World Academy of Science, Engineering and Technology
International Journal of Electronics and Communication Engineering
Vol:10, No:3, 2016

Open Science Index, Electronics and Communication Engineering Vol:10, No:3, 2016 publications.waset.org/10003729.pdf

Analysis of Time Delay Simulation in Networked Control System

Nyan Phyo Aung, Zaw Min Naing, Hla Myo Tun

Abstract**—**The paper presents a PD controller for the Networked Control Systems (NCS) with delay. The major challenges in this networked control system (NCS) are the delay of the data transmission throughout the communication network. The comparative performance analysis is carried out for different delays network medium. In this paper, simulation is carried out on Ac servo motor control system using CAN Bus as communication network medium. The True Time toolbox of MATLAB is used for simulation to analyze the effect of different delays.

Keywords**—**NCS, Time delay, CAN Bus, True time, MATLAB.

I. INTRODUCTION

NETWORKED CONTROL SYSTEMS (NCSs) have received much attention recently [1]–[5]. In control system when traditional feedback system is closed via communication channel which may be shared with other nodes outside the control system, it is defined as networked control system (NCSs) [6], [7]. So NCS is the system in which a control loop is closed via a shared communication network. Fig. 1 shows the typical model of the NCSs [8]. There are two major types of control systems that utilizes the communication network (i) shared network control system and (ii) remote control systems. Shared network resources can greatly reduce the complexity of connections between sensors to controllers and controllers to actuators. The place where central controller is installed is termed as "local site" while the place where the plant is installed is defined as "remote side". Two general approaches for designing NCSs. these are (i) hierarchical structure and (ii) direct structure. The use of a shared network in the feedback path has advantages of low installation cost, reducing system wiring, simple system diagnosis and easy maintenance. However, some inherent shortcomings, such as bandwidth constraints, packet delays and packet dropouts, will degrade performance of NCSs or even cause instability. Stability analysis of NCSs is investigated in [9]–[11], and stabilizing controllers are designed in [12], [13]. To overcome these challenges various researchers have designed different types of controllers. J. Nilsson et al. [12] designed the stochastic optimal LQG controller and studied the effect of networked induced delay in NCS with an assumption that networked induced delay is less than the sampling period.

The design of direct structure where a sensor and a controller are directly connected to the network is discussed in this paper. Apart from this the networked delay considered in this paper is greater than or equal to sampling interval. The objective of this paper is to analyze the effect of designed discrete controller in the presence of fixed delays under CAN Bus as a network medium. Within this limit the plant output will be controllable. The interest of this study is to show the importance of the networked delay used in different networks to control remote applications. There are many challenges that has to be taken care while designing of NCSs. The main challenge is to deal with time delay compensation for obtaining optimal performance of the system. The introduction of the time delay degrades the performance of the system and makes it unstable. In literature available there are various types of controllers designed for time compensation schemes. The general block diagram of NCSs model contain plant, controller, actuator and sensor is shown in Fig. 1.

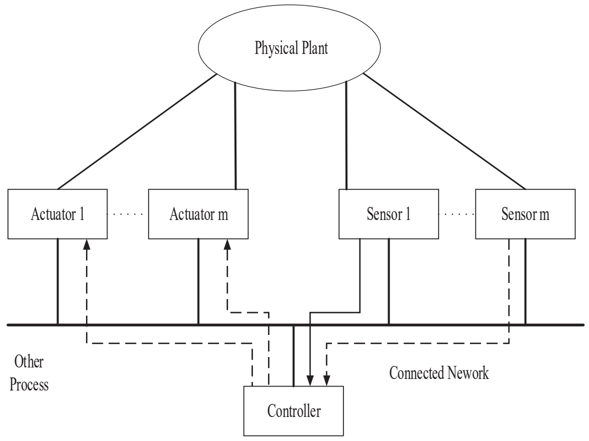


Fig. 1 General block diagram of NCS

The new model of NCSs was based on networked induced delay and data packet drop out in the transmission. This paper focuses on the controller design for compensating various communication delays which exist in the networked control system. The networked medium used is CAN Bus. The control law of the proposed discrete time PD controller is designed by considering the effects of delay parameters within the network. The proposed controller is discrete time PD controller whose control law is designed by considering the effect of delay parameters within the network. Three different cases are elaborated using this controller (i) network used with ideal conditions, (ii) Output response under different time delays.

Nyan Phyo Aung is with the Department of Electronic Engineering, Mandalay Technological University, Myanmar; nyanphyoaung@gmail.com).
Zaw Min Naing was with the Department of Research and Innovation, Ministry of Science and Technology, Myanmar (e-mail: drzaw290615@gmail.com).
Hla Myo Tun is with the Department of Electronic Engineering, Mandalay Technological University, Myanmar (e-mail: kohlamyotun@gmail.com).

Open Science Index, Electronics and Communication Engineering Vol:10, No:3, 2016 publications.waset.org/10003729.pdf

II. RELATIONSHIP BETWEEN DELAY AND NCS

The NCS works with a network, therefore the data transfers between the controller and the remote system will induce network delays in addition to the controller processing delay. Fig. 2 shows the delay model of NCSs. As discussed earlier this model represent the direct structure in which the controller communicates with sensor and actuator over the network [14].

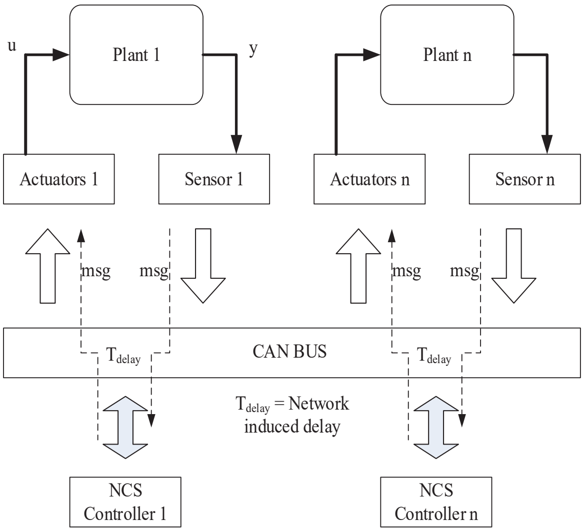


Fig. 2 Proposed model of Networked Control System

The signal takes some amount of time to transmit the data from sensor to controller and controller to actuator. In other words, the delay appears when exchange data activity between sensors, actuators, and controllers through the networks. This dynamic activity will affect the performance of control system designed without considering it and even stability of the system. Besides, the signal processing and computational delays that depending on the scheduling protocol should be taken. The amount of time delay taken by signal for transmission depends upon communication network protocol. The total time delay in NCSs is classified in three sections as discussed in [15]

(i) Data transfer delay between sensor to controller, τ_k^{sc}

(ii) Computational delay generated at the controller, τ_k^{c}.

(iii) Data transfer delay between controllers to actuator, τ_k^{ca}.

where k indicates the number of time instants (k=1,2, ... n).

The communication delay is the sum of two components i.e. sensor to controller τ_k^{sc} and controller to actuator τ_k^{ca}. The control delay is the sum of the computational delay and the communication delay.

$$\tau_k = \tau_k^{sc} + \tau_k^{c} + \tau_k^{ca} \tag{1}$$

The direction of the data transfer as the sensor to controller delay τ_k^{sc} and the controller to actuator delay τ_k^{ca} can be seen in Fig. 2. The delays are computed as

$$\tau^{sc} = t^{cs} - t^{se} \tag{2}$$

$$\tau^{ca} = t^{rs} - t^{ce} \tag{3}$$

where t^{se} the time instant that the remote system encapsulates the measurement to a frame or a packet to be sent, t^{cs} is the time instant that the controller starts processing the measurement in the delivered frame or packet, t^{ce} is the time instant that the main controller encapsulates the control signal to a packet to be sent, and t^{rs} is the time instant that the remote system starts processing the control signal. In fact, both network delays can be longer or shorter than the sampling time Ts. The control delay can be fixed or variable.

Designing of the system with variable control delay is more complicated rather than fixed control delay. The sampling period of sensor is denoted by Ts and that of controller and actuator is denoted by Tc. According to the sampling rates, the networked control system is classified in two categories (i) Single-rate NCSs (ii) Multi-rate NCSs.

- Definition 1: When the sampling periods of the sensor, controller and actuator in networked control systems are the same then such NCSs is called single-rate networked control systems.
- Definition 2: If the sampling periods of the sensor, controller and actuator in networked control systems are different then such NCS is called as multi-rate networked control systems. Single-rate NCSs model is consider in the paper.

III. CONTROLLER AREA NETWORK

The communication network is the backbone of the NCS, therefore, the best network used in this work is the Controller Area Network (CAN) which is a multi-master serial bus that broadcast messages to all nodes in the network system. CAN was defined in International Standardization Organization (ISO) as a serial communication bus to replace the complex wiring harness with a two wire bus. The CAN system offers a transmission speed of up to 1 Mbit/s with error detection method for effective transmission [16].

A. Communication Mechanism

CAN uses carrier sense multiple access protocol with collision detection (CSMA/CD) and arbitration on message priority as its communication protocol? This communication protocol allows every node in CAN to monitor the bus network in advance before attempting to transmit a message. When no activity occurs in the network, each node has the same opportunity to transmit a message. Each type of message has a unique message identifier which serves as the name of the message as well as the priority code of the message. Lower numbered identifiers correspond to higher priority messages.

The message identifier is the first component of a message sent by a node to the bus. When a node wants to transmit a message it checks the bus to see if there is a message on the bus. If not, the node will begin its transmission immediately. If

Open Science Index, Electronics and Communication Engineering Vol:10, No:3, 2016 publications.waset.org/10003729.pdf

there is a message on the bus, then transmissions delayed until the current message is completed. Once the node finds the bus idle transmissions begin immediately. Transmitted messages are broadcasted to the bus and all nodes on the network are constantly listening to the bus. When a message appears on the bus, every node checks the message identifier. If this message identifier matches a receiving object header in the node, then the node will retrieve the message otherwise the message is ignored. When two or more nodes begin their transmission at the same time, a collision will occur. As two message identifiers are sent to the bus bit by bit, the dominant ‘0’ bit will overwrite the recessive ‘1’ bit, so the identifier with the smaller binary value will overwrite the identifier with larger value. While a node is transmitting, it listens to see if the bit on the bus is the same as it is sending. If what it hears is different from what it sent, a collision has occurred and it will stop its transmission immediately. The higher priority message will continue until it is completed sending package. After the higher priority message is completed, the lower priority message will try to transmit again. If there are other higher priority messages waiting to be transmitted, the lower priority message will lose arbitration again and again until all higher priority messages are done sending the package.

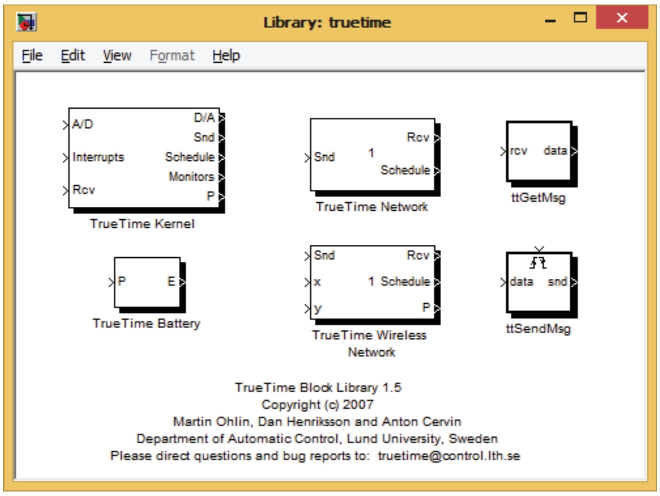
B. Scheduling of CAN Messages

CAN bus use the data frame identifier to express the source of information and the priority. The frames are broadcasted on the bus and each station can decide if the message content is relevant by examining the message identifier of received frames. These identifiers have to be assigned statically during the design phase of the bus system to avoid ambiguity in the interpretation of the frame content. A global view of the complete system is needed in this process.

C. TRUETIME

TRUETIME is a MATLAB/Simulink-based simulator for real-time control systems. TRUETIME facilitates co-simulation of controller task execution in real-time kernels, network transmissions, and continuous plant dynamics. The kernel block simulates a real-time kernel executing user-defined tasks and interrupt handlers. The various network blocks allow node (kernel blocks) to communicate over simulated wired or wireless networks.

The TRUETIME network block simulates medium access and packet transmission in a local area network. Six simple models of networks are supported: CSMA/CD (e.g. Ethernet), CSMA/AMP (e.g. CAN), Round Robin (e.g. Token Bus), FDMA, TDMA (e.g. TTP), and Switched Ethernet. The usage of the wireless network block is similar to and works in the same way as the wired one. To also take the path-loss of the radio signals into account, it has x and y inputs to specify the true location of the nodes. Two network protocols are supported at this moment: IEEE 802.11b/g (WLAN) and IEEE 802.15.4 (ZigBee) [17]. The block library consists of the TRUETIME Kernel block that simulates a real-time kernel executing user defined tasks and interrupt handlers, the Network block that allows nodes to communicate over simulated network, a couple of standalone interface blocks and of the Battery block that allows modeling of battery driven operation. TRUETIME library block in MATLAB is as shown in Fig. 3.



TrueTime Block Library 1.5
Copyright (c) 2007
Martin Ohlin, Dan Henriksson and Anton Cervin
Department of Automatic Control, Lund University, Sweden
Please direct questions and bug reports to: truetime@control.lth.se

Fig. 3 TRUETIME Tool Box

IV. DEVELOPMENT OF SIMULINK MODEL FOR NCS WITH AC SERVO MOTOR MODEL

The motor is comely used as actuator in control system design. In this paper, two phase ac servo motor is acted as plant in this simulation [18]. The two-phase ac servomotor is probably the most commonly used type of servomotor. The ac servomotor is a two-phase induction motor having its two stator coils separated by "90" electrical degrees with a high-resistance rotor. A control signal is applied to one phase while the other phase (the reference winding) is supplied with a fixed signal that is phase-shifted by 90-degree relative to the control signal. The motor is used primarily for relatively low-power applications. The reference filed voltage is denoted as er(s) and the control filed voltage is denoted as Vc(s). The developed torque of this motor is proportional to Vc(s), er(s) and the since of the angle between er(s) and Vc(s) The block diagram of, two-phase control field, rotational actuator controlled AC motor is shown in Fig. 4.

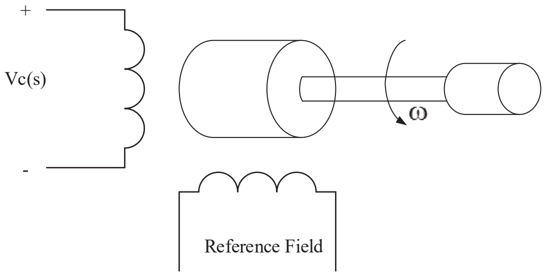


Fig. 4 The block diagram of two phase ac servo motor

In general, transfer function of motor is:

$$\frac{\theta(s)}{V_c(s)}=\frac{K_m}{s(\tau s+1)} \qquad (4)$$

Open Science Index, Electronics and Communication Engineering Vol:10, No:3, 2016 publications.waset.org/10003729.pdf

where

$$\tau\text{-}J/(b\text{-}m)$$

m-slope of linearized torque-speed curve (normally negative)

The variable parameters of the servo motor system are considered as: $\tau = 1$ and $K_m = 1000$.

The transfer function of servo motor which is used as plant in NCS are

$$Gp(s) = \frac{1000}{s(s+1)} \quad (5)$$

A. Calculation of PD Gain

PD Controller is used as control algorithm for controller node in this simulation. So controller parameter is required to calculate [19]. The transfer function of controller is

$$G_c = K_p + K_d s. \quad (6)$$

In this paper, AC servo motor acts as plant in the Simulink model. The transfer function of motor is shown in (5). First, P controller's gain Kp is first considered by:

$$G(s) = \frac{1000}{s(s+1)} \times K \quad (7)$$

The equation is changed into characteristic equation and solved by root criteria. So the value of gain kp for the controller p must be greater than zero (kp>0). And next calculation is PD controller gain by

$$G(s) = \frac{1000}{s(s+1)} \times \left(Kp + Kds\right) \cdot \quad (8)$$

By using Routh-Hurwitz, the value of Kd must be greater than 0.001. So the gain of PD controller is considered from this point as shown in Table I.

V. PROPOSED SIMULINK MODEL OF NCS BASED ON CAN BUS

CAN bus based Networked control system can be simulated using Matlab/Simulink with TRUETIME toolbox. TRUETIME is good tools used for experimental platform for research on dynamical real-time controls system, by taking into account the effects of the execution of the control task and the data transmission on the controlled system dynamics. It provides computer and network block which execute user define thread, such as scheduling policies, controller task, network interface task, I/O tasks and message communication. TRUETIME allow the researchers to study the compensation schemes that adjust the control algorithm based on measurement of actual timing variation.

In order to perform the model of CAN-based networked control system of the simulation platform is established by using DC motor connected to CAN, controlled by one controller node with PD control algorithm as shown in Fig. 5. The testing of controller parameters for simulation is as show in Table I.

TABLE I
PD CONTROLLER PARAMETERS

Parameter	Meaning	Values			
		Case I	Case II	Case III	Case IV
K_P	Proportional value	1	1.5	1.5	1.5
K_D	Differential Value	0.02	0.02	0.035	0.05

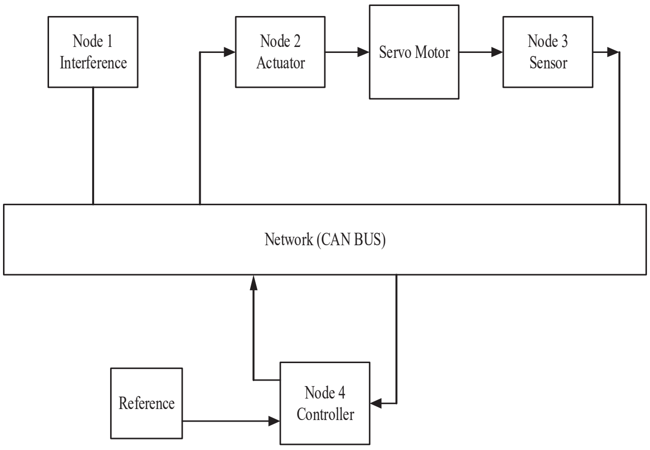


Fig. 5 Simulink Model of Networked Control System

VI. SIMULATION RESULTS

The simulation is carried out on this network specifications and simulation parameters. They are as follows.

- The network: CAN BUS
- Data Rate (bits/s) = 80000
- Minimum frame size (bits) = 40
- Loss probability = 0
- Sampling interval h = 10 m sec
- In this paper three cases are elaborated.
- Network is used with ideal conditions,
- Output response under different time delays.

The entire simulation is carried out using single-rate NCS means sensor, controller and actuator are working at same sampling interval. As discussed above PD controller consists of two tuning parameters Kp and Kd. Simulation are tested with four cases of PD parameter as shown in Table II. By proper tuning, the values of parameters came out to be Kp = 0.31 and Ti = 0.55. Figs. 6, 9 show the comparison of servo motor versus time and reference input signal versus time under different controller parameters. The simulation results of various controller parameters according to Table III. From these results, case 3 is less overshoot than case1 and case2. Fig. 9 shows the simulation result for case 4 parameter. In this case4, the output is underdamp case. So, it is the worst results in the simulation cases. And the simulation results of case 3 parameter get the best result as shown in Fig. 3. In four cases, the case 3 result is more suitable than any others. By proper tuning, the values of PD controller parameters came out to be Kp = 1.5 and Kd = 0. 035.So this value is selected and the

Open Science Index, Electronics and Communication Engineering Vol:10, No:3, 2016 publications.waset.org/10003729.pdf

second delay task which is between sensor and controller is studied. From this simulation works, the delays in the network can be found by change the data speed or the baud rate of the CAN bus. Therefore, the result and analysis can be studied in four Cases as shown in Table II. The following figures show Time versus Response of system.

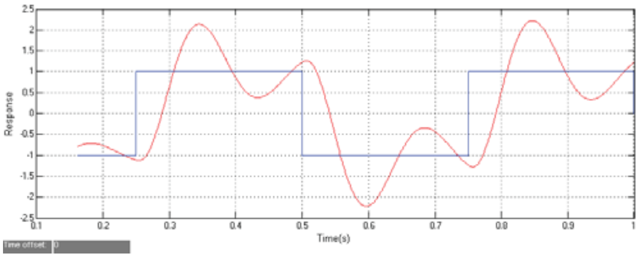


Fig. 6 The Simulation Result with Case 1 PD parameter

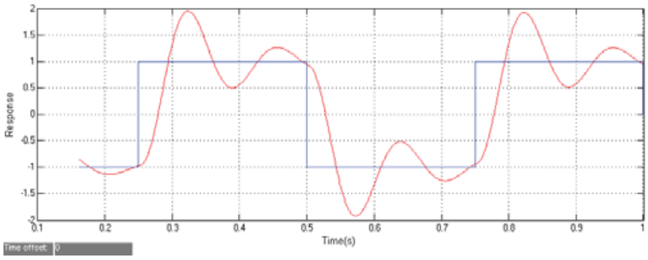


Fig. 7 The Simulation Result with Case 2 PD parameter

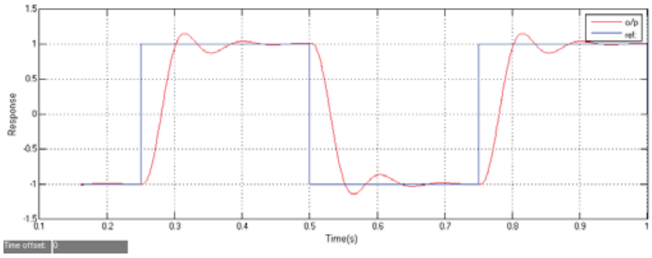


Fig. 8 The Simulation Result with Case 3 PD parameter

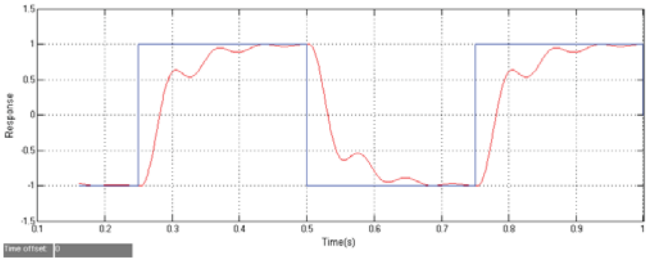


Fig. 9 The Simulation Result with Case 4 PD parameter

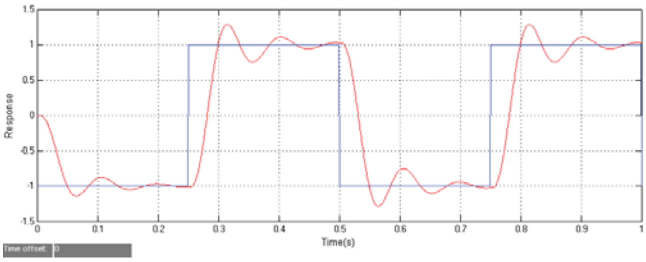


Fig. 10 Case A: Sensor-Controller Delay Simulation Result

Figs. 10, 13 represent the relationship of servo motor versus time and reference signal versus time under different delay conditions. The delay parameter is described in Table II. Fig. 10 shows the simulation of result with 0.0016 delay. From this figure, the delay causes more overshoot than response compare with result for case 3 in Fig. 8. Data speed of CAN does not change but delay is changed in case B 0.0024 value. The result is more overshoot compare with case A result as shown in Fig. 11. It is about 1.5. When CAN data speed is reduced to 40kbps and delay value is changed to 0.0032, it can be observed the response of servo motor become degrade due to longer sensor-controller delay as shown in Fig. 12. When the delay value is altered as case D, the response of DC motor is fulgurated in Fig. 13.

TABLE II
DELAY VALUE OF SENSOR TO CONTROLLER

Case	Node	Speed (kps)	Sensor-controller delay
A	Sensor	80	0.0016
B	Sensor	80	0.0024
C	Sensor	40	0.0032
D	Sensor	40	0.0048

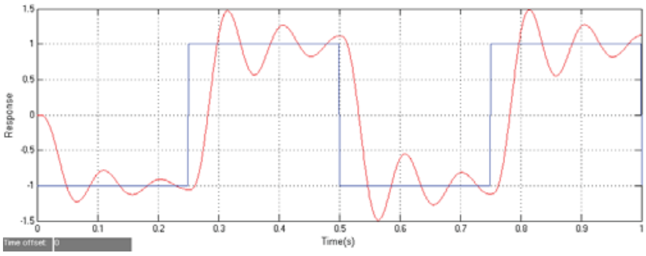


Fig. 11 CaseB: Sensor-Controller Delay Simulation Result

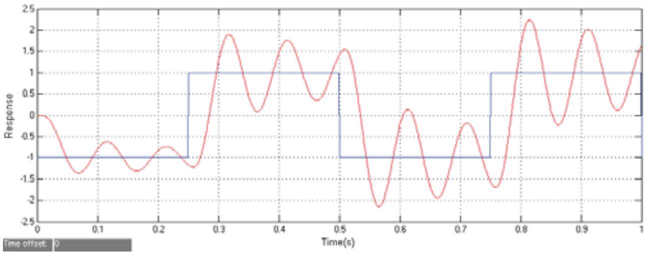


Fig. 12 Case C: Sensor-Controller Delay Simulation Result

Open Science Index, Electronics and Communication Engineering Vol:10, No:3, 2016 publications.waset.org/10003729.pdf

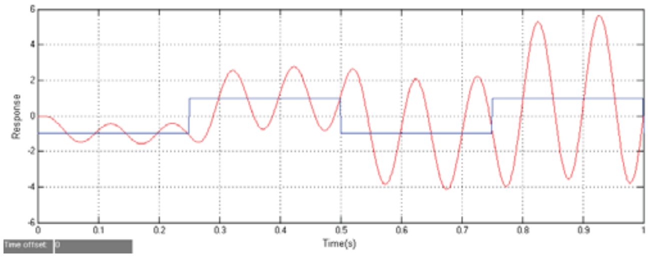


Fig. 13 Case D: Sensor-Controller Delay Simulation Result

Under the delay conditions the response of servo motor is controlled 0.0016 delay with less peak overshoot and nominal Offset as shown in Fig. 10. From the results it can be seen that as delay changes the settling time and peak overshoot also increases. If we considered Figs. 12, 13 in which control delay 0.0032 and 0.0048, the output response of the system as well as control input gets degraded. Thus it is observed that as the time delay increases the output response degrades accordingly. Comparison of response of servo motor under different delay conditions with CAN BUS as a networked medium is shown in Table III. Comparison is done in terms of time domain specifications i.e. rise time, peak overshoot, and settling time. All these times pacifications are calculated from the graph.

TABLE III
COMPARISON OF RESULTS FOR PD CONTROLLER PARAMETER AND DIFFERENT DELAY VALUES

case	PD parameters				Delay (sensor-controller			
	1	2	3	4	A	B	C	D
rise time	0.28s	0.23s	0.23s	0.24s	0.25s	0.27s	0.29s	ud
overshoot	2.3	2	1.3	ud	1.3	1.5	2	ud
setting time	0.48	0.42	0.38	ud	0.39	0.39	ud	ud
performance	bad	good	better	bad	better	good	bad	worse

Ud=undefined

VII. CONCLUSION

The sensor-controller delay can be studied from the simulation. Based on the results, analysis the effect of delay to performance of the control system. The result from Case A with 80kbps data speed have a constant time delay rather than Case 2 with 80kbps data speed resulting longer time delay. While for Case D, with the data speed at 40kbps and delay that is longer than case B. It is concluded that delay in CAN network is influenced by CAN transmission speed and performance.

REFERENCES

[1] Y. Tipsuwan and M.Y. Chow, “Control methodologies in networked control systems,” *Control Eng. Practice*, vol.11, no.10, pp.1099–1111, 2003.

[2] J.R. Moyne and D.M. Tilbury, “The emergence of industrial control networks for manufacturing control, diagnostics, and safety data,” *Proc. IEEE*, vol.95, no.1, pp.29–47, 2007.

[3] L. Schenato, B. Sinopoli, M. Franceschetti, K. Poolla and S.S. Sastry, “Foundations of control and estimation over lossy networks,” *Proc. IEEE*, vol.95, no.1, pp.163–187, 2007.

[4] W.-A. Zhang and L. Yu, “Output feedback stabilization of networked control systems with packet dropouts,” *IEEE Trans. Automatic Control*, vol.52, no.9, pp.1705–1710, 2007.

[5] H. Ishii, “*H∞* control with limited communication and message losses,” *Systems and Control Letters*, vol.57, no.4, pp.322–331, 2008.

[6] WEI Ling, XUE Ding-yu, E Da-zhi "Some Basic Issues in Networked Control Systems", Second IEEE Conference on Industrial Electronics and Applications, 2007, pp. 2098-2102.

[7] I. A. Sheikh, B. Bandyopadhyay and A. J. Mehta "A Brief Survey on Various Issues of Networked Control System", 1st International Conference on Current Trends in Technology, Nirma University Ahmedabad, 2010.

[8] T.C. Yang, “Networked control system: a brief survey”, IEE Proc-Control Theory Appl., Vol, 153, No.4, July 2006.

[9] W. Zhang, M.S. Branicky and S.M. Phillips, “Stability of networked control systems,” *IEEE Control System Magazine*, vol.21, no.1, pp.84–99, 2001.

[10] L.A. Montestruque and P.J. Antsaklis, “Stability of model-based networked control systems with time-varying transmission times,” *IEEE Trans. Automatic Control*, vol.49, no.9, pp.1562–1572, 2004.

[11] P.V. Zhivoglyadov and R.H. Middleton, “Networked control design for linear systems,” *Automatica*, vol.39, no.4, pp.743–750, 2003.

[12] J. Nilsson, B. Bernhardsson and B. Wittenmark, “Stochastic analysis and control of real-time systems with random time delays,” *Automatica*, vol.34, no.1, pp.57–64, 1998.

[13] L. Zhang, Y. Shi, T. Chen and B. Huang, “A new method for stabilization of networked control systems with random delays,” *IEEE Trans. Automatic Control*, vol.50, no.8, pp.1177–1181, 2005.

[14] Sumit Vardhan, Rajiv Kumar, "Simulations for Time delay Compensation in Networked Control Systems", Journal of Selected Areas in Telecommunications, 2011, pp. 38-43.

[15] Chen, H.; Tian, J. *Research on the Controller Area Network*. International Conference on Networking and Digital Society, Vol. 2, pp.251 - 254, 2009.

[16] Hu Shousong, Zhu Qixin "Stochastic optimal control and analysis of stability of networked control system with long delay", Automatica, 2003, 39(11), pp. 1877-1884.

[17] R.C.Dorf and R.H.Bishop "Modern Control Systems", Pearson Prentice Hall, Upper Saddle River, 11th edition, NJ, 2008.

[18] Za'er S. Abo-Hammour, Othman M.K. Alsmadi, Adnan M. Al-Smadi, Maha I. Zaqout and Mohammad S. Saraireh "ARMA model order and parameter estimation using genetic algorithms", Mathematical and Computer Modelling of Dynamical Systems, 2012, 18(2), pp. 201-221.

[19] Kannan M. Moudgalya "Digital Control", John Wiley and Sons Ltd., England, 2007.