

Research Article

A Hybrid PID-Fuzzy Control for Linear SISO Systems with Variant Communication Delays

Adrian-Bogdan Hanchevici, Monica Patrascu, and Ioan Dumitrache

Department of Automatic Control and Systems Engineering, University Politehnica of Bucharest, 060042 Bucharest, Romania

Correspondence should be addressed to Adrian-Bogdan Hanchevici, adrian.hanchevici@upb.ro

Received 11 April 2012; Accepted 30 April 2012

Academic Editor: Salvatore Sessa

Copyright © 2012 Adrian-Bogdan Hanchevici et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We propose one networked control strategy for linear SISO systems affected by variant communication delays. The purpose of this approach is to adjust, by using the fuzzy logic, the command provided by the PID controller. The input for the fuzzy logic controller (FLC) is represented by the delay and the variation of delay, and the output is used to adapt the PID controller's command to the new value of the communication delay which occurs in the network. Simulation and implementation results are presented. The results show performance improvement when using our control strategy. The authors present the implementation for an axis of a 3D crane, using the Ethernet/IP (TCP/IP) communication protocol.

1. Introduction

Networked Control Systems (NCSs) are a new type of feedback control systems, where the information from the sensors about the controlled parameters and the commands for the actuators are exchanged through a communication network like Ethernet, CAN, and wireless. In Figure 1 is presented the general architecture of a NCS [1].

In Figure 1, S_1 to S_n represent the sensor nodes, A_1 to A_n are the actuator nodes, and C_1 to C_n are the controller nodes. There is no compulsory to be as many controllers as actuators or sensors. The main problems of this networked control systems are the variant communication delays and package lose (information lose).

In [2, 3] are presented methods to reduce the traffic through the network and so it drops the possibility to appear communication delays. For this purpose, event-based control is used, which means that the control signals are not modified until a condition is met.

According to the communication delay must be used the proper control strategy to ensure the desired QoC (quality of control). Moreover, the parameters of the network can be adapted (QoS: quality of service) by analyzing the performances of all closed-loop systems that are sharing the network [1].

In [4] are presented the challenges and opportunities in Wireless Sensor/Actuator Networks (WSAN). One main issue described is that the WSAN has to support the QoS necessary for the applications which are using the WSAN.

When we want to control a process with variant parameters, we can use the multimodel control strategy. This solution has the advantage of breaking a difficult problem into several easy problems to solve [5–8].

The adaptive control systems represent another strategy that is used when the process has structural and parametric modifications. In this case, the controllers have the capability to identify the modifications of the process and to tune their parameters, accordingly [6, 9–15].

In [16, 17] were used the genetic algorithms for online tuning of the PID controller for linear SISO systems with random communication delay. The communication delay was included in the process model.

Recently were developed new concepts which are using a combination of multimodel and adaptive control. Viable solutions were proposed for the online control of complex processes characterized by nonlinear models [18, 19].

The paper is organized as follows. In Section 2, we start by presenting the proposed control strategy. Section 3 describes the case study which includes the presentation of the experimental setup and the mathematical model of the

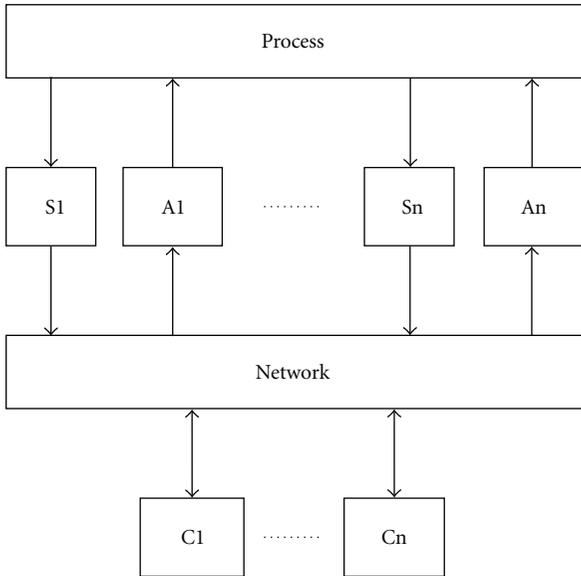


FIGURE 1: The architecture of a networked control system.

process, and also the simulation and experimental results. Finally, Section 4 contains the conclusions and the directions for future research.

2. Proposed Control Strategy

2.1. Networked Control Systems. A network of sensors and actuators is a group of sensors and actuators that are geographically distributed and interconnected. The sensors gather information about the process and the actuators react to this information by performing the appropriate actions. These networks allow such a management system to monitor and manipulate the behavior of a process. The quality of communication between nodes (Quality of Service (QoS)) plays an important role in the network.

Networks of sensors are used to gather information about a process and then submit those to interested users. Sensors cannot interfere with the process and in most applications it is not enough just to make the monitorization of parameters, also it is needed to perform actions on the process. This need for action makes necessary the coexistence in the network of sensors and actuators. The quality of communication between the nodes of the network must be provided to ensure user satisfaction with system services.

From the end user point of view, applications that use networks have their own specific requirements on the quality of communication offered by the network infrastructure. In some cases, the information has to be exchanged very fast between nodes and must be complete.

Depending on the type of target application, quality of communication within the network can be characterized by robustness, availability, security, and so forth. Some parameters can be used to measure the satisfaction level of services, such as bandwidth, delay, and loss of information.

Bandwidth is the amount of information carried within a period of time. In general, as long as the bandwidth of the network is greater, the system performance is better. Those

nodes that frequently generate large amounts of data, such as video cameras, often require high bandwidth.

Delay is the time required for a packet to get from sender to recipient. Applications that are sensitive to network delays are those that require delivering data packets in real time. Real time does not mean fast computation or high speed communication. A real time communication system is unique because it must work with speed to enable the synchronization requirements. Packet loss rate is the percentage of packets that are lost in the transmission process. It can be used to represent the probability to lose the packets. A package may be lost due to congestion and less successful connection.

These networks are not simple sensor network because of coexistence of sensors and actuators. Sensor nodes generally have low cost, low power consumption, and small size and are equipped with limited processing module and communication (for instance, enables communication with other nodes at low speeds). Due to the limitation of communication capacity, the bandwidth is limited. In particular, energy conservation is very important to extend the operation of the network, because it is not desirable to change the batteries of a sensor when it is operational. Nodes with actuators generally have a greater capacity for processing and communication. Also it is allocated more energy.

In the presence of resource constraints, network communication quality may suffer due to lack of computing resources and/or communication. For example, a number of nodes which want to transmit messages via the same network must compete for limited bandwidth that the network is able to provide. As a consequence, some data will suffer significant delays, resulting in a low quality of communication. Because of limited memory, some data packets may be lost before the node can send them to the destination. It is very important that the given limited resources available are to be used effectively.

Sensors and actuators do not have the same resource constraints. Because they were designed by using different technologies and with different purposes, they are not similar in several aspects, such as processing capacities, communication, and functionality. In a large-scale system, hardware and network technologies used in the network may be different from one subsystem to another. This is true because there are very few standard sensors and dedicated networks and actuators such of products in the market, often have different features. This heterogeneity of the network makes use of high level of available resources to be very difficult. Consequently, efficient use of resources cannot be maximized in most situations.

2.2. Hybrid PID-Fuzzy Control. The control strategy presented in this paper is proposed for linear SISO systems which have variant communication delays because of the communication network.

The control strategy is based on fuzzy logic. The fuzzy logic controller is used to adjust the command provided by the PID controller.

In Figure 2 is presented the structure of the control strategy using the fuzzy logic, where: y_k represents the controlled variable, r_k is the setpoint for the controlled variable, u_k is the control signal, $\hat{\tau}$ represents the estimated delay, and $\hat{\dot{\tau}}$ is the change in delay. The delay estimator module performs the estimation of the delay and change in delay. On every sampling time is performed the estimation of the current communication delay and the adjustment the PID controller's command (u_k^0) according to the fuzzy logic (Δu_k).

In order to determine the current communication delay, the controller requests data from the S (sensor) node. The controller knows that when there is no communication delay, the sensor node will provide the requested data after a certain period of time, t_1 . When there is the communication delay, the sensor node will provide the requested data in a higher period of time, t'_1 , where $t'_1 = t_1 + \tau'$. We have considered the case when the sensor and actuator nodes are in the same geographical area, which means that the communication delay between the sensor node and controller is the same with the communication delay between the controller and the actuator node. After all this considerations, the communication delay of the closed-loop system on a moment of time is $\tau = \tau' + \tau'$ and we have included this in the process model. We have analyzed the delay introduced by command computation and after comparing it with the communication delay, we have decided that it can be neglected.

3. Case Study

3.1. Experimental Setup and Crane Model. The nonlinear SISO system taken into study in this paper is the Oy axis of a 3-dimensional (3D) Crane. This electromechanical system has a complex dynamic behavior, being controlled from a process computer. The 3D Crane setup consists of a payload (lifted and lowered in the Oz direction by a motor mounted on a cart) hanging on a pendulum-like lifeline. The cart is mounted on a rail, giving the system capability of horizontal motion in the Ox and Oy directions. Thus, the payload can move in three dimensions. As actuators, the system makes use of three RH158.24.75 DC motors. Data acquisition is performed by means of five encoders that measure five process variables: the three coordinated of the payload in space and two deviation angles of the lifeline, with a resolution of 4096 pulses per rotation (ppr) for the spatial coordinates and 0.0015 rad for the deviation angles. A breakout box contains an interface module that amplifies the control signals which are transmitted from the PC to the DC motors [20].

For our application (3D Crane), the data acquisition is performed by five sensor nodes. The actuation is performed by three actuator nodes (DC motors).

On each sensor and actuator node there is an ATMEL microcontroller, namely ATMEGA32, an ENC28J60 module, and a clock signal generator. The microcontrollers from

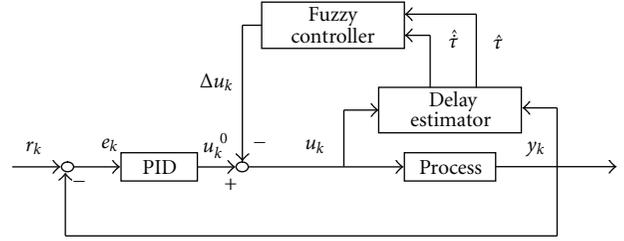


FIGURE 2: The control strategy.

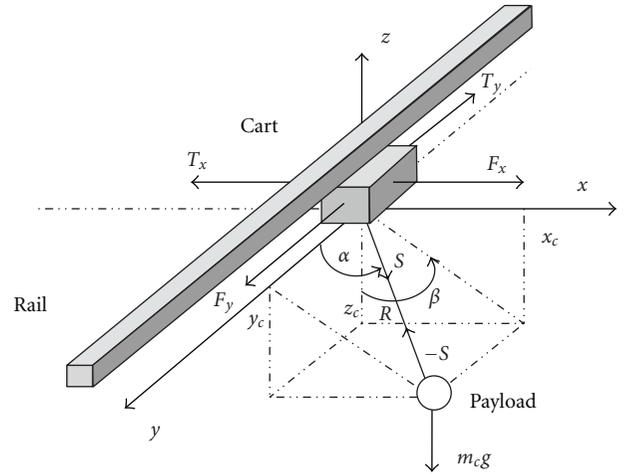


FIGURE 3: The crane system [20].

actuation nodes are used to convert the controller outputs into PWM signals used to command the motors [21]. The microcontrollers from sensor nodes count the rising and falling edges of the signals provided by the encoders and send the values from the counters in digital format to the process computer. The ENC28J60 module is used for the communication between the process computer and the sensor nodes, also between the process computer and the actuation nodes. The clock signal generator sets the working frequency of the microcontroller at the value of 16 MHz.

The 3D Crane is a MIMO system, having three input variables (control signals for the three DC motors) and five output variables (encoder signals). The nonlinear characteristic is given by the pendulum-like motion of the payload in space, with a variable line length. Figure 3 (INTECO (2000)) presents the crane system, where: x_w (not represented) is the distance of the rail with the cart from the center of the construction frame; y_w (not represented) is the distance of the cart from the center of the rail; R is the length of the lifeline; α represents the angle between the Oy axis and the lifeline; β represents the angle between the negative direction on the Oz axis and the projection of the lifeline onto the Oxz plane; m_c is the mass of the payload; m_w is the mass of the cart; m_s is the mass of the moving rail; x_c, y_c, z_c are the coordinates of the payload; $S = F_R - T_R$ represents the reaction force in the lifeline acting on the cart; F_x is the force driving the rail with cart; F_y is the force driving the cart along the rail; F_R is the force controlling the length of the lifeline; T_x, T_y, T_R are friction forces.

If we define the state variables and the relations of them we get a mathematical model as

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & s_5^2 \beta^2 + \dot{\alpha}^2 + Z & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & \mu_1 s_5 s_7 \\ 0 & 0 & 0 \\ 1 & 0 & \mu_1 c_5 \\ 0 & 0 & 0 \\ -c_5 & -s_5 c_7 & -(1 + \mu_1 c_5^2 + \mu_1 s_5^2 s_7^2) \end{bmatrix} \cdot \begin{bmatrix} N_1 \\ N_2 \\ N_3 \end{bmatrix}, \quad (1)$$

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & s_5 s_7 & 0 \\ 0 & 0 & 1 & 0 & c_5 & 0 \\ 0 & 0 & 0 & 0 & 0 & -s_5 c_7 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix},$$

where the following notations have been used: $x_1 = x_w$; $x_2 = \dot{x}_1$; $x_3 = y_w$; $x_4 = \dot{x}_3$; $x_5 = R$; $x_6 = \dot{x}_5$; $\mu_1 = m_c/m_w$; $\mu_2 = m_c/(m_w + m_s)$; $N_1 = (F_y - T_y)/m_w$; $N_2 = (F_x - T_x)/(m_w + m_c)$; $N_3 = (F_R - T_R)/m_c$; $s_5 = \sin \alpha$; $c_5 = \cos \alpha$; $s_7 = \sin \beta$; $c_7 = \cos \beta$; g is the gravitational acceleration; $Z = (g s_5 c_7)/R$ is a nonlinear function.

A simulation model of the system was implemented. Details of this model can be found in [20].

3.2. Simulation and Experimental Study. The nonlinear SISO system taken into study in this paper can be approximated by the rational s -transfer function:

$$H_P(s) = \frac{B(s)}{A(s)} = \frac{K_P}{s(T_P \cdot s + 1)}, \quad (2)$$

where K_P represents the gain and T_P describes the dynamics of the process. The communication delay is included in the process model and $H_P(s)$ becomes

$$H'_P(s) = \frac{B(s)}{A(s)} e^{-\tau \cdot s} = \frac{K_P}{s(T_P \cdot s + 1)} e^{-\tau \cdot s}, \quad (3)$$

where τ is variant.

We have considered the linear approximation of the process, as presented above. This approximation was made experimentally. After making this linearization, we have tuned the PID controller, for the case when there is no communication delay ($\tau = 0$ sec). The tuning was made offline, in a simulated environment.

After the offline tuning was achieved, the PID controller has been tested, in simulation. We have continued and considered the communication delay $\tau = 0.6$ seconds and $\tau = 1.0$ seconds. Below, in Figure 4, are presented the simulated responses of the system using the same PID

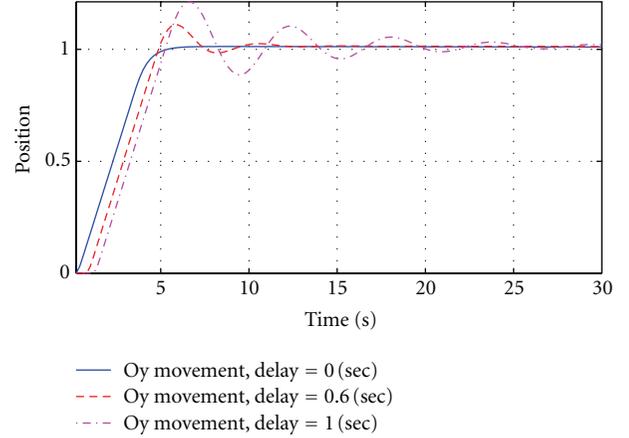


FIGURE 4: Simulated responses using the same PID controller and different communication delays.

Command u		Delay								
		P0	P1	P2	P3	P4	P5	P6	P7	P8
Change in delay	P4	P6	P7	P8						
	P3	P5	P6	P7	P8	P8	P8	P8	P8	P8
	P2	P4	P5	P6	P7	P8	P8	P8	P8	P8
	P1	P3	P4	P5	P6	P7	P8	P8	P8	P8
	Z	P2	P3	P4	P5	P6	P7	P8	P8	P8
	N1	P1	P2	P3	P4	P5	P6	P7	P8	P8
	N2	P0	P1	P2	P3	P4	P5	P6	P7	P8
	N3	P0	P0	P1	P2	P3	P4	P5	P6	P7
N4	P0	P0	P0	P1	P2	P3	P4	P5	P6	

FIGURE 5: The rule base of the fuzzy logic controller.

controller and different communication delays. By analyzing the three responses, we can see that there is a difference between them, in terms of settling time and overshoot, while there is no difference in terms of steady state error.

This mismatch is generated by the fact that when the communication delay appears, the PID controller designed for the case when there is no communication delay is not able to perform as requested and the closed-loop system's performances are getting worse as the communication delay is getting higher values.

In order to prevent the decrease of performances of the closed-loop system, we have adopted the control strategy presented in Figure 2.

The rule base of the fuzzy logic controller used in simulation and experimental study is presented in Figure 5.

For the delay and command are used the P0–P8 (positive) membership functions, and for the change in delay are used the N4, N3, N2, N1 (negative), Z (zero), P1, P2, P3, and P4 membership functions.

For our study, we have considered that the highest bound for the delay is 2 seconds.

The system works with a sampling time of 0.2 seconds.

TABLE 1: Simulation study analysis.

	Settling time(s)	Overshoot (%)	Steady state error (%)
	O _y	O _y	O _y
PID (fixed delay)			
$\tau = 0.0$ sec (S)	4.4	1.3	0
$\tau = 0.4$ sec (S)	6.2	6.2	0
$\tau = 0.6$ sec (S)	7.0	11.0	0
$\tau = 1.0$ sec (S)	18.6	21.3	0
$\tau = 1.4$ sec (S)		unstable	
$\tau = 1.8$ sec (S)		unstable	
$\tau = 2.0$ sec (S)		unstable	
PID and fuzzy (fixed delay)			
$\tau = 0.0$ sec (S)	4.8	1.3	0
$\tau = 0.4$ sec (S)	4.8	1.9	0
$\tau = 0.6$ sec (S)	5.0	3.7	0
$\tau = 1.0$ sec (S)	8.6	8.6	0
$\tau = 1.4$ sec (S)	10.2	12.4	0
$\tau = 1.8$ sec (S)	12.0	12.2	0
$\tau = 2.0$ sec (S)	13.2	9.3	0
PID (variant delay)			
S1 ($\tau_{\max} = 2.0$ sec)	24.2	23.8	0
S2 ($\tau_{\max} = 2.0$ sec)	31.2	26.8	0
S3 ($\tau_{\max} = 2.0$ sec)	8.8	17.1	0
S4 ($\tau_{\max} = 2.0$ sec)	16.0	27.2	0
S5 ($\tau_{\max} = 2.0$ sec)	10.0	42.9	0
S6 ($\tau_{\max} = 2.0$ sec)	9.2	25.3	0
S7 ($\tau_{\max} = 2.0$ sec)	14.2	41.0	0
PID and fuzzy (variant delay)			
S1 ($\tau_{\max} = 2.0$ sec)	10.8	16.5	0
S2 ($\tau_{\max} = 2.0$ sec)	9.8	9.0	0
S3 ($\tau_{\max} = 2.0$ sec)	9.4	7.4	0
S4 ($\tau_{\max} = 2.0$ sec)	9.2	12.3	0
S5 ($\tau_{\max} = 2.0$ sec)	11.2	24.7	0
S6 ($\tau_{\max} = 2.0$ sec)	6.8	2.5	0
S7 ($\tau_{\max} = 2.0$ sec)	9.4	9.9	0

TABLE 2: Experimental study analysis.

	Settling time(s)	Overshoot (%)	Steady state error (%)
	O _y	O _y	O _y
PID (fixed delay)			
$\tau = 0.0$ sec (E)	3.8	0.0	0
$\tau = 0.4$ sec (E)	3.2	3.9	0
$\tau = 0.6$ sec (E)	5.0	10.2	0
$\tau = 1.0$ sec (E)	11.6	31.5	0
$\tau = 1.4$ sec (E)	44.0	49.3	0
$\tau = 1.8$ sec (E)		unstable	
$\tau = 2.0$ sec (E)		unstable	
PID and fuzzy (fixed delay)			
$\tau = 0.0$ sec (E)	4.4	0.0	0
$\tau = 0.4$ sec (E)	4.0	0.0	0
$\tau = 0.6$ sec (E)	4.0	0.1	0
$\tau = 1.0$ sec (E)	6.4	6.2	0
$\tau = 1.4$ sec (E)	7.8	7.5	0
$\tau = 1.8$ sec (E)	6.8	0.0	0
$\tau = 2.0$ sec (E)	7.0	0.0	0
PID (variant delay)			
E1 ($\tau_{\max} = 2.0$ sec)	3.2	3.1	0
E2 ($\tau_{\max} = 2.0$ sec)	16.8	30.9	0
E3 ($\tau_{\max} = 2.0$ sec)	13.2	34.5	0
E4 ($\tau_{\max} = 2.0$ sec)	14.8	27.5	0
E5 ($\tau_{\max} = 2.0$ sec)	9.6	24.6	0
E6 ($\tau_{\max} = 2.0$ sec)	9.0	36.3	0
E7 ($\tau_{\max} = 2.0$ sec)	11.4	42.8	0
PID and fuzzy (variant delay)			
E1 ($\tau_{\max} = 2.0$ sec)	5.4	1.9	0
E2 ($\tau_{\max} = 2.0$ sec)	8.4	14.2	0
E3 ($\tau_{\max} = 2.0$ sec)	8.2	17.6	0
E4 ($\tau_{\max} = 2.0$ sec)	8.2	8.6	0
E5 ($\tau_{\max} = 2.0$ sec)	4.6	2.1	0
E6 ($\tau_{\max} = 2.0$ sec)	6.2	0.0	0
E7 ($\tau_{\max} = 2.0$ sec)	4.2	2.3	0

The process computer used has the following hardware configuration: Intel(R) Core(TM)2 Duo CPU E7200 @ 2.53 GHz 2.53 GHz, 2.00 GB of RAM. As an operating system, it was used Microsoft Windows XP Professional Version 2002 Service Pack 3. Also, we have used the Matlab.

In Tables 1 and 2 are presented the cases analyzed in this paper.

- (i) PID (fixed delay) S, E represent the performances of the system obtained in simulation and experimentally after the tuning of the PID controller was achieved and when were considered different communication delays.
- (ii) PID and fuzzy (fixed delay) S, E represent the performances of the system obtained in simulation

and experimentally when is used the proposed control strategy and when were considered different communication delays.

- (iii) PID (variant delay) S, E represent the performances of the system obtained in simulation and experimentally after the tuning of the PID controller was achieved and when the communication delay has variant values between 0 and 2 seconds.
- (iv) PID and Fuzzy (variant delay) S, E represent the performances of the system obtained in simulation and experimentally when is used the proposed control strategy and the communication delay has variant values between 0 and 2 seconds.

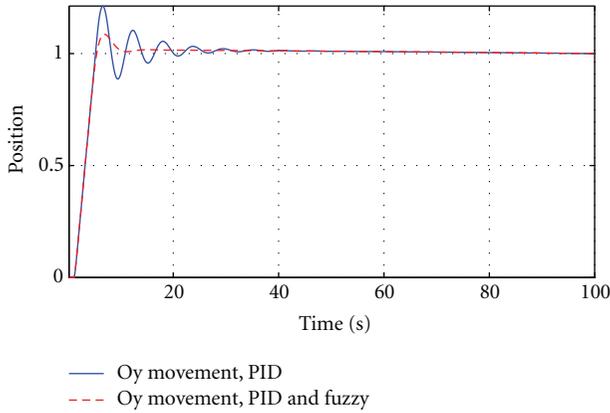


FIGURE 6: Simulated responses for both situations when the delay is fixed ($\tau = 1.0$ sec).

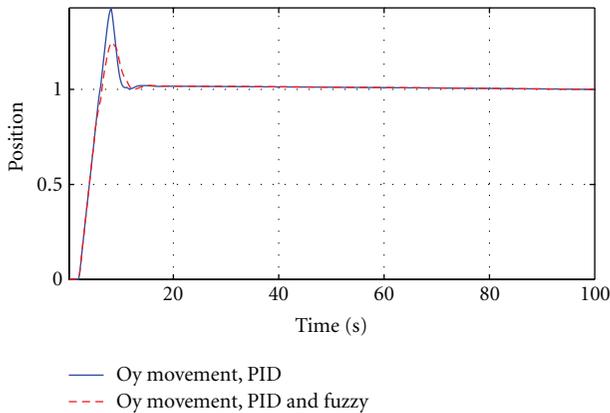


FIGURE 7: Simulated responses for both situations with variant delay (case S5).

In Figures 6 and 7 are described the simulated responses of the system for the case when is used only the PID controller, and for the case when is used the fuzzy logic adjustment. The responses from Figure 6 were obtained for fixed delay ($\tau = 1.0$ sec), and those from Figure 7 for variant delay. The distribution of the variant delay is presented in Figure 8 (case S5).

For the experimental study we have considered the Ethernet/IP (TCP/IP) communication protocol. As in simulation, we have considered the communication delay $\tau = 0.6$ seconds and $\tau = 1.0$ seconds. In Figure 9, are presented the experimental responses of the system using the same PID controller and different communication delays.

As in simulation, also in experimental environment, the PID controller designed for the case when there is no communication delay is not able to perform as requested and the closed-loop system's performances are getting worse as the communication delay is getting higher values. The experimental results are presented in Figure 9.

In Figures 10 and 11 are presented the experimental responses of the system for the case when is used only the PID controller, and for the case when is used the fuzzy logic adjustment. The responses from Figure 10 were obtained for fixed delay ($\tau = 1.0$ sec), and those from Figure 11 for variant

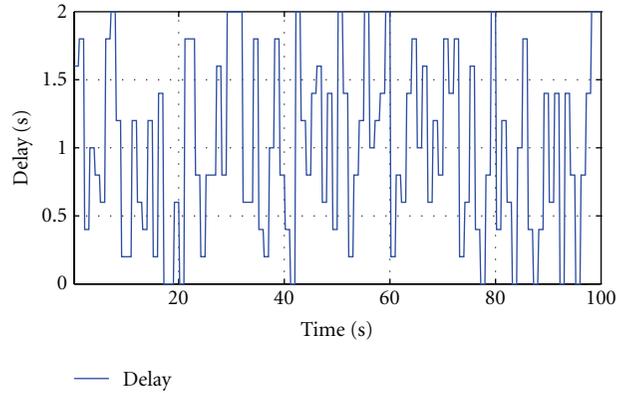


FIGURE 8: The distribution of the delay (case S5).

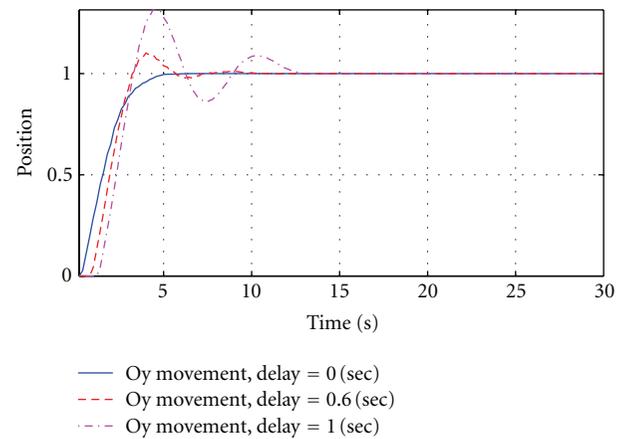


FIGURE 9: Experimental responses using the same PID controller and different communication delays.

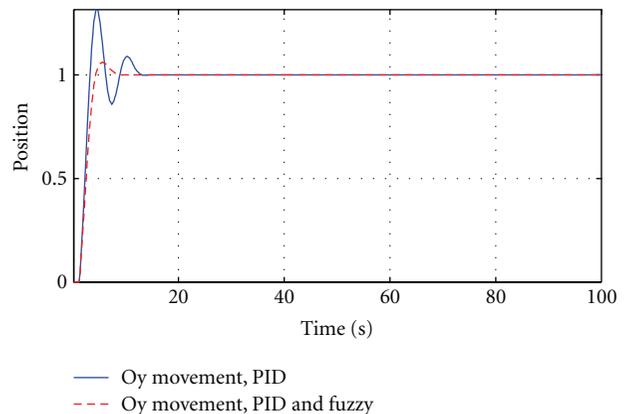


FIGURE 10: Experimental responses for both situations when the delay is fixed ($\tau = 1.0$ sec).

delay. The distribution of the variant delay is presented in Figure 12 (case E7).

By analyzing the responses presented in Figures 4, 5, and 6 and in Figures 9, 10, and 11 we can see that the performances of the closed-loop system are improved when is used our proposed control strategy, in terms of overshoot

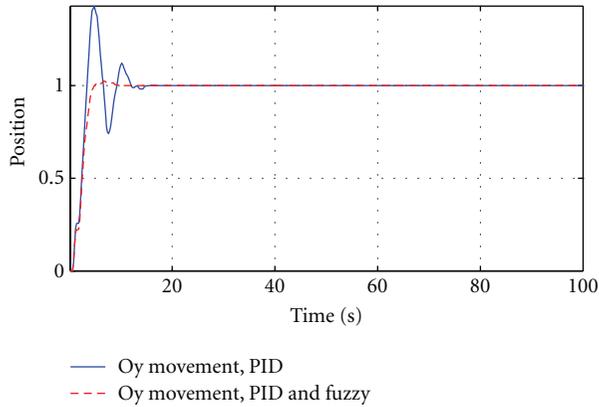


FIGURE 11: Experimental responses for both situations with variant delay (case E7).

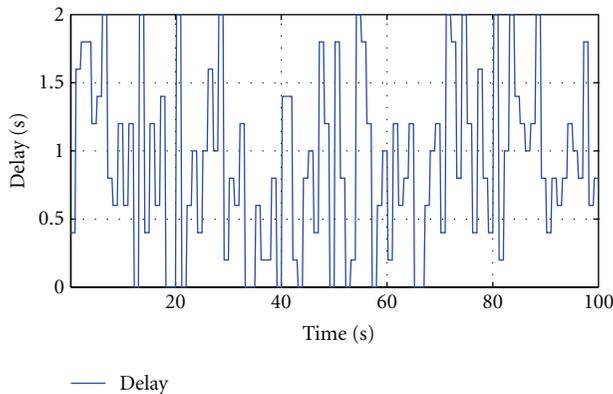


FIGURE 12: The distribution of the delay (case E7).

and settling time, while there is no difference in terms of steady state error.

4. Conclusions

In this paper, we proposed a networked control strategy for linear SISO systems affected by variant communication delays. By using a fuzzy logic controller, it is adjusted the command provided by the PID controller. As input for the fuzzy logic controller are used the delay and the change in delay. By using the output of the FLC, the PID controller's command is adjusted accordingly to the current value of the communication delay.

By analyzing the simulation and experimental results, we can see that the performances of the closed-loop system are improved when is used our proposed control strategy. For experimental analysis it was used an axis of a 3D Crane and the Ethernet/IP (TCP/IP) communication protocol.

In future work, we will use another controller for nonlinear processes.

Acknowledgments

The work has been funded by the Sectoral Operational Programme Human Resources Development 2007–2013 of the

Romanian Ministry of Labour, Family and Social Protection through the Financial Agreement POSDRU/88/1.5/S/61178. Figure 3 and the nonlinear crane model are presented with the courtesy of INTECO.

References

- [1] C. Aubrun, D. Simon, and Y. Song, *Co-design Approaches for Dependable Networked Control Systems*, John Wiley & Sons, New York, NY, USA, 2010.
- [2] D. Lehmann and J. Lunze, "Event-based control with communication delays," in *Proceedings of the 18th International Federation of Automatic Control World Congress (IFAC '11)*, pp. 3262–3267, Milano, Italy, 2011.
- [3] M. Mazo and P. Tabuada, "Decentralized event-triggered control over wireless sensor/actuator networks," *IEEE Transactions on Automatic Control*, vol. 56, no. 10, pp. 2456–2461, 2011.
- [4] F. Xia, "QoS challenges and opportunities in wireless sensor/actuator networks," *Sensors*, vol. 8, no. 2, pp. 1099–1110, 2008.
- [5] M. Athans, S. Fekri, and A. Pascoal, "Issues on robust adaptive feedback control," in *Proceedings of the 16th Triennial World Congress of International Federation of Automatic Control (IFAC '05)*, pp. 547–577, Prague, Czech Republic, July 2005.
- [6] I. Dumitrache, *Automatica. Volumul 1*, Editura Academiei Romane, Bucuresti, Romania, 2010.
- [7] M. Patrascu and A. B. Hanchevici, *Sisteme Avansate de Conducere. Indrumar de laborator*, Editura Politehnica Press, Bucuresti, Romania, 2011.
- [8] R. Toscano, "Robust synthesis of a PID controller by uncertain multimodel approach," *Information Sciences*, vol. 177, no. 6, pp. 1441–1451, 2007.
- [9] K. J. Aström and B. Wittenmark, *Adaptive Control*, Addison-Wesley, Reading, Mass, USA, 1989.
- [10] I. Dumitrache, *Ingineria Reglării Automate. Volumul 1*, Editura Politehnica Press, Bucuresti, Romania, 2010.
- [11] I. Dumitrache, *Ingineria Reglării Automate. Volumul 2*, Editura Politehnica Press, Bucuresti, Romania, 2010.
- [12] G. Tao, *Adaptive Control Design and Analysis*, Wiley-Interscience, John Wiley & Sons, 2003.
- [13] R. Iserman, *Digital Control Systems. Volume 11*, Springer, Berli, Germany, 1991.
- [14] J. J. E. Slotine, *Nonlinear Control Systems*, Prentice Hall, 1993.
- [15] L. V. Thomas and J. W. Grantham, *Nonlinear and Optimal Control Systems*, John Wiley & Sons, 2000.
- [16] A. B. Hanchevici and I. Dumitrache, "Online tuning of PID controller for linear SISO system with random communication delay by using genetic algorithms," in *Proceedings of the IFAC Conference on Advances in PID Control*, Brescia, Italy, 2012.
- [17] A.B. Hanchevici and I. Dumitrache, "Intelligent PID control for linear SISO system with random communication delay by using online genetic algorithms," in *Proceedings of the 11th IFAC/IEEE International Conference on Programmable Devices and Embedded Systems*, Brno, Czech Republic, 2012.
- [18] S. Fekri, M. Athans, and A. Pascoal, "Issues, progress and new results in robust adaptive control," *International Journal of Adaptive Control and Signal Processing*, vol. 20, no. 10, pp. 519–579, 2006.
- [19] S. Fekri, M. Athans, and A. Pascoal, "Robust multiple model adaptive control (RMMAC): a case study," *International Journal of Adaptive Control and Signal Processing*, vol. 21, no. 1, pp. 1–30, 2007.

- [20] InTeCo, *3D Crane*, InTeCo, Krakow, Poland, 2000.
- [21] S. F. Barrett and D. J. Pack, *Atmel AVR Microcontroller Primer: Programming and Interfacing*, Morgan & Claypool Publishers, Seattle, Wash, USA, 2008.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

