Proceedings of Sixth International Conference on Machine Learning Cybernetics, Hong Kong, 19-22 August 2007

COMPLETE COMPENSATION FOR TIME DELAY IN NETWORKED CONTROL SYSTEM BASED ON GPC AND BP NEURAL NETWORK

TIAN-KUN WANG[1], LI-HUI ZHOU[2, 3], PU HAN[3], QIAN ZHANG[3]

[1]School of Energy and Power Engineering, North China Electric Power University, Beijing 102206
[2]Guo Dian Nan Jing Automation Co. LTD, Nanjing 210003, China
[3]School of Control Science and Engineering, North China Electric Power University, Baoding 071003, China
E-MAIL: wangtiankun@gmail.com

Abstract:

A new framework is proposed to cope with the uncertain time delay of networked control system. Event-clock-driven controller nodes, together with clock-driven sensor nodes and actuator nodes are required in this framework. Queuing Strategy is introduced both in controller nodes and actuator nodes while the time delay between controller node and actuator node is compensated by multi-step control increment given by the algorithm of General Predictive Control. An output error prediction model is built using BP neural network to deal with the time delay between sensor node and controller node. The principle of this model is to revise the predictive output of general predictive control model using predictive error signal; if the value of time delay exceeds the upper limit, controller nodes will immediately produce the control strategies adopting the revised predictive output, and thus the compensation for time delay between sensor nodes and controller nodes would be accomplished. Simulation experiments are practiced over Ethernet network which embraces both kinds of time delay. It is proved that the scheme of complete compensation remains a good control performance.

Keywords:

Networked Control System (NCS); Time delay; General Predictive Control (GPC); BP neural network; Queuing Strategy (QS)

1. Introduction

A closed loop connected through network is defined as Networked Control System (NCS). In 1999, Gregory C. Walsh in Maryland University initially stated the concept of NCS[1], which brought extensive attention and had been applied in complex dynamic processes, such as in automobiles, the manufacturing process of airplanes, industrial processes and so forth. The modules in NCS (such as sensor nodes, controller nodes, and actuator nodes) exchange information over network. The introduction of network makes it more complex to analyze and design systems, and the random transfer time delay would produce inevitably while information is passing through network, which will degrade the performance of the system or even destabilize it. A .Ray and his cooperators adopted a method of state estimation to analyze and process the acquired information from the angle of statistics. However, this method is based on the fact that the transfer time delay is less than a sampling period, which is incapable of being used in the otherwise situation, thus making it a conservative method; Luck and Ray proposed that buffer areas were set in controller nodes and actuator nodes and synchronized sampling was adopted betweens the nodes, thus transferred the stochastic delay into securing delay. This method makes it easier to design the controller nodes, but the setup of the buffer areas artificially prolongs the delay .So it is too conservative as well; G.P. Liu advised that general predictive control should be utilized in controller nodes[2], which can only be used to compensate the Controller-to-Actuator delay τ_{ca} and ignores the Sensor-to-Controller delay τ_{sc}, therefore when τ_{sc} exists, the compensation effect is unsatisfied.

In this paper, a distributing NCS model with event -clock-driven controller nodes, clock-driven sensors nodes and actuator nodes is proposed over Ethernet network which embraces both τ_{ca} and τ_{sc}. The method discussed in this paper makes the best use of multi-step control increment given by the algorithm of General Predictive Control (GPC) and error model BP algorithm to predict the revised signal, and queuing strategy is introduced in controller nodes and actuator nodes to select the control signal correctly, so that complete compensation for τ_{ca} and τ_{sc} are accomplished.

2. The model of networked control system

The typical structure of NCS is composed of a network

1-4244-0973-X/07/$25.00 ©2007 IEEE

module and several intelligent nodes. Fig.1 shows the major two kinds of transfer delay in NCS: Controller-to-Actuator delay τ_{ca} and Sensor-to-Controller delay τ_{sc}.

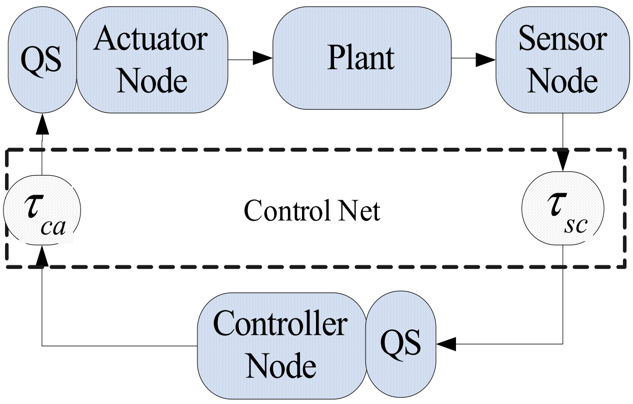

Figure 1. Model structure of NCS with queuing strategy

According to the different driving way in nodes, the modeling way under the network environment is different in NCS. Because of the clock-driven way, the intelligent nodes must be strictly synchronized. However, due to the existence of the deterministic time delay in network, it is hard to be accurately synchronized in NCS. If, however, the controller nodes adopt the event-driven way instead of the time-driven one, the waiting time of data sampling will be saved. In this way, the network transferring delay would be reduced, and the situation of null sampling and data loss would be avoided when the time-driven way in controller nodes is adopted, consequently increasing the efficiency rate of feedback data However, if the event-driven way is adopted completely in controller nodes, it is possible that controlling quantum will be deficient or even the system will be destabilized when the network transfer delay is relatively long. Thus, it is proposed that the controller nodes adopt the way of event-clock-driven way. Controller nodes and actuator nodes are in the same special position in NCS, so data transferring between them needn't to pass the complicated network and it can be reckoned that network delay does not exist between them, thus resulting accurate synchrony. Therefore, sensor nodes and actuator nodes both adopt the clock-driven way with securing period.

Figure.2 shows the information of transfer time sequence in closed-loop networked control system. The basic principles: event-driven way is adopted when controller nodes are normal; when the sampling result is sent through network passage to controller nodes, they immediately execute control arithmetic with a maximum transfer delay T_{max},when the transfer delay τ_{sc} between sensor nodes and controller nodes is longer than T_{max}, controller nodes automatically start up control arithmetic in time-driven way and send out the control signal. As is shown in Fig.2, controllers nodes have not received the measuring signal sent from sensor nodes within the range of [*k-2, k-1*], so an estimated control signal is sent automatically at the time *k-1*. Besides, suppose the controlled variable calculated by the sampling signal y_k at k-th time is u_k, we can see from the Figure2 that u_k may arrive later than u_{k+1} or even u_{k+2} to actuator nodes, thus resulting in mess of time sequence. If not handled in time, the uncertainty of transfer delay may cause serious damage to control systems. In order to solve these problems, GPC arithmetic is introduced in controller nodes. Actuator nodes are processed with queuing strategy, and the control information in the queue can renew the sequence automatically according to time tags, so there will be abundant and optimized control information in actuator nodes. When the value of time delay is longer than T_{max}, actuator nodes will automatically send out signals using the optimal control information processed in queue. In order to deal with the time delay between sensor nodes and controller nodes, queuing strategy is also introduced in controller nodes. In the queue of controller nodes, the newest output predictive signal revised by BP neural network is consistently saved and the revision method is: by using historical data and judging from the past predictive deviation signal, an error prediction model is build based on BP neural network, using the output of the error model to revise the predictive output signal in GPC. When there is time delay and it is longer than T_{max}, by using predictive output signal revised from the error predictive signal, the arithmetic of GPC is utilized to get the optimal control rules, so that the transfer delay between sensor nodes and controller nodes is well solved and thus improve the robustness of the system.

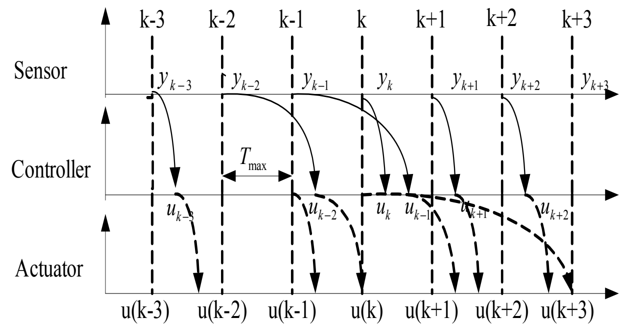

Figure 2. Transfer sequence graph of system information

The schemes of compensation for the two kinds of system delay discussed above will be analyzed as follow.

3. The compensation for Controller-to-Actuator delay

3.1. General predictive control in controller nodes

General predictive control is based on parameter model, and the predictive model of which is controlled auto-regressive integrated moving average model (CARIMA). For multi-step predictive time domain P and control time domain L, the vector descriptive way of output predictive signal is:

$$Y_P(K+1)=G\Delta U(K)+Fy(k)+H\Delta u(k) \qquad (1)$$

In the formula：

$$Y_P(K+1)=\left[y_P(k+1),y_P(k+2),\cdots,y_P(k+P)\right]^T$$

$$\Delta U(K)=\left[\Delta u(k),\Delta u(k+1),\cdots,\Delta u(k+M(1))\right]^T$$

$$F=\left[F_1,F_2,\cdots,F_P\right]^T,\quad H=\left[H_1,H_2,\cdots H_P\right]^T,$$

$$G=\begin{bmatrix} g_0 & & & \\ g_1 & g_0 & & \\ \vdots & \vdots & \ddots & \\ g_{M-1} & g_{M-2} & \cdots & g_0 \\ g_M & g_{M-1} & \cdots & g_1 \\ \vdots & \vdots & & \vdots \\ g_{P-1} & g_{P-2} & \cdots & g_{P-M} \end{bmatrix},$$

Suppose the reference trace is:

$$Y_d(K+1)=\left[y_d(k+1),y_d(k+2),\cdots,y_d(k+P)\right]^T,$$

The aim of GPC is to make the tracking error of predictive output to reference trace to the least, so the performance index can be decided as follows:

$$J=E[(Y_P(K+1)-Y_d(K+1))^T(Y_P(K+1)-Y_d(K+1)) +\lambda U^T(K)U(K)]$$

where Y_d is the expectation value, λ is the weighted factor, and from equation $\partial J/\partial U(K)=0$, we can get the optimal control law as follows[3]:

$$\Delta U(K)=(G^TG+\lambda I)^{-1}G^T(Y_d(K+1)-F_y(k)-H\Delta u(k-1)) \qquad (2)$$

3.2. The control increment queue of predictive control in actuator nodes

The arithmetic GPC can calculate the control increment sequence from the time *k* to *k+L-1* each time:

$$\Delta U(K)=\left[\Delta u(\mathrm{k}),\Delta u(\mathrm{k}+1),\cdots\Delta u(\mathrm{k}+L-1)\right]$$

In traditional closed-loop control system, there is no data delay or loss, so the optimal control signal $u(\mathrm{k})=u(\mathrm{k}-1)+\Delta u(\mathrm{k})$ is only adopted and the later L-1 control increment signal can be abandoned. In order to compensate the time delay τ_{ca} in NCS, a buffer queue QS is set in actuator nodes to save the control increment signal $\Delta U(K)$ .Suppose the maximum signal of τ_{ca} is $T_{max}=m\times T$, where T is the sampling period, suppose the maximum length $L=m$, and the length of queue in actuator nodes is m.

As shown in Fig.2, for the time when the signal of control increment signal reaches actuator nodes is stochastic while actuator nodes send out control signals periodically, the renewal program of QS is divided into two parts[4]: event-driven and clock-driven, the renewal of the former generates when control increment sequence $\Delta U(K)$ reaches actuator nodes, the renewal of the latter happens after actuator nodes send out control signal, the specific renewal arithmetic can be found in reference [4]. In this way, optimal or hypo-optimal control increment signal is consistently kept in QS, and actuator nodes can send out optimal or hypo-optimal controlling value in each sampling period.

4. The compensation for Sensor-to-Controller delay

4.1. The queue of revised predictive output in controller nodes

Because controller nodes operate in event-clock-driven way, to reduce the influence induced from transfer delay further, buffer queue QS is set at controller nodes and the queue length is decided by the length of delay τ_{sc}. Revised predictive output signal $\hat{Y}_P(K)$ is kept in buffer queue QS, $\hat{Y}_P(K)=\tilde{Y}_P(K)+Y_E(K)$, in which $\hat{Y}_P(K)$ is the actual output predictive value, and can be calculated with $\Delta U(k)$ according to Eqs.(1). $Y_E(K)$ is given by trained error predictive model of BP neural network, and thus provide adequate information for GPC arithmetic, improve predictive accuracy. Controller nodes adopt event-driven way in normal condition. When network transfer delay is longer than T_{max}, controller nodes will automatically start the controlling arithmetic corresponding to the revised output signal $\hat{Y}_P(K)$ of the corresponding time, and send out control signal to actuator nodes at the same time.

4.2. The calculation of the error predictive value in BP neural network

Suppose BP neural network as the predictive model, expected output and input controlling value of the system as the input sample, and the difference between actual output and predictive output as the sample of output. The connective weighted factor in network will be constantly revised to make the mapping described by network approach the input-output relation step by step, and the network can be used as the error model after the study process[5].

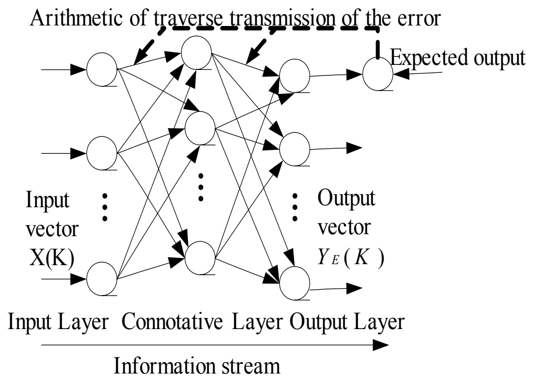

Figure 3. BP network schematic plan

Choose three-layer BP network as showing in Fig.3, and Sigmoid function is choose in the middle layer while the nodes of output layer adopts linear function. Input vector:

$$X(K) = [y_d(k), y_d(k-1), \cdots, y_d(k-n-1), u(k-1), \cdots, u(k-m-1)]^T$$

Where $y_d(*)$ is the expected output value of the system, $u(*)$ the control signal, n and m the ranks of expected output and input respectively.

Output vector is the error predictive value:

$$Y_E(K) = [y_e(k), \cdots, y_e(k-j)]^T,$$

$$y_e(k-j) = y(k-j) - y_p(k-j/k-j-d)$$

$j = 1, 2, \cdots, N$,Where $y(k-j)$ is the actual output at the time k-j, $y_p(k-j|k-j-d)$ the predictive output at the time k-j-d, d the delay in the system, N the amount of trained samples.

Utilize the arithmetic of traverse transmission of the error to adjust the connective weighted factor. The trained neural network can be used to predict error.

Mark the predictive value as $y_e(k+j|k)$, the revised predictive output of the system according to errors is:

$$\hat{y}_p(k+j|k) = \tilde{y}_p(k+j|k) + y_e(k+j|k) \quad (3)$$

Substituting Eqs.(1) into the above one as the predictive value, the optimal control increment vector is as follows:

$$\Delta U(K) = (G^T G + \lambda I)^{-1} G^T (Y_d(K+1) - F_y(k) - H\Delta u(k-1) - Y_E) \quad (4)$$

When there is Sensor-to-Controller time delay τ_{sc} between sensor nodes and controller nodes and it is shorter than T_{max}, the controller nodes automatically start the control arithmetic in event-driven way according to Eqs.(2) and send out the control signal. When Sensor-to-Controller time delay τ_{sc} is longer than T_{max}, the controller nodes automatically start the control arithmetic in clock-driven way according to Eqs.(4). In both above cases, the renewal of the sequence of QS in both controller nodes and actuator nodes is going on at the same time. The optimal output value compensated by BP neural network is used as the control increment signal sending to the actuator nodes. The delay τ_{sc} is efficiently compensated and the level of accuracy has been enhanced[6].

5. Simulation experiment of the compensation for NCS delay

5.1. Design of the experiment

The scheme above accomplishes the complete compensation for Controller-to-Actuator delay τ_{ca} and Sensor-to-Controller delay τ_{sc}. To verify the controlling effects, the simulation experiment over Ethernet network is designed.

Because Ethernet network adopts the network protocol of CSMA/CD, when there is relatively heavy load on the network, besides, when the collision produces, using arithmetic BEB to give out waiting time, thus cause the uncertain delay on the network.

The networked control system based on North China Electric Power University campus-wide network is designed in Figure.4.

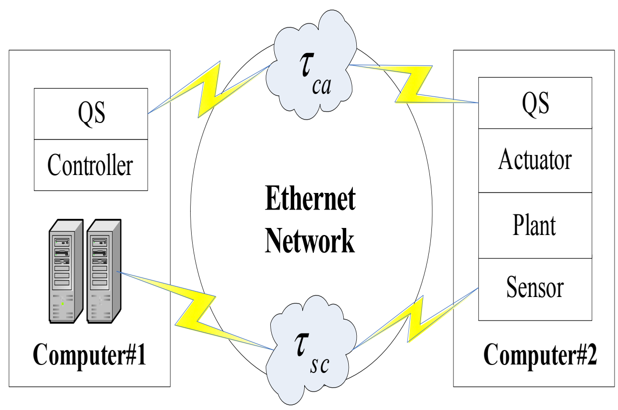

Figure 4. NCS based on Ethernet network

Use two computers in the lab as the net nodes, one as intelligent sensor/actuator, the other as controller. Because NCEPU campus-wide Ethernet is a commonly used network and there is great flow over it, there should be scholastic waiting if collision is detected before data package sends out, thus resulting in network delay. In computer1 there mainly operates the programs of the communication and calculation of controlled plant. The communication programs apply the Socket technology, and support TCP/IP protocol, and communicate in point-to-point credible way, receive the controlling signals from computer2 and send out calculated output signals; In computer2 there mainly operates control and communication programs, and the control programs adopt the arithmetic of general predictive control and BP neural network error compensation, the communication programs are used to receive output information and send out control information. Mixed programming in the languages of Matlab and VC++ is used in designing the programs, besides, the function library of matrix computing MATCOM is also used.

5.2. Simulation results

Suppose the controlled plant is:

$$y(k)-0.496585y(k-1)=0.5u(k-2)+\xi(k)/\Delta$$

$\xi(k)$ is the white noise uniformly distributed within the range of [-0.2,0.2], p=6,L=m=2. The respond curve after the application of complete compensation is shown in Fig.5.

When the state of the network is relatively bad, and the communication load is relatively heavy, the scheme of completely compensation has the quick response time, smaller overshoot and satisfied control effects.

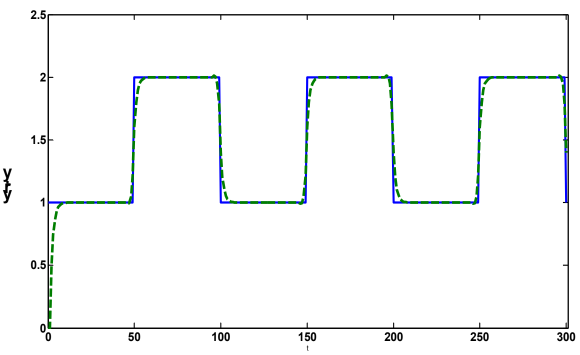

Figure 5. simulation results of complete compensation in NCS

6. Conclusion

For the uncertain transfer time delay existing in NCS, the paper discusses the methodology to use multi-step predictive control increment plus queue sequencing to compensate for Controller-to-Actuator time delay, and to use the value revised by BP neural network error predictive model to compensate for the Sensor-to-Controller time delay, thus accomplishing the complete compensation for time delay in NCS. The feasibility of this methodology is verified via the campus-wide simulation experiment and it is proved that the scheme of complete compensation remains a good control performance.

Acknowledgements

This paper is supported by the research projects fund funded for teachers of North China Electric Power University. The item number is 200512016.

References

[1] Antsaklis P. and Baillieul, J. “Special Issue on Networked Control Systems”. IEEE Trans. on Automatic Control, 2004, 49(9): 1241-1243.

[2] G.P,Liu, J.X.Mu.,D.Rees, “Networked Predictive Control of Systems With Random Communication Delay” Control 2004, September 2004.

[3] ShiMing Yu, MaYing Yang, Li Yu. “Predictive Compensation for Stochastic Time Delay in Network Control Systems”, ACTA Automatica Sinica Vol.31,No.2 231-238 March 2005.

[4] Can Yang, Shanan ZHU, Wanzeng Kong,, Liming Lu, “Application of generalized predictive control in networked control system”, Journal of Zhejiang University SCIENCE A Vol.1,No.2, 225-233 July 2006.

[5] Aiguo Wu, Dongqing Wang, Jin Liang. “Application of Generalized Predictive Control With Neural Network Error Correction in Networked Control System”. Proceedings of the 5th World Congress on Intelligent Control and Automation, Hangzhou 1387-1390 June 15-19, 2004.

[6] Dong Xia, Ping Li, Shuyan Ding, “Generalized Predictive Control Based on Error Correction Using Improved Neural Network”, Journal of Computer Simulation Vol.21, No.12, 143-145 December 2004.