Design of Networked Control System with Discrete-Time State Predictor over WSN

Hyun-Chul Yi, Hyoung-Woo Kim, and Joon-Young Choi

***Abstract*—We design a networked control system (NCS) with discrete-time state predictor where the communication between the controller output and the plant input takes place over a wireless sensor network (WSN). In order to measure time delays between the controller output and the plant input in real time, we design an algorithm to measure round trip time (RTT) between WSN nodes, and implement it into TinyOS of WSN. By using the measured time delays, we construct the discrete-time state predictor to compensate the time delays between the controller output and the plant input in real-time. For the real time experiment, we simulate the dynamic plant model, the controller, and WSN interface using Real-Time Windows Target provided in MATLAB. The WSN interface in the Simulink model consists of serial ports, which connect the controller output and the plant input with WSN nodes. The experiment results show that the time delays between the controller output and the plant input are precisely measured in real time; the discrete-time state predictor appropriately compensates the time delays; and the stability is achieved in the closed-loop of the NCS.**

***Index Terms*—Delay, discrete-time state predictor, NCS, WSN.**

I. Introduction

Network control systems (NCSs) are spatially distributed systems in which sensors, actuators, and controllers exchange I/O information through a shared band-limited digital communication network. NCSs have been finding applications in a broad range of areas such as mobile sensor networks, remote surgery, haptics collaboration over the Internet, automated highway systems, and unmanned aerial vehicles [1], [2]. In particular, wireless networked control systems (WNCSs) are more promising because the deployment of wireless networks allows fully mobile operation, flexible installation and rapid deployment while reducing maintenance costs [3]. The key issue in NCSs is how to deal with the network delay and packet loss because it is well-known that even a small amount of time delay in the feedback loop of a control system can make the whole system oscillating or unstable [4].

On the other hand, advances in pervasive computing, communication and sensing technologies are leading to the emergence of wireless sensor networks (WSNs), and significant efforts have been made for fundamental network operations and various applications of WSNs. In [5], a networked control system is designed over WSNs and a state estimation algorithm is proposed based on the Extended Kalman Filter. However, in [5], the time delay between sensors and controllers is not measured but just assumed to be a bounded value that is dependent on the sampling period of sensors.

In this paper, we design an NCS with discrete-time state predictor where the communication between the controller output and the plant input takes place over WSNs. In order to measure time delays between the controller output and the plant input in real time, we design an algorithm to measure round trip time (RTT) between WSN nodes, and implement it into TinyOS of WSN. By using the measured time delays as a parameter, we construct the discrete-time state predictor [6], [7] to compensate the time delays between the controller output and the plant input in real time. The discrete-time state predictor suitably compensates measured time delays in the feedback loop. The experiment results show that the time delays between the controller output and the plant input are precisely measured in real time; the discrete-time state predictor appropriately compensates the time delays; and the closed-loop of the NCS is made to be stable.

II. RTT Measurement in WSN

In order to measure RTT between two WSN nodes, we need to measure the time difference between Packet Send (RTT Start) and Ack Receive (RTT End) as shown in Fig. 1.

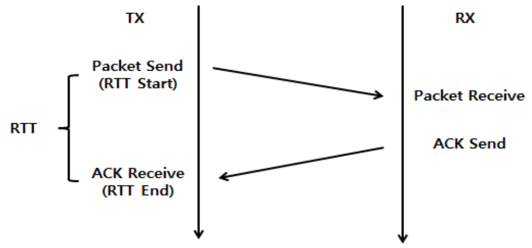

Fig. 1. Outline for RTT measurement.

The time of Packet Send is set right before sending a packet and the time of Ack Receive is set immediately after receiving an ACK packet from the receiving node. The time when an ACK packet arrives can be obtained by using ACK component in TinyOS [8]-[10]. Specifically, right before sending a packet, the sending node calls ACK.requestAck() routine. Then, the sending node confirms the arrival of an ACK packet by calling ACK. was Acked() routine while processing an ACK packet. In this way, we can measure RTT for each packet between two WSN nodes in real time.

Manuscript received October 29, 2013; revised December 20, 2013. This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(2010-0022878).

H. C. Yi and H. W. Kim are with the Department of Electrical and Computer Engineering, Pusan National University, Korea (e-mail: {hcy, hwkim0314}@pusan.ac.kr).

J. Y. Choi is with the Department of Electrical Engineering, Pusan National University, Korea (e-mail: jyc@pusan.ac.kr).

DOI: 10.7763/JACN.2014.V2.91

```
SReceive.receive() {     // Serial Packet Receive
   if (NODE_ID == 0) {
   paket->rtt = rtt / 2;
   ACK.requestAck(&packet);

    rttstart = GetNow();

   AMSend.send(); //WSN Packet Send
   }
}
AMSend.sendDone() {    // WSN Packet SendDone
   if(ACK.wasAcked(packet)) {
     rttend = GetNow();
     rtt = rttend - rttstart;
   }
   else {
     rtt = 0;
   }
}
```

Fig. 2. Pseudocode for RTT measurement

Fig. 2 shows a psuedocode for RTT measurement, which is implemented into TinyOS of WSN. As soon as a serial data packet containing the controller output value is received, we call the SReceive.receive routine, where the received packet is sent and the ACK packet is requested to another node in WSN. Then, the AMSend.sendDone routine is called to calculate the RTT based on the time when the ACK packet arrives.

III. Networked Control System with Discrete-Time State Predictor

The whole structure of the designed NCS with discrete-time predictor is depicted in Fig. 3. The plant model in Fig. 3 is described by the following discrete-time linear time-invariant system:

$$X(k+1) = AX(k) + BU(k-D), \quad (1)$$

where $X(k) \in R^n$ is state, (A,B) is controllable pair, and the input signal $U(k) \in R$ is delayed by D units of time. Suppose that a static state feedback control has been designed for the system (1) with no delay (i.e., with $D = 0$) such that $U(k) = KX(k)$ is a stabilizing controller, that is the matrix $A+BK$ is Schur stable. Then, our wish is to have a control that achieves

$$U(k-D) = KX(k), \quad (2)$$

which can be alternatively written as

$$U(k) = KX(k+D), \quad (3)$$

and it is non-implementable since it requires future values of state. However, the D-step ahead predictor is designed in [6]-[7] as

$$X(k+D) = A^D X(k) + \sum_{j=k-D}^{k-1} A^{k-j-1} BU(j), \quad (4)$$

which yields the implementable predictor feedback controller

$$U(k) = K\left[A^D X(k) + \sum_{j=k-D}^{k-1} A^{k-j-1} BU(j) \right], \quad (5)$$

where K is the state feedback gain.

Once the system input delay D is known, the discrete-time state predictor can compensate the time delay D and makes the whole closed-loop system stable. In Fig. 3, Node 1 receives the controller output value and transmits it to Node 2. In Node 1, the RTT between Node 1 and 2 is measured and the time delay between Node 1 and 2 is calculated as RTT/2 in real time. Node 1 generates a combined packet containing the current time delay as well as the controller output value, and sends the generated packet to Node 2 through WSN. When the packet from Node 1 arrives at Node 2, the combined packet is separated into the delayed input value and the time delay. Then, the delayed input value is applied to the plant input, and the separated time delay is used as the time delay D for the discrete-time state predictor in real-time.

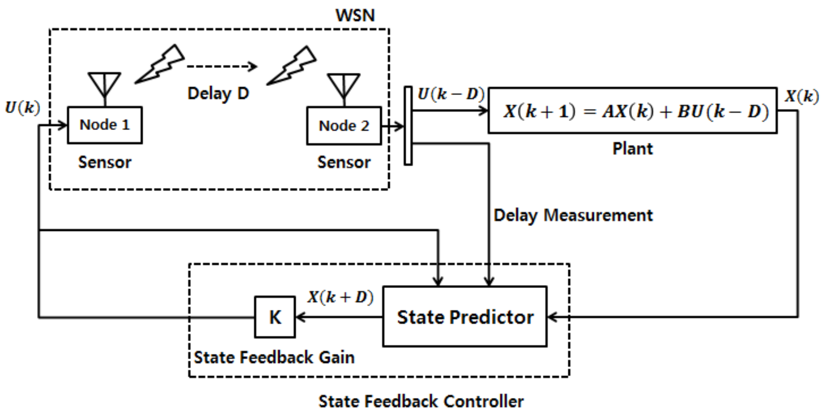

Fig. 3. Structure of NCS with discrete-time predictor over WSN.

IV. Experiment

For the real-time experiment of WSN, we adopt Real-Time Windows Target in MATLAB that provides a real-time engine for executing Simulink models on MS Windows PC. Crossbow sensor nodes, IRIS platform[11], are used for WSN. The plant and the discrete-time state feedback controller in Fig. 3 are modeled in Simulink, and the controller output feedback is connected to WSN nodes through serial ports. We select the plant with the state equation as $A = 1.2$ and $B = 1$ in (1). The state feedback gain is selected as $K = -2$ and the sampling time as 50ms.

The state predictor feedback controller in experiment can be written as follow.

$$U(k) = -2\left[(1.2)^{D} X(k) + \sum_{j=k-D}^{k-1} (1.2)^{k-j-1} U(j) \right]. \quad (6)$$

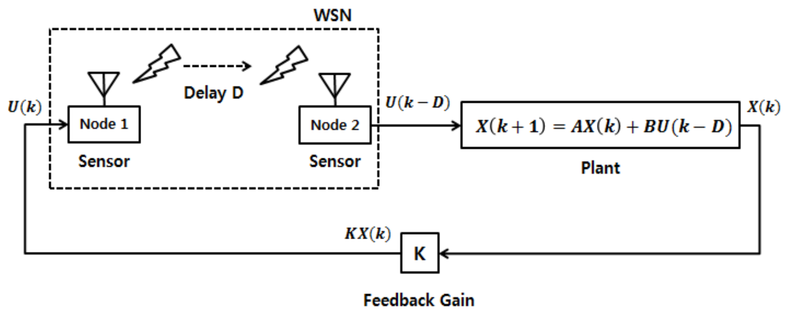

Fig. 4. NCS over WSN without discrete-time predictor.

In order to show the validity of the designed NCS, we conduct two cases of experiments. The first case is the NCS over WSN without the discrete-time state predictor, which is depicted in Fig. 4. The second case is the NCS over WSN with the discrete-time state predictor, which is designed in this paper as shown in Fig. 3.

Fig. 5, the experiment result of the first case, shows that the time delays due to WSN destroy the stability of the system and the system state oscillates and eventually explodes. Fig. 6, the experiment result of the second case, shows that the discrete-time state predictor based on the measured time delays in real time compensates time delays and the system state converges to a steady state value. Fig. 6 also verifies that the RTT measurement algorithm appropriately measures RTT between Node 1 and 2 in real-time and Fig. 7 shows the measured time delays at each sampling time.

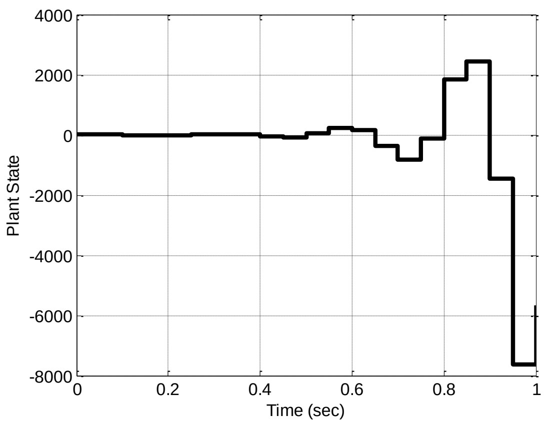

Fig. 5. Response without discrete-time state predictor.

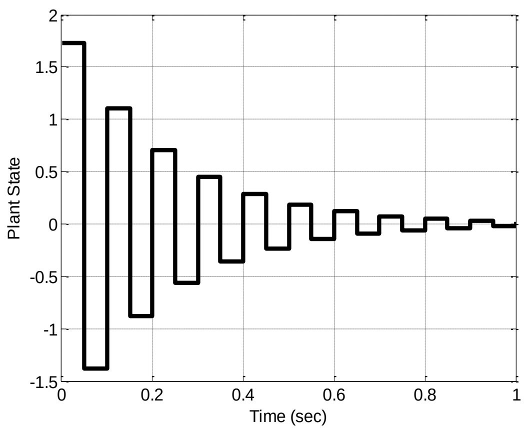

Fig. 6. Response with discrete-time state predictor.

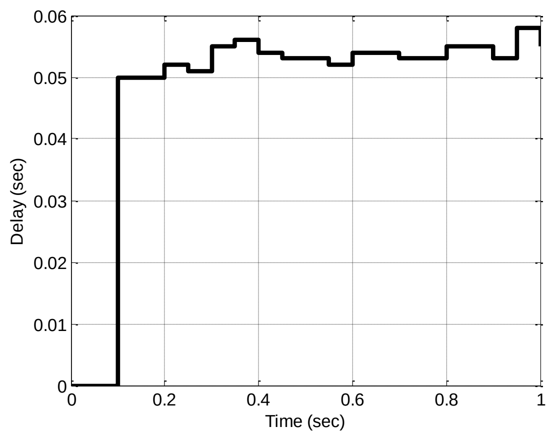

Fig. 7. Measured time delay

V. Conclusion

We design an NCS with discrete-time state predictor where the controller output and the plant input exchange I/O information over WSN. Different from other approaches, time delays are measured in WSN node and transferred to the discrete-time state predictor in the controller in real time, which allows the discrete-time state predictor to precisely compensate time delays. By actively measuring time delays in real-time, we achieve that the NCS with the discrete-time state predictor over WSN is stable and responsive to variation of time delays in WSN. Even though we adopt only two WSN nodes and one feedback controller in the experiments, the designed NCS framework can be extended for NCS consisting of multiple WSN nodes and multiple feedback controllers.

In the design of the proposed NCS, we do not consider the packet loss that is known as another main disturbance that might destroy the stability in the NCS. It is obvious from the nature of discrete systems that the designed NCS tolerate a single time of packet losses for a short duration. As a future work, we will address the compensation technique for periodic or long duration packet losses in the NCS.

References

[1] J. P. Hespanha, P. Naghshtabrizi, and Y. Xu, "A survey of recent results in networked control systems," *IEEE*, vol. 95, no. 1, pp. 138-162, Jan. 2007.

[2] U. Premaratne, S. K. Halgamuge, and I. M. Y. Mareels, "Event triggered adaptive differential modulation: a new method for traffic reduction in networked control systems," *IEEE Transactions on Automatic Control*, vol. 58, no. 7, pp. 1696-1706, Jul. 2013.

[3] M. Drew, X. Liu, A. Goldsmith, and K. Hedrick, "Networked control system design over a wireless LAN," in *Proc. 44th Conf. Decision and Control and European Control Conf.*, Dec. 2005, pp. 6704-6709.

[4] Y. A. Millan, F. Vargas, F. Molano, and E. Mojica, "A wireless networked control systems review," in *Proc. Robotics Symposium, 2011 IEEE IX Latin American and IEEE Colombian Conference on Automatic Control and Industry Applications*, Oct. 2011, pp. 1-6.

[5] H. Li, X. Li, J. Wang, and H. Ma, "Design of networked control system based on wireless sensor networks," in *Proc. ICIA '09. International Conference on Information and Automation*, June. 2009, pp. 22-24.

[6] G. C. Goodwin and K. S. Sin, *Adaptive Filtering Prediction and Control*, *Information and System Science Series*, Prentice Hall, 1984.

[7] R. Lozano, P. Castillo, P. Garcial, and A. Dzul, "Robust prediction-based control for unstable delay systems: Application to the yaw control of a mini-helicopter," in *Proc. 42nd IEEE Conference on Decision and Control*, Dec. 2003, vol. 4, pp. 9-12.

[8] Tiny OS source. [Online]. Available: http://www.tinyos.net/

[9] T. Zhu, Z. Zhong, T. He, and Z. Zhang, "Achieving efficient flooding by utilizing link correlation in wireless sensor networks," *Networking, IEEE/ACM Transactions on*, vol. 21, no. 1, pp. 121-134, Feb. 2013.

[10] R. Gao, H. Zhou, and G. Su, "Structure of wireless sensors network based on tiny OS," in *Proc. 2011 International Conference on Control, Automation and Systems Engineering*, July. 2011, pp. 1-4.

[11] MEMSIC. [Online]. Available: http://www.memsic.com/

Hyun-Chul Yi received the B.S. degree in computer science engineering from Pusan National University, Pusan, Korea in 2010 and the M.S. degree in electronic and electrical engineering from Pusan National University, Pusan, Korea in 2012. He is now a Ph.D. candidate in electrical and computer engineering at Pusan National University, Korea since 2012. His research interests are embedded system and wireless sensor network.

Hyoung-Woo Kim received the B.S. degree in electronic and electrical engineering from Pusan National University, Pusan, Korea in 2012. He is now a M.S. candidate in electrical and computer engineering at Pusan National University, Korea since 2012. His research interests are nonlinear control and power systems.

Joon-Young Choi received the B.S., M.S. and Ph.D. degrees in electronic and electrical engineering from Pohang University of Science and Technology (POSTECH), Pohang, Korea in 1994, 1996 and 2002, respectively. From 2002 to 2004, he worked as a senior engineer at Electronics and Tele -Communication Research Institute (ETRI), Daejeon, Korea. From 2004 to 2005, he worked as a visiting associate in the Departments of Computer Science and Electrical Engineering at California Institute of Technology (CALTECH), Pasadena, CA. Since 2005, he has been an associate professor in the School of Electrical Engineering at Pusan National University, Pusan, Korea. His research interests include nonlinear control, internet congestion control, embedded systems and automation.