HAL
open science

Adaptation of scheduling policy parameters for controlling networked systems

Idriss Diouri, Jean-Philippe Georges, Eric Rondeau

To cite this version:

Idriss Diouri, Jean-Philippe Georges, Eric Rondeau. Adaptation of scheduling policy parameters for controlling networked systems. 3rd IFAC International Workshop on Networked Control Systems : Tolerant to Faults, Jun 2007, Nancy, France. hal-00169165

HAL Id: hal-00169165
https://hal.science/hal-00169165v1

Submitted on 31 Aug 2007

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.

ADAPTATION OF SCHEDULING POLICY PARAMETERS FOR CONTROLLING NETWORKED SYSTEMS

I. Diouri, JP. Georges, E. Rondeau

Centre de Recherche en Automatique de Nancy
Nancy-Université, CNRS
BP 239, F-54600 Vandoeuvre lès Nancy, France
<firstname.name>@cran.uhp-nancy.fr

Abstract: The performances of networked control systems depend on the Quality of Service (QoS) offered by the network. The objective of this paper is to analyse from experimentations, different approaches for dynamically adapting the bandwidth allocation of network according to the Quality of Performances (QoP) required by the controller. The switched Ethernet architectures implementing WRR (Weighted Round Robin) policy are especially studied. *Copyright © 2007 IFAC*

Keywords: Networked Control Systems, Control of Network, Scheduling policy, Quality of Service.

1. INTRODUCTION

The point to point architectures initially used for controlling remote and distributed industrial applications were progressively replaced by networked systems for mitigating the wiring costs (Gallara et *al.*, 1984), and for increasing the flexibility and modularity of control systems. Basically, fieldbuses based on deterministic protocols enable to ensure a level of quality of service required for running the networked applications. Currently, the industrial requirements are evolving. The closed industrial network architectures interesting to easily guarantee the communication performances, but exchanging only process information are not adapted to implement new functionalities such as E-Maintenance, Tele-operation, etc. Moreover and in order to reduce network equipment costs, the industrial networks have to support both traffic dedicated for controlling the industrial process, and messages generated by any other applications of the enterprise.

Thus, the fieldbuses have to support transport protocols for ensuring new industrial services over Internet and they have to propose sufficiently bandwidth for transferring real-time traffic and other time unconstrained traffic.

But, the bandwidth does not solve all the problems for controlling industrial applications, because it does not guarantee a predefined performance for critical messages. It means, it is necessary to implement mechanisms inside the network devices enabling to split the bandwidth according to the time constraints required by each application. The idea is to allocate sufficient bandwidth for strongly time constrained messages. Thus, the network devices have to implement, for example, scheduling policies, such as Earliest Deadline First (EDF) (Liu and Layland, 1973) (Hoang *et al.*, 2002) and Weighted Round Robin (WRR) (Demers *et al.*, 1989) (Katevenis *et al.*, 1991), for managing the communication buffers in considering the priorities associated to messages.

In this paper, Ethernet network is studied, because it is more and more implemented in real-time systems (Airbus A380, etc.). The choice was based also on the fact that it is an open network (by using a TCP/IP stack) offering an important bandwidth and because it is commonly used in the Networked Controlled Systems (NCSs). The CSMA/CD protocol used by the Ethernet (IEEE standard 802.3) leads to a non determinism access due to collisions. To avoid this problem, switched Ethernet networks combined with full-duplex mode as defined in the IEEE standard 802.1D (IEEE, 1998) is used.

The evolution of Ethernet to segmented architectures and the definition of the Virtual Local Area Networks (VLAN) have led to the birth of new standards set (802.1D/p, 802.1Q) in which new encapsulation fields are added to the classical frame (IEEE Computer Society, 2003). One of these fields is specified in order to support 8 priority levels associated to 8 types of applications (voice, video, network management, best effort, etc.). The number

of Classes of Service may be different to the number of priority levels, and also different for each port. That is why the standard also recommends a mapping between classes, priority and ports queues.
The Ethernet switches implementing classification of service (CoS) can be modelled with the network calculus theory for estimating bounded end-to-end delays (Georges *et al.*, 2005) which are essential information in industrial environments. But currently the scheduling policies implemented on the switches are often limited to strict priorities or WRR techniques. WRR is selected in this study because it avoids famine problems specific to the strict priority.
The objective of this paper is to analyse the behaviour of Networked Control Systems (NCS) using Ethernet network in taking into account dynamically the traffic variations. In NCSs the regulation loops are closed by the communication network (Zhang et al., 2001), and it is necessary to ensure the stability of the process control, taking into account the performances of the network (Branicky et al., 2003) (figure 1). The network generates delays (Lian et al., 2003), when the controller sends a new reference to the process and it induces also delays when the controller monitors the system state. The delay can be infinite when messages are lost.

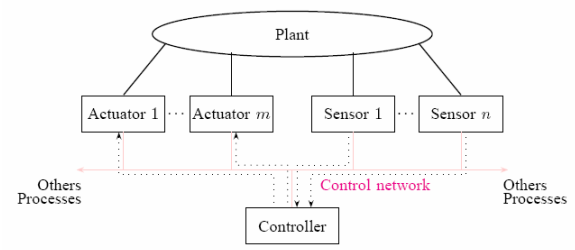

Fig. 1. A typical NCS setup and information flows

Thus, the behaviour study of NCS is mainly based on a stability analysis. And there are two strategies to compensate instabilities due to the network perturbations. From the control point of view, it is important to take into account these network induced delays in the control law design (Lian, 2001; Cervin et al., 2003). In this case the network is considered as passive. The second strategy consists in adapting network parameters to maintain a delay compatible with the constraints required by the application. More sophisticated approach (called co-design approach) could also be to compensate delays both at the controller level and at the network level in a coordinated way.
In this paper, the second approach is investigated. It means, the control law is fixed and the network parameters are dynamically adapted. The goal is to ensure the process stability in maintaining a sufficient level of Quality of Service by tuning the WRR parameters implemented inside Ethernet switches.
This research is mainly based on experimentations, because the works on NCS are often theoretical and use simulations for validation, but without using an experimental feedback.
The paper is composed of 4 sections. The section 2 details the experimental platform developed for analysing the behaviour of a NCS under network perturbations.
The section 3 proposes different triggers coming from the network or from the control system to adapt dynamically the network performances. A comparison between these approaches is finally presented in the conclusion.

2. EXPERIMENTAL PLATFORM

2.1 Introduction.

The experimental platform (figure 2) is constituted of a switched Ethernet network connecting four computers (running under Linux operating systems) representing respectively the controller, the process and two traffic generators. The role of the traffic generator (MGEN) is to study an "open" NCS sharing the network resource with other unspecified applications generating non real-time communications. These additional exchanges tend to overload the network and then to mitigate the performances of time constrained traffic between the controller and the process. The WRR policy implemented in the Ethernet switches enables to split the traffic in several classes according to their time constraints for guaranteeing a minimal service for each traffic class. Two traffic classes are defined: the first one corresponds to the traffic between the controller and the process (packet size = 48 bytes) and the second one is the background traffic (packet size = 1500 bytes) sent by the traffic generators to both the controller and the process. The ports of the Ethernet switch are configured at 10Mb/s in Full-Duplex mode.

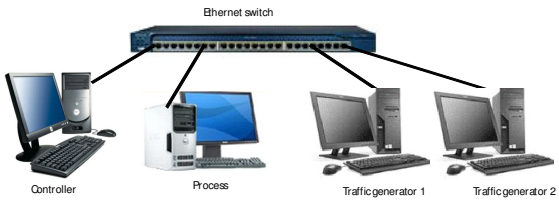

Fig. 2. Experimental platform

The NCS architecture used in the experimentations induced network delays when the controller sends new reference to the process and when the controller collects measures from the process. These delays have an impact on the performances of the system to be controlled in term of response time and in term of stability. To analyze the platform behavior, several algorithms are implemented on the controller collecting information on the system performances produced both by the process and by the network.
One algorithm enables to dynamically show the process behavior and one algorithm measures the current delays induced by the network between the controller and the process for observing the time constrained communications.
The measures of delay are based from a procedure ensuring the synchronization of all the device clocks using the IEEE 1588 protocol. Before sending a message, the computers tag it with the current time. The computer receiving the message has just to

compare the tag information associated to each message with its clock to measure the end-to-end delay.

2.2 Description of the process and the controller.

In evaluating the effects of the network on the control system performance, the following model of a real time process and a nominal controller were used (time in ms):

$$P(s) = \frac{2}{(s+5)(s+0.2)}$$

$$C(s) = \frac{K_P s + K_I}{s}, \quad K_P = 0.5508, \quad K_I = 0.4529$$

This controller was considered to be a nominal controller. The controller parameters for the nominal controller were obtained by minimizing the integral of the square errors (ISE) for the system. This delay corresponds to the delay of RT messages from the controller to the process added to the one from the process to the controller.

2.3. Influence of network induced delay on NCS

The experimental platform enables to observe the behavior of NCS in taking into account the network activity. In a first experiment, the network is a passive resource since the WRR policy is not implemented. The Background Traffic (BT) is fixed at 8 Mb/s. The figure 3.a. shows the background traffic impact on the control system. The Figure 3.b. describes the RT delays.

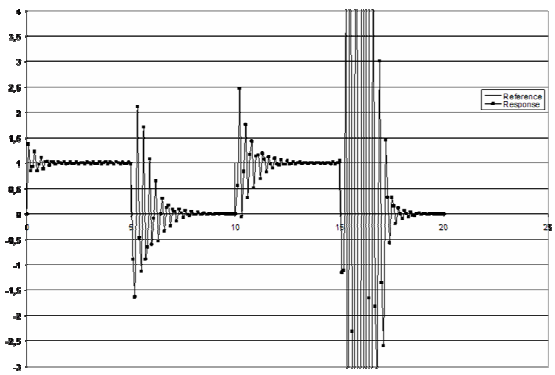

Fig. 3.a. System behaviour

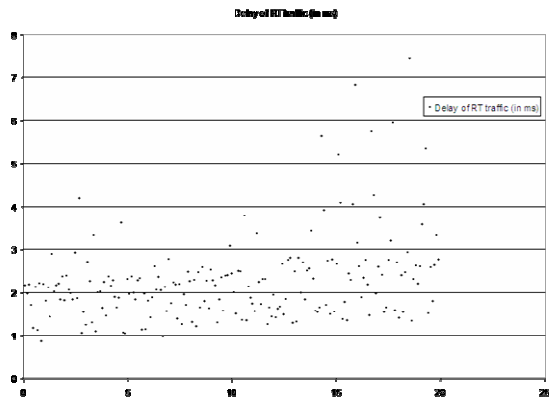

Fig. 3.b. RT Delays

The figure 3.b. shows that the background traffic generates an important jitter on the RT delays. The consequence is that when the delay of RT messages stays for a long period higher than the sampling period, the system becomes instable. This problem is observed in the figure 3.a. at t=15 s. In an other hand, the delays of RT traffic are rarely lower than the sampling time. Thus, the system has an unpredictable behaviour because of the important delays.

The interest of this experiment is to show, of course, that the stability of the system depends on the Quality of Service offered by the network, but also to point out the problem of delay variations. It means that the network performances can be adapted to the application constraints during a long period, but sometimes the congestion in Ethernet switch buffers increases (for example due to a burst of messages) and then increases the waiting time of messages inside the Ethernet switch.

2.3 Conclusion.

The conclusion is that a network with a large bandwidth is not sufficient to control real-time applications when the network is shared by other applications. It is necessary to be able to split the bandwidth according to the Quality of Control required by the applications.
Two problems have now to be solved. The first one is to estimate the levels of Quality of Service offered for each application in avoiding to generate famine state for the non real-time traffic. The second one is to find pertinent inputs to adjust dynamically the bandwidth allocation.
In NCS, the goal is to guarantee the stability of the system depending on the Quality of service. One input for maintaining the stability could be to measure the delays of time constrained messages and when these delays increase significantly, the network has to offer more bandwidth to these messages. A second input could be the error provided by the system controller. In this case, when the error is too important, the network has also to provide more bandwidth to the real-time traffic.
In the section 3, these two approaches are analysed and commented. Then, a third approach combining the two indicators is also studied. A fourth approach, based on third one and on an estimation of the bandwidth necessary to guarantee the system stability, is finally experimented.

3. CONTROL OF NETWORK

3.1 Control of network by measuring delay

In this approach, the IEEE 1588 protocol has to be implemented for measuring delays. The goal is to monitor the performances of time constrained messages and to maintain a delay compatible with the real-time application requirements. When the RT delays increase, the bandwidth weight for RT messages has to be incremented. The weight is a value comprised between 1 and 255 and the increment (respectively the decrement) is achieved

with a step of 10. This step will be adopted for all the approaches in this paper.

In this experiment, the increment procedure is triggered by a predefined threshold on the RT delay. The controller parameters for the nominal controller were obtained for a delay equal to 2 ms, a sampling period. Thus, this value has been chosen as the threshold. The algorithm to control the network is then both to progressively increment the background traffic weight and to mitigate the RT traffic weight when the RT delay is lower than the threshold and inversely when the RT delay is higher than the threshold. The figure 4.a shows the behaviour of the system by dynamically tuning the WRR weights in the Ethernet switch (figure 4.b) according to the delays (figures 4.c.). The RT delays were globally reduced comparing to the experiment where there is not any control of network (figures 3.b.). The RT delays of stay globally under 2,3 ms (90 % of the measured delays), and the mean of RT delays is equal to 1,73 ms. In this experiment, the system response reaches quickly the references (except for t=15s where the system waits 2,1 s before reaching the reference) and comparing with the results obtained without controlling network the overshot is mitigated at 94 % in the worst case. The figure 4.b. illustrates that this method gives a large (respectively a small) bandwidth to the background traffic (respectively to the RT traffic). After t = 1 s, the background traffic weight is always higher than the RT one. The mean weight offered to the RT traffic is 40,25 and the mean one offered to the background traffic is 213,75. These two weights offer a mean bandwidth equal to 9,94 Mbit/s for the background traffic.

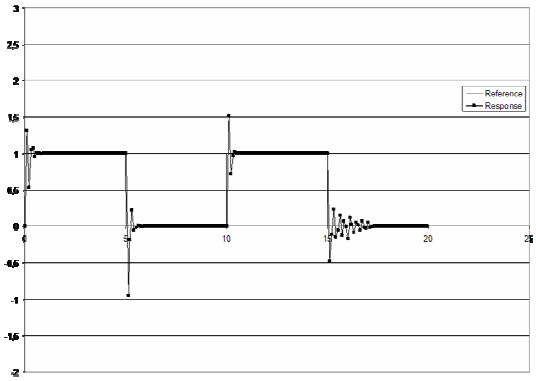

Fig. 4.a. System behaviour (case of delay compensation)

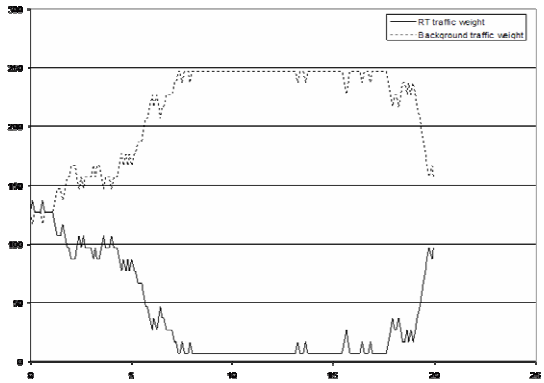

Fig. 4.b. WRR tuning (case of delay compensation)

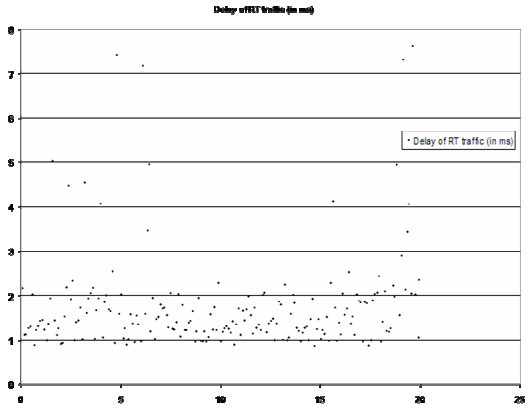

Fig. 4.c. Delay of RT messages (case of delay compensation)

To conclude, the interest of this approach is that it is a predictive method since it makes the corrections on the network before the occurrence of process instability matters.

3.2 Control of network by observing errors

In the second approach, the control of network depends on the difference between the reference and the response of the system. The analysis of both figures 5.a. and 5.b. shows the interaction between errors and network parameters.

This approach is easier to implement than the one used in the section 3.1, since the weights of WRR scheduling are tuned just by analysing the error. The implementation of IEEE 1588 protocol is not necessary.

The network is modified only when a system dysfunction is observed. The error threshold chosen is 5 %. The algorithm to control the network consists then in progressively incrementing the background traffic weight and decrementing the RT traffic weight when the error is less than the threshold and inversely if it is higher than the threshold. It also avoids to continually modifying the network parameters when the delay has no effect on the system and it offers a large bandwidth to the background traffic. The mean weight offers to the background traffic is equal to 230,8 and the mean one offered to the RT traffic is equal to 23,2. These 2 weights correspond to a mean background traffic bandwidth equal to 9,96 Mbit/s. Finally, in this approach, the relationship between the network correction and the process stability is directly connected. But with this method, the anticipation is not possible: the compensation on network is achieved when the matter is occurred. The experiment shows also that the maximum overshot (72% at t=15,1 s) is lower than in the case of delay compensation The RT delays (figure 5.c.) are globally lower than 2,6 ms (for 90 % of the measures) and the mean of RT delays is equal to 1,75 ms.

Finally, it is important to note, that the instability cause is not necessary due to the network degradation but can be generated by other factors. If the observation is based only the error and the matter is

not due to the network, the compensation action has no effect to improve the system performance.

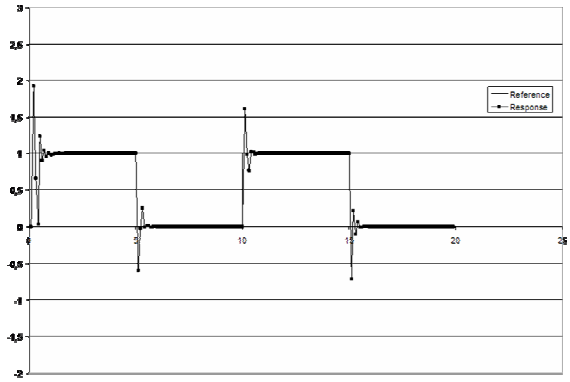

Fig. 5.a. System behaviour (case of error compensation)

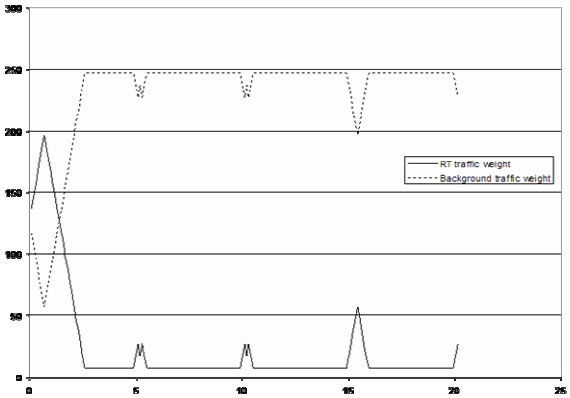

Fig. 5.b. WRR tuning (case of error compensation)

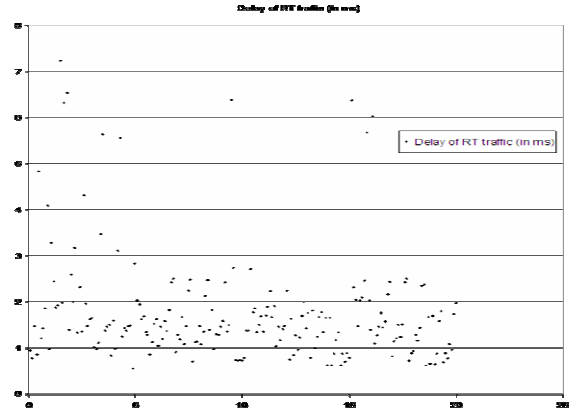

Fig. 5.c. RT Delays (case of error compensation)

3.3 Network control based on both RT delays and on the overshot

The combination of the two previous approaches enables to both anticipate system degradation by analysing the delay variations and to adapt network according to the current system state. The IEEE 1588 protocol has to be implemented for the delays measurements.

This approach is based on two thresholds: the RT delays have to be less than 2ms and the error has to be lower than 5%. When at least one of the two thresholds is reached, the weight for RT messages is progressively incremented and the one for background traffic is decremented and inversely if the two thresholds are not reached.

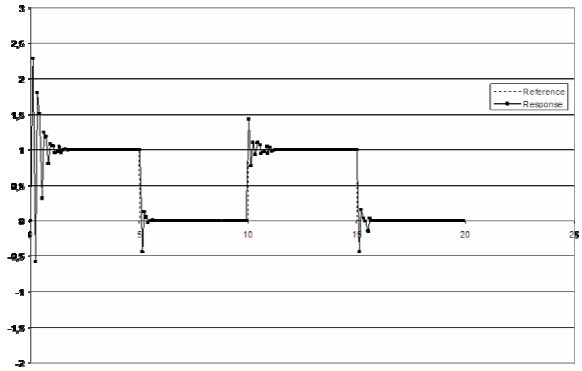

Fig. 6.a. System behaviour (case of delay and error compensation)

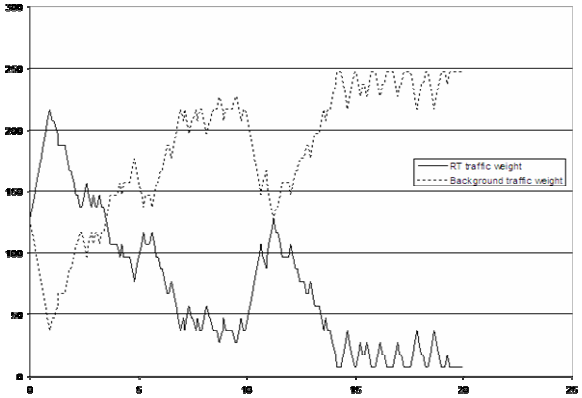

Fig. 6.b. WRR tuning (case of delay and error compensation)

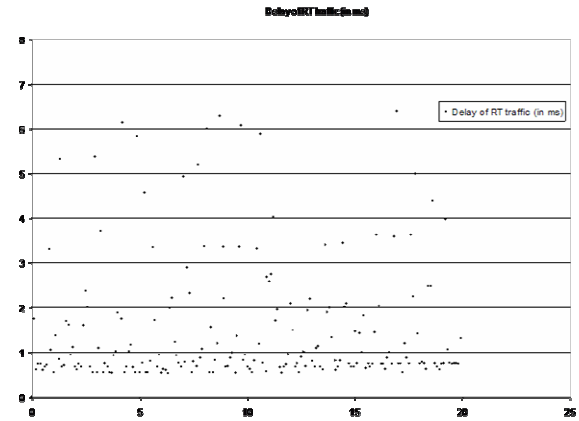

Fig. 6.c. Delay of RT traffic (case of delay and error compensation)

The figure 6.a shows the system behaviour by using dynamically the WRR policy implemented in the Ethernet switch according both to the RT delays and to the errors. The RT delays (figures 6.c.) stay under 3,3 ms for 90 % of the measures (more than 50 % of the measures are lower than 1 ms) and the mean of RT delays is equal to 1,52 ms, which is lower than the two previous cases. In this experiment, the system response always reaches quickly the references (except the first one where it takes 1,1s to reach the stability margin) and comparing with the results obtained using one threshold, the overshot is mitigated (44 % in the worst case after t = 1 s). In another hand, the figure 6.b. illustrates that this method offer a large (respectively a small) bandwidth to the background traffic (respectively to the RT traffic). The mean weight given to the background traffic is 182,5 and the mean one offered to the RT traffic is 71,5. These two weights correspond to a

mean bandwidth equal to 9,8 Mbit/s for the background traffic.
The interest of this network control based on the two thresholds is that it anticipates and reacts to the system response since it enables to avoid the occurrence of significant delays and errors.

3.4. Approach using the network calculus theory

This approach is the same as the previous one, but when at least one of the threshold is reached (the overshot or the RT delay), the weights are immediately set to pre-estimated weights by using the network calculus theory. In (Diouri & al, 2007) the upper bound delay can be obtained according to the values associated to w_1 and w_2.
The expression is :

$$\overline{D} = Max\left[\left(w_2\overline{\tau} + \frac{L}{C}\right), \left(w_2\overline{\tau} + \frac{\sigma}{C}\frac{w_1 L + w_2\overline{L}}{w_1 L}\right)\right]$$

Where:
w_1 (Respectively w_2) is the RT (background) traffic weight.
L (Respectively $\overline{L}$) is the RT (background) traffic frame length.
σ is the maximum amount of traffic that can arrive in a burst.
ρ is an upper bound on the long-term average rate of the traffic.

In the context of this paper:

$$\sigma = L \text{ and } C = 10Mbit/s.$$

Thus,

$$\overline{D} = w_2\overline{\tau} + \frac{L}{C}\frac{w_1 L + w_2\overline{L}}{w_1 L} = w_2\overline{\tau} + \frac{L}{C} + \frac{w_2\overline{L}}{w_1 L}$$

The algorithm enables to maximize the bandwidth allocated to the background traffic and to ensure that the upper bound delay from the controller to the process (or from the process to the controller) will be lower than $\overline{D}$. In this study, it is chosen that the delay upper-bound should be inferior to one sampling period. The pre-estimated weights (w_1 =130 and w_2=1) correspond to a safeguard used when the error is upper than 5% (error threshold) or when the RT delay is higher than 2 ms (RT delay threshold). The idea is to guarantee the system stability, in maintaining at the same time the maximum bandwidth for the background traffic. The algorithm consists then on progressively incrementing the background traffic weight and on decrementing the RT traffic weight if the error and the delay thresholds are not reached and if the delay supported by the control traffic is decreasing.
If the delay is increasing but the thresholds are still not reached, the weights are kept.

But finally if the error or the delay measurement is higher than the defined threshold, the weights are immediately set to pre-estimated values by using the safeguard.

The experiment shows that the maximum overshot (figure 7.a.) is 42 % (after t = 1s) and is lower than the others approaches. Also, the RT delays (figure 7.c.) stay globally under 2,9 ms (for 90 % of the measures) and the mean of RT messages delays is equal to 1,5 ms, which is lower than the others cases. In another hand, the mean weight offered to the background traffic is equal to 15,9 and the mean one given to the RT traffic is equal to 115,7. These two weights give a mean bandwidth equal to 8,1 Mbit/s for the background traffic.

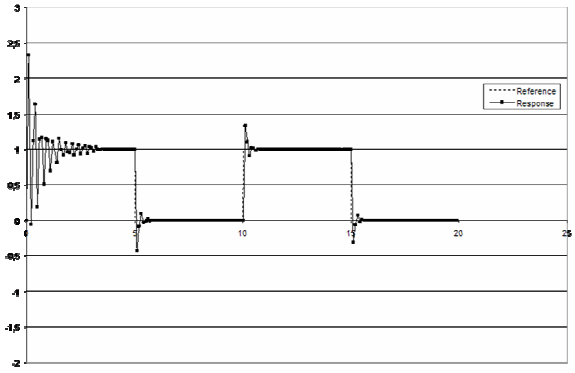

Fig. 7.a. System behaviour (case of compensation based on the Network calculus)

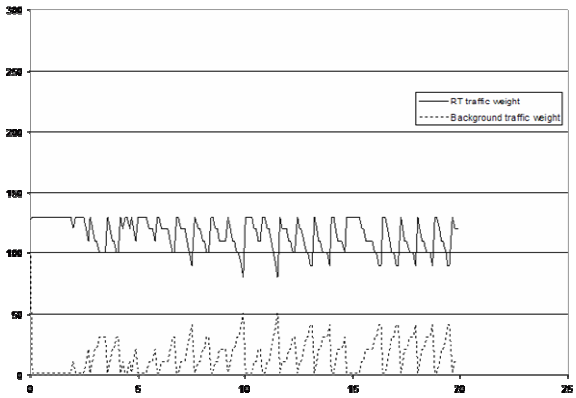

Fig. 7.b. WRR tuning (case of compensation based on the Network calculus)

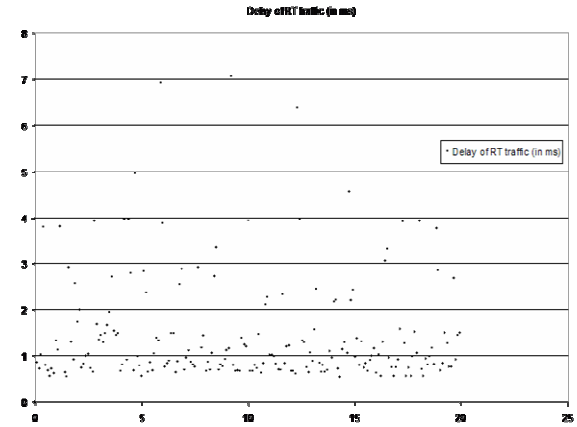

Fig. 7.c. RT Delays (based on the Network calculus)

4. CONCLUSION

The table 1 collects the whole results obtained in the different experiments. Table 1 contains performance index of each compensation method relatively to the paper objective (provide to the background traffic the maximum bandwidth in guaranteeing the process stability). The bandwidth indicator is estimated regarding to the mean cycle length and to the worst case for the background traffic. The comparison between the approaches cannot be generalised but gives a trend on how network compensation could be applied to networked control systems. The results presented in this paper correspond to results generally observed in others similar experiments.

Network compensation based on only delay enables to anticipate instability occurrences but without a straight relationship with the system behaviour. The table 1 shows that each compensation method has improved the control performance compared to networked systems without classification of service.

Network compensation based on only error signal ensures a link between the network performances and the controller performances, but the anticipation is not possible. The table 1 shows for these two approaches same results concerning the network performances and better results for the second approach on the controller performances.

The coupling of the two inputs: error and delay for controlling network improves the results both on the network and the process. In the fourth approach using the network calculus for pre-estimating the network configuration (used from disturbance detection either on the network or on the process) tends to penalise the bandwidth offered to the background traffic. But in this case, the controller has the guarantee to obtain the QoS required by the application. To note that anyway the network calculus penalty for the background traffic is limited by taking into account error and delay signals.

In conclusion, the network compensation has to be based both on network and process information. The perspective of this work is now to propose optimal strategy enables to tune the network parameters according to the application requirements, the state of the process and the state of the communication system. Petri Nets or other formal approaches could be useful to specify such strategy. In the context of hard real-time systems, the estimation of network configuration by using the network calculus is necessary even if the background traffic is less served by the schedulers. The problem, in this case, is to be able to earlier detect anomalies for avoiding instability of the system.

One another perspectives of this research is to couple network adaptation with controller adaptation to better minimize the impact of the network on the process to be remote controlled.

Type of compensation	Mean of RT delays (in ms)	Maximum overshot (in % after t=1s)	Bandwidth allocated to background traffic (in Mbit/s)
Without CoS	2,27	793,8	-
Delay	1,73	94	9,94
Error	1,75	72	9,96
Delay + error	1,52	44	9,8
Network calculus + error+ Delay	1,5	42	8,1
Network calculus	-	-	1,9

Tab.1. Results of the three methods

ACKNOWLEDGEMENTS

The authors wish to acknowledge funding support for this research under the European Union 6th Framework Program contract No. IST-2004-004303 "Network Control Systems: Tolerant to Faults" (NeCST).

REFERENCES

Branicky M., V. Liberatore, and S. Philips (2003). Networked control system co-simulation for co-design. In: *American Control Conference*. Vol. 4, pp. 3341-3346.

Demers, A., S. Keshav and S. Shenker (1989). Analysis and simulation of a fair queueing algorithm. *In: ACM SIGCOMM Computer Communication Review*. Vol. 19. pp. 1–12.

Diouri I., J.-P. Georges, E. Rondeau (2007). Accommodation of delays for NCS using classification of service. In : *IEEE International Conference on Networking, Sensing and Control ICNSC'07*. London, United Kingdom.

Gallara, D. and J.-D. Decotignie (1984). Groupe de réflexion FIP : proposition d'un système de transmission série multiplexée pour les échanges entre des capteurs, des actionneurs et des automates réflexes. Livre blanc, Ministère de l'Industrie.

Georges, J.P., E. Rondeau and T. Divoux (2005). Classification of service for networked control systems. In: *1st NeCST Workshop on Networked Control System Tolerant to faults. Ajaccio, France*.

Hoang, H., M. Jonsson, A. Larsson, R. Olsson and C. Bergenhem (2002). Deadline first scheduling in switched real-time Ethernet – Deadline partitioning issues and software implementation experiments. In : *1st International Workshop on Real-Time LANs in the Internet Age (RTLIA'02)*. Vienne, Autriche. pp. 68-71.

IEEE (1998). IEEE standards for information technology - Telecommunications and

information exchange between systems – Local and metropolitan area networks – Common specifications - Part 3: Media Access Control (MAC) bridges. ANSI/IEEE Std 802.1D, Edition 1998.

IEEE Computer Society (2003). IEEE Standards for local and metropolitan area networks - Virtual bridged local area networks. IEEE standard 802.1Q, Edition 2003.

Katevenis, M., C. Sidiropoulos and C. Courcoubetis (1991). Weighted round-robin cell multiplexing in a general purpose ATM switch chip. IEEE Journal on Selected Areas in Communications 8(9), 1265–1279.

Lian, F-L. (2001). Analysis, design, modeling and control of networked control systems. PhD thesis. Department of Mechanical Engineering, University of Michigan.

Lian, F.-L., J.R. Moyne and D.M. Tilbury (2001). Performance evaluation of control networks. IEEE Control Systems magazine 21(1), 66–83.

Liu, C.L and J.W. Layland (1973). Scheduling algorithms for multiprogramming in a hard-real-time environment. In : *Journal of the ACM (JACM)*, 20(1), 46-61.

Zhang, W., S. Branicky and S. Philips (2001). Stability of networked control systems. In: *IEEE Control Systems magazine*, vol. 21, pp. 84-89.