Asian Journal of Information Technology 15 (24): 5075-5083, 2016
ISSN: 1682-3915

Gain Scheduling of the PID Controller Time Delay System Using Hybrid Methodology for Network Control System

[1]Isra'a L. Salim, [1]Osama Ali Awad and [2]C.T. Kalaivani
[1]College of Information Engineering, Al-Nahrain University, Baghdad, Iraq
[2]Saveetha Engineering College, Chennai, India

Abstract: Networked Control System (NCS) has recently been a hotspot in the research fields of control theory and control engineering application where control system components (controller and controlled system or plant) are connected via a shared network. Taking a third-order DC motor as the plant and using the true time toolbox in MATLAB, a NCS Model is designed with Ethernet network as the communication channel. Through the simulation, different traffic scenarios on Ethernet network are studied. Different load conditions induce time-varying delays between measurements and control. To face these variations a delay-dependent gain scheduling Proportional-Integral-Derivative (PID) controller has been designed a hybrid technique for gain scheduling of the PID controller time delay system. Based on the training dataset the AI technique will be trained and the gain parameters of the PID controller will be scheduled. By using the proposed technique, the PID controller will be tuned and the stability of the system will be maintained. The ABC technique is one of the non linear programming techniques which can constructed that approximately fits the data from the available solution space. So, the fit poles, the optimal controller gains and parameters will be determined to the system. The proposed hybrid technique is implemented in MATLAB working platform and the tuning performance is evaluated.

Key words: Ethernet, gain scheduling, NCS, PID controller, true time toolbox

INTRODUCTION

A Networked Control System (NCS) is one type of distributed control system is defined as a control system where the control loops are closed via a network, this network is shared with other nodes outside the control system. NCS has many advantages comparing with traditional control systems: improve the flexibility, simpler, efficient, quick and easy maintenance and systems are remotely controlled. Thus, NCS reduces installation and maintenance costs However, there are many problems along with NCS itself because of the uncertainties in network communication. The obvious problem is the network-induced time delay which has a great impact on the stability and the performance of the NCS. Moreover, due to the uncertainty of the network transmission and the restriction of communication bandwidth and data packet, there exist some problems such as data packet dropout. Regardless of the type of the network, time delay naturally brings negative effect on NCS stability and performance and even results in unstable control system. The delayed packets may be more harmful to system stability than packet loss. There are three types of time delay in NCS:

- Processing time delay (τ_p)
- Time delay from sensor to controller (τ_{SC})
- Time delay from controller to actuator (τ_{CA})

Hence, the total time delay in NCS can categorize by the following shown in Fig. 1:

$$RTT = \tau_{SC} + \tau_{CA} + \tau_p \tag{1}$$

Where, RTT is the total time delay to send the sensor packet from sensor node, until the controller packet has received at the actuator node. The problem under focus in this RESEARCH is the loss of Quality of Performance (QoP) arising when conventional controllers (particularly: Proportional Integral Derivative (PID)) are implemented in NCS subject to time-varying delays between measurements and actuation. A delay-dependent Gain

Corresponding Author: Isra'a L. Salim, College of Information Engineering, Al-Nahrain University, Baghdad, Iraq

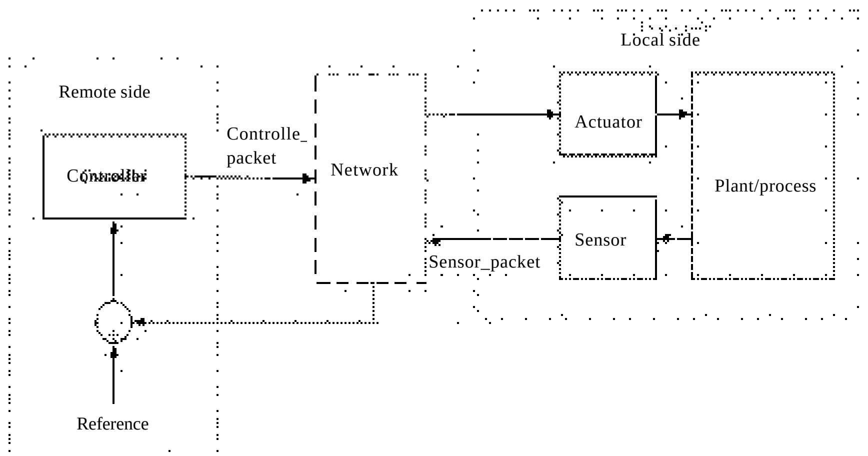

Fig. 1: General structure of NCS

Scheduling PID (GSPID) controller structure is to be designed and compared with the conventional PID controller based on a specific performance analysis criterion.

In order to apply delay-dependent controller methodologies, the time delay measurement must be carried out for each sample. One option is to synchronize nodes. It's known that high accuracy in clock synchronization between network nodes will lead to high network bandwidth consumption, i.e., effecting Quality of Service (QoS) of network and consequently resulting in more time delay because the network channel not idle most of the time.

In different regions, Time-Delay Systems (TDS) come across with engineering, biology and economics (Farkh *et al*., 2009; Park *et al*., 2010). A time delay is a foundation of unsteadiness and fluctuations in a system (He *et al*., 2004). Two kinds of time delay systems are there retarded and neutral (Wu *et al*., 2004). In TDS, where time-delays present between the functions of input to the system and their resulting outcome, can be indicated by Delay Differential Equations (DDEs) (Sipahi *et al*., 2007). Systems with delays indicate a class in limitless size mostly applied for the modeling and the study of transport and propagation phenomena (Tsang *et al*., 2005). Time-delays in control loops commonly mortify system presentation and confuse the study and plan of feedback controllers (Agnihotri and Waghmare, 2014).

The steadiness of time-delay systems is a setback of recurrent interest due to a number of functions of communication networks in biology and population dynamics (Papachristodoulou *et al*., 2007). Normally, steadiness study of time-delay systems can be categorized into two types. One is the delay-dependent stability study which comprises the data on the size of the delay and one more is the delay-independent stability study (Zhang *et al*., 2001; Moon *et al*., 1999). The delay independent stabilization offers a controller which can soothe the system irrespective of the size of the delay (Singh and Ouyang, 2013). On the other hand, the delay dependent stabilizing controller is worried with the size of the delay and regularly presents the upper bound of the delay (Fridman and Shaked, 2002).

Dead time is one of the influenced the presentation and steadiness of time-delay systems (Hamed and Issa, 2011). Time delay ceaselessly subsists in the measurement loop or control loop, hence it is more hard to control this type of process (Shekher *et al*., 2012). In order to enhance the control presentation, a few novel control technologies, like predictive control, the neural type of an artificial neutral delay in a control loop (Hamidian and Jalali, 2011). The PID controller is applied in a broad collection of problems like automotive, instrumentation, motor drives etc (Hagiwara *et al*., 2009). This controller presents feedback, it has the capacity to remove steady-state offsets via connected action and it can wait for the future by a derived action. As of structural plainness and capacity to work out a lot of practical control problems of the system (Liu *et al*., 2007). PID controller offers robust and reliable presentation for most of the systems if the PID parameters are adjusted suitably. A number of delay-independent adequate situations for the asymptotic steadiness of neutral delay differential systems have been applied (Chang *et al*., 2011).

The long-established PID compensator was applied for the tracking and stabilization of the loops design system. Conventional PID controllers are not appropriate for nonlinear systems and higher-ordered and time-delayed systems (Rao *et al*., 2010). The presentations of the PID controllers awfully depend on the precise knowledge of the system time delay. The stabilizing PID controller can reduce the computation time and it keeps away from the time-consuming stability inspection (Shamsudin *et al*., 2014).

Literature review: Numerous related works are already available in literature which based PID controller gain scheduling. Some of them reviewed here.

Le *et al.* (2015) have proposed the problem of determining the stabilizing controller gain and plant delay ranges for a general delay system in feedback configuration. Such a problem admits no analytical solutions in general. Instead, the condition of the loop Nyquist plot's intersection with the critical point was employed to graphically determine stability boundaries in the gain-delay space and stability of regions divided by these boundaries is decided with the help of a new perturbation analysis of delay on change of closed-loop unstable poles. As a result, all the stable regions are obtained and each stable region captures the full information on the stabilizing gain intervals versus any delay of the process. Their method was applicable to both stable and unstable processes of any order with or without the right-half plane zeros.

Anwar and Pan (2015) have presented a PID controller design method for the integrating processes based on frequency response matching. Two approaches were proposed for the controller design. In the first approach, a double feedback loop configuration was considered where the inner loop was designed with a stabilizing gain. In the outer loop, the parameters of the PID controller are obtained by frequency response matching between the closed-loop system with the PID controller and a reference model with desired specifications. In the second approach, the design was directly carried out considering a desired load-disturbance rejection model of the system. In both the approaches, two low frequency points are considered for matching the frequency response which yield linear algebraic equations, solution of which gives the controller parameters.

Vesely and Ilka (2015) have developed a novel approach to robust gain-scheduled controller design. Their design procedure was based on the robust stability condition developed for an uncertain LPV system model. The feasible design procedures were obtained in the form of BMI or LMI.

Lee *et al.* (2014) have discussed about a gain tuning method for variable PID controllers. First, the equivalence relationship between a discrete PID control and a discrete back stepping control with time-delay estimation and nonlinear damping was clarified and the variable gains of the PID controller were automatically tuned with a nonlinear damping component. The nonlinear damping terms directly affect system performance and make PID gains vary.

Ngo and Shin (2015) have proposed a new technique to calculate the dynamic gains of nonlinear systems represented by Fuzzy Basis Function Network (FBFN) models. The dynamic gain of an FBFN could be approximated by finding the maximum of norm values of the locally linearized systems or by solving a non-smooth optimal control problem. From the proposed gain calculation techniques, a novel Adaptive Multilevel Fuzzy Controller (AMLFC) with a maximum output scaling factor was presented. To guarantee the system stability, a stability condition was derived which only requires that the output scaling factor of the AMLFC be bounded.

Motivation for the research: The review of the recent research work shows that, the PID controller design has attracted considerable attention in recent years. Because of the fact that, better performances of the closed-loop can be achieved by using such kinds of controllers. In many industrial applications, the PID controller is used due to the simplicity of operation, design and low cost. But, the gain scheduling process of PID controller is one of the complexes for making stability system. Many PID gain scheduling methods have been developed but the Ziegler Nichols method is one of the classical PID gain scheduling technique. But, the Ziegler-Nichols method is still extensively used for determining the parameters of PID controllers. Also, construction of a mathematical model design becomes sometimes costly and these scheduling rules often do not give appropriate values because of modeling errors. The linear programming method is effectively for scheduling the PID gains. But, it is an iterative feedback scheduling; so, the computational time and the complexity of the gain scheduling are high. Hence, the linear programming method is suitable for specific case of gain scheduling that is linear objective function. In literature very few works are presented to solve this problem and the presented works are ineffective. These draw backs have motivated to do this research work.

MATERIALS AND METHODS

Here, I have intended to propose a hybrid technique for gain scheduling of the PID controller time delay system. The proposed hybrid technique will be the

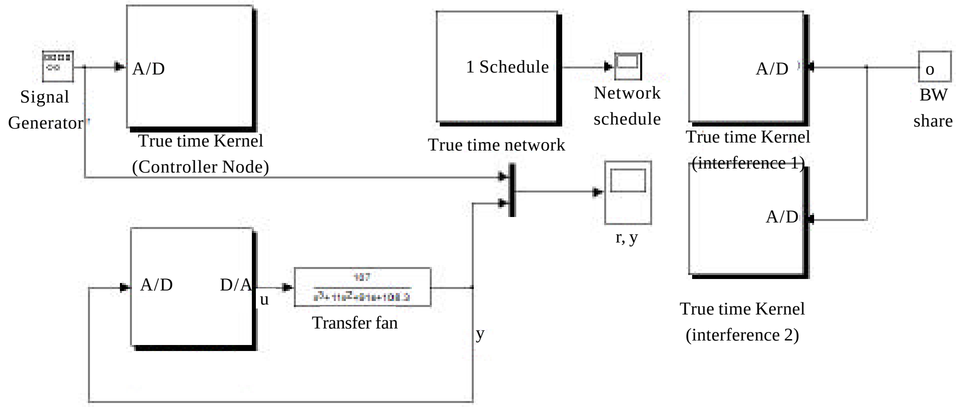

Fig. 2: The designed NCS model represented with truetime and MATLAB

combination of the Artificial Bee Colony (ABC) algorithm and Artificial Intelligence (AI) technique. The ABC algorithm will be optimized the gain parameters and generates the training dataset based on the variation of the output response of the system. Here, the ABC utilizes least square error is the objective function. Based on the training dataset the AI technique will be trained and the gain parameters of the PID controller will be scheduled. By using the proposed technique, the PID controller will be tuned and the stability of the system will be maintained. The ABC technique is one of the non linear programming techniques which can constructed that approximately fits the data from the available solution space. So, the fit poles, the optimal controller gains and parameters is determined to the system. The proposed hybrid technique is implemented in MATLAB working platform and performance is evaluated.

The designed NCS Model and controllers: For designing embedded control system and NCS model, the true time kernel and true time network blocks are used with the help of other MATLAB blocks. With the decreasing price, high data rate, widely use, increasing software and applications and well-established infrastructure, Ethernet network has been widely used in industrial for controlling applications (Fig. 2).

The local side consists of a true time kernel (Sensor/Actuator node) connected directly with the plant via A/D and D/A ports. This kernel implement two tasks: sensor and actuator tasks, both share the same local memory storage. The sensor task is a time-base mode of operation, i.e., sensor will sense the plantevery 0.1 sec directly from A/D input port and storing the time of sensing, named as sensingtime, along with a packet Identifier (ID) for this sensor_packet. This sensor_packetis designed to contain: packet ID, plant measurement and the pervious measured RTT. When the preparation of sensor_packetis done, the sensor will send it to the controller node over network with 72 byte packet size.The 0.1 sec sampling time is chosen for the reasons: best for both QoP of control and QoS of network, the process will be sensed about 15 times during its rise time (i.e., rise time/sampling time = 15).

The actuator task is responsible of doing D/A conversion of control signal to actuate the plant with the new received value. The actuator task is an event-based mode of operation, i.e., an interrupt is set to listen to network interface for any receiving controller_packet. When the actuator task is triggered, it will capturing the time of receiving controller_packet, named responsetime and examine the controller_packet ID. Actuator task will search the localmemory for matching IDs. When is found, the associated sensingtime is retrieved and its field is cleared. Then the calculation of a new RTT value is done by:

RTT = response time-sensing time (sec)

The remote side consists of a TrueTime Kernel (Controller node) connected directly with a reference or set point input via A/D port. The reference input is chosen as a square wave form with frequency equals to 0.05 Hertez. The scheduling policy for this kernel is chosen as prioFP. This kernel implement the controller task which is either implement the PID controller algorithm or the proposed GSPID controller algorithm. The controller task is event-based mode of operation, i.e., an interrupt is set to listen to network interface for any receiving sensor_packet. When the controller task is triggered, it will do the following steps sequentially:

- Reading or pulling the sensor_packet from network interface and reading reference value from A/D input port

- Examine the packet ID and the RTT value
- Implement the desired controller algorithm to produce control signal

Preparing the controller_packet. This packet is designed to contain: packet ID which is the same received sensor_packet ID and the control signal. Sending the controller_packet to the actuator node over network with 72 byte packet length

Because the controller node will receive an old value of RTT, the controller will use a forecasting technique for predication of the next RTT value, the Forecasted RTT (FRTT). The Moving Average (MA) is one type of the time series models [MA2], it takes the average of a defined window size as a forecast. The simplest form of MA, for example forecasting the RTT, can be represented by the following equation:

$$RTT_{t+1} = \left(RTT_t + RTT_{t-1} + RTT_{t-2} + RTT_{t-ws-1} / ^{ws}\right) \quad (1)$$

In order to generate load on the Ethernet network, Two TrueTime Kernel are used to send packets over network with different and hence a time-varying delay will induced in the communication channel. Four parameters define the operation mode for each interference node:

- maxP: the maximum packet size
- Bwshare: a fraction of the occupied bandwidth by each node
- w: weighting index for the parameter BWshare, w is random number between 0 and 1
- nodes: Number of computer nodes emulated by the interference node, each interference node represents 249 computer node so the total nodes in this network will be 500 (including control system nodes)

In this way, each interference node will send a packet with size maxP×BWshare×w and repeated for nodes time. Obviously, when a high value of BWshare is used, high network traffic will be generated and vice versa.

PID controller: PID controllers are commonly used, over 95% of the controllers in industrial applications are PID controller (or some form of PID controller like P, PI or PD controller) (Christiansen *et al*., 2005). Large factory may have thousands of them, in instruments and laboratory equipment. The PID con troller is delay independent and therefore the MA technique is not taken into considering while designing a PID controller. Because PID controller is implemented as a computer-based, the continuous time PID controller has to be approximated by a discrete time PID controller described by Karl and Richard, 2010):

$$u(t_k) = P(t_k) + I(t_k) + D(t_k)$$

$$P(t_k) - Kpe(t_k) \quad (2)$$

$$I(t_{k+1}) - 1(t_k) - K_i T_{Se}(t_k)$$

$$D(t_k) = -K_d \frac{y(t_k) - y(t_k - 1)}{T_s}$$

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

Where:

$P(t_k)$ = The proportional term, the integral term
$I(t_k)$ = The derivative term
T^k = Denotes the sampling instants
t_k-1 = The previous state
T_s = The sampling period
$u(t_k)$ = The control signal output
$e(t_k)$ = The error
$r(t_k)$ = The reference
$y(t_k)$ = The received measurement

The PID tunable gains are: proportional gain K^p, integral gain Ki and derivative gain Kd.

GSPID controller: Gain scheduling controller is one of the most simple and popular nonlinear controller techniques. The GSPID controller can be viewed as a multi-structure PID controller which the PID controller switch its parameters based on the scheduling variable. The scheduling variable for NCS is usually the time delay of the network, where the time delay is continuously changed based on the current data traffic in the network. To design a GSPID controller the following steps are considered:

- Dividing the network time delays into groups at selected operating points from 1-n
- A separate PID controller is designed for each operating point
- The final control action is computed from these local PID controllers as given below:

$$u(t_k) = \sum_1^n K_n P_n(t_k) + I_N(t_k) + D_n(t_k) \quad (3)$$

where, Kn is the scheduling variable which is defined by:

$$K_n = \begin{cases} 1 & \tau n \le \tau c < \tau n+1 \\ 0 & \text{otherwise} \end{cases}$$

where, τ_n is the network time delay. τ_c is te current or the actual network time delay Simply, when the current time delay is belong to a specific group, switch to the PID controller for that group. During online operation, the GSPID controller make an automatic switching between these stored local PID controllers. In this way, to implement the GSPID controller within a computer-based control system, a look-up table is used to stores and recalls these PID controllers.

RESULTS AND DISCUSSION

By altering the BWshare value, different load scenarios can be obtained. Three scenarios for Ethernet network are studied and RTT is calculated online inside Sensor/Actuator node based on Eq. 3:

Unloaded network scenario: Te control system nodes (Controller and Sensor/Actuator nodes) are the only active nodes in the NCS, in this case the BWshare value is set to zero and no packets will be send from interference1 and interference 2 nodes. The RTT plot and its histogram for this scenario with 200 sec simulation run are showed in Fg. 2-3, respectively.

Medium loaded network scenario: Te BWshare value is set to 0.4, the packet length send from interference1 and interference2 nodes will be (0.4×1518×w) byte. The RTT plot and its histogram for this scenario with 200 sec simulation run are showed in Fig. 5-6, respectively.

High loaded network scenario: The BWshare value is set to 0.9, the packet length send from interference1 and interference 2 nodes will be (0.9×1518×w) byte. The RTT plot and its histogram for this scenario with 200 sec simulation run are showed in Fig. 7-8, respectively.

The scenarios features can be summarized as follows: for unloaded Ethernet network, the time delay is almost constant at 5.6152 ms which is only the processing time at the controller node plus the transmission time to send a minimum length of an Ethernet packet (sensor_packet and controller_packet). In this case, Ethernet network is an ideal network for transferring control system packets with a very low time delay because of its high data rate. For medium loaded Ethernet network, except the sensor_packet and controller_packet there are other packets will be send over network and one or more nodes will try to transmit at the same time which results in transmission collision and the nodes will be backed off and retransmit later. This will results in time-varying delays. The RTT for medium load could be bounded as ≤0.1 sec with a possibility of random high spikes. The last scenario shows that RTT values for high loaded Ethernet network are random and could not be bounded, 1.6 sec as a maximum value. For comparison between PID and GSPID controllers, the PID controller of is first tested with these scenarios. For the plant in Eq. 2 PID gains are tuned automatically using pidtune MATLAB function without considering the network time delay in closed loop control system. The obtained PID

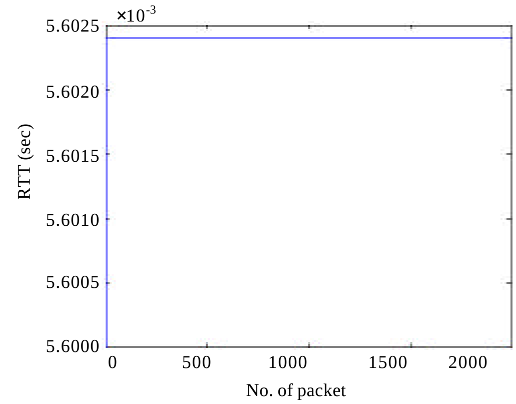

Fig. 3: RTT values for unloaded Ethernet network scenario

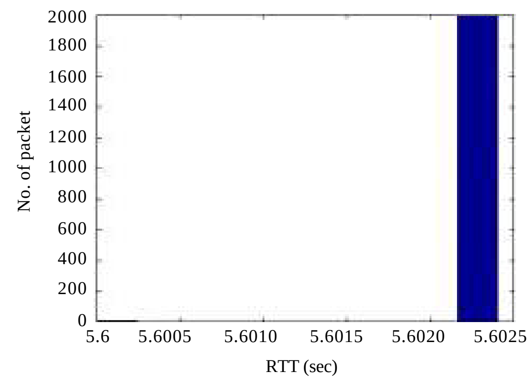

Fig. 4: RTT histogram for unloaded Ethernet network scenario

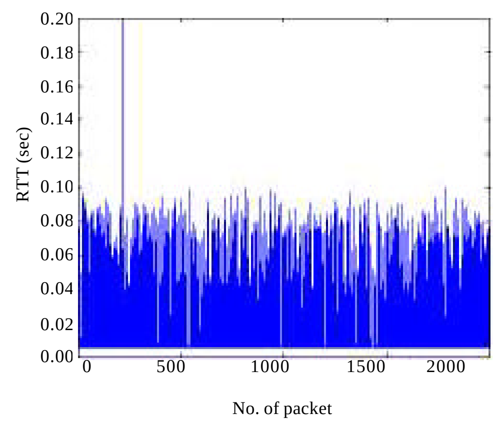

Fig. 5: RTT values for medium loaded Ethernet network scenario

gains: = 2.114, = 3.504 and = 0.083. The response of PID controller for 150 sec simulation run, along with control signal for unloaded (from 0-50 sec), medium loaded (from 50-100 sec) and high loaded (from 100-150 sec) network traffic scenarios with respect to the square wave form input reference is shown in Fig. 9. Because GSPID controller is delay-dependent, the MA technique is implemented inside controller node for calculating the FRTT. From the graphical analysis of the RTT values for different traffic scenarios for Ethernet network, it is found that spikes and high values of RTT does not last for more than five samples. The MA model of GSPID controller with equal to five is used here for modeling RTT time series. In order to implement the gain scheduling process, the FRTT values are divided into six groups:

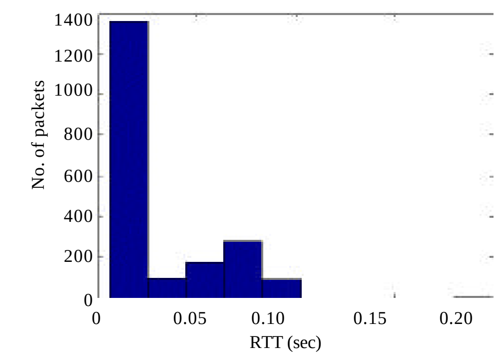

Fig. 6: RTT histogram for medium loaded Ethernet network scenario

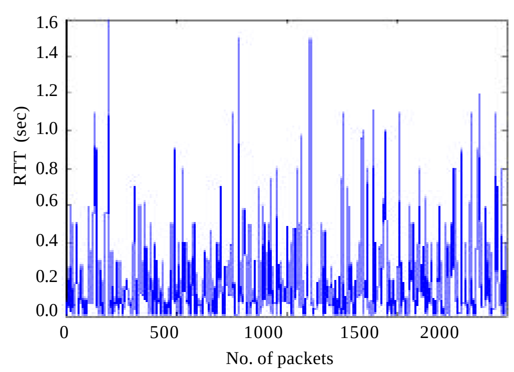

Fig. 7: RTT values for high loaded Ethernet network scenario

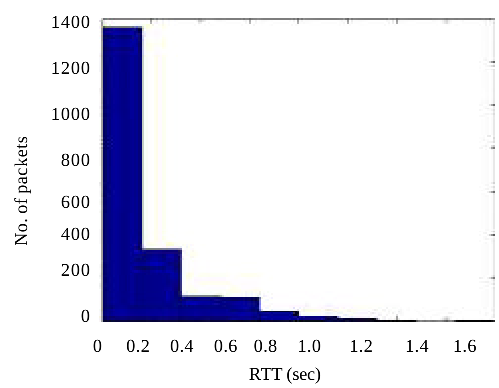

Fig. 8: RTT histogram for high loaded Ethernet network scenario

- Group one: (0.005 = FRTT<0.1)
- Group two: (0.1= FRTT<0.2)
- Group three: (0.2 = FRTT<0.3)
- Group four: (0.3 = FRTT<0.4)
- Group five: (0.4 = FRTT<0.5)
- Group six: (0.5 = FRTT<0.6 and more)

The time-varying delays are assumed to be constant or have a small variations within each group. Table 1-2 shows the PID gains (automatically obtained using pidtune MATLAB function) for the median value of each group of the FRTT.

The response of GSPID controller for 150 sec simulation run, along with control signal for unloaded (from 0-50 sec), medium loaded (from 50-100 sec) and high loaded (from 100-150 sec) network traffic scenarios with respect to the square wave form input reference is shown in Fig. 10.

Generally the goal of any controller algorithm is simply achieving good reference tracking, thus the analysis of controller performance could depends on the observation of the plant measurement with respect to the

Table 1: Setup of network parameters

Items	Network parameter	Values
1	Type of network	Ethernet
2	Data rate	10 Mbps
4	Minimum packet size	72 byte
5	Maximum packet size	1518 byte

Table 2: PID gains for the six FRTT groups

	PID gains		
FRTT value	K_P	K_I	K_D
0.05	1.211	2.010	0
0.15	1.174	1.468	-0.081
0.25	0.843	1.185	0
0.35	0.861	0.904	-0.201
0.45	0.319	0.644	0
0.55	0.181	0.501	0

Table 3: Performance analysis criteria for PID and GSPID controllers

Criteria/controller type	PID	GSPID
Integral time absolute error	4600	1500
Maximum overshoot	4.266	1.574
Maximum undershoot	-5.537	-1.811

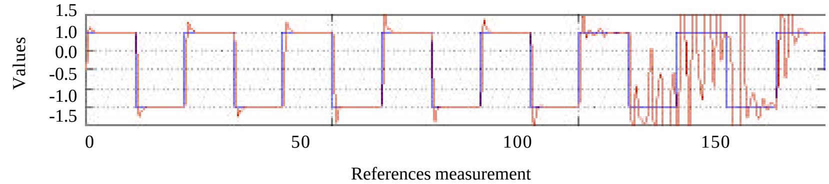

Fig. 9: PID controller response

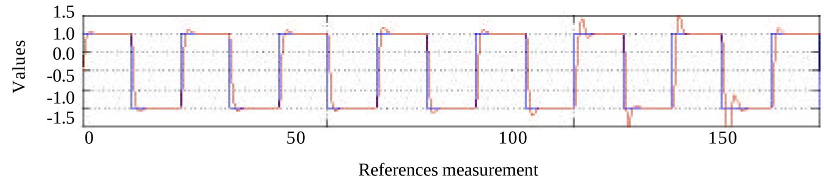

Fig. 10: GSPID controller response

input reference. Besides this criterion, many other performance analysis criteria are chosen to test the controllers performance.

Integral time absolute error: The accumulative sum of the absolute value of error multiplied by time over time

Maximum overshoot: The maximum positive value of the system response.

Maximum undershoot: The maximum negative value of the system response. The performance analysis criteria for PID and GSPID controllers are summarized in Table 3.

As shown the PID controller has high values of IAE, MO and MU. With PID controller the stability cannot be guaranteed. While GSPID controller significantly improves the closed loop system response with significantly decreases IAE, MO and MU and guarantee stability.

CONCLUSION

In this study, a NCS model is built using TrueTime toolbox and ordinary MATLAB blocks. The network setup considers a remote side (Controller node) and a local side (Sensor/Actuator node connected with the plant). The ethernet network is chosen as communication channel, the network is shared with other nodes that sending packets with randomlyvarying delays. The simulation shows the results for different ethernet load scenarios. As a conclusion for the designed NCS, the use of intelligent components in control systems can significantly improves the performance. Since, the modern controllers are implemented as a computer-based, the sensor, actuator and controller can be implemented as tasks in real time operating systems which can take advantage of the huge memory provided by the computer. Assuming that sensor and actuator of NCS are implemented within one node and both have a common local clock has the advantage to eliminate the need for synchronization between controller and sensor nodes. The calculation of RTT is done by the cooperation between sensor and actuator tasks. The delay-independent PID controller is no more suitable for NCS field. A delay-dependent GSPID controller structure is proposed.

REFERENCES

Agnihotri, S.P. and L.M. Waghmare, 2014. Regression model for tuning the PID controller with fractional order time delay system. Ain Shams Eng. J., 5: 1071-1081.

Anwar, M.N. and S. Pan, 2015. A frequency response model matching method for PID controller design for processes with dead-time. ISA. Trans., 55: 175-187.

Chang, P.K., J.M. Lin and K.T. Cho, 2011. Ziegler-nichols-based intelligent fuzzy PID controller design for antenna tracking system. Proceedings of the International Multiconference on Engineers and Computer Scientists, March 16-18, 2011, IAENG, Hong Kong isBN: 978-988-19251-2-1, pp: 1-6.

Christiansen, D., C. Alexander and R. Jurgen, 2005. Standard Handbook of Electronic Engineering. 5th Edn., McGraw Hill, New York, USA. isBN: 9780071500845, Pages: 2200.

Farkh, R., K. Laabidi and M. Ksouri, 2009. PI control for second order delay system with tuning parameter optimization. Int. J. Electr. Electron. Eng., 3: 1-7.

Fridman, E. and U. Shaked, 2002. An improved stabilization method for linear time-delay systems. IEEE. Trans. Autom. Control, 47: 1931-1937.

Hagiwara, T., K. Yamada, I. Murakami, Y. Ando and T. Sakanushi, 2009. A design method for robust stabilizing modified PID controllers for time-delay plants with uncertainty. Int. J. Innov. Comput. Inf. Control, 5: 3553-3563.

Hamed, B. and W. Issa, 2011. A modified internal model control for unstable-time delayed system. Int. J. Eng. Adv. Technol., 1: 56-62.

Hamidian, H. and A.A. Jalali, 2011. Calculation of PID controller parameters for unstable first order time delay systems. Int. J. Sci. Eng. Res., 2: 70-75.

He, Y., M. Wu, J.H. She and G.P. Liu, 2004. Parameter-dependent lyapunov functional for stability of time-delay systems with polytopic-type uncertainties. IEEE. Trans. Autom. Control, 49: 828-832.

Karl, J.A. and M.M. Richard, 2010. Feedback Systems: An Introduction for Scientists and Engineers. Princeton Univeristy Press, USA isBN: 978-0-691-13576-2, Pages: 387.

Le, B.N., Q.G. Wang and T.H. Lee, 2015. Development of D-decomposition method for computing stabilizing gain ranges for general delay systems. J. Proc. Control, 25: 94-104.

Lee, J.Y., M. Jin and P.H. Chang, 2014. Variable PID gain tuning method using backstepping c ontrol with time-delay estimation and nonlinear damping. IEEE. Trans. Ind. Electron., 61: 6975-6985.

Liu, X.G., M. Wu, R. Martin and M.L. Tang, 2007. Delay-dependent stability analysis for uncertain neutral systems with time-varying delays. Math. Comput. Simul., 75: 15-27.

Moon, Y.S., P. Park and W.H. Kwon, 1999. Robust stabilization of uncertain linear systems with time-delay. Inst. Control Autom. Syst. Eng., 1: 1144-1148.

Ngo, P.D. and Y.C. Shin, 2015. Gain estimation of nonlinear dynamic systems modeled by an FBFN and the maximum output scaling factor of a self-tuning PI fuzzy controller. Eng. Appl. Artif. Intell., 42: 1-15.

Papachristodoulou, A., M.M. Peet and S.I. Niculescu, 2007. Stability analysis of linear systems with time-varying delays: Delay uncertainty and quenching. Proceedings of the 2007 46th IEEE Conference on Decision and Control, December 12-14, 2007, IEEE, Gif-sur-Yvette, France isBN: 978-1-4244-1497-0, pp: 2117-2122.

Park, M., O.M. Kwon, J.H. Park and S. Lee, 2010. Delay-dependent stability criteria for linear time-delay system of neutral type. World Acad. Sci. Eng. Technol., 70: 1014-1018.

Rao, K.G., B.A. Reddy and P.D. Bhavani, 2010. Fuzzy Pi and integrating type fuzzy PID controllers of linear, nonlinear and time-delay systems. Int. J. Comput. Appl., 1: 144-264.

Shamsudin, M.A., R. Mamat, S.M. Ayob and S.W. Nawawi, 2014. Simplified partial state fuzzy-PID control on nonlinear wheel balancing robot. Int. J. Machanical Mechatronics Eng., 14: 65-73.

Shekher, V., P. Rai and O. Prakash, 2012. Comparison between classic PID, integer order PID and fuzzy logic controller for ceramic infrared heater: Analysis using MATLAB/Simulink. Int. J. Eng. Adv. Technol., 1: 222-227.

Singh, K.V. and H. Ouyang, 2013. Pole assignment using state feedback with time delay in friction-induced vibration problems. Acta Mech., 224: 645-656.

Sipahi, R., N. Olgac and D. Breda, 2007. Complete stability map of neutral type first order-two time delay systems. Proceedings of the 2007 Conference onAmerican Control, July 9-13, 2007, IEEE, Udine, Italy isBN: 1-4244-0988-8, pp: 4933-4938.

Tsang, K., A.B. Rad and W. Chan, 2005. Iterative feedback tuning for positive feedback time delay controller. Int. J. Control Autom. Syst., 3: 640-645.

Vesely, V. and A. Ilka, 2015. Design of robust gain-scheduled PI controllers. J. Franklin Inst., 352: 1476-1494.

Wu, M., Y. He and J.H. She, 2004. New delay-dependent stability criteria and stabilizing method for neutral systems. IEEE. Trans. Autom. Control, 49: 2266-2271.

Zhang, J., C.R. Knopse and P. Tsiotras, 2001. Stability of time-delay systems: Equivalence between Lyapunov and scaled small-gain conditions. IEEE. Trans. Autom. Control, 46: 482-486.