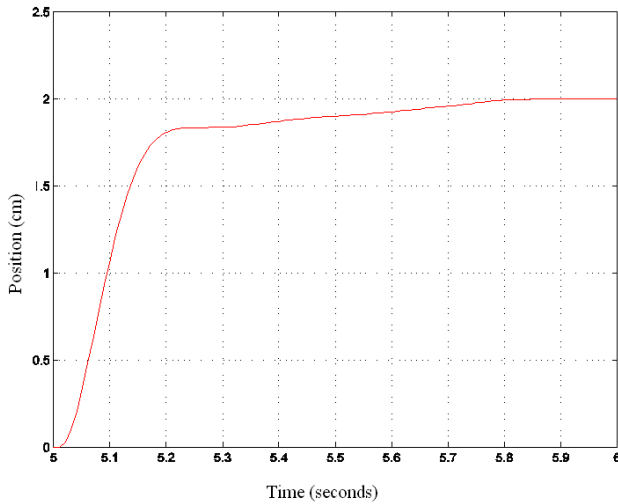
Another measurement is, *Ratio*, the ratio between the network bandwidth required by a  $(m, k)$  dropping policy and the one required by the ideal configuration:

$$Ratio = \frac{m}{k}$$

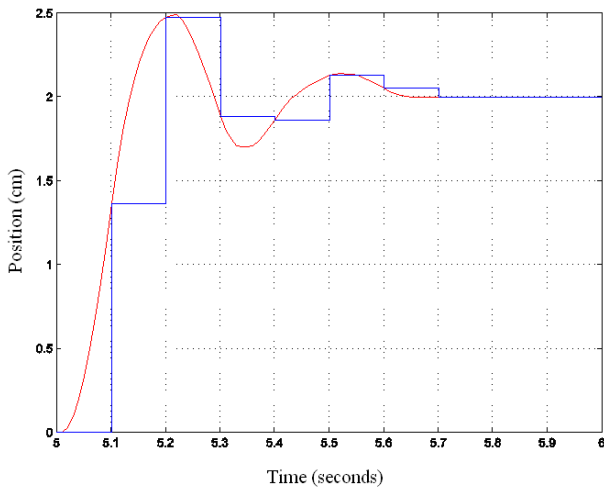
In the following sections we show that both the  $(m, k)$ -firm constraint and the distribution of packet drops have an important impact on performance of control system.

##### 4.2. Worst-case configuration for a given value of $k$

The first experiments aim to compare the performance obtained for the best distribution of  $(l, 10)$ -pattern ( $\{1000000000\}$ ) and for the ideal case, i.e.  $(k, k)$ -pattern. For example, under a  $(l, 10)$ -firm constraint, the *Ratio* is 10% while the performance of control system suffers a degradation  $Q = 84\%$ . That means that the price to decrease significantly the required bandwidth is an important control effort.



**Figure 4. Response time to the reference under  $(k,k)$ -firm constraint (ideal situation)**



**Figure 5. Response time to the reference under  $(1,10)$ -firm constraint ( $\{1000000000\}$ )**

As we can see in figures 4 and 5, which plot the response time of cart system with respectively  $(k,k)$ -firm constraint (ideal case) and  $(1,10)$ -firm constraint, the cart system suffers a larger overshoot and settling time under  $(1,10)$ -firm constraint compared with the ideal case.

#### 4.3. Performance degradation for $(2,10)$ -pattern

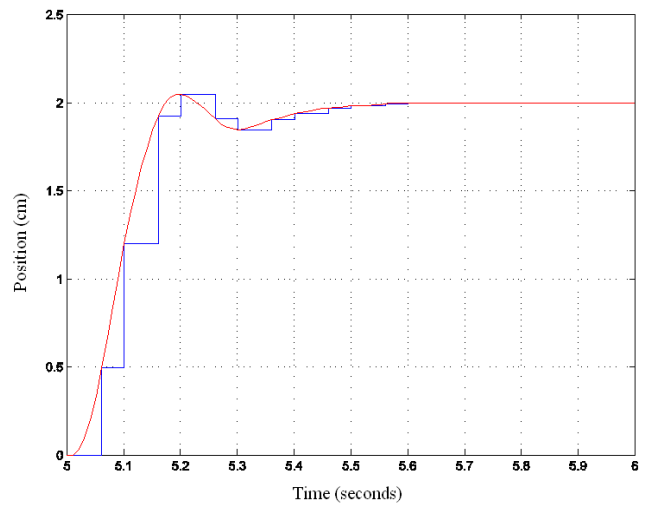
We adopt a  $(2,10)$ -firm constraint. Results are presented in Table 2.

We can see that, under the  $(2,10)$ -pattern  $\{1000001000\}$ , the performance degradation,  $Q$ , of the control system is reduced to 23.74% while the implementation requires 20% of network bandwidth of the ideal system. The response time of the cart system in this case is given in figure 6. It shows clearly a less

overshoot and a smaller settling time than for  $(1,10)$ -pattern ( $\{1000000000\}$ ) illustrated by figure 5.

**Table 2. Impact of sample drop distribution on the performance of the control system for  $k = 10$  and  $m = 2$**

$(m,k)$ -pattern	LQR cost	Performance degradation $Q(\%)$
1100000000	$9.652 \cdot 10^{-3}$	62.95
1010000000	$7.680 \cdot 10^{-3}$	53.00
1001000000	$6.760 \cdot 10^{-3}$	46.20
1000100000	$6.257 \cdot 10^{-3}$	35.40
1000010000	$5.818 \cdot 10^{-3}$	25.80
<b>1000001000</b>	<b><math>5.721 \cdot 10^{-3}</math></b>	<b>23.74</b>
1000000100	$5.944 \cdot 10^{-3}$	28.54
1000000010	$6.400 \cdot 10^{-3}$	38.40
1000000001	$7.150 \cdot 10^{-3}$	54.00



**Figure 6. Response time under  $(2,10)$ -firm constraint and  $(m,k)$ -pattern ( $\{1000001000\}$ )**

#### 4.4. Performance degradation for $(3,10)$ -pattern

Table 3 presents the results corresponding to a sequence of  $(3,10)$ -pattern experiments.

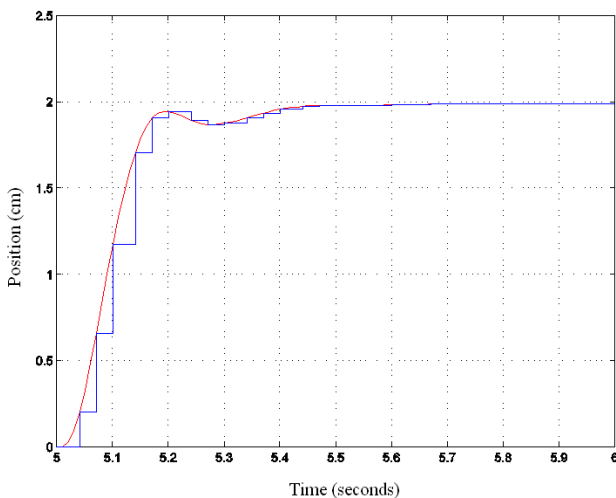
The performance degradation is less than 10% when the packet delivery sequence is given by  $\{1000100100\}$ , and the control system requires only 33.3% of the network bandwidth compared with the original system without any sample drop. The response time of the cart system in this case is given in figure 7 showing, for this

pattern, a better performance than the one obtained in the two previous cases.

Note that, in our experiments, the distribution of sample drops according to the approach proposed by Ramanathan [2] (uniform distribution: {1001001000}) does not provide the best performance of control.

**Table 3. Impact of sample drop distribution on the performance of the control system for  $k = 10$  and  $m = 3$**

$(m,k)$ -pattern	LQR cost	Comments
1100000100	$5.605 \cdot 10^{-3}$	
1100000010	$5.879 \cdot 10^{-3}$	
1100000001	$6.540 \cdot 10^{-3}$	Performance degradation worst than the one obtained under $(2,10)$ -constraint {1000001000}
1010001000	$5.299 \cdot 10^{-3}$	
1010000100	$5.354 \cdot 10^{-3}$	
1010000010	$5.677 \cdot 10^{-3}$	
1010000001	$6.286 \cdot 10^{-3}$	Performance degradation worst than the one obtained under $(2,10)$ -constraint {1000001000}
1001001000	$5.312 \cdot 10^{-3}$	Pattern of Ramanathan
1001000100	$5.482 \cdot 10^{-3}$	
1001000010	$5.453 \cdot 10^{-3}$	
1000101000	$5.099 \cdot 10^{-3}$	
1000100100	$5.080 \cdot 10^{-3}$	Q=9.90%



**Figure 7. Response time under  $(3,10)$ -firm constraint and  $(m,k)$ -pattern {1000100100}**

Furthermore, some distributions of sample drops for  $(3,10)$ -firm constraint give a performance degradation that is worse than the one obtained under  $(2,10)$ -pattern {1000001000}. We therefore propose that at the design stage of the control application, the  $(m,k)$ -firm constraints and  $(m,k)$ -patterns for each control system should be judiciously chosen by taking into account both the requirement in terms of performance of control system and the requirement in terms of bounded network bandwidth rather than using a fixed model.

## 5. Conclusions

In Networked Control Systems (NCS), it is more and more common to have a single network shared by several applications. This is particularly true with the spread use of industrial Ethernet in factory automation. How to efficiently manage the network quality of service (QoS) during its congestion periods is a key issue. In our study, as a radical way to deal with the network congestion, we proposed to selectively drop packets (sampled data). Of course, this dropping policy may degrade the control performance to some extent but should never jeopardize its stability. In order to control the performance degradation, a  $(m,k)$ -pattern is used.

In this paper, by taking a multiple variable closed loop control system (control of cart movement) as a case study, we investigated the effect of sample drops according to a  $(m,k)$ -pattern on the QoS. The relationship between the stability region of the control system and the value of  $k$  is established. The performance (LQR cost) degradation is also identified in function of the value of  $m$  and the distribution of the  $m$  over  $k$ . It has been shown that the judicious distribution of sample drops plays a critical role in reducing the control system performance degradation. Based on the above results, we can conclude that selectively dropping samples under judiciously chosen  $(m,k)$ -pattern consists in an interesting way to handle the network QoS.

For a given value of  $k$ , the fact that the control performance degradation depends not only on the value of  $m$  but also on their distribution tells us the existence of a tight relationship between the system current state variables value (or their derives) and the sample importance. So as a future work, we plan to formalize this relationship and use it to dynamically assign the priorities for network packet scheduling. Another future work, also towards to the dynamic QoS control, aims at the establishment of a general theorem linking the control stability to any  $(m,k)$  couple, so that dynamic sample drop can be performed based on the lastly realized  $k$ -sequence [7].

## References

- [1] Quan, G., and X. Hu, "Enhanced Fixed-priority Scheduling with  $(m,k)$ -firm Guarantee", *Proc. of 21st IEEE Real-Time Systems Symposium*, pp.79-88, Orlando, Florida, (USA), November 27-30, 2000.
- [2] Ramanathan P., "Overload management in Real-Time control applications using  $(m,k)$ -firm guarantee", *IEEE Transactions on Parallel and Distributed Systems*, 10(6): 549-559, Jun 1999.
- [3] Ling Q and M.D. Lemmon, "Robust performance of soft real-time networked control systems with data dropouts", *Proc. Of the IEEE Conference on Decision and Control*, 2002.
- [4] Ling, Q., and M.D. Lemmon, "Soft real-time scheduling of networked systems with dropouts governed by a Markov chain", *Proc. of the American Conference on Control*, 2003.
- [5] Zhang, W., M. S. Brannicky, and S. M. Philips, "Stability analysis of networked control systems", *IEEE Control System Magazine*, 21(1): 84-89, Feb 2001.
- [6] Jia, N., Song, Y.Q. and Lin, R.Z. , "Analysis of networked control system with packet drops governed by  $(m,k)$ -firm constraint". *Proc of the 6th IFAC international conference on fieldbus systems and their applications (FeT'2005)*, Puebla (Mexico), November 2005.
- [7] Hamdaoui M. and P. Ramanathan, "A dynamic priority assignment technique for streams with  $(m, k)$ -firm deadlines", *IEEE Transactions on Computers*, 44(4), 1443-1451, Dec.1995.