Networked Control Systems: From Independent Designs of the Network QoS and the Control to the Co-desig

Ye-Qiong Song

Networked Control Systems: From Independent Designs of the Network QoS and the Control to the Co-design

Ye-Qiong Song

LORIA - Nancy University - INPL
Campus Scientifique - BP 239 - 54506 Vandoeuvre-les-Nancy - France
song@loria.fr

Abstract: Taking a network QoS designer's point of view, this paper firstly reviews some of the recent advances on the NCS (Networked Control Systems) design then analyzes its *requirements* in terms of network QoS guarantees and its *capacity* to tolerate network performance variation. Current deterministic QoS design approach (including traffic schedulability analysis) may lead to network resource over-provisioning problem since worst-case scenario is often dictated to network QoS designers by the control application. We show that integrated control and network QoS co-design consists in a better solution to this problem. Taking into account the current available results, we see that more research efforts remain to do on control and network QoS co-design and on the development of on-line adaptive QoS mechanisms.

Keywords: Networked control system, Network, Quality of Service, Co-design.

1. INTRODUCTION

Process control is the most important class of real-time applications that industrial networks and especially fieldbuses support. Considering the distributed sensors, controllers and actuators interconnected by a network, and bearing in mind that deterministic or upper bounded end-to-end packet response time must be guaranteed, great efforts have been made since 1980 to develop various but IEC standardized solutions (Thomesse 2005), (Zurawski 2005). They are mainly MAC protocols but also application layer standards allowing the application interoperability. Real-time communication is ensured essentially at MAC layer. The main paradigm used to conduct those developments is assuming that messages to be transmitted by the network have deadlines that a network must meet, otherwise the solution is considered as invalidate since the consequence of missing a deadline is considered catastrophic for the application. Although this scenario could be true for real-time critical alarm reporting for example, for the most of control applications, occasional violation of a message deadline constraint can often be tolerated thanks to the closed-loop control robustness.

So looking for absolutely guaranteeing the deadlines of all the control loop related data exchanges is neither necessary nor optimal from resource utilization point of view. New design approaches should be developed. The resulting system that one can imagine is an adaptive one. Not only the network should be able to dynamically allocate necessary resources to a control application whenever needed, but also the application should accept some network QoS degradation resulting in a degraded but still acceptable QoC (Quality of Control). For a control application, dynamic network resources allocation can be done using network QoS adaptation mechanisms (e.g. priority re-allocation) according to the observation of the application related parameters such as the sensor-to-actuator delay, process state output deviation or some control loop cost function (Boughanmi *et al.* 2009), (Juanole and Mouney 2007). When the network QoS adaptation reaches its limit (especially during network overload periods), control applications must be adapted by adjusting for example the control loop sampling period (Eker *et al.* 2000), (Cervin 2003), (Marti *et al.* 2004), (Simon *et al.* 2005), (Antunes *et al.* 2007), or both the sampling period and control gain (Jia *et al.* 2007), (Felicioni *et al.* 2008). This results in a more radical solution to network overload problem than the QoS adaptation since the application traffic is reduced by increasing the sampling period.

All those approaches are co-design ones as they consider at the same time not only the network QoS but also the control application quality (QoC). In fact, for most of control applications, relaxing hard real-time constraint is not only possible thanks to the closed-loop control robustness, but also desirable in order to optimize the network resources allocation when they are shared by several applications.

The goal of this paper is to present some new trends toward the adaptive real-time NCS (networked control systems). In particular, integrated control and network QoS co-design approach is presented as an interesting direction.

The rest of the paper is organized as following. Section 2 gives a short review of the NCS and defines the co-design approach. Section 3 presents some control application-aware network QoS adaptation mechanisms. Section 4 gives some insights on the QoC adaptation for dealing with system overload (mainly on processor sharing but can be applied to network sharing). Finally Section 5 concludes the paper.

2. Networked control systems

Figure 1 shows a general NCS architecture in which the communication between sensors, actuators and controllers occurs though a multi-purpose network often shared by several control loops (Fig. 1)

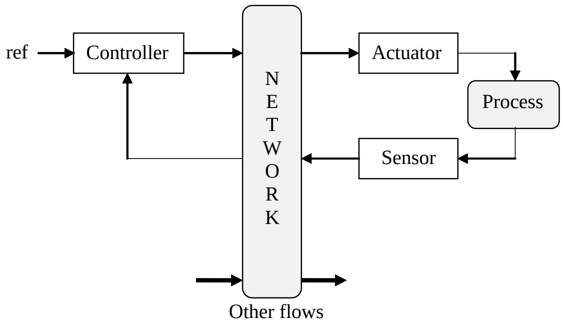

Fig. 1. *General architecture of NCS*

It is not so different from the fieldbus systems we are used to deal with in the factory automation community. However the focus is quite different. In fact, the fieldbuses or more generally speaking the industrial networks aim at providing general purpose real-time communication without considering one specific control application. The NCS is a research topic initiated by the automatic control community. There are two objectives: control over networks and control of networks. The former deals with control law design problems taking into account the network characteristics (e.g. delay and packet loss) while the latter focuses on the control of the network QoS by applying feedback control theory. There is abundance of research works dealing with NCS. Interested readers can refer to (Antsaklis and Baillieul, 2007) and (Zampieri 2008) for further details.

In this section we only focus on the control over networks. Readers can refer to (Antsaklis and Baillieul, 2007) for rather complete details on NCS.

2.1 Control loop robustness and QoC

Control systems are often considered as typical examples of hard real-time systems where deadline violation is strictly forbidden. Ensuring deadline meet has been the main preoccupation of the real-time scheduling theory, where the deadline constraint is directly deduced from the sampling period. With NCS, ensuring strict deadline meet is much more difficult because of the network QoS fluctuation, especially in case that the network is shared by several different applications. Fortunately experiments show that this hard deadline assumption may be false for closed-loop control. In fact, any practical feedback control system is designed to obtain some stability margin and robustness with respect to the process parameters uncertainty. This also provides some robustness with respect to timing uncertainties. It means that closed-loop control systems are able to tolerate some sampling period deviations and occasional data loss without losing the stability. In general, a commonly shared idea is that lower is the control period, better is the control performance. In (Cervin 2003), the control performance variation has been experimentally checked using an inverted pendulum with a LQ controller for different sampling period values. It has been shown that the control system remains stable although the performance degradation. In (Moyne and Tilbury 2007), an illustrative chart (Fig. 2) has been given to show the importance of choosing a good sampling period which gives trade-off between the control performance and the related network load.

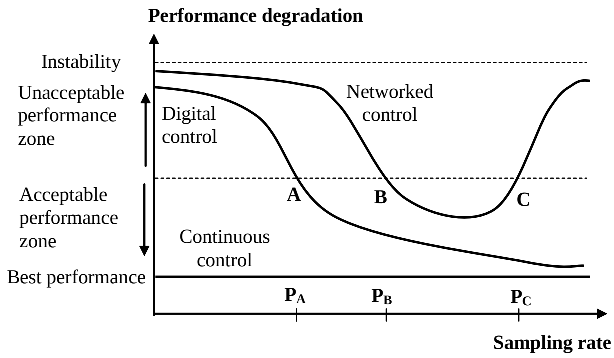

Fig. 2. *Performance comparison of continuous control, digital control and networked control vs. sampling rate*

From Fig.2, we can see that the control performance is acceptable for a range of sampling rate from P_B to P_C. This is to say that the network QoS can be variable correspondingly, giving thus a larger solution space to a network QoS designer. To characterize the quality of a control system, two important criteria are used: stability and control performance.

Let's consider a simple linear process being described by the following equations.

$$dx(t) = Ax(t) + Bu(t) + Gv(t)$$
and (1)
$$y(t) = Cx(t)$$

Where *x(t)* represents the process (plant) state, *y(t)* the output signal, $u(t)$ its input (command signal); $v(t)$ is white Gaussian noise disturbing the process state, which assumed to be independent and with zero mean. *A*, *B*, *C* and *G* are constant matrix describing the process dynamic.

And consider a proportional controller with gain *L*:

$$u(t) = Lx(t) \quad (2)$$

The stability is of primary important for a control loop. There exist several stability criteria (Aström and Wittenmark 1997). However the main idea is that the system state variation should be kept within a certain limits.

Typical performance criteria for feedback control loops include overshot to a step reference, steady-state tracking error, phase margin or time-average tracking error. Variance of the state variables *x(t)* is often used as a cost function to which the variance of the control effort *u(t)* is also added as shown in equation 3.

$$J = \frac{1}{H}\int_0^H (x^T(t)Qx(t) + u^T(t)Ru(t))dt \qquad (3)$$

Where Q and R are quadratic weight matrix representing the importance of the elements in the vectors *x(t)* and *u(t)* respectively. H is the time horizon during which the cost function is calculated. A great value of J indicates thus either a great deviation of the process state to a desired state or a great control effort to bring the state to its reference value. The system is instable when J tends to infinite.

In general, the QoC is defined as the control performance which can be measured using for example the cost function defined by equation 3.

2.2 NCS co-design

Most of the research works on NCS are from automatic control point of view and focus on the robust control loop design which takes into account the network induced delays and data loss. Network delays are assumed either constant (by input data buffering) or randomly distributed following a well-known probability law. Data losses are assumed following Bernoulli process. This paper does not have intention to review those results since very good reviews can be found in (Antsaklis and Baillieul, 2004), (Antsaklis and Baillieul, 2007), (Zampieri 2008).

From those works we can see that the assumptions on network delay or data loss pattern seldom consider the actual network characteristics and the possible QoS mechanisms which are specific from one type of network to another. Indeed, using a prioritized bus like CAN, a switched Ethernet or a wireless sensor network will result in fundamentally different QoS characteristics. This point is of primary importance especially when the network is shared by several control loops and other applications whose exact characteristics are often unknown at the control loop design step. In this case the traffic scheduling has a great impact on both delay variation and packet loss, which in turn impact on the quality of control (stability and performance) of the control loops.

One solution to this problem is tightly coupling the control with network during the design step which is called co-design approach (Branicky *et al.* 2003). One important issue towards this objective is the co-design tool development capable of modelling and simulating both the control and network parts. Mathlab/Simulink based TrueTime (Ohlin *et al.* 2007) is one of the useful tools for this purpose.

2.3 Control of networks

Control of networks has the same objective than the traditional network QoS design. However they are different in terms of the used approaches. Traditional network QoS design approach uses rather static resource allocation principle by scheduling the different resource demands of the different network data flows (or applications). The control of networks approach is based on the dynamic feedback control of the resource allocation to the network data flows for maintaining their desired QoS. This requires the monitoring and on-line modification of the network parameters, which are not easily implementable considering the current layered communication protocols. A typical example is the control of the network congestion by using a stochastic controller LQG (Altman, 1999). For a condensed review of the recent works, readers can refer to (Zampieri, 2008).

Using feedback control theory to deal with congestion and resource allocation of wireless networks is an interesting and challenging issue. Link capacity between two nodes is time varying. Fortunately current wireless protocols like IEEE802.15.4 and IEEE802.15.4a (UWB) provide cross-layer optimization possibility allowing dynamic network parameter changes such as transmit power, path selection based on multiple criteria (e.g. LQI, ED, end-to-end delay requirement, time-slot and channel allocation in IEEE802.15.4) (Li *et al.* 2007). Figure 3 gives the idea of control and network QoS co-design by adding the control of the networks. This architecture should result in an adaptive NCS. However further research efforts still remain to do since mathematic modelling of the network behaviours is not easy.

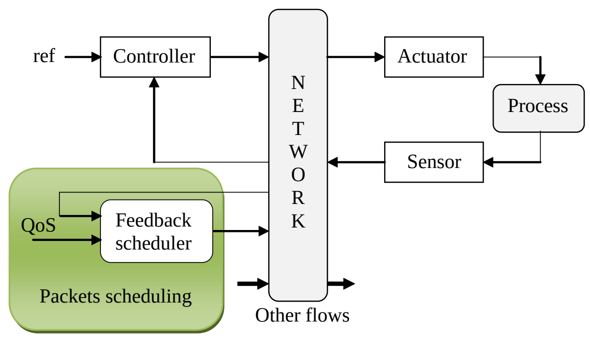

Fig. 3. *Control and network QoS co-design*

In the next section, we choose to only review some simpler QoS adaptation mechanisms without necessarily using feedback control theory.

3. Application-aware dynamic QoS adaptation

Dynamic network QoS adaptation aims two objectives: adaptation to the network operating condition and to the change of the application requirements. QoS adaptation to the network operating condition is normally ensured by the existing QoS mechanisms and will not be reviewed in this paper.

NCS can use network QoS adaptation mechanisms (e.g. priority re-allocation) according to the observation of the application related parameters such as the sensor-to-actuator delay, process state output deviation or some control loop cost functions such as what is define by equation 3. There are several works on this direction which can give inspiration for further adaptive QoS mechanisms development.

3.1 Dynamic CAN message priority allocation according to the control application needs

In (Juanole and Mouney 2007), a hybrid CAN message priority allocation scheme is proposed. This scheme is inspired from the mixed traffic scheduling scheme defined by (Zuberi and Shin 1997). The idea is to separate CAN identifier bits into two fields: one for dynamic priorities and one for static priorities. In the normal case, messages are assigned with static priorities. When there is a urgent transmission need, dynamic priority field can then be used giving thus higher priority to the urgent transmission. The main contribution of the work of (Juanole and Mouney 2007) resides in the exhibition of a link between the hybrid priority scheme and the control loop performance. The studied NCS is similar to that of Fig. 1. The process to control is a DC-servo process and the controller is PD (Proportional Derivative). Following cost function is used which interests in the difference between the reference *r(t)* and the process state output signal *y(t)* :

$$J = \frac{1}{H}\int_0^H t(r(t) - y(t))^2\,dt \qquad (4)$$

The key issue is to define a relation to translate the control need in terms of the QoS (mainly message transmission delay) into a dynamic priority. In this work authors proposed to use the control signal *u* as the indicator to trigger dynamic priority allocation. The triggering is done by the controller node when it receives the CAN message that the sensor node sends at each sampling period. According to the sensor data, the controller calculates the new value of *u*, and decide to choose a new priority to send the command data if necessary. So the important result in this work is that the priority to be used to send a command data through CAN network is a function of the control signal *u* so that the network QoS is dynamically adjusted to accommodate to the dynamic control application needs.

3.2 Dynamic priority in IEEE802.15.4/Zigbee according to the NCS needs

IEEE802.15.4/Zigbee based wireless sensor network technology is becoming attractive for industrial communication (Willig 2008), (Mouney *et al.* 2008). Wireless NCS is a new and important topic that should be further investigated. In (Boughanmi *et al.* 2009) we proposed a simple dynamic priority scheme, which for the moment only provides two priorities. The key idea explored is that when the QoC of the controlled system is not sufficient, the quality of service of the network is dynamically adapted by giving higher priority (thus less delay) to the important control data transmission. This adaptation is realized by modifying the macMinBE parameter of the MAC protocol of the IEEE 802.15.4. For illustrating the approach, a cart position control system is considered (Fig. 4) and its control loop is given in Fig. 5.

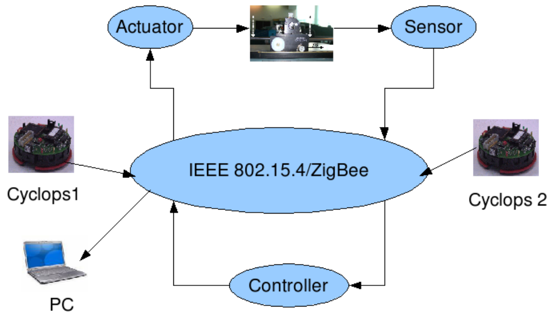

Fig. 4. *IEEE 802.15.4/Zigbee using CSMA/CA*

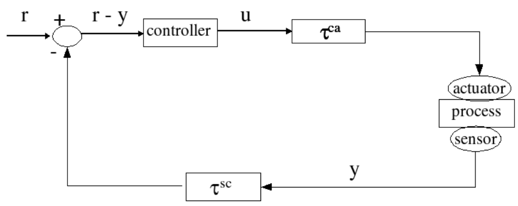

Fig. 5. *Control loop*

In this example, CSMA/CA MAC instead of GTS has been chosen. In order to make CSMA/CA support different priorities, in (Boughanmi *et al.* 2009) particular protocol parameter configuration is suggested which consists in choosing different upper ranges of the backoff exponent for the control loop related nodes when the QoC begins to decrease. The network is shared by other applications generating an additional workload. In normal network operating condition (i.e. light workload), all the nodes have the same priority. By increasing the external workload, the traffic generated by the control loop is disturbed resulting more packet delay or loss, which in turn impact the QoC. In this case, it is preferable to increase the priority of the control loop related traffic.

For observing the QoC variation the control error $e = |r - y|$ is used. The error should be bounded by a threshold to ensure the required QoC to the plant. This threshold depends on the controlled process and on the reference value if there is any. If *e* > *threshold*, the system is considered to be in a critical situation and action has to be taken. The trouble is caused either by the controlled system itself or by the network (overloaded network). Action should be taken on the network for a certain period of time through offering more resources to the wireless NCS. If the situation is still not improved (there is a problem in the control loop), action should be taken on the control loop by changing the sampling period for example.

Fig. 6 and 7 show respectively the simulated system response with the same workload without and with the dynamic QoS adaptation. The effectiveness of the adaptation can be easily seen.

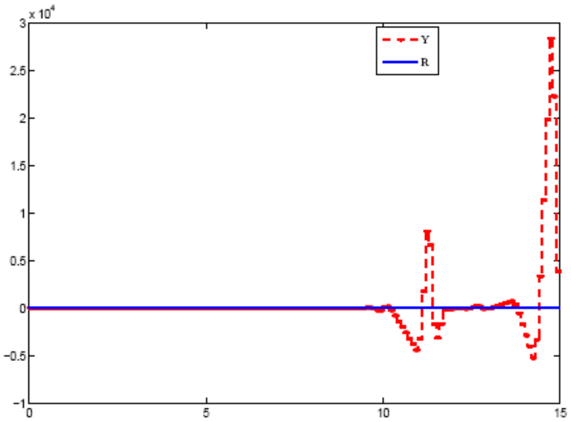

Fig. 6. *System response using CSMA/CA*

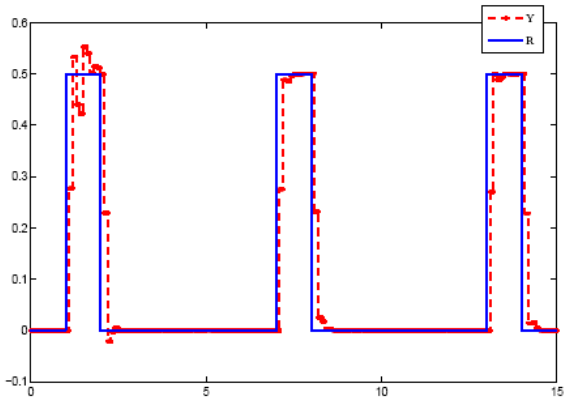

Fig. 7. *System response using dynamic QoS adaptation with threshold = 0.5*

It is worth noting that with IEEE802.15.4, it is indeed also possible to get real-time communication feature by using GTS (Guaranteed Time Slots) (Koubâa *et al.* 2006) (Koubâa *et al.* 2007), (Francome 2007). Further effort should be made to find a concrete way to dynamically allocate the GTS slots according to the QoC variation. In fact, the standard mechanism only allows the nodes to ask to the coordinator for obtaining the desired GTS slots by sending their demands via CSMA/CA. This may result in important demand latency.

3.3 Discussions

A lot of other related works contribute to this topic but they will not be reviewed here because of space limitation (Walsh et al. 2002). Considering the important position of Switched Ethernet in industrial communications, we just cite the work of (Diouri *et al.* 2007) that dealt with the dynamic allocation of bandwidth share in Ethernet switches with WRR (Weighted Round-Robin) scheduler according to both observed delay and the QoC (difference between the reference and the process state). For this purpose, the bandwidth share (i.e., the weight assigned to each data flow or Ethernet switch port) is defined as a function of the sensor to actuator delay and the current QoC level.

Instead of looking for meeting deadlines by static network resource allocation, the works reviewed in this section present a new approach which consists in dynamically allocate network resource according to the control application needs. This leads to more effective resource utilization since network resource can be used by other applications during the period that a specific control application does not need. For embedded systems with resource constraint, this application-aware QoS design approach allows also avoid resource over-provisioning problem as the resource is no longer dimensioned for the worse-case. In summary, dynamic QoS adaptation according to network operating condition and QoC requirements effectively improve the total QoC of the NCS. However for each type of network, one has to find not only the critical QoC condition for triggering the QoS adaptation, but also the corresponding mechanisms to act on the network parameters (Mouney *et al.* 2008). This calls for more research effort.

4. QoC adaptation to network operating conditions

Network QoS adaptation can enhance the NCS performance until some limit. Beyond this limit, the control application should re-adjust its own parameters to continue to run even in a degraded mode. This can happen during network overload in NCS where a network is shared by several control loops. A common approach to deal with this problem is to dynamically change the sampling period. In fact the works reviewed in the previous section have in most of the case a control loop with variable sampling period because of the no constant network induced delay. The difference is that in previous section, we suffer from this sampling period variation and only act on the network QoS (by dynamic resource re-allocation) but not on the control loop itself. In this section we review some representative works that dynamically change the control loop parameters when, for example, the underlying network can no longer provide requested QoS. We choose to present two types of solutions: explicit sampling period adjustment and indirect sampling period adjustment which is based on selective packet drops according to (m,k)-firm model (Hamdaoui and Ramanathan 1995).

4.1 Explicit sampling period adjustment approach

The first work dealing with sampling period adjustment is reported in (Seto *et al.* 1996). The problem considered is the optimization of the QoC of a set of control loops sharing a same resource (processor). The QoC of each control loop is measured by the cost function (equation 5) which is similar to equation 3 but it is a function of the sampling period *h.*

$$J(h) = \lim_{H\to\infty} \int_0^H (x^T(t)Qx(t) + u^T(t)Ru(t))dt \qquad (5)$$

The problem to solve is then the calculation of the optimal sampling periods minimizing the weighted sum of the cost function of each control loop:

$$\min_{(h_1,\ldots,h_n)} \sum_{i=1}^{n} w_i J(h_i) \quad (6)$$

under constraint of the schedulability:

$$\sum_{i=1}^{n} C_i / h_i \le A \quad (7)$$

Where w_i is used for weighting the importance of a control loop with respect to the others. C_i is the execution time of the control task i. $0 < A \le 1$ is the processor utilization threshold guaranteeing the tasks schedulability and whose value depends on the used scheduling policy.

This method has been extended in (Eker *et al.* 2000) and (Cervin 2003). On-line adaption has been proposed to accommodate to the application configuration change. This is achieved using a regulator that monitors the current configuration of the application. As soon as a configuration change trend is detected by this regulator, the optimal periods for the new configuration are calculated according to an approximate method of the equations 5 to 7 (because of the high computing complexity of those equations). Furthermore, in (Henriksson and Cervin 2005) the process state has been included in the optimal period calculation. This allows to favor control loops whose state variables is experiencing great changes comparing to those whose state variables are more stable.

Although current existing works only deal with control tasks sharing a cpu, similar principle should also be applicable to control loops sharing a network.

4.2 Sampling period adjustment with selective packet dropping

Adjusting sampling period could lead to implementation difficult for general purpose network since at network overload situation, the period change decision must still be transmitted to the sensor nodes via network, implying sometimes complicated mechanisms. In (Jia *et al.* 2007) and (Felicioni *et al.* 2008) we proposed an indirect sampling period adjustment method which selectively drops some sampling packets in case of network (or processor) overload. The sampling period of a control loop is therefore increased although this alternative can only adjust the period by the multiple of the nominal one. The result is less accuracy than the direct sampling period adjustment method but it often implies less implementation complexity.

It is well known that data losses have a direct impact on the QoC. But most importantly is to avoid long term consecutive data losses (Felicioni *et al.* 2009). One way to avoid long term consecutive data losses is to use (m,k)-firm model introduced by (Hamdaoui and Ramanathan, 1995). This model can be used, for instance, to indicate that at least m out of any k consecutive sampling packets has to be sent to the controller within their deadline, where m and k are two positive integers with $m \le k$ (the case where $m=k$ is equivalent to the ideal case, which is noted by (k,k)-firm and corresponds to the hard real-time constraint). If a control system is designed to accept a control performance degradation until k-m deadlines misses (or equivalently packet losses) among k consecutive ones (this can be justified by the observation that most control systems can tolerate misses of the control law updates to a certain extent), the system can then be designed according to the (m,k)-firm model to offer the variable levels of QoC between (k,k)-firm (ideal case) and (m,k)-firm (worst case) with as many intermediate levels as the possible values there are between k and m. This results in a control system with graceful degradation of control performance (Ramanathan 1999).

In (Felicioni *et al.* 2008), we considered N physical plants with one dedicated controller implemented as a real-time task for controlling each plant. Each instance of a task is responsible for carrying out the control law computation and has a deadline by which it is expected to complete its computation. We consider a centralized implementation of all the controllers as shown in Fig. 8. Note that the same approach applies to the case where N control loops share a same network.

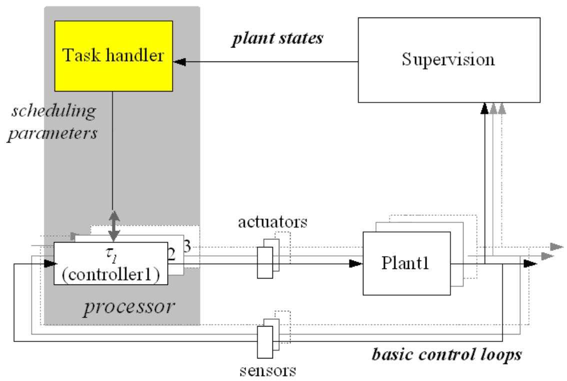

Fig. 8. *System architecture with N control loops sharing a same processor*

At any time, there are n activated plants (with $n \le N$), i.e. n tasks must be computed. This raises the problem of the schedulability of these n tasks. Each task is under (m_i,k_i)-firm constraint. Assume that each task τ_i has period h_i (corresponding to the nominal sampling period of the related control loop) and the worst case execution time C_i. The value of k_i corresponds to the maximum number of consecutive non-execution instances that the ith control loop can accept before going to the instability. m_i corresponds to the desired QoC. The instances of each task are partitioned into two sets: the mandatory instances and the optional instances. An instance of τ_i, activated at time aT_i, for $a = 0,1,\ldots$ is classified as mandatory if

$$a = \left\lfloor \left\lceil \frac{am_i}{k_i} \right\rceil \frac{k_i}{m_i} \right\rfloor \quad (8)$$

and as optional, otherwise.

The control tasks are scheduled using the fixed priority policy. The mandatory instances of all the tasks are assigned the rate-monotonic priorities. That is, the mandatory instances of τ_i are assigned a higher priority than the mandatory instances of τ_j if $h_i<h_j$. The optional instances are

assigned the lowest priority. Following sufficient condition ensures the schedulability.

Given a task set ($\tau_1, \tau_2 \ldots \tau_n$) such that $h_1<h_2< ..<h_n$. Let:

$$n_{ij} = \left\lceil \frac{m_j}{k_j} \left\lceil \frac{h_i}{h_j} \right\rceil \right\rceil \tag{9}$$

If $C_i + \sum_{j=1}^{i-1} n_{ij} C_j \le h_i$ *for all* $1 \le i \le n$, *then the* (m_i, k_i)*-firm constraint of each task* τ_i *is satisfied.*

In (Jia *et al.* 2007), the cost function defined by equation 5 has been adapted to include also the distribution of m_i among k_i instances and an optimization problem is formulated similar to that of (Seto *et al.* 1996) but with equation 9 as the schedulability constraint. Moreover a method is given to calculate the optimal control gain for each value of m_i.

For giving a concrete insight, let's consider a system composed of four control loops for controlling four carts and sharing a same processor (Jia et al. 2007). At a system configuration change, the task handler (Fig.8) receives the information about the number of tasks sharing the processor and the actual execution time C_i of each task, and deduce the new (m_i,k_i)-firm constraint as well as the corresponding optimal control gain for each control task by resolving the optimization problem.

Fig. 9 shows the simulation trace of the system without instance dropping according to (m,k)-firm. At t = 2, τ_3 is turned on. Together with τ_1 and τ_2, the pre-emption due to these tasks makes the execution of τ_4 impossible. As a result, the cart system $Cart_4$ becomes instable since all its instances miss their deadline.

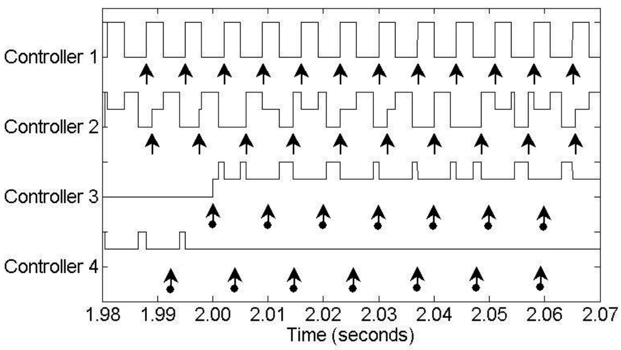

Fig. 9. *Close-up of schedule at t=2 under traditional scheduling*

Fig. 10 shows the system configuration change at t = 2. The (m,k)-firm constraints of the tasks τ_1, τ_2, τ_3 are adjusted to respectively (2,5), (4,8), and (3,10)-firm. Note that the task deadline violations after the (m,k)-firm constraint adjustments in Fig. 10 is due to the transient overload, however, the overload condition is removed rapidly.

Compared with the traditional scheduling approach, the controllers perform much better. The $Controller_4$ is stable.

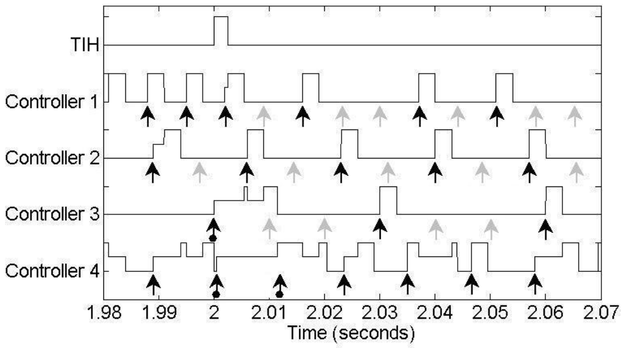

Fig. 10. *Close-up of schedule at t = 2 under scheduling approach with (m,k)-firm constraint regulation*

5. Conclusion

NCS is a subset of the DCS (Distributed Control Systems) on which the factory automation community has worked for long time. But NCS is also a particular DCS application with special features implied by the control loops specific behaviours. Especially with the recent results obtained on NCS, we see that control loops are often robust and can tolerate network performance variation until certain extent. So specifying more precisely the QoS requirement levels of a control application and their distribution in time could give opportunity for the network QoS designer to propose more resource-utilization-efficient solutions. This requires of course a deep knowledge on both control application and the underlying network technology. It calls for developing the integrated control and network QoS co-design methods.

In this paper we presented the control and network QoS co-design approach. Two important points are presented. One is the control application-aware dynamic QoS adaptation. It allows providing the QoS to the applications only when needed, releasing thus network resource to other applications. Another is to make the control application adaptive to the network operating condition variation. It leads to more robust NCS. So the advantage of this integrated co-design approach is on the one hand the great improvement of the system robustness, and on the other hand the minimisation of the resources necessary for meeting the required QoC. This minimisation is of special interest for autonomous embedded systems.

Networked Control Systems form a growing field and call for the development of integrated approaches requiring multidisciplinary skills in control, real-time computing and communication protocols. More research efforts remain to be provided on control and network QoS co-design and on the development of on-line adaptive QoS mechanisms, especially for wireless NCS. Another interesting direction is applying feedback control theory to the network QoS control. Further research efforts are necessary for the mathematic modeling of pertinent network behaviors before applying to a control loop. Its interaction effect with the application control loop has also to be investigated.

REFERENCES

Altman E., T. Basar, and R. Srikant, "Congestion control as a stochastic control problem with action delays," *Automatica*, vol. 35, pp. 1937-1950, Dec. 1999.

Antsaklis P. and J. Baillieul (editors), "Special issue on networked control systems", *IEEE Trans. On Automatic Control*, VOL.49, NO.9, September 2004.

Antsaklis P. and J. Baillieul (editors), "Technology of networked control systems", *Proceedings of the IEEE*, VOL.95, NO.1, Jan. 2007.

Antunes A., P. Pedreiras, L. Almeida and A. Mota, "Dynamic rate and control adaptation in networked control systems", *5th International Conference on Industrial Informatics*, Vol. 2, Vienna, Austria, June 2007.

Aström, J. and Wittenmark, B., (1997). *Computer-controlled Systems: Theory and Design*. Prentice-Hall International Inc, second edition.

Boughanmi N. , Y.Q. Song and E. Rondeau, "Online adaptation of the IEEE 802.15.4 parameters for wireless networked control systems", *IFAC FeT2009*, Ansan, Korea, May 2009.

Branicky M.S., V. Liberatore and S.M. Phillips, "Networked control systems co-simulation for co-design", *American control conference*, Denver, USA, Vol. 4, pp3341-3346, June 2003.

Cena G., A. Valenzano and S. Vitturi, "Hybrid wired/wireless networks for real-time communications", *IEEE industrial electronics magazine*, Vol. 2, No. 1, pp8-20, March 2008.

Cervin A., *Integrated control and real-time scheduling*, PhD thesis, dept. of automatic control, Lund Institute of Technology, Sweden, April 2003.

Diouri I., J.P. Georges and E. Rondeau, "Accommodation of delays for NCS using classification of service", *International conference on networking, sensing and control*, London, UK, April, 2007.

Eker J., P. Hagander, and K. E. Årzén, "A feedback scheduler for real time control tasks", *Control Engineering Practice*, Vol. 8, pp. 1369-1378, 2000.

Felicioni F., N. Jia, F. Simonot-Lion, and Y.Q. Song, "Optimal on-line (m,k)-firm constraint assignment for real-time control tasks based on plant state information", *13th IEEE ETFA*, Hamburg, Germany, 2008.

Felicioni F., F. Simonot, F. Simonot-Lion and Y.Q. Song, "Optimal control with packet drops in networked control systems", *IFAC FeT2009*, Ansan, Korea, May 2009.

Francomme J., G. Mercier, and T. Val, "Beacon synchronization for GTS collision avoidance in an IEEE 802.15.4 meshed network" *IFAC FET'07*, Toulouse, France, 7-9, November 2007.

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.

Henriksson D. and A. Cervin, "Optimal on-line sampling period assignment for real-time control tasks based on plant state information", *44th IEEE Conference on Decision and Control and European Control Conference ECC 2005*, Seville, Spain, Dec. 2005.

Jia N, Y.Q. Song and F. Simonot-Lion, "Graceful Degradation of the Quality of Control through Data Drop Policy", *European Control Conference 2007 (ECC'07)*, Kos, Greece, July 2007.

Juanole G. And G. Mouney, "Networked control systems: definition and analysis of a hybrid priority scheme for the message scheduling", *IEEE RTCSA 2007*, Daegu, Korea, August 21-23, 2007.

Koubâa A., A. Cunha, and M. Alves, "A time division beacon scheduling mechanism for IEEE 802.15.4/Zig-Bee cluster-tree wireless sensor networks", *19th ECRTS*, July 2007.

Koubâa A., M. Alves, and Y. Song B. Nefzi. "Improving the IEEE802.15.4 slotted CSMA/CA for time-critical events in wireless sensor networks", *RTN 2006 Satellite Workshop to ECRTS 2006*, Dresden, Germany, July 2006.

Li Y., Chen C. S., Song Y.-Q., Wang Z., "Real-time QoS support in wireless sensor networks: a survey", *7th IFAC FeT'2007*, Toulouse, France, Nov. (2007)

Marti P., J. Yepez, M. Velasco, R. Villa, and J.M. Fuertes, "Managing quality of control in network based control systems by controller and message scheduling co-design", *IEEE transactions on industrial electronics*, Vol. 50, pages 1159-1167, Dec. 2004.

Mouney G., J. Guy, and C. Christophe, "On the implementation of one process control application type through a network considering three LANs: CAN, WiFi, ZigBee", *17th IFAC World Congress*, July 6-11 2008.

Moyne J. R. and D. M. Tilbury, "The emergence of industrial control networks for manufacturing control, diagnostics, and safety data", *Proceedings of the IEEE, special issue on Technology of networked control systems*,VOL.95, NO.1, pp29-47, Jan. 2007.

Ohlin M., D. Henriksson, and A. Cervin. *TrueTime1.5-Reference Manual*. Department of Automatic Control Lund University, 2007.

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.

Seto D., J. P. Lehoczky, L. Sha and K. G. Shin. "On task schedulability in real-time control system", Proceedings of the *17th IEEE Real-Time Systems Symposium*, pages 13-21, Washington, DC, Dec 1996.

Simon D., D. Robert, and O. Sename, "Robust control/scheduling co-design : application to robot control", *IEEE RTAS'05*, San Francisco, USA, Mar. 2005.

Thomesse J.P., "Feildbus technology in industrial automation", *Proc. IEEE*, Vol. 93, No. 6, pp1073-1101, June 2005.

Walsh G. C., H. Ye, and L. Bushnell, "Stability analysis of networked control systems", *IEEE Transactions on Control Systems Technology*, number 10(3): 438-446, 2002.

Willig A., "Recent and emerging topics in wireless industrial networks: A selection", *IEEE Trans. on Indus. Informatics,* Vol. 4, No. 2, pp102-124, May 2008.

Willig A., K. Matheus and A. Wolisz, "Wireless technology in industrial networks", *Proc. IEEE*, Vol. 93, No. 6, pp1130-1150, June 2005.

Zampieri S., "Trends in networked control systems", 17th *IFAC World congress*, pp2886-2894, Seoul (Korea), July 6-11, 2008.

Zuberi K. and K. G. Shin, "Scheduling messages on controller area network for real-time cim applications," *IEEE Transactions on Robotics and Automation*, vol. 13, no. 2, pp.310–314, 1997.

Zurawski R. (editor), *The industrial communication technology handbook*, Taylor&Francis, CRC press, 2005.