International Journal of Control, Automation, and Systems (2011) 9(2):285-293 http://www.springer.com/12555
DOI 10.1007/s12555-011-0210-7

Resilient Control for Wireless Networked Control Systems

Kun Ji and Dong Wei

Abstract: This paper proposes an approach to quantify the concept of resiliency in terms of Quality of Control (QoC) of a control system. Based on this concept, an intelligent resilient control algorithm (RCA) is presented for wireless networked control systems (WNCS) to maintain operational normalcy in face of wireless interference incidents, such as Radio Frequency (RF) jamming and signal blocking. The proposed algorithm closes the control loop with wireless sensors feasible by significantly increasing control system's tolerance to data packet loss and delay caused by wireless interference. The proposed algorithm, along with other well developed wireless technologies, has the potential to enable implementing wireless sensors widely in the next generation of industrial automation and control systems.

Keywords: Resilient control, NCS, wireless interference, wireless sensors.

1. INTRODUCTION

Resiliency is the property of a material to absorb energy when it is deformed elastically and then, upon unloading to have this energy recovered. A broad definition of Resilient Control Systems is "… those that tolerate fluctuations via their structure, design parameters, control structure and control parameters" [1]. Many literatures discuss the definition, properties, measurements, research areas and possible applications can be found in conference proceedings [2,3]. One definition of the resilient control system, "… one that maintains state awareness and an accepted level of operational normalcy in response to disturbances, including threats of an unexpected and malicious nature", is proposed in [4]. We understand that a resilient control system is able to provide and maintain acceptable performance and functionalities of the controlled process and system in the face of undesirable incidents. The resiliency of a control system should be measured by 1) static response: for a given incident, how much performance in terms of production or quality it would lose; 2) dynamic response: when the incident is removed, how long does it take to get back to its original performance [5].

In this paper, we discuss the resilient control problem for wireless networked control systems (WNCS). A networked control system (NCS) is a distributed control system (DCS) whose components (sensors, controllers, actuators, etc.) are distributed using digital network technology. The use of this technology brings with its important advantages, such as low cost, improved usage of resources, simplicity of maintenance, and error diagnosis, and above all, the flexibility of reconfiguring the different components [6]. Consequently, NCSs have been finding applications in a broad range of areas such as, tele-surgery [7], remote control over the Internet [8,9], and automated highway systems and un-manned aerial vehicles [10]. Nevertheless, there are also disadvantages in this type of systems: implementing closed-loop control in a communication network leads to delay and data loss that inevitably can degrade its performance even lead to instability [10-15]. There has been a trend towards the implementation of NCS using wireless network (IEEE802.11) since wireless network is increasingly being adopted as a low level sensor and control network technology [13]. This type of NCSs is called wireless networked control systems. There is significant interest in wireless solutions for industrial and manufacturing environments where DCS is used. The challenges for closing the control loop over wireless networks are inherited from the nature of NCS and wireless network itself. Wireless networks in nature are subject to interference such as RF jamming and signal blocking which causes link failures and further degrading the performance of WNCSs, i.e., quality of control (QoC). Therefore, a practical resilient control algorithm (RCA) is needed to maintain the operational normalcy (i.e., certain guaranteed QoC) throughout the wireless link failure, which is the motivation of the research presented in this paper. The emphasis is on WNCSs using wireless sensors and having an unreliable wireless link affected by both data packet loss and delay in the sensor feedback loop. The data packet loss and delay are assumed random in nature and unknown in advance. To improve the QoC of NCSs, there are basically two kinds of approaches: 1. improving the network communication reliability by modifying network framework or protocol; 2. robust controller design to cope the network unreliability. The approach proposed in this paper is different from those two since it requires neither the change of network framework/protocol nor the controller design. It can be

Manuscript received January 15, 2010; revised August 30, 2010; accepted November 24, 2010. Recommended by Editorial Board member Young Soo Suh under the direction of Editor Young-Hoon Joo.
Kun Ji and Dong Wei are with the Corporate Research, Siemens Corporation, 755 College Rd East, Princeton, NJ 08540, USA (e-mails: {ji.kun, dong.w}@siemens.com).

implemented on the existing control framework to improve the system resiliency against network unreliability. The major ideas presented in this paper come from the delay and packet loss compensation algorithms presented in [6] and later work.

2. PROBLEM STATEMENT

2.1. Resiliency index

In this paper, a new index called Resiliency Index (RI) is defined to indicate how resilient the control system is and try to quantify the concept of resiliency in terms of control performance. The generic Resiliency Index (RI) of a control system is defined as:

$$RI_{generic} := \frac{T_{incident}(QoC)}{\Delta QoC}, \tag{1}$$

where $T_{incident}(QoC)$ is the maximum incident time during which the system still maintains operational normalcy, i.e., meets certain QoC requirement. ΔQoC is the performace degradation. This definition is further illustrated in Fig. 1. Note that C.P. in Fig. 1 represents control performance. The difference between the maximum performance and the minimum performance during incident is defined as the ΔQoC. The rationale behind Eqn. (1) is that a control system has higher resiliency than others if it can tolerate longer incident time or deliver better QoC (i.e., less performance degradation) during the same incident. Based on the definition of the RI, a practical resilient control algorithm is proposed in this paper to deal with wireless link/sensor failure in WNCSs.

2.2. WNCS implementation challenges

There has been a growing interest in adopting wireless sensors for industrial automation and control applications, in particular, for distributed control systems in which the communication infrastructure uses wireless networks [17]. For example, in many retrofitting projects of old buildings, wireless temperature sensors can be added, without adding more cabling work in wall or ceiling if wired sensors were used, to improve efficiency of energy usage of Heating Ventilating and Air Conditioning (HVAC) systems. However, there are many challenges for closing the control loop over wireless networks: closed-loop control requires that data flows from the sensors to the controller, reliably; wireless networks are subject to interference and cannot guarantee the timely flow of data; and disruption in feedback data degrades

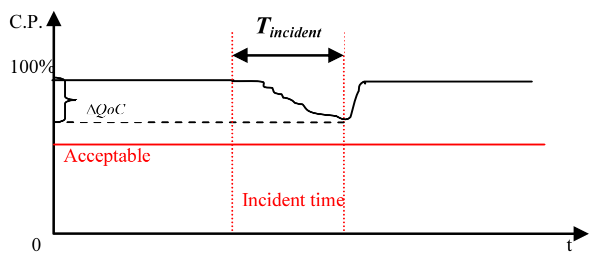

Fig. 1. Illustration of generic Resiliency Index (RI).

control performance and even leads to system shutdown. In summary, the problem here is that there are possible incidents, e.g., link failures due to RF interferences, the sensor data received by the controller are disrupted which may lead to unnecessary and costly system shutdown, as depicted in Fig. 2. To address this problem, a model-prediction based resilient control algorithm (MPRCA) is proposed in this paper, as shown in Fig. 3. The MPRCA performs sensor data filtering and prediction based on a Modified Kalman Filter and provides alarm signal if the confidence level of the filtered or predicted sensor data are below certain tolerable threshold because of long-time link failure or sensor failure. The goal of this MPRCA is to enable resilient control against wireless link issue due to RF interferences to avoid unnecessary system shutdown. The risk assessment and alarm mechanism of the MPRCA works as a quality gauge and provides risk assessment of the wireless sensor data. It constantly monitors the data confidence level and takes specific actions (e.g., sending alarm if link failure or sensor failure happens for a certain period of time) when the risk becomes excessive. The users can set up the confidence level, the error

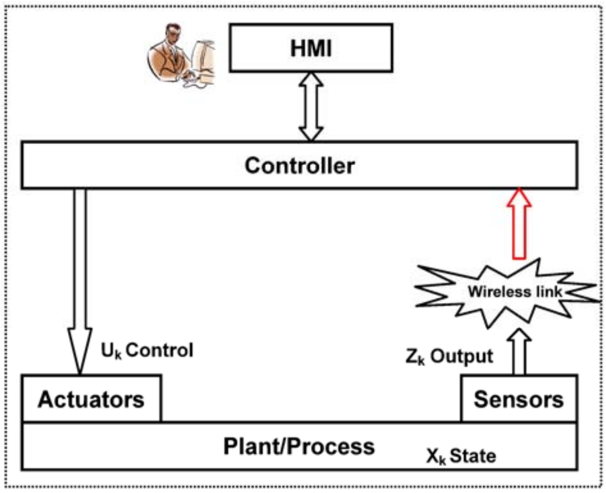

Fig. 2. A WNCS with wireless link in the control loop.

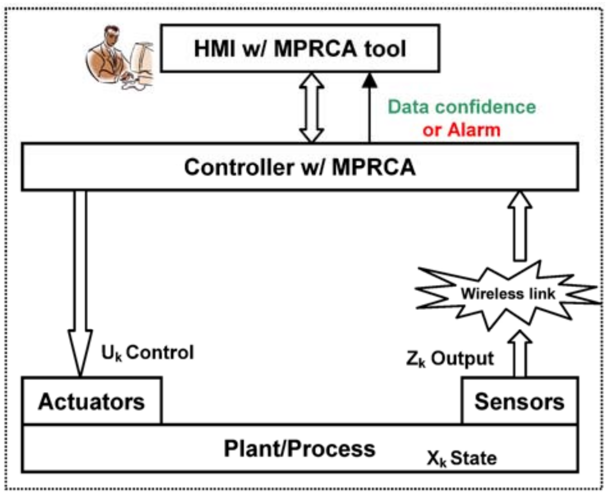

Fig. 3. A WNCS with proposed MPRCA.

tolerance level, and etc. through a Human Machine Interface (HMI).
The NCS shown in Fig. 2 can be described by

$$x_{k+1} = Ax_k + Bu_k + w_k,\\ z_k = Cx_k + v_k, \tag{2}$$

where $x_k \in R^n$ is the state, $u_k \in R^m$ is the control input and z_k is the controlled process output. *A, B* and *C* are the corresponding system matrices with appropriate dimensions. The random variables w_k and v_k represent the process and measurement noise, respectively. They are assumed to be independent of each other, and with following normal probability distributions.

$$w \sim N(0,Q),\\ v \sim N(0,R). \tag{3}$$

3. INTELLIGENT RESILIENT CONTROL ALGORITHM

The proposed MPRCA shown in Fig. 3 has three basic functionalities:
1) Sensor Data Filtering
2) Sensor Data Prediction
3) Risk Assessment & Alarm Mechanism.

3.1. Sensor data filtering

Based on Kalman Filtering [16], predicted state and error covariance for (2) are

$$\hat{x}_{k+1}^- = A\hat{x}_k + Bu_k,\\ P_{k+1}^- = AP_kA^T + Q, \tag{4}$$

where $\hat{x}_{k+1}^-$ is the state prediction and P_{k+1}^- is the error covariance before correction. When new sensor measurement z_{k+1} is available, then the above state prediction and error covariance are updated as

$$K_k = P_{k+1}^- C^T (CP_{k+1}^- C^T + R)^{-1},\\ \hat{x}_{k+1} = \hat{x}_{k+1}^- + K_k(z_{k+1} - C\hat{x}_{k+1}^-),\\ P_{k+1} = (I - K_kC)P_{k+1}^-,\\ P_k \underset{k\to\infty}{=} P_{\min}. \tag{5}$$

The Kalman Filter is essentially a set of mathematical equations that implement a prediction-correction type estimator which minimizes the estimated error covariance and generates optimal estimate of the desired system states. Note that, we assume the applications the proposed algorithm can apply are linear control systems, therefore Kalman filter can be applied. The applicability of the Kalman Fitler is certainly an interesting topic but not the focus of this paper.

3.2. Sensor data prediction

If sensor measurement z_{k+1} are missing due to wireless link failure caused by RF interference, then the predicted sensor data are

$$K_k = P_{k+1}^- C^T (CP_{k+1}^- C^T + R)^{-1},\\ \hat{x}_{k+1} = \hat{x}_{k+1}^- + K_k(z_{k+1} - C\hat{x}_{k+1}^-),\\ P_{k+1} = (I - K_kC)P_{k+1}^-,\\ P_k \underset{k\to\infty}{=} P_{\min}. \tag{6}$$

Time-out scheme, as discussed in [8], is used in the MPRCA to guarantee timely delivery of sensor data. The MPRCA sends the filtered sensor data to the controller if the new sensor data are available. If no new sensor data are available within the timeout threshold, the MPRCA provides the predicted sensor data to the controller together with an updated confidence level to indicate how good the prediction is. If the new sensor data z_{k+1} arrived after the timeout threshold, but before the end of the next sampling period, it will still be used to update or correct the current prediction as in (7). If the new sensor data arrived after the next sampling period, it is considered being lost. The basic idea is that older data gets discarded since the newer data is more accurate to reflect the real situation of the system for the controller. Therefore, in the MPRCA, sensor data packet delay and loss are dealt with in a unified way.

$$K_k = P_{k+1}^- C^T (CP_{k+1} C^T + R)^{-1},\\ \hat{x}_{k+1} = \hat{x}_{k+1} + K_k(z_{k+1} - C\hat{x}_{k+1}),\\ P_{k+1} = (I - K_kC)P_{k+1}. \tag{7}$$

If consecutive sensor measurements are missing due to link failures, then the corresponding consecutively predicted sensor measurements are

$$\begin{aligned}\hat{z}_{k+1} &= CA\hat{x}_k + CBu_k\\ &= CA\hat{x}_k^- + CAK_{k-1}(z_k - C\hat{x}_k^-) + CBu_k,\\ \hat{z}_{k+2} &= CA\hat{x}_{k+1} + CBu_{k+1}\\ &= CA^2\hat{x}_k^- + CA^2K_{k-1}(z_k - C\hat{x}_k^-) + CABu_k + CBu_{k+1},\\ &\ldots\\ \hat{z}_{k+n} &= CA^n\hat{x}_k^- + CA^nK_{k-1}(z_k - C\hat{x}_k^-) + \sum_{j=1}^{n} CA^{j-1}Bu_{k+j-1},\end{aligned} \tag{8}$$

where n is the number of the consecutively missing sensor measurements. It is intuitive that the error covariance will increase when the predicted sensor data are kept being used instead of the real measurements. Fig. 4 shows a generic relation between link failure and the

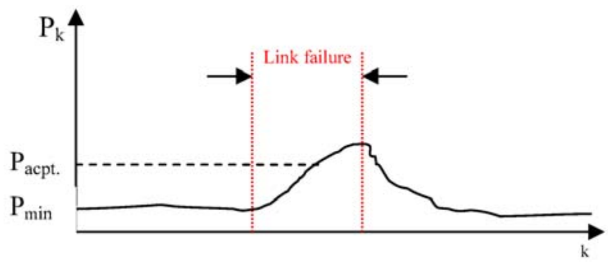

Fig. 4. Error covariance vs. link failure.

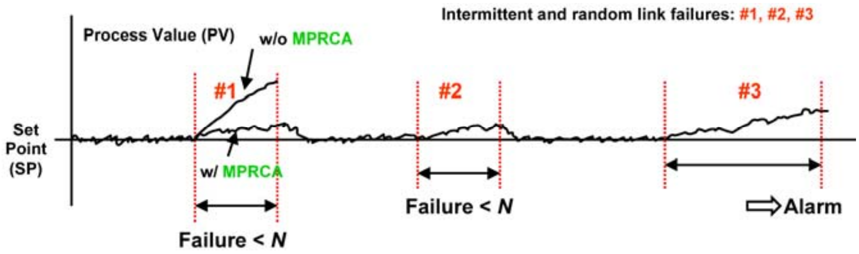

Fig. 5. MPRCA design principle.

estimation error covariance. The confidence level of the predicted sensor data will become lower and lower because of extrapolating from missing sensor data. There needs to be a risk assessment and alarm mechanism to constantly monitor the data confidence level and take specific actions when the risk becomes excessive.

With this sensor data prediction algorithm, the issues of sensor data packet dropout, sensor data delay and jitters caused by the unreliability of wireless network are dealt with in a unified way. Controller computation can also cause delay which affects the control performance, but this is not the focus of this paper. Since now days the controllers usually have very high speed CPUs, e.g., SIMENS S7 PLCs, the computation delay can be ignored comparing with the delay caused by network data transmission.

3.3. Risk assessment & alarm mechanism

For (8), the error covariance of the predicted value is shown in Table 1. Note that n is the number of the consecutively missing sensor measurements

3.4. Data confidence level and QoC

The definition of sensor data confidence level has to be considered together with the control system QoC. The confidence level of the sensor data is defined in such a way that it is related to the system control performance. Most of the industrial process controls are regulator controls, i.e., the controlled process output value (PV) should follow the system set point (SP). Confidence level of sensor data is defined as the probability (Pro.) of that the difference between PV and SP is less than a certain PV error tolerance value a. This can be described as

$$\begin{aligned}\{Confidence\ Level\} &= \{Control\ Performance\}\\ &:= \left\{ \text{Pro.}\left(\left\| \frac{SP-PV}{SP} \right\| \le a \right) \right\}.\end{aligned} \tag{9}$$

Table 1. Estimated error covariance.

Predicted Data	Error Covariance P(n)
$\hat{x}_{k+1}^{-}$	P_{k+1}^{-}
$\hat{z}_{k+1}$	$CP_{k+1}^{-}C^T + R$
$\hat{x}_{k+2}$	$AP_{k+1}^{-}A^T + Q$
…	…
$\hat{x}_{k+n}$	$A^{n-1}P_{k+1}^{-}(A^T)^{n-1} + A^{n-2}Q(A^T)^{n-2} + ...AQA^T + Q$

The control system QoC can be defined based on above definition of control performance as

$$\begin{aligned}\{QoC\} &:= \{Confidence\ Level\ \ge b\}\\ &= \left\{ \text{Pro.}\left(\left\| \frac{SP-PV}{SP} \right\| \le a \right) \ge b \right\},\end{aligned} \tag{10}$$

where b is the user confidence tolerance value, i.e., QoC threshold. Poor QoC, i.e., the data confidence level or control performance becoming lower than what is desired or expected can be defined as:

$$\begin{aligned}\{Confidence\ Level < b\} &= \left\{ \text{Pro.}\left(\left\| \frac{SP-PV}{SP} \right\| \le a \right) < b \right\}\\ &= \left\{ \text{Pro.}\left(\left\| \frac{SP-PV}{SP} \right\| > a \right) \ge 1-b \right\}.\end{aligned} \tag{11}$$

The confidence level can be used to indicate how good the prediction is and provides real time control performance information as defined in (9). When the confidence level drops below the threshold, as described in (11), alarm can be raised to notify the user. This continuously updating information of control performance or alarm can be sent to the HMI as shown in Fig. 3. This information also can be used in the control law design to further improve the control performance, but controller or control law design is not the focus of this paper. Actually, the controller design can be independent to this resilient algorithm. The resilient algorithm proposed in this paper can be used with existing controller design to provide a continuously flow of sensor data.

3.5. Design principle

The MPRCA is designed to tolerate short-time link failures to avoid costly system shut down. The design principle of this MPRCA is depicted in Fig. 5 with three possible failure scenarios. Two short-time link failures #1 and #2, and one long-time link failure #3 are assumed to illustrate the design principal. If there is no link failure, the controlled process value is close to the set point. When the short-time link failure (e.g., #1 or #2) occurs, with MPRCA, the system performance is acceptable, without MPRCA, the system may be out of control. When long-time link failure occurs (e.g., #3), the alarm is triggered if the MPRCA detects that the number of consecutive sensor data packet loss has past a threshold value N. MPRCA also enables users to configure this threshold value N based on their defined QoC including confidence tolerance or QoC threshold b as defined in (10) and process value error tolerance a as defined in (9). It is intuitive that looser tolerance (low QoC requirement) will get bigger N and tighter tolerance (high QoC requirement) will get smaller N.

3.6. Finding N

The threshold N indicates how many sensor measurements loss that the control system can tolerate due to the incident of wireless link failure. The conceptual expression of finding this N is to find the smallest value of N such that the confidence of prediction becomes lower than desired, i.e., $\text{Pro.}\left(\left\|\frac{SP-PV}{SP}\right\| \le a\right) < b.$

Numerical calculation can be done by solving the following equation:

$$P(N) = A^{N-1}P(A^T)^{N-1} + A^{N-2}Q(A^T)^{N-2} + \ldots + AQA^T + Q,$$

$$\int_{a\cdot\|SP\|}^{\infty} \frac{1}{\sqrt{2\pi \cdot tr\left[CP(N)C^T\right]}} \exp\left(-\frac{x^2}{2\cdot tr\left[CP(N)C^T\right]}\right) dx \ge (1-b). \tag{12}$$

Through the HMI, the user can define system model, QoC requirements including PV error tolerance a and confidence tolerance b. Then the MPRCA can compute the smallest value of N as the threshold value for the alarm setting.

3.7. Resiliency index of MPRCA

We can apply the generic RI definition in Eqn. (1) to the proposed MPRCA and one possible definition of the Resiliency index of WNCS with MPRCA can be

$$RI_{MPRCA} := \frac{NT}{a(1-b)}, \tag{13}$$

where N, a and b are defined in (8), (9) and (10), respectively. T is the sampling period of the control system. This definition of resiliency is further illustrated in Fig. 6. Control performance defined in (9) and QoC defined in (10) are used to evaluate the system resiliency.

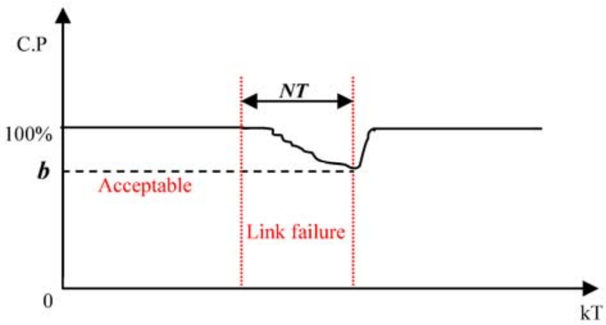

Fig. 6. Illustration of the RI of MPRCA.

From (13) and Fig. 6, it is clear that bigger N indicates higher resiliency which means the system can tolerate longer link failure without jeopardizing control performance. For the same N, smaller a or bigger b indicates higher resiliency which means that better control performance can be expected comparing with systems with lower resiliency in the presence of certain link failure.

4. CASE STUDY

Two applications were developed to verify the MPRCA algorithm proposed in the previous section.

4.1. Light intensity control

4.1.1 Demo setup

The first example is a SISO (Single Input Single Output) control system: light intensity control, which is a typical closed loop feedback control system. The purpose of this demo is to show the impact of losing sensor data and demonstrate that the MPRCA can improve control system resiliency with respect to wireless link failures. Fig. 7 shows the demo setup. The process to be controlled is the light intensity in the circled area. Wireless light intensity sensor is used to feed the light intensity information back to the controller. There are two light bulbs. The bulb #1 is the main light source. It is driven by the actuator, and controlled by the controller that is implemented in a laptop. The bulb #2 is working as an independent light disturbance which is randomly changing. The control objective of this system is to keep the light intensity constant in the presence of light disturbance. The MPRCA function is implemented in the laptop with the controller. Wireless gateway receives the sensor data from the wireless sensor and provides them to the MPRCA and controller. Another laptop directly connected to the wireless sensor is working as a monitor only to display the original sensor data, and it does not participate in any control tasks. So there are two sensor data display windows.

The monitor laptop displays the original sensor data before they are transmitted through the wireless link. The controller laptop displays the received sensor data from the wireless link. To build up a strong and steady RF interference source, a WiFi video streaming set up is used. This video streaming set up is placed in the same room and causes strong RF interference to the light intensity control system. When the video is being streamed through WiFi, it causes a lot of RF jamming to the wireless sensor data transmission and there are many link failures between the wireless light sensor and the wireless gateway, i.e., there are many sensor data delays and losses in the closed control loop.

The transfer function model of the process shown in Fig. 7 is

$$G(s) = \frac{0.66}{1+0.125s} \tag{14}$$

with sampling time of 0.125 second the corresponding

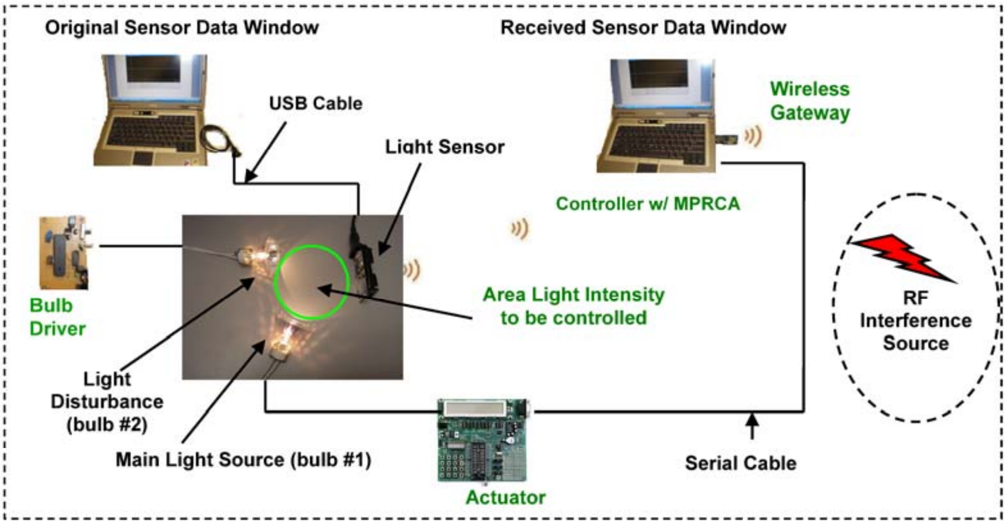

Fig. 7. Light intensity control demo setup.

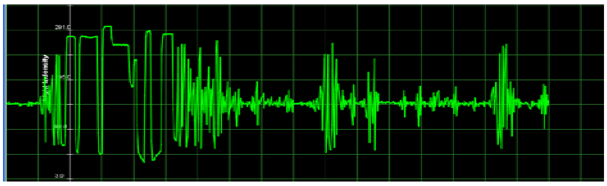

Fig. 8. Source sensor data (MPRCA off).

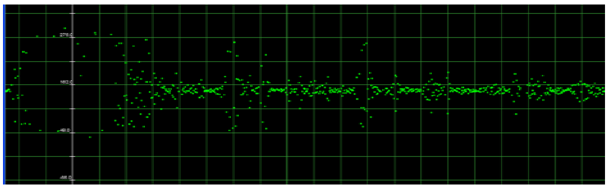

Fig. 9. Received sensor data (MPRCA off).

system parameter A = 0.367879, B = 0.079015, and C = 5.28.

4.1.2 Experiment results

1) MPRCA turned off

In the first experiment, the MPRCA was turned off, when the video streaming was turned on, the light intensity control system went out of control and the main light bulb just kept flashing, as shown in Figs. 8 and 9. In Fig. 9, the gaps between points clearly show that sensor data were lost. In this scenario, the controller uses the previously received sensor data if new sensor data is not available.

2) MPRCA turned on

In the second experiment, the MPRCA was turned on, when the video streaming was turned on. The light intensity control system maintained its stability and the light intensity was under control, as shown in Figs. 10 and 11. Note that in Fig. 11, the red points denote the predicted sensor data provided by the MPRCA when original sensor measurements are lost due to link failures.

This demo verifies that the MPRCA can significantly improve system resiliency with respect to sensor data packet loss. The system with the MPRCA turned off can

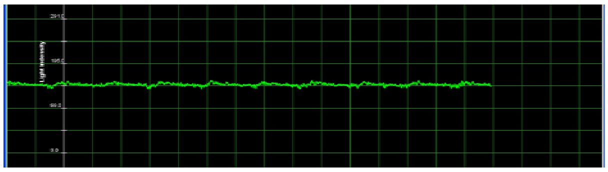

Fig. 10. Original sensor data (MPRCA on).

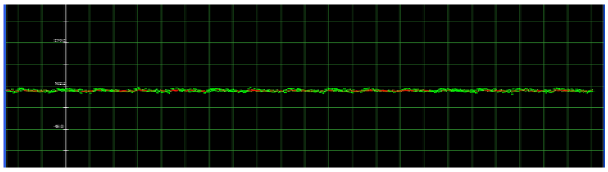

Fig. 11. Received sensor data (MPRCA on).

only tolerate about 20% packet losses, while with the MPRCA turned on it can tolerate about 80% packet losses.

4.2. Distillation column simulation

4.2.1 Simulation setup

The second example is a MIMO system: distillation column which is very commonly used in refineries and chemical plants, as shown in Fig. 12. This distillation column is simplified to a 2 by 2 MIMO system of composition control, although the real model is very complex. Other control loops, e.g., levels, temperature and pressure control are controlled by separate controllers. These controllers are usually designed separately and omitted here for the sake of simplicity. Two materials are separated because of different boiling points. The two input variables are reflux flow ***L*** and boil up flow ***V***. This is a so called LV-configuration. The output variables are distillate product composition ***xD*** and bottom product composition ***xB***. The controller is a dynamic matrix controller (DMC) which is a type of model predictive controller (MPC). This example demonstrated that the alarm mechanism is a necessary component of the MPRCA. The simulation is implemented in MATLAB/Simulink. The system parameters come for a practical process model which contains 4 by 4

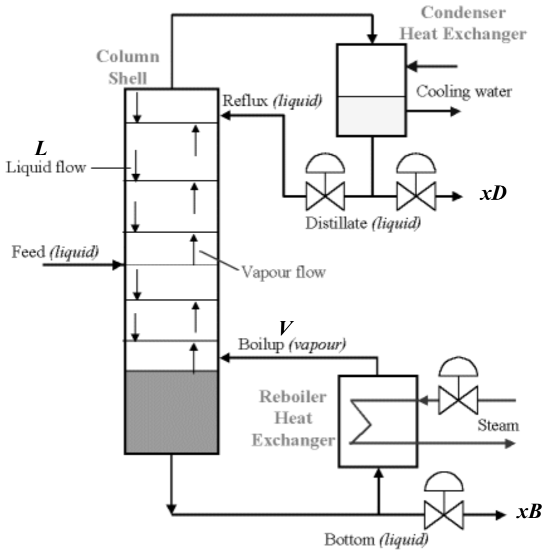

Fig. 12. Simple distillation column controlled with the LV-configuration.

matrices with very long floating point numbers. The detailed system model and controller model are omitted here for the sake of brevity.

4.2.2 Simulation results

Simulation results are shown in Figs. 13 and 14 (in the next page). Only one output variable ***xD*** is shown here, the bottom product ***xB*** has similar result. Three link failures were introduced, the first one had a relatively short failure N_1 = 50, the second and the third ones had longer time N_2 = 90. The system output without the MPRCA is shown on the top of Fig. 13. The system output with the MPRCA is shown on the bottom of Fig. 13. To show the difference, the alarm mechanism was not set for the second link failure and was set for the third link failure. Fig. 13 clearly shows that the MPRCA can improve system resiliency with respect to link failures. Fig. 14 is a closer look at these three link failures. The PV error tolerance a is set as 5%, and the user confidence tolerance b is set as 99%. The MPRCA alarm configuration tool (12) computed the minimum N as 55. For the first link failure, N_1 was less than 55, thus the PV didn't go out of bound. For the second failure, N_2 is larger than 55, the alarm mechanism was not set. It is shown in Fig. 14 that the PV went out of the error bound. For the third failure, we set the alarm and when alarm was trigged, we resumed the link and the PV did not go out of bound. This example demonstrated that the alarm mechanism is a necessary supplement of the MPRCA, since predicted data cannot be used continuously without having any idea about how good the prediction is and how it will affect the control system performance.

5. CONCLUSIONS

This paper proposed a quantitative definition of "resiliency" which can be used for better evaluation of the performance of so-called "resilient control systems". Based on this definition, an intelligent resilient control algorithm MPRCA for WNCSs with wireless sensors in the feedback loop is proposed. The objective of the MPRCA is to improve the resiliency of control systems that use wireless communication between sensors and the controller. Within this MPRCA, wireless sensor data are filtered or predicted by using Kalman Filtering theory depending on availability. The Risk Assessment & Alarm Mechanism automatically determines the risk of extrapolating from missing sensor data due to link failures and takes specific actions when the risk becomes excessive. Two applications were developed to verify and validate the feasibility and effectiveness of the

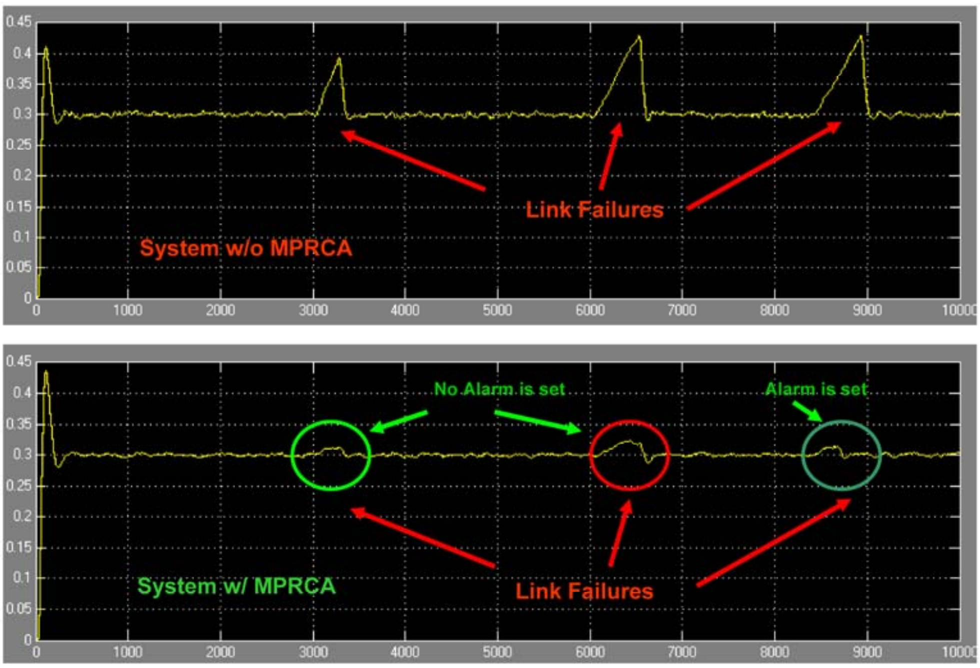

Fig. 13. MPRCA performance on distillation column.

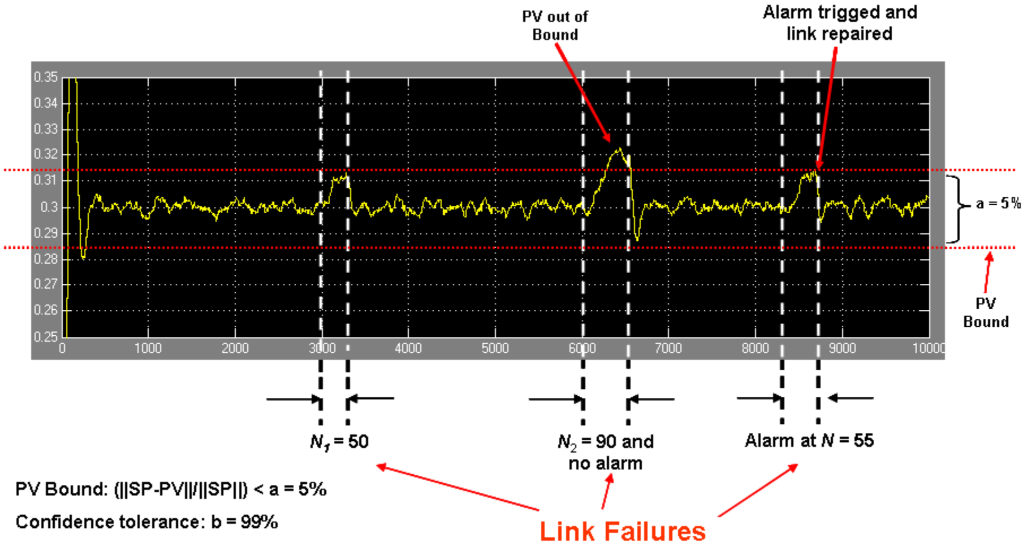

Fig. 14. Closer look at the link failures with and without alarm set.

proposed algorithm. The results show that the proposed method can improve the control system performance in terms of resiliency with respect to wireless link failure. We anticipate that this technology can be used for next generation of industrial automation and control systems over wireless networks.

This paper is only focused on sensor data packet delay and loss, and addresses these issues accordingly. The system resiliency is defined based on QoC during undesirable incidents. However, further study is needed on how fast the system comes back to its original performance after the incident is removed.

REFERENCES

[1] S. M. Mitchell and M. S, Mannan, *Designing Resilient Engineered Systems*, Chemical Engineering Progress, vol. 102, no. 4, pp. 39-45, April 2006.

[2] *Proceedings of the 1st International Symposium on Resilient Control Systems*, Idaho Falls, ID, Sep. 2008. https://secure.inl.gov/isrcs2008/default.aspx.

[3] *Proceedings of the 2nd International Symposium on Resilient Control Systems*, Idaho Falls, ID, August, 2009. https://secure.inl.gov/isrcs2009/default.aspx.

[4] C. G. Rieger, D. I. Gertman, and M. A. McQueen, "Resilient control systems: next generation design research," *Proc. of the 2nd IEEE Conference on Human System Interaction*, pp. 636-636. Catania, Italy, May 2009.

[5] D. Wei and K. Ji, "Resilient industrial control system (RICS): concepts, formulation, metrics, and insights," *Proc. of the 3rd IEEE International Symposium on Resilient Control Systems*, pp. 15-22, Aug. 2010.

[6] K. Ji and W.-J. Kim, "Real-time control of networked control systems via ethernet," *International Journal of Control, Automation, and Systems*, vol. 3, no. 4, pp. 591-600, Dec. 2005.

[7] C. Ming, T. Wang, W. Chou, S. Luan, Y. Zhang, and Z. Tina, "Remote surgery case: Robot-assisted teleneurosurgery," *Proc. of IEEE Int. Conf. Robot. & Auto.*, vol. 1, pp. 819-823, Apr. 2004.

[8] W.-J. Kim, K. Ji, and A. Srivastava, "Network-based control with real-time prediction of delayed/lost sensor data," *IEEE Trans. on Control System Technology*, vol. 14, no. 1, pp. 182-185, Jan. 2006.

[9] K. Ji, W.-J. Kim, and A. Srivastava, "Internet-based real-time control architectures with time-delay/packet-loss compensation," *Asian Journal of Control*, vol. 9, no. 1, pp. 47-51, March 2007.

[10] J. P. Hespanha, P. Naghshtabrizi, and Y. Xu, "A survey of recent results in networked control systems," *Proc. IEEE*, vol. 1, pp. 138-162, Jan. 2007.

[11] G. C. Walsh, Y. Hong, and L. G. Bushnell, "Stability analysis of networked control system," *IEEE Trans. Contr. Syst. Technol.*, vol. 10, no. 3, pp. 438-446, May 2002.

[12] W. Zhang, M. S. Branicky, and S. M. Phillips, "Stability of networked control systems," *IEEE Contr. Syst. Mag.*, vol. 21, no. 1, pp. 84-99, Feb. 2001.

[13] N. J. Ploplys, P. A. Kawka, and A. G. Alleyne, "Closed-loop control over wireless network," *IEEE Contr. Syst. Mag.*, vol. 24, no. 3, pp. 58-71, Jun. 2004.

[14] G. N. Nair, F. Fagnani, S. Zampieri, and R. J. Evans, "Feedback control under data rate constraints: an overview," *Proc. IEEE*, vol. 95, pp. 108-137, Jan. 2007.

[15] K. Ji and W.-J. Kim, "Stabilization of networked control system with time delays and data-packet losses," *European Journal of Control*, vol. 13, no. 4, pp. 1-8, Sep. 2007.

[16] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering*, vol. 82D, no. 1, pp. 35-45, 1960.

[17] M. Rabi and K. H. Johansson, "Event-triggered strategies for industrial control over wireless networks," Invited paper, *Proc. of Wireless Internet Conference*, Maui, HI, USA, 2008.

Kun Ji received his B.S. and M.S. degrees in Precision Instrumentation and Engineering from Tsinghua University, Beijing, China in 1999 and 2002, respectively, and his Ph.D. degree in Mechanical Engineering from Texas A&M University, College Station, in 2006. He has been with Siemens Corporate Research, Princeton, where currently he is a research scientist. His research interests focus on analysis, design of networked control systems, real-time control systems, wireless sensor networks and resilient control systems.

Dong Wei is a research scientist at Siemens Corporate Research, Princeton, New Jersey. He has worked at Siemens for 10 years. He received his Bachelor degree in Electrical Engineering from Tsinghua University in Beijing, China. He received his master and Ph.D. degrees from New Jersey Institute of Technology, both in Electrical Engineering. He has worked on factory automation system, PLC, motion control, Human-Machine Interface, drive system, and industrial communication networks for more than 10 years. Dr. Wei has 20 publications, including book chapters, journal papers. He is also an active reviewer of IEEE Transactions on Smart Grid, IEEE Transactions on Automation Science and Engineering, IEEE Transactions on Industrial Informatics, IEEE Transactions on System, Man, and Cybernetics, Computer in Industry, Journal of the Network and Systems Management, etc.