ROBUSTNESS EVALUATION OF AN ADAPTIVE CONTROL METHODOLOGY FOR NETWORKED CONTROL SYSTEMS

Eduardo Paciência Godoy, epgodoy@yahoo.com.br
Robson Rogério Dutra Pereira, robson.pereira@gmail.com
Arthur José Vieira Porto, ajvporto@sc.usp.br
Department of Mechanical Engineering, São Carlos School of Engineering - EESC,University of São Paulo - USP
Avenida Trabalhador São carlense, 400 CEP 13566-590, São Carlos, São Paulo, Brazil

Ricardo Yassushi Inamasu, ricardo@cnpdia.embrapa.br
EMBRAPA - Brazilian Agricultural Instrumentation Research Corporation
Rua XV de Novembro, 1452 CEP 13560-970, São Carlos, São Paulo, Brazil

Abstract. *Recently networked control systems (NCS) have become one of the research topics in control that continuously receives an increasing attention. This is due to the widely application of fieldbus based distributed control systems in nowadays industrial control solutions. The NCS concept differs from the traditional fieldbus systems in that the controller and the plant are physically separated and connected through an industrial network. New control methodologies have been developed in order to mitigate the degenerative effects of the messages sampling time, information lost on the network and time variable network delays on the NCS performance and stability. However, an important stage in these NCS controllers development, which is sometimes neglected, is its robustness assessment. The robustness is the ability of a controller to maintain good performance for the NCS even in the presence of significant disturbances or NCS parameter variations. In the present paper, the robustness of an adaptive control methodology developed for CAN-based NCS is evaluated. The performance of the adaptive control methodology, which automatically changes the NCS sampling time in accordance with the NCS output, is analyzed for different NCS configurations used on a CAN-based NCS research platform. The results lead to the conclusion that the developed control methodology is reliable for application to NCS based on the CAN protocol, providing an acceptable performance for the NCS even in the worst-case scenario analyzed.*

Keywords: networked control system, messages sampling time, performance evaluation

1. INTRODUCTION

Recent applications of distributed fieldbus based control systems demonstrate a new approach for the use of industrial networks. In this type of application, named Networked Control Systems (NCS), the controller, the sensor and the plant are physically separated and connected through a communication network (Yang, 2006) as shown in Fig. 1. The control signal is sent to the controller by a message transmitted over the network while the sensor samples the plant output and returns the information to the controller also transmitting a message over the network.
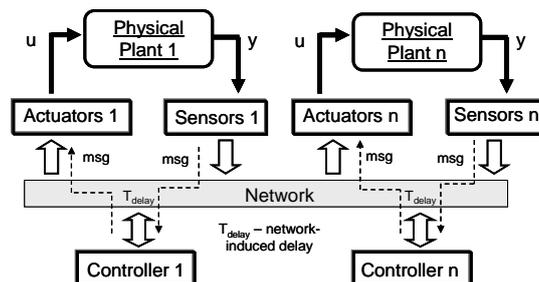

Figure 1. Structure of a NCS

Networked control systems impose additional problems inherent in control applications that are usually difficult to meet due to the variations and uncertainties introduced by the communication network: delays, *jitter*, bandwidth limitations and packet losses (Baillieul and Antsaklis, 2007). The network-induced delays and data packet dropouts occur when sensors, actuators, and controllers exchange data across the network. The characteristics of network-induced delays are mainly determined by the protocol used in the NCS (Lian et al., 2002). Moyne and Tilbury (2007) describe the components of the network-induced delays in NCS and present the possible effects of the communication network in the performance and stability of NCS. Because of the network delays variability, the NCS can present characteristics of time variant systems doing the analysis and design of these systems even more complicated. Other problems related to the NCS are the correct choice of the network configuration parameters such as the network

bandwidth, messages data length and messages sampling time. These parameters also influence the network performance (Lian et al., 2002). The NCS must sample and transmit data at a sampling time appropriate to achieve required performance metrics. However, if this sampling time is higher than the network bandwidth available, the network becomes overloaded, originating additional network delays and jitter, and causing packet losses and errors transmissions (Al-Hammouri et al., 2008). Therefore, NCS generally must meet two main criteria: bounded network delay and guaranteed transmission; that is, a message should be transmitted successfully within a bounded network delay. Unsuccessfully transmitted or large network delay in messages from a sensor to an actuator may deteriorate system performance or make system unstable (Godoy et al., 2010b).

Currently two main research directions can be distinguished in the NCS area. One focused in the development and use of specific tools to simulate the operation and to ease the analysis of the network influence in the performance and stability of the NCS (Cervin et al., 2003, Torngren et al, 2006; Al-Hammouri et al., 2008). And other in the development of control and design methodologies to handle the network effects improving the performance and guaranteeing the stability of the NCS (Zhang et al., 2001; Tipsuwan and Chow, 2003; Hespanha et al., 2007). An important stage in the development of control methodologies for NCS, which is sometimes neglected, is its robustness assessment. The robustness is the ability of a control strategy to maintain good and acceptable performance for the NCS even in the presence of significant disturbances or NCS parameter variations.

In the present paper, the robustness of an adaptive control methodology developed for CAN-based NCS is evaluated. The performance of the adaptive control methodology, which automatically selects the NCS sampling time in accordance with the NCS output, is analyzed for different NCS configurations and conditions of CAN network load. The results, obtained through experiments on a CAN-based NCS research platform, lead to the conclusion that the developed control methodology is reliable for application to NCS based on the CAN protocol, providing an acceptable performance for the NCS even in the worst-case scenario analyzed.

2. CONTROL METHODOLOGIES FOR NCS: SAMPLING TIME X EXECUTION PERIOD

The development of NCS presents new challenges for the application of traditional design and control techniques. For NCS, the control design shall considerate, simultaneously, several inherent factors to NCS such as the effects of sampling time, time variant network delays and loss of information on the network. Therefore, the development and application of new control methodologies is required for NCS (Baillieul and Antsaklis, 2007). These control methodologies have been developed based on different types of networks and communication protocols in conjunction with different approaches to handle the degenerative effects related to NCS. According to Tipsuwan and Chow (2003), a control methodology is required to minimize the effects of these factors in NCS while maintaining its performance and guaranteeing its stability.

The most common approach for the design and implementation of NCS consist in the periodic execution of a control algorithm, by defining a fixed execution period (time-driven) for the controller or by defining a fixed sampling time for the sensor messages (event-driven) (Gupta and Chow, 2010). Both techniques use in a static manner the available bandwidth for data communication without considering other factors such as the momentary network load and variations on the controlled systems. However, results of recent papers have proved that using dynamic definition techniques of this execution period may result in a better control performance for the NCS when compared to using consolidated techniques for static definition or fixed execution period (Cervin et al., 2010). Two current trends can be identified for control methodologies that use dynamic definition of this execution period (Camacho et al., 2010).

The first tendency uses techniques to change the controller execution period in accordance with the dynamics of data measured from the NCS such as the output and the network load. The main goal of these approaches is to improve the NCS control performance using efficiently all available network bandwidth for data transmission. Martí et al. (2010) present a technique that can improve the performance of a set of CAN-based NCS by additionally transmitting non periodic control messages, proportionally to the CAN network bandwidth available in that moment.

The second tendency uses event-based techniques for the controller triggering, which originates non periodic executions of the controller and therefore non periodic messages transmission. The goal of these methods is to decrease the network load at the same time ensuring control performance and stability requirements. Camacho et al. (2010) present an experimental investigation about the development and implementation of self-triggered controllers for NCS. In this paper, in addition to standard tasks performed every execution cycle, the controller also uses an algorithm to calculate when the next execution cycle will be triggered. The results of the paper show similar control performance for the NCS with self-triggered controller in comparison to a periodic controller, yet obtaining a small decrease in the CAN network load.

In order to handle the effect of the message sampling time in CAN-based NCS and using the ideas of the cited techniques for dynamic definition of the controller execution, Godoy et al. (2010c) have developed a new adaptive control methodology for CAN-based NCS. This control methodology developed shares the same goal of the event triggering technique, that focus on decreasing the network load at the same time ensuring the required control performance. Nevertheless, the adaptive control methodology innovates and differs from the cited since it uses the concepts of another technique, which dynamically defines the sampling time in accordance with NCS data. This

concept is used to achieve a greater reduction in the CAN network load which represents advantage of the developed adaptive control methodology against the others.

The control methodology (Godoy et al. 2010c) is mainly based on an adaptive discrete time PID controller that automatically manages the NCS sampling time according to its output data. With the dynamic definition of the message sampling time transmitted by the sensor, a dynamic change on the controller execution period is obtained at the same way. Additionally, a dynamic change in the transmission of control message to the NCS is also obtained. This dynamic definition in the transmission of messages in the NCS therefore provides a reduction in the CAN network load and minimizes possible degrading effects while ensuring the control performance and stability required. In this paper this adaptive control methodology developed for CAN-based NCS is evaluated to figure out its reliability, efficiency and robustness of application even under worst case conditions of operation.

3. ADAPTIVE CONTROL METHODOLOGY FOR NCS APPLICATION

One of the main problems in CAN-based NCS is related to the messages sampling times (Godoy et al. 2010b). This is due not only because the great impact of this parameter on the performance of the NCS, as also by this parameter being quite related to the CAN network utilization. If the NCS sampling time is very fast, there may be saturation of the communication network. In this situation, where the CAN network presents high utilization rate and becomes overloaded, large network delays are induced and errors in transmitting messages become constant, degrading the performance and may make the NCS unstable.

Therefore, it is important to deal correctly with the sampling time when developing CAN-based NCS. To handle this issue, Godoy et al. (2010c) have developed a new control methodology which is able to adapt the NCS sampling time in accordance with the NCS output and a few user defined parameters. As described the desired objective of this control methodology, that uses a PID controller, is to reduce the network utilization rate while maintaining the NCS performance and stability.

3.1. Discrete Time PID Controller as a Basis

According to Eriksson (2008), controllers for NCS cannot be designed with continuous time control theory because the resulting performance is unsatisfactory. Thus, the PID controller designed to be used in the control methodology for CAN-based NCS was a discrete-time PID controller derived with the backward derivative approximation with setpoint weighting, reference off, filtering on derivative part and Anti-Windup of the integrative part (Godoy et al. 2010b). Figure 3 shows the block diagram of the developed controller with filtering constant of the derivative part (N) and weighting constants (B and C).

The controllers for NCS have to handle the network delay effects in the systems. Based on the flexibility and large application, a PID controller was defined to be designed for the systems in the NCS platform. The controller is a discrete-time PID controller derived with the backward derivative approximation and with setpoint weighting, reference off and filtering on derivative part. This controller has been applied for NCS with good results (Eriksson, 2008). In this work, an Anti-Windup of the integral action was added to this controller to work with saturation of the actuators where the parameter T_t is equal to $\sqrt{T_i.T_d}$ for PID controllers and equal to T_i for PI controllers. Thus, the resulting discrete-time controller with constant sampling time (h) and constant filtering of the derivative part (N) is showed in the Fig. 2:
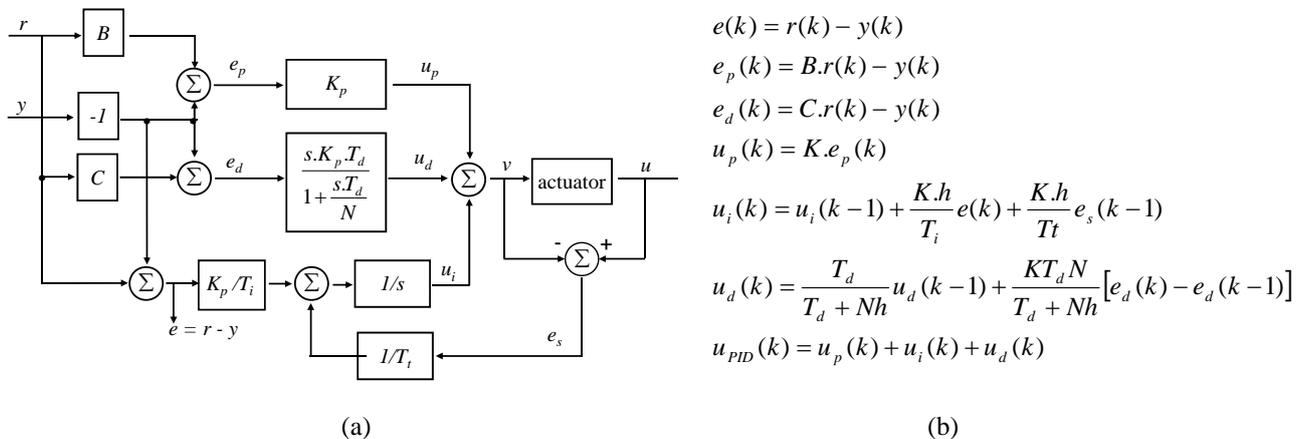

$$e(k) = r(k) - y(k)$$

$$e_p(k) = B.r(k) - y(k)$$

$$e_d(k) = C.r(k) - y(k)$$

$$u_p(k) = K.e_p(k)$$

$$u_i(k) = u_i(k-1) + \frac{K.h}{T_i}e(k) + \frac{K.h}{Tt}e_s(k-1)$$

$$u_d(k) = \frac{T_d}{T_d + Nh}u_d(k-1) + \frac{KT_dN}{T_d + Nh}[e_d(k) - e_d(k-1)]$$

$$u_{PID}(k) = u_p(k) + u_i(k) + u_d(k)$$

(a)   (b)

Figure 2. Discrete-Time PID Controller for NCS: (a) Block Diagram (b) Routine for Parameters Implementation

3.2. Adaptive Control Methodology

The principle of operation of the adaptive control methodology (ACM) applied in this paper is shown in Fig. 3.
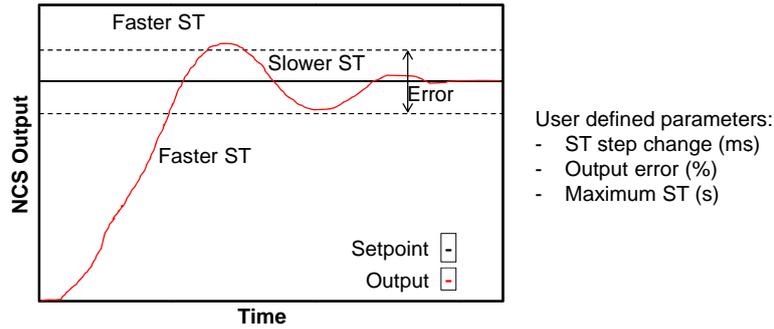
Figure 3. Application Structure for the Adaptive Control Methodology for CAN-Based NCS

It is important to understand that the control methodology starts to operate automatically with a change at the NCS setpoint. According to the Fig. 3, the user needs to set three parameters for the application of the control methodology. The maximum sampling time (ST) which can be used in the NCS, the ST step change (ms) during the NCS operation, which defines the value for the increase or decrease of the ST at each controller period and the output error (%), which determines the area related to the final output value required for the NCS that will be used by the control methodology. During the NCS operation, the ST initial value is faster in order to achieve an initial good output and performance. If the NCS output is within the area designated by the output error, the controller automatically reduce the NCS sampling time (making it slower) while maintaining the performance and guaranteeing the stability of the NCS. Consequently, the CAN-based network utilization will decrease too. If the NCS output is outside the designated area, the controller perform the contrary action, increasing the NCS sampling time (making it faster).

4. NETWORKED CONTROL SYSTEM (NCS) RESEARCH PLATFORM

The platform used in this paper has been developed by the Department of Mechanical Engineering of the University of São Paulo at São Carlos for NCS research and experimentation (Godoy et al. 2010a). The communication network used for the control systems integration and information exchange is the CAN protocol. The architecture of the NCS platform is shown in Fig. 4. Common control systems used in the industrial area such as DC motor velocity and position control, temperature control, tank level control and belt conveyor control are selected for the platform.

Each of the defined control systems has a microcontroller based electronic control unit (ECU), which is responsible for the sensor data acquisition, actuation in the plant and communication with the CAN-based network. Two desktops with LabVIEW and PCI-CAN interfaces from National Instruments are used for the development of the control methodologies. The architecture proposed has high flexibility for the research and teaching of NCS. In addition to the aimed tasks for NCS such as analysis, modeling, simulation and control, the platform provides capabilities to study and applies advanced techniques for NCS development. Among these techniques are the real-time evaluation, hardware-in-the-loop simulation and rapid control prototyping.
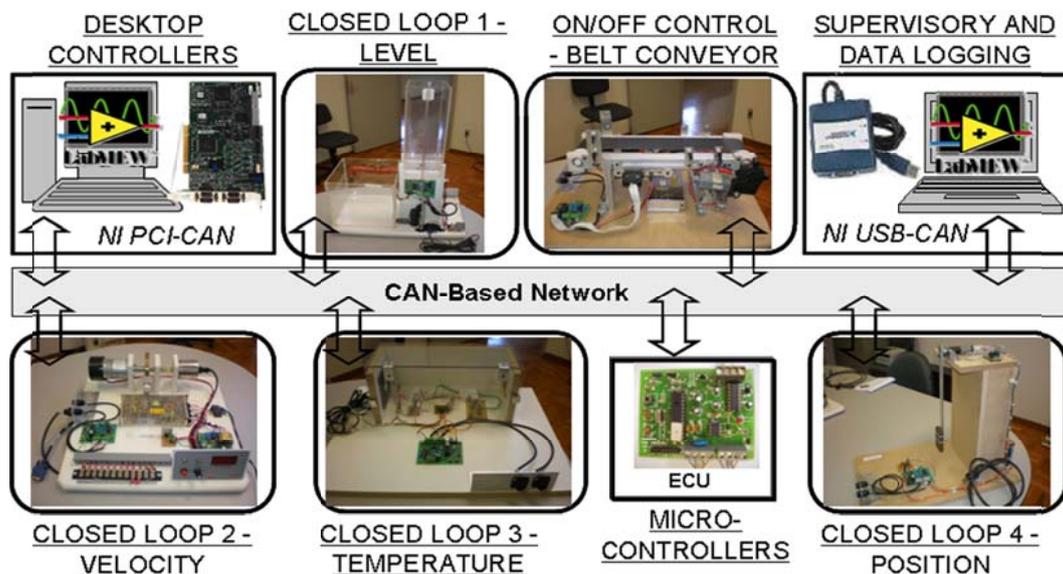

Figure 4. CAN-Based NCS Research Platform

For each of the NCS that composes the platform, the time-driven sensor node samples the plant or process periodically and sends the samples to the controller node over the CAN network. Upon receiving a sample, the controller computes a control signal which is sent to the actuator node, where it is subsequently actuated. The threads executing in the controller and actuator nodes are both event-driven. All the closed loop control systems in the platform are sharing both limited CAN network bandwidth and controller CPU. The competition for these constrained resources will certainly increase the network delays of the control loops and degrade the overall performance of the NCS in the platform.

The choice of the CAN protocol (Johansson et al. 2005) for the communication network is due to its low cost of development and large acceptance in the industrial and academic areas. The CAN was developed to interconnect electronic control units (ECU) in automotive area, but recently it has also been applied in many other networked applications. Currently, CAN-based networks are applied as a solution for distributed systems in several areas, such as robotic, automated manufacturing and process control, and they are also used in proprietary architectures such as Device Net and CAN Open. As described in Johansson et al. (2005), in CAN-based networks data are transmitted and received using messages that carry data from a transmitting node to one or more receiving nodes. An identifier, unique throughout the network, labels each message and its value defines the priority of the message to access the network. The CAN protocol is optimized for short messages and uses a CSMA/CD with NDBA (Carrier Sense Multiple Access / Collision Detection with Non-Destructive Bitwise Arbitration) arbitration access method. The bit stream of a transmission is synchronized on the start bit, and the arbitration is performed on the following message identifier, in which a logic zero is dominant over a logic one.

5. CONTROL METHODOLOGY APPLICATION AND ROBUSTNESS EVALUATION

To evaluate the applicability, versatility and robustness of the adaptive control methodology and also demonstrate its benefits of application, several experiments have been conducted for different configurations of NCS using the NCS research platform. For all tests performed, the five NCS of the platform were kept in operation, in order to achieve a situation with significant message traffic (periodic and non-recurrent) on the CAN network, consistent with the reality of networked industrial applications. In addition, this configuration assures a bigger utilization of the CAN network and enhances the results analysis. An initial message sampling time (ST) of 10ms was configured for the sensors of four NCS, while the NCS for belt conveyor has an event based control (presence of pieces).

The parameters defined for the application of the ACM and for the discrete time PID controller used in each NCS of the research platform are presented in Table 1. Additionally, the following parameters were used: CAN network speed of 250kbit/s, a value of N=10 for the constant filtering, C=0 and B=1 for the weighting constants.

Table 1. Parameters Used in the Evaluation Experiments of the Adaptive Control Methodology

NCS	Velocity	Position	Level	Temperature
Parameters of the Adaptive Control Methodology				
Maximum ST (s)	0,2	0,2	0,4	0,8
ST Step Change (ms)	5	5	5	10
Output Error (%)	10	5	5	10
Parameters of the Discrete Time PID Controllers				
Proportional Gain - K	0.07	3	10	15
Integrative Time – Ti	0,65	-	0,55	7,5
Derivative Time - Td	-	0,001	0	0,1

With the application of the ACM simultaneously for more than one NCS, the graph of the variation on the CAN network load will show the changes during the experiment time. So, each listed change will be related to the moment in which the ACM started to operate on a given NCS, automatically managing its sampling time (ST). The first experiment was conducted with the application of the ACM, at the same time, for four NCS of the platform. The compiled results presented jointly in Fig. 5 demonstrate the efficiency of the ACM by a significant reduction in the CAN network load.

Figure 5 presents for the four NCS of the platform, a graph showing the NCS control performance to a step response (top graphs) and a graph showing the change of the NCS sampling time (intermediate graphics) during the operation of the ACM. In addition, a graph showing the variation in the CAN network load during the experiment (bottom graph) is also presented.
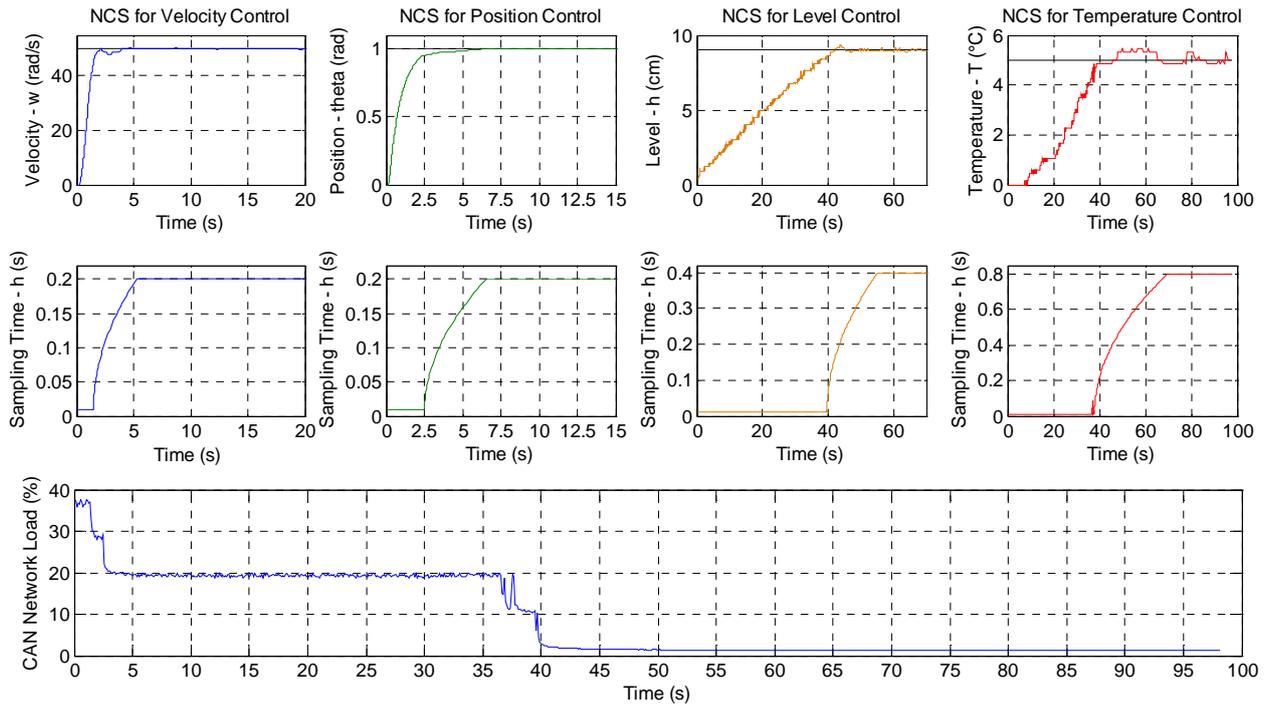
Figure 5. Application of the Adaptive Control Methodology (ACM) for all the NCS of the Platform: NCS Step Responses, Sampling Times (ST) Variation and CAN Network Load Variation

As can be seen in the results of the NCS for velocity control (blue graphs), the NCS sampling time was automatically increased until 200ms since the ACM started operating. With this increase in the NCS sampling time, the CAN network load decreased to approximately 10%, while the NCS performance was kept constant, as shown in the NCS step response graph in Fig. 5. Summing this same reduction effect for the others NCS, the final result is even more significant, as shown in the CAN network load variation graph. In this case the reduction in the CAN network load is from 37% to 2%, without sacrificing any NCS control performance or reliability. The results obtained with this experiment prove the effectiveness of the control methodology developed for application in CAN-based NCS.

Figure 6 presents the comparison among the NCS control performance for different sampling times including fixed values and the adaptive control methodology (ACM) developed. The objective of this analysis is to check the difference among the fixed and variable approaches for the NCS sampling time. And also to use the results to verify if there is advantage in the application ACM implemented.
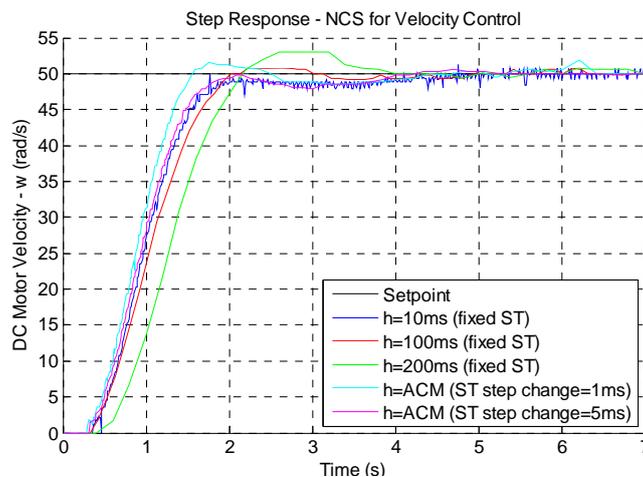

Figure 6. Comparison among Step Responses of the NCS for Velocity Control using Constant Sampling Times and with the Adaptive Control Methodology (Variable Sampling Time)

Analyzing the curves of Fig. 6, it is possible to verify that there is no disadvantage or performance degradation in using the adaptive control methodology (ACM) instead of the constant sampling times. On the contrary, a slight

improvement in the NCS time response (rise time) can be noticed, mainly in the case of the ACM with ST step change of 1ms. The comparison results for the others NCS provided the same conclusion and have been omitted for simplification. The obtained results prove the efficiency of the ACM for application in CAN-based NCS and also demonstrate its great versatility of application for different types of NCS.

The application of the adaptive control methodology (ACM) is important by the fact that usually in NCS, the network characteristics, such as network delays, are time variant (for example depending on message traffic, number of devices connected). Because of that, an NCS operating with high (or slow) constant sampling times can at some point loses its control requirements and become more oscillatory. On the other hand, if the NCS sampling time is too small (or fast), the CAN network presents high utilization rate and becomes overloaded, inducing larger network delays (waiting time for message contention) and even preventing the connection of more devices on the network.

To demonstrate the importance of the ACM for CAN-based NCS and also to evaluate its robustness of application, an experiment was carried out comparing the results for the NCS of the platform to four different configurations and CAN network conditions. Table 2 resumes the parameters and conditions of operation for the experiments. The experiment 3 is the same experiment which the results are presented in Fig. 5.

Table 2. Configuration of the Experiments for ACM Importance and Robustness Evaluation

Experiment	NCS Sampling Time Approach	CAN Network Condition
1	Fixed h = 10ms	Without extra traffic
2	Fixed h = 10ms	With extra traffic = 55% of the CAN load
3	Using ACM	Without extra traffic
4	Using ACM	With extra traffic = 55% of the CAN load

In accordance with the Table 2, for each experiment conducted two main configurations were used. The NCS sampling time that defines the approach used for the selection of the ST for the NCS of the platform. And the CAN network condition that defines the insertion of an extra message traffic on the CAN network equal to approximately 55% of its load. This extra traffic was obtained with the connection in the CAN network of a computer configured to transmit repeatedly extra messages increasing the CAN network load. The objective of this extra traffic is to evaluate the NCS operating in severe operation conditions of the CAN network. The experiment 1 have been performed with a fixed ST of h=10ms for all NCS and without extra traffic on the CAN network. In this situation the CAN network load measured was equal to 37% during the experiment. In the experiment 2, the same fixed ST of h=10ms was used but with the extra traffic in the CAN network. In this situation, the CAN network load during the experiment was equal to approximately 92%, configuring a severe condition of operation for the NCS of the platform. The experiments 3 and 4 repeated the experiments 1 and 2 with the application of the ACM. The idea is to compare the results to evaluate the benefits of the application of using the ACM developed. It is important to show that the experiment 4 represents the worst case condition for application of the ACM. In this experiment the CAN network load is initially 92% and all the four NCS of the platforms use the ACM, originating a great change in the transmission of messages.

The graphs of the Fig. 7 and Fig. 8 present the comparison among the four experiments of the Table 2 and simplify the discussion and the evaluation of the results. In these graphs, the legends "with" or "without network load" refers to the experiments conducted with the extra message traffic using 55% of the CAN network load.
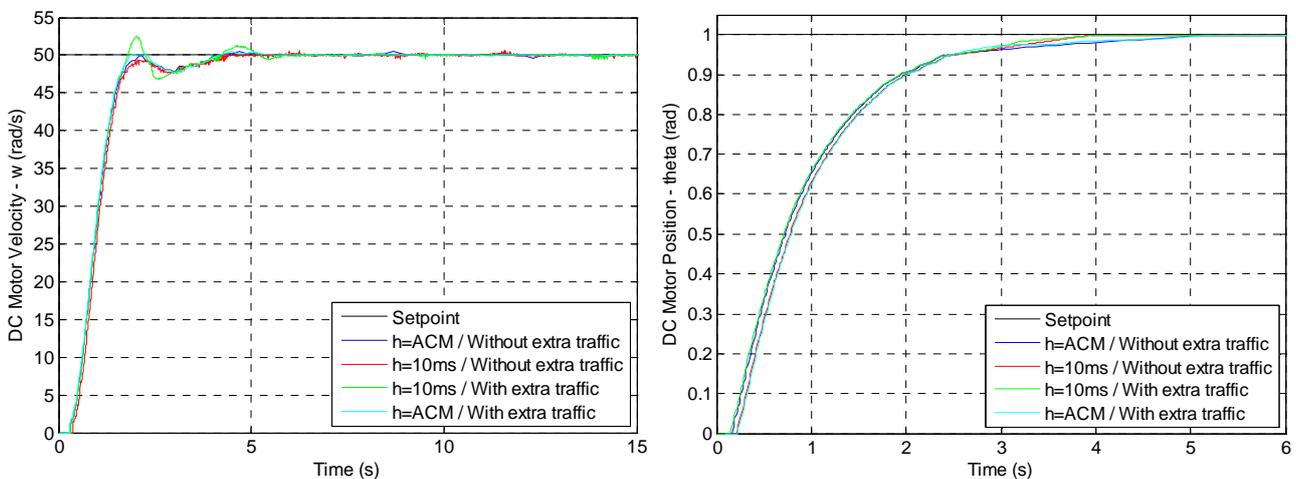

Figure 7. Comparison among Step Responses using Constant Sampling Times and the Adaptive Control Methodology (Variable Sampling Time): (a) NCS for Velocity Control, (b) of the NCS for Position Control

Analyzing the results separately, the experiment 1 (h=10ms / without extra network load) demonstrated that a higher and fixed network load (37%), and consequently further delays, can degrade the control performance of the NCS. It is proved on the graph of the Fig. 8(a) that shows the NCS of level control becoming more oscillatory and tending to instability after the time t = 80s. The experiment 2 (h=10ms / with extra network load), carried out with the NCS operating with a severe condition of 92% of the CAN network load, demonstrated that the degradation in the NCS control performance is more pronounced according to the occupation of the CAN network. This can be explained by the great network delays, induced mainly by the messages contention and blocking time to access the CAN network. Observing the graphs of the Fig. 7(a) and Fig. 8(b), there is deterioration in the control performance of the NCS for velocity and temperature control, which had not been affected in the previous experiment 1. In the same way as in the experiment 1, the NCS for level control becomes more oscillatory and tends to instability.
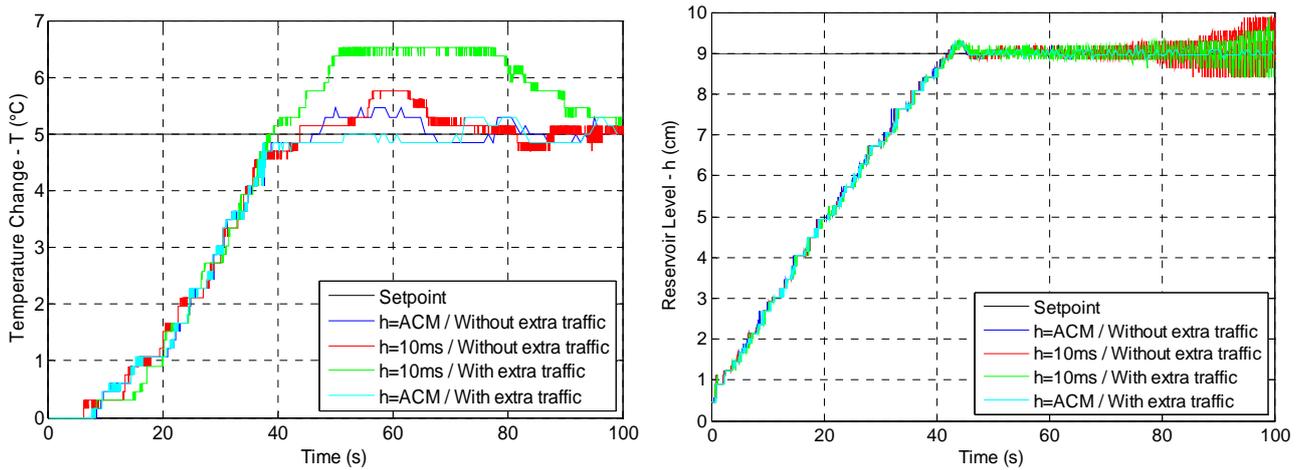

Figure 8. Comparison among Step Responses using Constant Sampling Times and the Adaptive Control Methodology (Variable Sampling Time): (a) NCS for Temperature Control, (b) NCS for Level Control

The results of the experiments 3 (h=ACM / without extra network load) and 4 (h=ACM / with extra network load) show the importance of the adaptive control methodology (ACM) for CAN-based NCS. The application of the ACM for the NCS of the platform provided an automatic way to dynamically select its sampling time (ST) according to the NCS output and to overcome the influence of the ST in the performance of the NCS. In addition to this change on the NCS sampling times and consequently with the reduction in the CAN network load and network delays, the ACM has also provide a way to guarantee the control performances for the NCS.

It is important to show that the experiment 4 represents the worst case condition for the application of the ACM. Observing the graphs of the Fig. 7 and 8 related to the experiment 4, the application of ACM provided acceptable control performance for the NCS even with the initial occupation of 92% of the CAN network load. These results demonstrate the robustness of the ACM even for the worst case condition of application. The graph of the Fig. 9 presents the variation of the CAN network load during the experiment 4. The automatic selection of the NCS sampling times originates a reduction of the CAN network load from 92% to 57% (representing the 55% related to the extra message traffic plus 2% related to the control traffic of the NCS of the platform). For the experiment 3, which the compiled results are presented in the Fig. 5, the reduction on the CAN network load was from 37% to 2%.
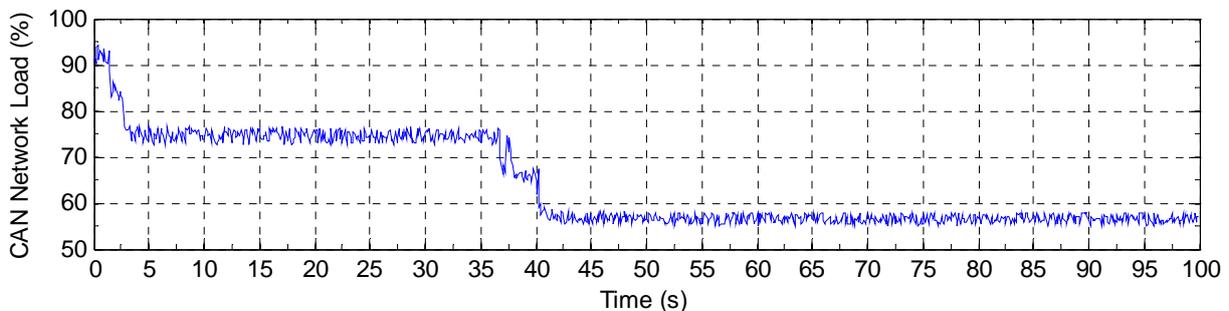

Figure 9. Variation in the CAN Network Load during the Experiment 4: Worst Case Condition for ACM Application

Figure 7(b) presents the results of the experiments for the NCS for position control. It is important to note that these results are different from the obtained to the others NCS. This NCS has not presented significant performance

degradation related to the increase in the CAN network load. It is believed that it happened because the CAN messages of this NCS have the highest priorities to access the CAN network of the platform, which consequently means smaller network delays. Another interesting fact concerns the application of the adaptive control methodology (ACM) for this NCS. Observing the graph of Fig. 7(b) in a close manner, it can be seen that the application of the ACM caused an increase of the NCS rise time.

In a nutshell, it could be verified by the experimental results that the use of fixed sampling times with the increase of the occupation of the CAN network degrades the control performance of NCS, making it more oscillatory. This conclusion is shown in the graphs of the Fig. 7(a), Fig. 8(a) and (b). However, these problems can be overcome with the application of the adaptive control methodology (ACM) developed in this paper. The results obtained from the experiments prove the reliability and the benefits of the application of the ACM. In addition, the results evidence the versatility and robustness of the ACM for application in CAN-based NCS. With the use of the ACM, a significant reduction in the CAN network load was obtained during the operation of the NCS in the platform, minimizing potential problems related to network delays and information lost on the network which can affect the performance and stability of the NCS.

Another advantage of the ACM is its full compatibility with industrial systems already deployed. New NCS can be developed using the ACM on systems which use NCS with fixed sampling times or even on distributed systems with industrial networks that do not have solutions based on the NCS technology.

6. CONCLUSIONS

An important stage in the development of control methodologies for NCS is the robustness analysis. The robustness is the ability of a controller to maintain good performance for the NCS even in the presence of significant disturbances or NCS parameter variations. This type of analysis is essential for NCS due to the impact on its performance and stability of several different parameters such as network delays, packet losses and message sampling time. For CAN-based NCS, another important parameter is the network load (the used percentage of the total capacity to transmit messages on the network). In this paper, the robustness of an adaptive control methodology (ACM) has been evaluated against different NCS configurations and network loads. The ACM automatically changes the NCS sampling time in accordance with the NCS output, reducing the CAN network load and maintaining the NCS performance and stability.

This evaluation was based on a series of experiments conducted in a CAN-based NCS research platform. The ACM versatility was verified through the performance analysis of different types of NCS using the ACM. The ACM provided the ability to be applied to more than one NCS simultaneously and also is fully compatible with industrial systems that do not have solutions based on the NCS technology. The results could also demonstrate that the performance and stability of CAN-based NCS can be degraded by the increase in the CAN network load, which induces greater network delays and causes packets losses.

Additionally, the robustness of the ACM has been proved by experiments conducted under extreme conditions of operation including the worst case scenario in which all NCS are using the ACM at the same time with the maximum CAN network load (92%). In all scenarios investigated, the ACM has been efficient to guarantee the NCS performance and stability when compared to the results of the NCS using fixed sampling times. Therefore, the results lead to the conclusion that the ACM is reliable for application to NCS based on the CAN protocol, providing a solution to overcome two verified problems in CAN-based NCS that are the message sampling time and network occupancy impact on the performance and stability of this system.

7. ACKNOWLEDGEMENTS

8. REFERENCES

Al-Hammouri, A., Branicky, M.S and Liberatore, V., 2008, "Co-simulation Tools for Networked Control Systems", Springer-Verlag, pp. 16-29, Berlin.
Baillieul, J. and Antsaklis, P.J., 2007, "Control and Communication Challenges in Networked Real Time Systems", Proceedings of IEEE Technology of Networked Control Systems, Vol. 95, No. 1, pp. 09-28.
Camacho, A., Martí, P., Velasco, M., Lozoya, C., Villa, R., Fuertes, J.M. and Griful, E., 2010, "Self-triggered networked control systems: An experimental case study", Proceedings of the 2010 IEEE International Conference on Industrial Technology (ICIT), Vina del Mar, Spain, March 14-17, p.123-128.
Cervin, A, Henriksson, Lincoln, Eker, J. and Arzen, K., 2003, "How does control timing affect performance? Analysis and simulation of timing using Jitterbug and TrueTime", IEEE Control System Magazine, Vol. 23, pp. 16-30.
Cervin, A., Velasco, M., Martí, P. and Camacho, A., 2010, "Optimal Online Sampling Period Assignment: Theory and Experiments", IEEE Transactions on Control Systems Technology, Vol. 99, pp.1-9, July.
Eriksson, L., 2008, "PID Controller Design and Tuning in Networked Control Systems", PhD Thesis, Dept. of Aut. and Systems Tech., Helsinki Univ. of Tech., Finland, 118p.

Godoy, E.P., Pereira, R.R.D., Scorzoni, F., Porto, A.J.V. and Inamasu, R.Y., 2010a, "CAN-based Platform for the Study and Experimentation on Networked Control Systems (NCS)", Proceedings of the 5th IFAC Symposium on Mechatronics Systems, Boston, USA.

Godoy, E.P., Porto, A.J.V. and Inamasu, R.Y., 2010b, "Applied Simulation to Evaluate the Quality of Control of Networked Control Systems", Proceedings of the 2010 IEEE International Conference on networking, Sensing and Control (ICNSC), Chicago, IL, USA, April 11-13.

Godoy, E.P., Porto, A.J.V. and Inamasu, R.Y., 2010c, "Sampling Time Adaptive Control Methodology for CAN-Based Networked Control Systems", Proceedings of the 2010 IEEE/IAS International Conference on Industry Applications (INDUSCON), 9, São Paulo, SP, Brazil, November 8 - 10.

Gupta, R.A. and Chow, M.Y., 2010, "Networked Control System: Overview and Research Trends", IEEE Transactions on Industrial Electronics, Vol. 57, No. 7, pp. 2527-2535, July.

Hespanha, J.P., Naghshtabrizi, P. and XU, Y., 2007, "A Survey of Recent Results in Networked Control Systems", IEEE Proceedings of the Technology of Networked Control Systems, Vol. 95, No. 1, pp. 138-162.

Johansson, K.H., Torngren, M. and Nielsen, L., 2005, "Vehicle applications of controller area network", Handbook of Networked and Embedded Control Systems, Ed. Birkhäuser, 25p.

Lian, F.L., Moyne, J.R. and Tilbury, D.M., 2002, "Network Design Consideration for Distributed Control Systems", IEEE Transactions on Control Systems Technology, Vol. 10(2), pp. 297-307.

Martí, P, Camacho, A., Velasco, M. and Gaid, M.E.M.B., 2010, "Runtime Allocation of Optional Control Jobs to a Set of CAN-Based Networked Control Systems", IEEE Transactions on Industrial Informatics, Vol. 6, No. 4, pp. 503-520, November.

Moyne, J.R. and Tilbury, D.M., 2007, "The Emergence of Industrial Control Networks for Manufacturing Control, Diagnostics, and Safety Data". IEEE Technology of Networked Control Systems, Vol. 95, No. 1, pp. 29-47.

Tipsuwan, Y. and Chow, M.Y., 2003, "Control Methodologies in Networked Control Systems", Control Engineering Practice, Vol. 11, No. 3, pp. 1099-1111.

Torngren, M., Henriksson, D., Arzen, K.E., Cervin, A. and Hanzalek, Z., 2006, "Tool supporting the co-design of control systems and their real-time implementation: current status and future directions", Proceedings of the 2006 IEEE International Symposium on Intelligent Control, Munich, Germany, 4-6 October, pp. 1173-1180.

Yang, T.C., 2006, "Networked control system: a brief survey", IEEE Proceedings of Control Theory and Applications", Vol. 153, No. 4, July, pp. 403–412.

Zhang, W., Branicky, M.S. and Phillips, S.M., 2001, "Stability of Networked Control Systems", IEEE Control System Magazine, Vol. 21(1), pp. 84–99.

9. RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.