Evaluation of Industrial Networked Control Systems Using Hardware-In-The-Loop Simulation

Thomas Mellios, Christina Kansoulidou and George Hassapis*
Aristotle University of Thessaloniki
Thessaloniki 541 24, Greece
mellios@auth.gr, kans@auth.gr and chasapis@eng.auth.gr

Abstract – The performance of advanced control algorithms implemented on networked control devices is highly dependent on the network delay. As most algorithms are not designed by taking this implementation parameter into consideration there is a need to evaluate the performance of these algorithms before commissioning them. In this work the hardware-in-the-loop simulation is proposed as a possible evaluation technique. The hardware-in-the-loop simulator is defined as the facility that consists of a computer-based simulation of the plant linked to the communication network and the actual control devices on which the algorithm will be implemented. In order to demonstrate the way of using such a facility for evaluating the performance of a control algorithm, an experimental hardware-in-the-loop simulator based on the Profibus-DP industrial network standard has been realized and the study of the LQC control of a cement milling circuit was undertaken.

Index Terms – Industrial control, Profibus, networked control.

I. INTRODUCTION

The control systems that have their control loops closed via a serial and common communication channel present certain advantages over the traditional point-to-point architectures. These advantages are the small volume of wiring, the distribution of the processing functions to many units, the low cost and the quick and easy maintenance. However, the time varying delay that is introduced during the transmission of sensor and actuator data may influence the performance of the control algorithms that are designed with conventional control theories. In these designs the assumption is made that measurements of the controlled variables are taken at sampling rates that are either steady or varying within certain limits. Because of the network-induced delays, these limits might not be always secured. This problem may cause deterioration in the response of the system controlled variables and drive the system to an unstable state.

Hence, the need arises to check the performance of control algorithms that have been designed with conventional control theories when they are implemented on networked control systems. This performance is greatly influenced by the type of the network and its traffic. So, there are networks with communication protocols that minimize the likelihood of having long packet delay and other that have predictable delay limits. As the networks of the second type seem to dominate the industrial sector it would be wiser to invent ways of assessing the performance of control algorithms that will be implemented on such networked control systems as reliably as possible. The manufacturers of Distributed and PLC-based control systems provide simulators that either simulate the functions of their controllers and the way they interact with the process dynamics [10], [11], [12]. or provide software implementations of their controllers on separate computers which can be linked with computer simulators of the controlled processes [13], [14]. As far as fieldbus networks are concerned, there seems to be a few systems that are able to simulate such networks. Delta V of Emerson [11] is one such system and the Simpow Simcomx ITS of ABB [10] is another one. However, these systems simulate the fieldbus functions on a computer and do not consider the use of the actual equipment of a real-life network. Believing that next to the implementation of the algorithm in the real field conditions is the hardware-in-the-loop simulation of the algorithm, we propose in this work a facility for assessing the performance of the control algorithms which involves the actual controllers and the network that will be used in the industrial field and a computer simulation of the controlled process. This facility is enhanced with a generator of network traffic, so that the controller induced traffic added to the generated traffic would allow studying the behavior of the controller under varying traffic loads. The traffic load imposed by the generator simulates the load inflicted on the network by other control applications that the same network is used for.

The proposed test facility consists of at least four units. The first unit is a PC-based computer station which hosts a process simulation package such as MATLAB [4] and is interfaced and linked to an industrial network, such as DP Profibus [1], [6]. It also has the necessary software that allows the transfer of data between the controllers and the process simulation package. The second unit is a Programmable Logic Controller (PLC) which simulates the traffic load of the control applications. The third unit is another PLC that absorbs the load and the fourth unit is also a PLC which is programmed to run the algorithm that controls the simulated plant. This facility is scalable and can be expanded to accommodate additional network stations that will implement the control of other variables of the simulated plant.

How this facility can be used to assess the performance of an algorithm on a networked control system is demonstrated by realizing the LQC networked control of two loops of a cement milling circuit. Two network parameters are adjusted and the performance of the considered LQC control algorithm was studied under specific network traffic. This demonstration shows the possible use of the facility to check whether a control algorithm can be implemented without driving the system to instability in an existing networked control system and tune the network to accommodate the insertion of the new algorithm.

II. THE DP PROFIBUS INDUSTRIAL NETWORK

Profibus is one of the available standards that have been approved by CENELEC [2] for networked control systems. The Profibus MAC protocol is a simplified version of the timed-token-bus protocol [3]. According to this protocol, the so called master stations of the network get the bus access when a data object called token is passed to them. During operation of the network the station with the token can either transmit data to the slaves that have been assigned to the specific master station or request the transmission of data from the slaves. The slaves cannot initiate any data transmission neither can receive and hold the token. The token is exchanged only between masters. Each master transmits data frames until it runs out of data frames or the time over which the token was held has reached a limit. Data between two masters are exchanged with the rules of the so called FMS or DP Profibus protocols. Master and slaves constitute the nodes of the network. The time delay of a message is defined as the difference between the time when the source node begins the process of sending a message and the time when the destination node completes reception of this message. This time delay must be less than the target rotation time, a network timing parameter that can be set by the user and expresses the time elapsed between two consecutive receptions of the token by a master station. A dominant part of the message time delay between two masters is the slot time which defines the maximum timing period that the sender of a package must wait for the receiver response. If the sender does not receive the response from the receiver within this timing period the sender repeats the transmission of the same package. By adjusting the token rotation time (ttr) and the slot time (tst) and keeping all the other timing parameters of the network at preset values is a first level tuning of the network is obtained. The purpose of the network tuning is to cope with the satisfaction of deadlines of tasks that have to be executed by the network stations.

III. THE HARDWARE-IN –THE-LOOP-SIMULATOR

The diagram in Fig. 1 shows pictorially the architecture of the developed facility for evaluating the performance of new DP-based controllers of a process which may be inserted to an existing Profibus network. Actually, it consists of four different nodes. Each node plays a specific role in the network and is loaded with software that is appropriate for its role. The first node is a PC and a master station in the network. It is used for simulating the process to be controlled as well as the sensors and the actuators of the controlled plant. It transmits the actuator values and receives the sensor measurements. The second node, being also a master station, is a Siemens S-700 PLC and generates network traffic similar to the one that is expected to be produced by the aggregated operation of all the other control functions except the ones that we are going to introduce. This traffic is exchanged with one of the slave stations which is a Siemens S-300 PLC. This traffic is produced by a generator which can be preset to send a number of packages, the data size of each package being determined statistically according to a Poisson distribution. When the first master holds the token it may establish a master-slave communication procedure with the other slave station of the network which is also a Siemens S-300 PLC. This slave PLC implements the algorithm that controls the variables of the simulated process. In the first station where the plant dynamics are simulated, the control system simulation environment of MATLAB [4] is installed. It communicates with the Siemens WINCC SCADA system [8] through the Excel program. The SCADA system sends and receives the output and input data to the plant model that is simulated in the MATLAB environment by incorporating services of the application layer of the DP protocol. In Figure 2 the architecture of the installed software is pictorially depicted.

III. THE NETWORKED CONTROL OF A CEMENT PLANT

Next, the way the developed hardware-in-the-loop-simulator has been used to evaluate the performance of the networked control of two critical parameters of a cement plant will be demonstrated. The considered control functions are implemented on an existing Profibus network which is supposed to be used also for the control of other variables of the same or other processes. First, a brief description of the control problem of the cement plant is presented. The basic process unit in the production of cement is the cement milling circuit. Such units are fed with raw material which after being ground is introduced into a high efficiency classifier and separated into tailings (refused part) and the finished product (accepted part). The tailings are fed back into the milling circuit. In the schematic diagram of Figure 3 the milling circuit principle is shown.

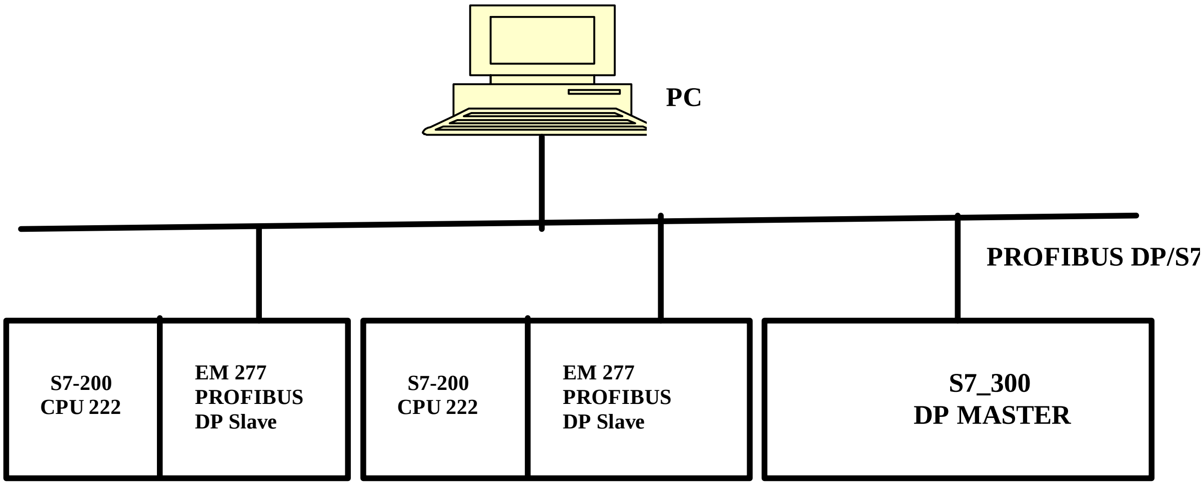

Fig.1: The architecture of the hardware-in-the-loop-simulator

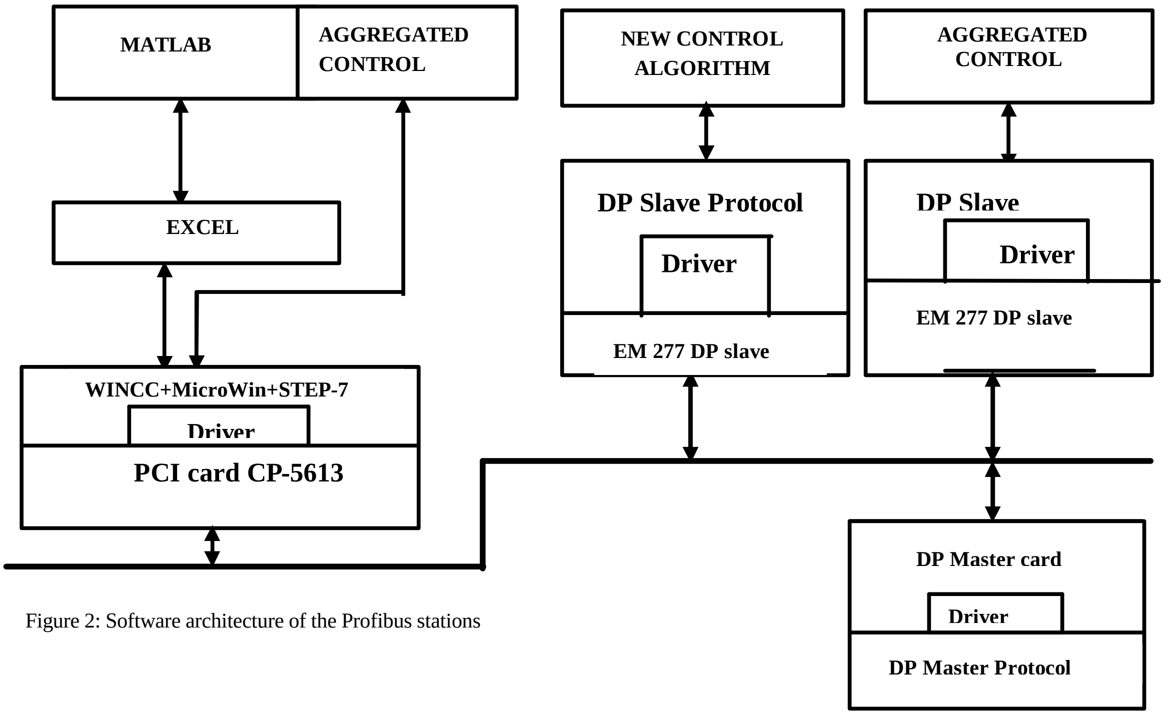

Figure 2: Software architecture of the Profibus stations

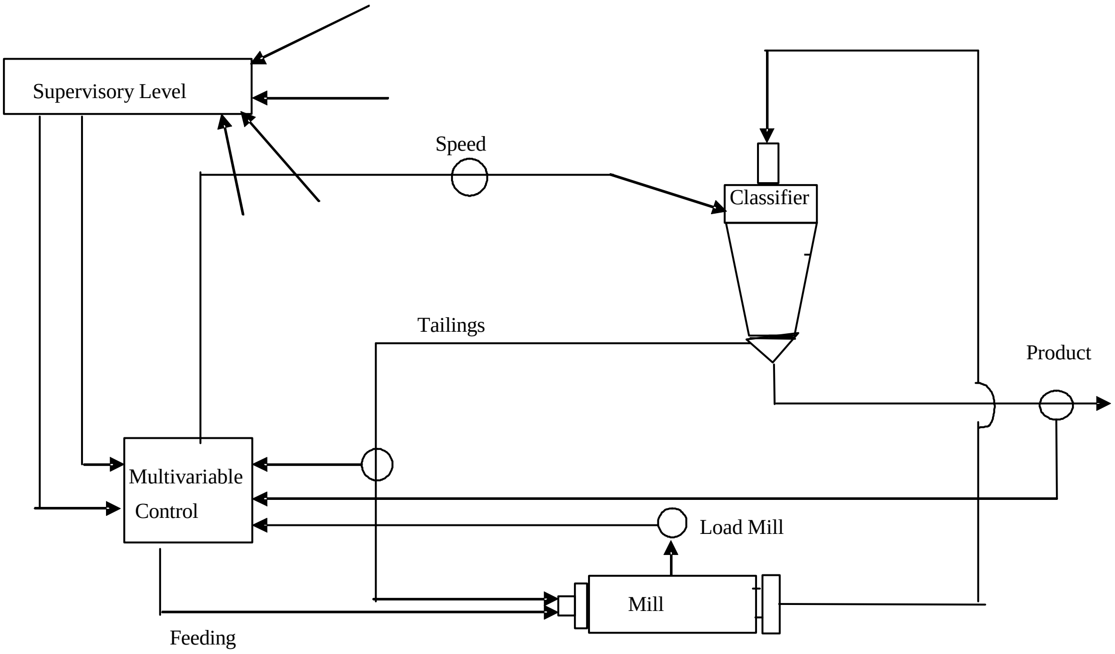

Figure 3: The milling circuit

The classification of the material is driven by the rotational speed and by the air flow rate of the classifier. The load in the milling circuit depends on the input feed (fresh feed plus tailings flow rate) and on the product flow rate that depends in a non-linear way on the hardness of the material. There is a different upper limit of the load for each hardness value which leads to circuit instability, while keeping the load at a low level will result in the fast wear of the mill internal equipment. Various control techniques have been introduced [5], [7] to deal with the instability problem. In one of these [7] which is based on the Linear Quadratic Control (LQC) two outputs, namely the product flow rate (y_f) and the load (z) are simultaneously controlled by using the two available inputs, that is the feed flow rate (u) and the classifier speed (v). For the implementation of the LQC algorithm on the networked control system one of the slave stations was programmed to perform this algorithm. In the first master station the plant dynamics without the LQC control were simulated in the MATLAB environment. During this simulation the solution of the differential equations is carried out at every sampling instant and then the data of the controlled variables y_f and z are sent to the slave station. The results of the algorithm execution based on the previous set of data that were transmitted to the slave are read. Then the token is passed to the other master station which generates data packages of sizes that are randomly selected according to the Poisson distribution. While the token is held by the second master the LQC computations are executed on the slave PLC by using the received data. Once the generated data are transmitted or the token holding time is elapsed the token is passed to the first master. This master polls the slave station which implements the control algorithm and reads the values of the manipulated variables u and v, computes the values of the controlled variables y_f and z and sends them back to the slave. There are certain known deadlines for sending and receiving data to and from the plant simulation, equal to the sampling rate of each algorithm. To meet these deadlines for a known network baud rate, the timing parameter of the maximum token rotation time (ttr) and the slot time (tsl) have been preset. If, however these figures are not correct or there are bursts of heavy traffic in the network, sensor measurements and controlled variables updates might not occur within the deadlines. This might drive the controlled variables either to deviations from the desired transient and steady state specifications.

Two experiments were conducted to show the effect of two different values of network timing parameters on the performance of the networked control of the cement plant under two different network loads. The baud rate in both experiments was 12Mbps. The results are presented in Fig. 4 and 5.. Curves (a) in Fig. 4 show the responses of the controlled variables for tsl=1000 tbit and ttr=46325 tbits for the load of 100 packages, curves (b) show the responses of the same variables for the same timing settings but for load of 1000 packages whereas curves (c) correspond to the responses when the network does not have any other load than the one of the cement control. The size of each package varied from 0-510 bytes, randomly defined by the Poisson distribution. In Fig. 5 the responses the same network loads are shown but the timing settings were tsl=100 tbit and ttr=24433 tbit.

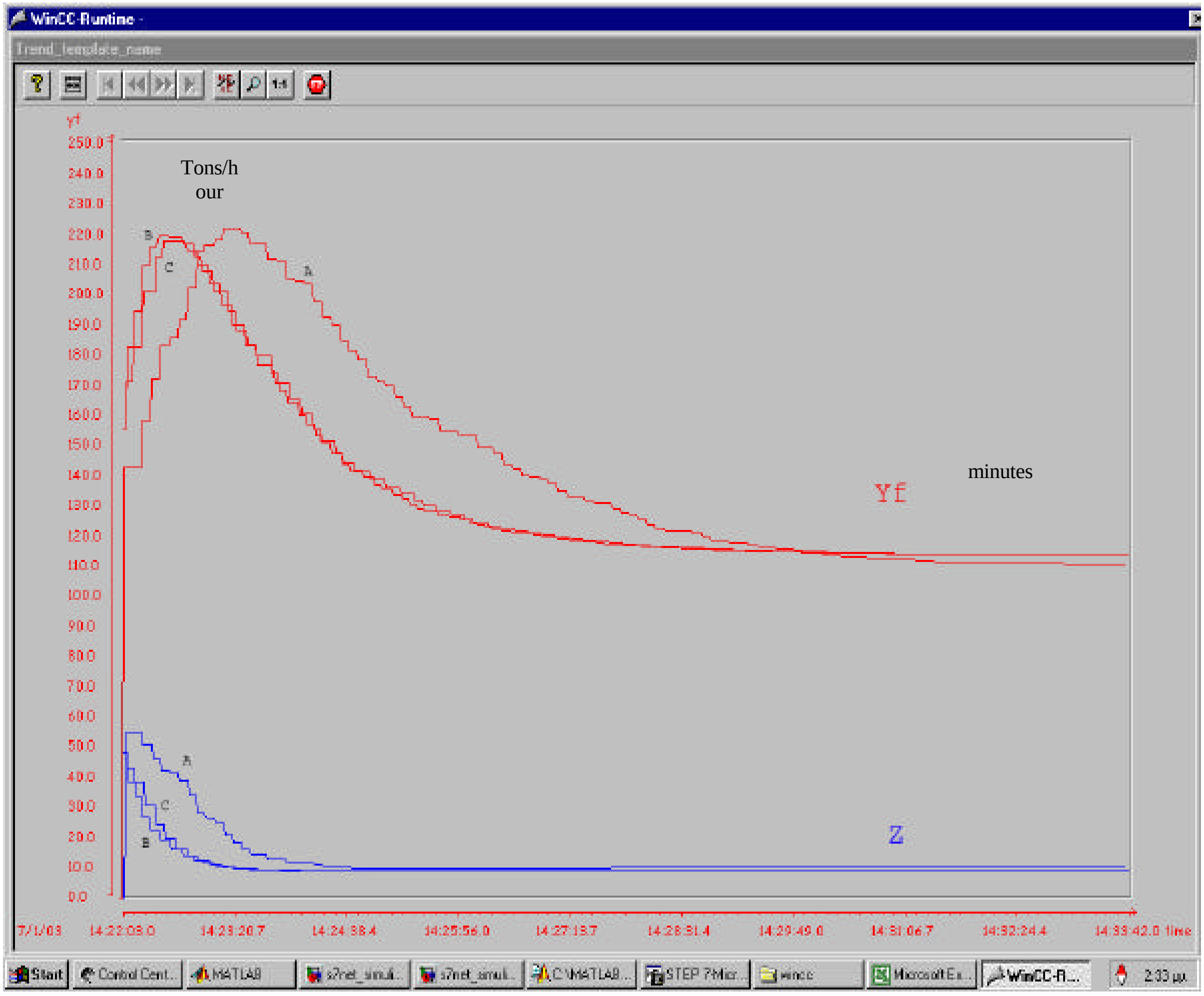

Figure 4: Plant response of the LQC control of the cement plant for bw=12 Mbps tsl=100 tbit and ttr=24333 tbit

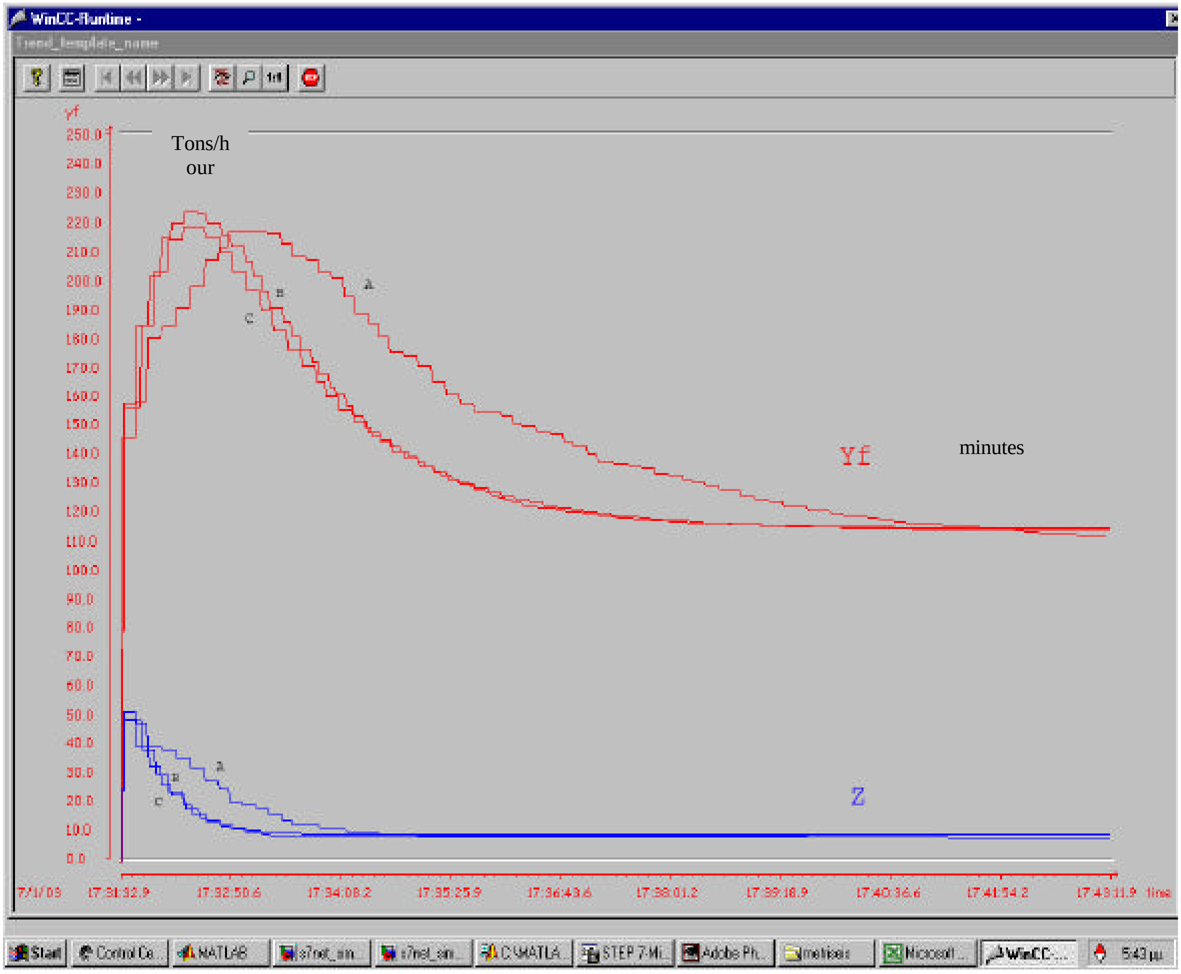

Figure 5: Plant response from the test-bed implementation of the LQC control of the cement plant for bw=12Mbps, tsl=1000 tbit and ttr=46325 tbit

V. PERFORMANCE EVALUATION OF THE NETWORKED CONTROL SYSTEM

From the curves of Figures 4 and 5 one can conclude that the network delay has a negative influence on the settling time of the plant variables when either the network load and/or the timing parameters of ttr and tst are decreased. This is quite an expected behavior is concerned because at small token rotation times and high network load the possibility of not completing the sampling of the sensors and the computation of the algorithm is quite high. Therefore, according to the Profibus operation the computations will be concluded by violating the ttr time. Consecutive violations added up over the time will result to omit one or more computations at certain sampling times, a condition that is known to lead to deterioration on the loop performance and instability. So, for a given load of the network, one can find by continuously reducing the timing parameters those values that drive the plant to instability and the values which make the plant to operate with the desired settling time, overshoot and steady state error.

VI. CONCLUSIONS

A hardware-in-the-loop simulator for networked control systems has been developed and tested. It consists of two master stations and two slave stations. Each slave is polled by each master station. The first master simulates the dynamics of the process that is controlled and the second generates load on the network in order to imitate the load that is generated by other applications that utilize the same network. This load is generated as a random sequence of data packages the size of which is determined by the Poisson distribution. These packages are absorbed by one of the slaves while the other slave performs the studied control algorithm. The network protocol implemented was the DP Profibus protocol. The conducted tests involved the quantitative comparison of the responses of the feed flow rate and load of a milling circuit of a cement plant to different network loads which can be attributed to the use of the network for other control operations in addition to those of the cement plant. The comparison has disclosed that as it was expected, the network delay influences the response and especially the settling time significantly at low token rotation times. The use of the facility can provide the necessary quantitative information which will allow somebody to trade off the network loading with more control functions at the expense of performance degradation.

REFERENCES

[1] DIN 19245, 1992. *The Profibus standard part I and II*, translated from German, Profibus Nutzerorganisation E.V.

[2] EN 50170, *General purpose field communication system*, 1996, CENELEC.

[3] Standard 802.4, *Token passing bus access method and physical layer specification*, IEEE, 1985.

[4] Matlab, *Control Systems ?oolbox, User’s Guide,* Math Works, 1996.

[5] Magni, G. Bastin, and Vincent Wertz, “Multivariable Nonlinear Predictive Control of Cement Mills”, IEEE *Transactions on Control Systems Technology*.vol. 7, no 4, pp. 502- 508, 1999.

[6] Tovar E. and Vasques F.,“Real-time Fieldbus communications using Profibus networks,” *IEEE Trans. Indust., Electron.*, vol.46, pp.1241-125. 1999.

[7] Van Breusegen V, Chen L., Bastin G., Wertz V., Werbrouck V., and de Pierpont C., “An industrial application of multivariable linear quadratic control to a cement mill circuit,” *IEEE Trans. Ind. Applicat*, vol.32, pp. 670-677, 1996.

[8] Siemens AG.,“WinCC Windows Control Center V3.1+ SP1,” 1994 – 1997.

[9] Pyyko J., Leppakoski, J., “A Novel Training and Engineering Simulation Concept for Power Plants,” ISA Product Code: TP01POW500, 2001.

[10] ABB, “Product Description Pages: Soft Controller,” http://www.abb.com

[11] EMERSON, “Process management Product Description Pages: Delta V,” http://www.emersonprocess.com

[12] Calerga, “Product Description Pages: Sysquake,” http://www.calerga.com

[13] Munger-Mynah, “Product Description Pages:Mimic,” http://www.munger.com

[14] Thayer, T., Smith, K., Brownlee, B., “Trends in Control System Vendor Solutions for Simulator Applications. Sim-Sci-Esscor,” ABB, 2004.