International Journal of the Korean Society of Precision Engineering Vol. 3, No. 3, July 2002.

Implementation of Networked Control System using a Profibus-DP Network

Kyung Chang Lee[1], and Suk Lee[2]

[1] School of Mechanical Engineering, Pusan National University, Pusan, South Korea
[2] School of Mechanical Engineering and Research Institute of Mechanical Technology, Pusan National University, Pusan, South Korea

ABSTRACT

As numerous sensors and actuators are used in many automated systems, various industrial networks are adopted for real-time distributed control. In order to take advantages of the networking, however, the network implementation should be carefully designed to satisfy real-time requirements considering network induced delays. This paper presents an implementation scheme of a networked control system via Profibus-DP network for real-time distributed control. More specifically, the effect of the network induced delay on the control performance is evaluated on a Profibus-DP testbed. Also, two conventional PID gain tuning methods are slightly modified for tuning controllers for the networked control system. With appropriate choices for gains, it is shown that the networked control system can perform almost as well as the traditional control system.

Keywords : Networked control system, Profibus-DP, Fieldbus, Network induced delay, Industrial network

1. Introduction

Due to extensive functionality and low cost, microprocessor are changing many industrial control systems into more intelligent digital systems. Especially, because of their excellent flexibility and adaptability, their applications are gaining acceptance in control and automation areas including process control, manufacturing automation, building automation, and machine tool control. As a system becomes more intelligent and flexible, the system requires more field devices such as sensors, actuators, and controllers that require some type of electrical connections because they are distributed over a certain area.

As the number of devices in the system grows, field devices need to exchange rapidly increasing amount of data among them. Conventionally, these devices are connected with point-to-point or direct connections where each piece of information is exchanged via at least one cable. This approach is not any more suitable for systems composed of many devices because the number of cables is increasing proportional to the square of the number of devices. In order to solve this problem, various serial communication networks have been designed and implemented to provide reliable and efficient communication paths for data exchange among the system component[1,2].

In general, data transmitted on an industrial network can be classified into two groups: real-time and non-real-time data. Non-real-time data do not have stringent time limits on their delays experienced during the data exchange. In contrast, real-time data have very strict time limits and the data value is diminished greatly as the delay grows larger. This real-time data can be further divided into periodic and asynchronous data depending on the periodic nature of the data generation. For example, the data for program download belongs to non-real-time data while digital control command and alarm signal is periodic and asynchronous real-time data, respectively. On many industrial networks, these data types are sharing a single network even though they have different requirements on communication. That is, the non-real-time data need assurance of delivery without error and duplication while the real-time data are concerned mostly on the time taken to reach the destination. Therefore, when building a distributed control and automation system, the designer must configure the network to satisfy the real-time requirements of data and design a remote controller that can withstand some delays in the network[3].

Such industrial networks are called fieldbuses that include Profibus, World FIP, Fieldbus Foundation, Controller Area Network (CAN), and LonWorks. Recently, International Electrotechnical Commission (IEC) announced IEC 61158 as its fieldbus standard for various applications including power plant, chemical industry and assembly line[4,5].

One of the most common applications of fieldbus is shown in figure 1. This type of system is often referred to as Networked Control System (NCS). As shown in the figure, the components of an NCS, i.e., actuators, sensors and controller are connected via a fieldbus, and all the information including reference input, plant output, and control command are exchanged through the fieldbus without any direct electrical connection. Recently, significant progress for NCS has been made by several researchers including Cavalieri(6) and Lee(7). Cavalieri et al. introduced the fully distributed control system, which focuses only on a performance evaluation of NCS. A research on plant control at Profibus-FMS is done by Lee et al., which focuses on control by adjustment of network performance parameter only.

This paper presents the structure of NCS by using a Profibus-DP network and investigates the cause of network induced delay. In addition, the performance of an NCS is compared to the traditional control system. This paper is organized into five sections including this introduction. Section 2 gives a brief summary of Profibus-DP. Section 3 presents the structure of the NCS and analyzes the cause of network induced delay. In section 4, the results of experimental evaluation of the NCS are described. Finally, conclusions are presented in section 5.

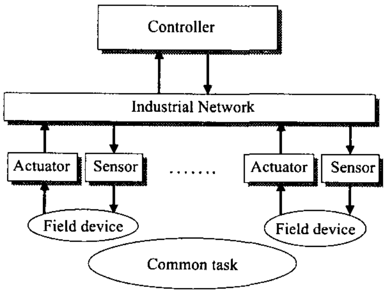


Fig. 1 A typical example of NCS

2. Profibus-DP Protocol

2.1 Overview of Profibus-DP Protocol

Figure 2 shows the architecture of Profibus-DP[8,9]. In order to satisfy the real-time requirement, it is designed to have only two layers, i.e., physical layer (PHY) and fieldbus data link layer (FDL), out of seven layers of the Open System Interconnection Reference Model (OSI RM). Profibus-DP uses polling method as its Medium Access Control (MAC) to avoid a large fluctuation of data latency. In addition, Profibus-DP has the fieldbus management layer 1 & 2 (FMA1/2) for network management, and the Direct Data Link Mapper (DDLM) for exchange between FDL and the user interface.

The physical layer is responsible for converting the data into transmission signals, propagating the signals to the receiver, and converting the signals back to the original data. Profibus-DP employs the bus topology and accommodates up to 32 nodes or stations in a segment. The maximum transmission speed is 12Mbps (mega bit per second). A station connected to Profibus-DP is classified either as a master station or as a slave station. The master station plays a role as a central controller that polls other slave stations during a pre-defined polling period. The slave station collects input signals from peripheral devices in order to send the information if

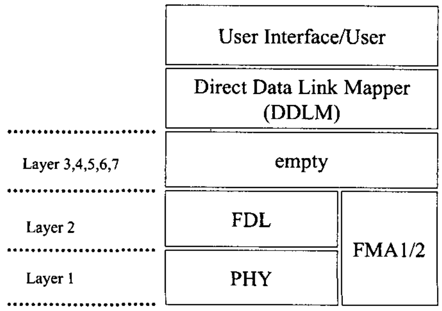


Fig. 2 Architecture of Profibus-DP protocol

requested by the master, and sends output signals to the connected devices according to the received data from the network. In general, the master station transmits data and requests to slave stations to transmit, while the slave station participates in communication only by request of the master station.

Figure 3 shows a generic communication model of Profibus-DP. A user interface where communication processes are running communicates with FDL via a dual port RAM (DPRAM). Each module of the user interface such as module 5 receives data from other stations by reading from the input buffer of the DPRAM and transmits data to other stations by writing into the output buffer. On the other hand, FDL receives a response frame from other stations and writes the frame into the input buffer of DPRAM, and then it reads a data from the output buffer and transmits to other stations. This connection via the DPRAM between application process and communication function makes Profibus-DP more suitable to a system that must satisfy the real-time requirement.

2.2 Performance of Profibus-DP

In order to evaluate the performance of Profibus-DP for networked control systems, a Profibus-DP testbed was established with several IBM-compatible PCs equipped with interface boards developed by Softing. As explained above, one PC is designated to be a master station while the others operate as slave stations.

Figure 4 shows the overall response time of Profibus-DP with varying number of slave stations. Here, the overall response time is defined as the difference between the time when a master station transmits the request frame to the first slave station and the time when the master station receives the response frame from the last slave station. In the experiment, the master transmits a two byte message for polling, and each slave responds with a two byte message. Here, message length is set to 2 bytes because it is sufficient to represent the control signal of a controller and the sensor data sent by the slaves. Also, the master station starts its polling cycle at every 10 ms (polling period of 10ms) that is fast enough for many networked control systems. In the figure, when the transmission speed of Profibus-DP is 1.5 Mbps (mega bit per second), and single slave station participates in communication, a overall response time is 215μ sec. When the number of slave stations increases, the overall response time increases linearly. Even when 12 slave stations participate in communication, the overall response time is below 3msec. The relation between the number of slaves and the overall response time is linear because there is no uncertainty such as collision of messages. When transmission speed of Profibus-DP is set to 12 Mbps and single slave participates in communication, the overall response time is 37μ sec. And, when 12 slaves participate in communication, the overall response time is below 1 msec. From these results, it can be said that Profibus-DP is efficient enough to support most real-time control systems.

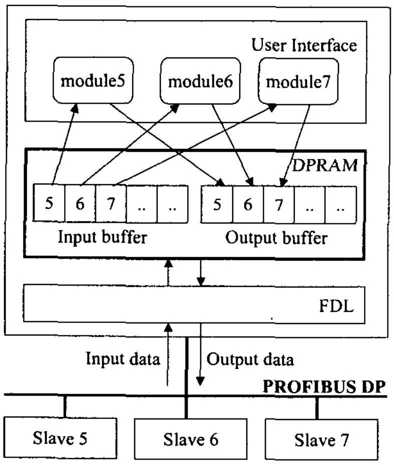


Fig. 3 Communication model of Profibus-DP

3. NCS via Profibus-DP

3.1 Structure of NCS

Figure 5 shows the structure of NCS based on a Profibus-DP network. In the figure, a master and slave

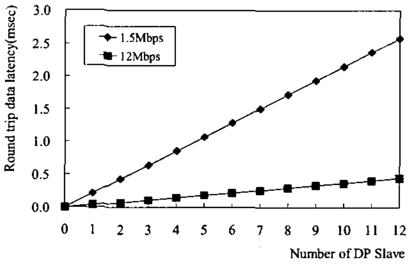


Fig. 4 Overall response time of Profibus-DP

stations consist of an input/output data process, a communication process, and an I/O instrumentation process. Communication is achieved by the communication cycle and instrumentation cycle in the NCS. In the communication cycle, the control signal from the master is sent to the slave station via DPRAMs and the communication processes. The plant output, i.e., the sensor value is sent from the slave to the master following the same path in the opposite direction. In the instrumentation cycle, the control signal is delivered to the plant while the plant output is received for its transmission by the I/O instrumentation process.

An operational procedure of the master station is as follows. As communication starts, the input/output data process using DP master service functions begins to operate for exchange of information with a slave station. The input data process reads a feedback signal from its DPRAM input buffer, and forwards to the output data process. And then, a control algorithm that is implemented in the output data process generates a control signal for the plant using the feedback signal, and writes to the DPRAM output buffer for transmission to the slave station.

At the same time, the input/output data process of the slave station begins to operate using DP slave service functions. The input data process reads a control signal from the DPRAM input buffer, and forwards to the output data process. The driver software within the output data process drives the plant following the control signal, and writes to the DPRAM output buffer in order to transmit feedback signal of the plant to the master.

3.2 Characteristics of network induced delay

As control and feedback signals are transmitted in the networked control system, some amount of network induced delay is inevitable because more than one stations are sharing the network. A signal must be delayed if there is another signal being transmitted on the network. This delay can be quite substantial and sometimes adversely affect the performance of the control system[10,11].

Figure 6 shows the transmission and reception time of each message in networked control system with Profibus-DP. Here, T_S is the communication period of the master station (equal to the sampling time of the networked control system), and T_P is the sum of processing time of the input/output process to read from or to write on to the DPRAM. The communication process takes the amount of time equal to T_T to prepare to send a message while it is assumed to take a negligible amount of time to receive one. Also, T_I is the processing time of the instrumentation process, and T_C is the time required

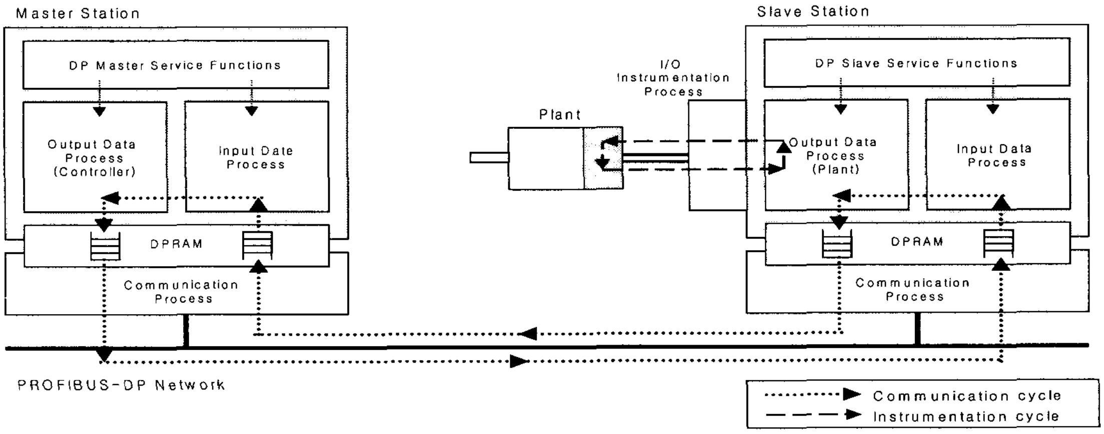
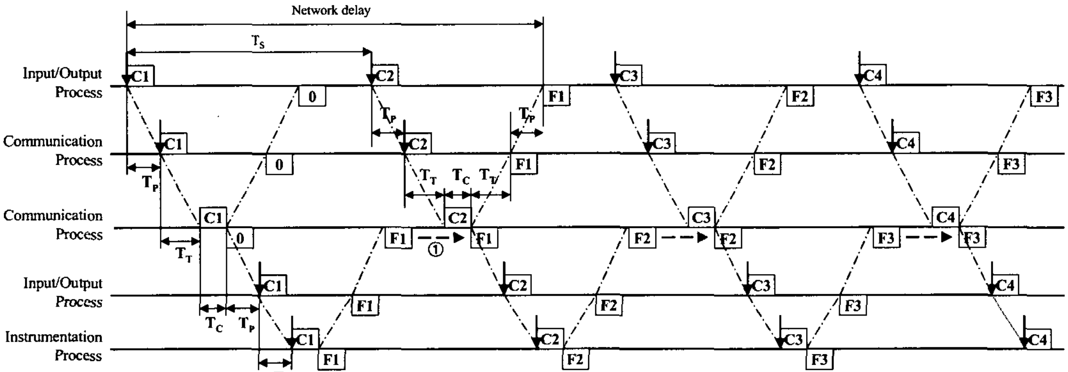


Fig. 5 Structure of application programs of remote feedback control system

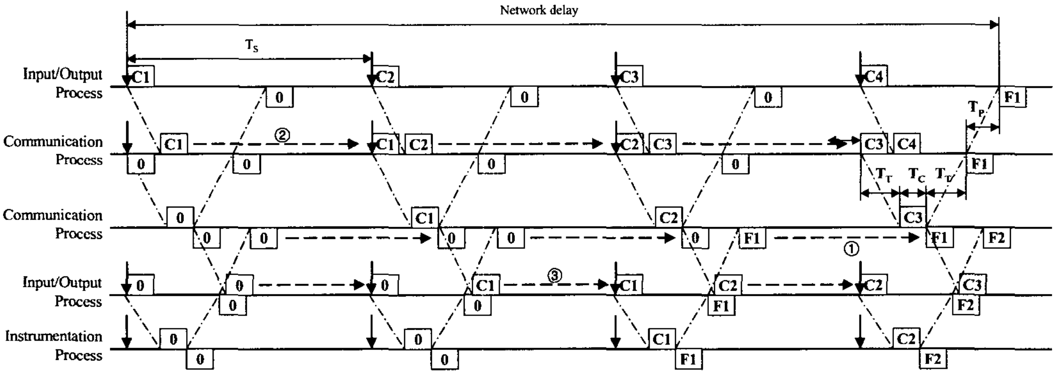
to transmit a message on the network. In the figure, the block above a horizontal line of each process is a frame that is transmitted from master to slave, and the block under the line is a frame that is transmitted in the reverse direction. Using the above definitions, the network induced delay can be explained as follows.

First, under a complete synchronization among the processes of a station, the network induced delay is caused by the processing time of each process and the polling mechanism of Profibus-DP as shown in Figure 6.a. The output data process of the master station generates a control signal (C1 frame) and finishes writing it on to DPRAM by TP, and then the communication process begins to transmit C1 after its processing in T_T. The communication process of the slave station receives C1 and writes C1 to DPRAM after the transmission time of T_C. And then, the communication process responds to the poll of the master station by transmitting a frame (initialization frame, 0) that was saved in the output buffer of slave station during initialization. And then, the input data process of the slave reads C1 by taking another T_P, and forwards to the instrumentation process. The instrumentation process drives the plant using the received control signal, and generates a feedback signal (F1 frame) by measuring sensor values. However, F1 is delayed until the next polling by the master station that will be right after the transmission of C2 (①). Therefore, the network induced delay is equal to $T_S + 2T_P + 2T_T + T_C$ from the generation of one control signal to the reception of the corresponding feedback signal.

Second, the network induced delay is caused by the lack of synchronization among processes. Figure 6.b shows the transmission procedure of frames under the lack of synchronization. The communication procedure for control and feedback signal is similar to figure 6.a. But, because the input/output data process and



(a) network delay due to processing time and polling



(b) network delay under lack of synchronization

Figure 6. Transmission sequence of Profibus-DP with network induced delay

communication process of master are not synchronized, C1 passed to the communication process from the input data process of the master is not transmitted immediately. It is transmitted at the next communication period of the communication process. Due to this reason, the network induced delay is increased by the time left till the next cycle (②). Also, because the input/output data process and instrumentation process of the slave are not synchronized, after the input/output data process receives the control signal from the communication process, it doesn't forward the control signal to the instrumentation process immediately, and forwards only at the next communication process. This results in additional delay (③) is occurred. Therefore, the total network induced delay is equal to $T_P + 2T_T + T_C + 3T_S$ from the generation of one control signal to the reception of the corresponding feedback signal.

In order to design a controller for networked control system, the designer should be aware of the nature of network induced delay that is more than the sum of processing delays. These delays may adversely affect the control systems performance even to the point of instability. Due to this reason, the controller should be specifically designed for the network induced delay.

4. Performance evaluation using a DC motor

4.1 Plant

A DC motor is used as a plant in order to evaluate the performance of the NCS. In order to interface with a PC, we use a 12 bit D/A converter to drive the motor and a 16 bit counter to measure the output of the encoder. The sampling time of the system is chosen to be 10 msec, and a reference input is given as 1,000 rpm for speed control.

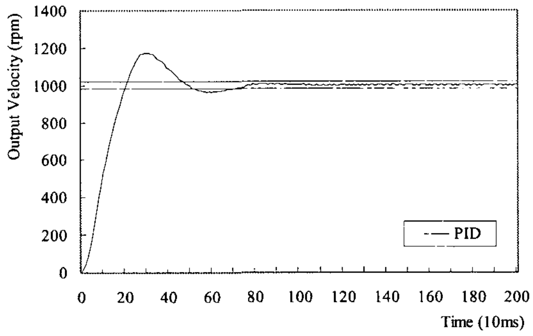


Fig. 7 Response using the PID controller without network

The transfer function of DC motor is modeled by step response method and is given below.

$$G(s) = \frac{y(s)}{u(s)} = \frac{543.47}{0.12s+1} \tag{1}$$

Here, the design specifications of the PID controller are as follows: 10 % percent overshoot, 0.8 sec 5% settling time, and the damping ratio of 0.5. Under these design specifications, the PID controller designed by the root locus method can be expressed as follows[12];

$$\begin{aligned} G_{PID}(s) &= K_d s + K_p + \frac{K_i}{s} \\ &= K_p\left(T_d s + 1 + \frac{1}{T_i s}\right) \\ &= \frac{0.0001s^2 + 0.011s + 0.2}{s} \end{aligned} \tag{2}$$

Figure 7 is a step response of the control system using the PID controller given in equation (2). In the figure, the percent overshoot is 7 %, and the settling time is 0.73 sec. From these results, we know that the designed PID controller satisfies the design specifications.

4.2 Experimental setup for NCS performance evaluation

Figure 8 shows the NCS based on a Profibus-DP network. In the experimental setup, a single remote controller and three DC motors are connected via a Profibus-DP network so that the remote controller can control three DC motors simultaneously through the network. This setup can be regarded as a system where

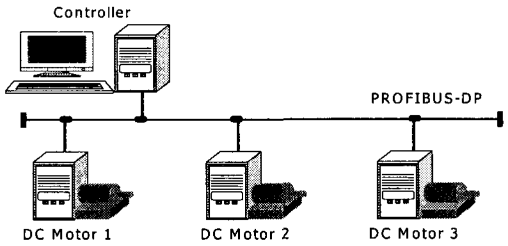


Fig. 8 Networked control system based on a Profibus-DP network

the central remote controller controls the actuator of each ais in a robot or a machining center through the fieldbus.

Each station has a network interface card (Softing PROFI-board) that is based on a Siemens SAB-C165 microprocessor along with two communication ASICs (Application Specific Integrated Circuit): ASPC2 and SPC3. Also, a transmission speed of Profibus-DP is selected to be 1.5 Mbps, and the length of data is 2 bytes. Here, the data length of 2 bytes is sufficient to represent a counter value of the encoder and the control signal to the motor. Also, the sampling time of DC motor is 10 msec, and the polling period of Profibus-DP (T_{DP}) is selected to be 2 msec because of the Softing's recommendation that T_{DP} must be smaller than $1/4 \cdot T_{MAPC}$ in order to mitigate the effect of network induced delay.

Figure 9 shows the motor response obtained by the previous PID controller given by equation (2). In the figure, the percent overshoot is increased to 16 % and the settling time is prolonged to 1.12 sec that do not satisfy the original design specifications. This deterioration of the performance is caused by the network induced delay. Therefore, in order to satisfy the design specifications, a procedure is needed to design PID controllers for NCS by considering the effect of the network induced delay.

4.3 Design of remote controller with network induced delay

From the viewpoint of the controller, the network and the motor can be treated as a part of the overall plant. Therefore, more appropriate controllers can be designed by using several techniques for time-delayed plants. These design methods can be grouped into a model based method and a non-model based method. The model based method includes Smith predictor[13], GAP (Generalized Analytical Predictor)[14], IMC (Internal Model Control)[15] while the non-model based method includes Ziegler-Nichols method[16], Cohen-Coon method[17], fuzzy predictor[18]. In general, the effectiveness of the model based method depends on the accuracy of the plant model. For NCS, the network delay is affected by the amount of network traffic and the internal synchronization, which makes it difficult, if not impossible, to predict the delay. Hence, the non-model based method seems to be a more appropriate choice over the model based method[19].

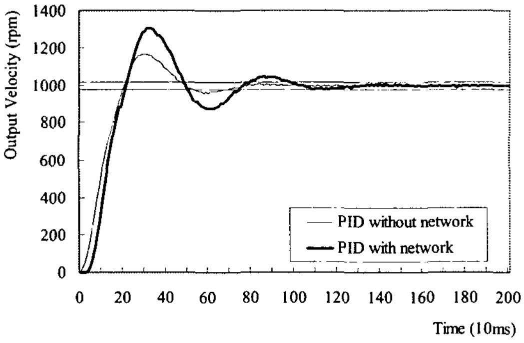


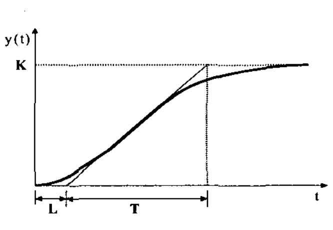
Fig. 9 Effect of network induced delay on a remote PID control system

4.3.1 Design by Ziegler-Nichols method

The Ziegler-Nichols method[16] is one of the most typical methods for PID gain tuning. Table 1 summarizes the guideline by Ziegler and Nichols based on the step response of a given plant. Although the step response of the motor combined with the network is a little different from that given in table 1, the Ziegler-Nichols method is modified to accommodate the network delays. Based on several trial and error procedures, it is found that the control system performs as well as the system without network when the time delay term L is set to the network induced delay. Because the network induced delay is known to be 3 or 4 sampling times and the sampling time is 0.01 second, the time delay L is chosen to be 0.03. Also, the time constant T is 0.12, and K is 543.47

Table 1 Ziegler-Nichols & Cohen-Coon tuning methods

	K_p	T_i	T_d
Z-N	$1.2T/KL$	2L	0.5L
C-C	$\frac{T}{KL}\left(\frac{16+3L}{12T}\right)$	$\frac{L[32+6(L/T)]}{13+8(L/T)}$	$\frac{4L}{11+2(L/T)}$



as given in equation (1). By following the formulae in table 1, K_p, T_i, and T_d are 0.008832, 0.06, and 0.015, respectively.

Figure 10 shows the response of the DC motor using the above gains. In the figure, the maximum overshoot is 17 % and the settling time is 0.7 sec. From these results, we may say that the modified Ziegler-Nichols method can be used to design a controller for NCS in order to compensate the adverse effect of the network induced delay.

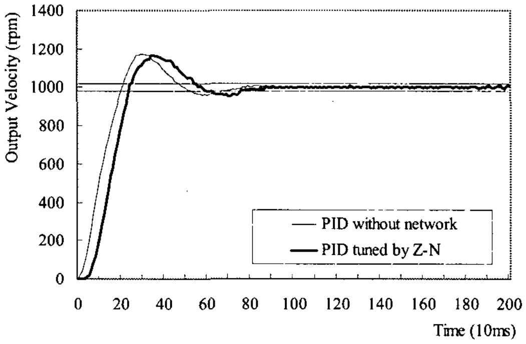


Fig. 10 Response of PID controller tuned by Ziegler-Nichols method

4.3.2 Design by Cohen-Coon method

The Cohen-Coon method[17], modified similar to the Ziegler-Nichols method, calculates the PID controller gains using the equations shown in table 1. By this method, K_p is 0.0102, T_i is 0.067 and T_d is 0.0104.

Figure 11 shows the response of the DC motor using the PID controller tuned by the Cohen-Coon method. The figure shows that the maximum overshoot is 20 % while the settling time is 0.8 sec. Although these results are not as good as the previous ones, these still satisfy the design specification. Therefore, the modified Cohen-Coon method is almost equivalent to the modified Ziegler-Nichols method in designing controllers for NCS.

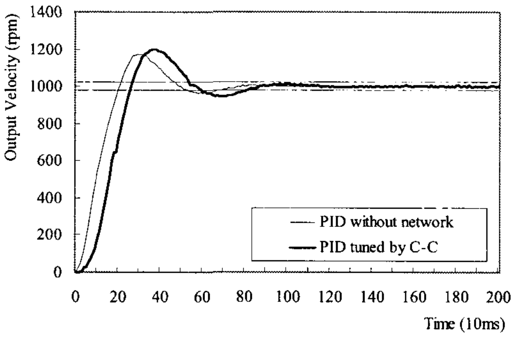


Fig. 11 Response of PID controller tuned by Cohen-Coon method

5. Conclusions

This paper presents an implementation method of the networked control system based on a fieldbus along with the analysis of the network induced delay. More specifically, an NCS for motor speed control is implemented by using a Profibus-DP network for performance evaluation. In addition, this research proposes the modification of traditional design methods for PID controllers in order to mitigate the adverse effect of network delays on control performance. The conclusions derived from the research are as follows.

1) From the observation made on the NCS experimental setup, there exists non-zero network induced delay for data exchange via the network. This delay causes the control system's performance to deteriorate. Therefore, the remote controllers for networked control system should be designed with the consideration of the network induced delay.
2) The causes for network induced delays in Profibus-DP include the waiting time to share the network medium among numerous nodes, processing time at each node, and lack of synchronism among the application processes. While the delay due to the first two causes are inevitable because the network and the processor have finite capacities, the application processes should be designed such that the lack of synchronism does not cause excessive delays.
3) With a slight modification, both Ziegler-Nichols method and Cohen-Coon method are able to design the controllers for NCS that are comparable to the controllers without network in terms of maximum overshoot and settling time.

References

1. Rembold, U., Nnaji, B. O., and Storr, A., Computer Integrated Manufacturing and Engineering, Addison-Wesley, 1993.

2. Ray, A., "Networking for Computer-Integrated Manufacturing," IEEE Network, Vol. 2, No. 3, pp. 40-47, 1988.
3. PNO, Profile for Variable Speed Drives, PROFIDRIVE, version 2.0, Profibus International, 1997.
4. Lee, K. C., Kim, D. W., Kang, S., and Lee, S., "On-line Fuzzy Performance Management of Profibus Networks," International Journal of the Korean Society of Precision Engineering, Vol. 1, No. 2, pp. 5-13, 2000.
5. Hong, S. H., and Park, T. J., "A Study on the Implementation of Fieldbus-Based Manufacturing Automation Systems," Journal of Korean Society of Precision Engineering, Vol. 16, No. 3, pp. 91-102, 1999.
6. Cavalieri C., Stefano, A.D., and Mirabella, O., "Impact of fieldbus on communication in robotic systems," IEEE Transactions on Robotics and Automation, Vol. 13, No. 1, pp. 30-48, 1997.
7. Lee, K. C., Kim, K. W., Kim, H. H., and Lee, S., "Implementation of Profibus-FMS Network for Real-Time Closed-Loop Control System," Journal of Control, Automation and Systems Engineering, Vol. 6, No. 5, pp. 92-98, 2000.
8. Popp, M., The Rapid Way to Profibus, Profibus International, 1997.
9. PNO, Technical Overview - Hints and Tips, http://www.profibus.com, 1999.
10. Walsh, G. C., and Hong, Y., "Scheduling of Networked Control Systems," IEEE Control Systems Magazine, Vol. 21, No. 1, pp. 57-65, 2001.
11. Wei, Z., Branicky, M. S., and Phillips, S. M., "Stability of Networked Control Systems," IEEE Control Systems Magazine, Vol. 21, No. 1, pp. 84-99, 2001.
12. Ogata, K., Modern Control Engineering, Prentice-Hall, 1990.
13. Smith, O. J., "Closer control of loops with dead time," Chemical Engineering Process, Vol. 53, No. 5, pp. 217-219, 1959.
14. Wellons, M. C., and Edgar, T. F., "The generalized analytical predictor," Industrial Engineering on Chemical Research, Vol. 26, No. 8, pp. 1523-1536, 1987.
15. Morari, M., and Zafiriou, E., Robust process control, Prentice-Hall, 1989.
16. Ziegler, J. G., and Nichols, N. B., "Optimum Settings for Automatic Controllers," Transactions on ASME, Vol. 64, 1942.
17. Cohen, G. H., and Coon, G. A., "Theoretical consideration of retarded control," Transactions on ASME, Vol. 75, 1953.
18. Jang, M. J., and Chen, C. L., "Fuzzy successive modelling and control for time-delay system," International Journal on Systems Science, Vol. 27, No. 12, pp. 1483-1490, 1996.
19. Zhao, F., Ou, J., and Du, W., "Pattern-based fuzzy predictive control for a chemical process with dead time," Engineering Applications of Artificial Intelligence, Vol. 13, pp. 37-45, 2000.