IET Cyber-Physical Systems: Theory & Applications

Research Article

Novel cyber fault prognosis and resilience control for cyber–physical systems

ISSN 2398-3396
Received on 18th September 2018
Revised 12th January 2019
Accepted on 29th January 2019
doi: 10.1049/iet-cps.2018.5061
www.ietdl.org

Shanshan Bi[1] ✉, Tianchen Wang[2], Lei Wang[3], Maciej Zawodniok[1]

[1]*Electrical and Computer Engineering Department, Missouri University of Science and Technology, Emerson Electric Co. Hall, 301W. 16th St., Rolla, MO, 65401, United States*

[2]*Computer Science and Computer Engineering Department, University of Notre Dame, Notre Dame, IN, 446556, United States*

[3]*Department of Computer Science, Union College, 807 Union Street Schenectady, NY, 12308, United States*

✉ *E-mail: sbn65@mst.edu*

Abstract: Cyber–physical systems (CPSs) consists of a network, computation, and physical process. Embedded networks, which deliver control and sensing signal, can potentially affect CPSs performance. However, the degradation of physical system performance caused by the embedded networks is frequently oversimplified with strong assumptions. The proposed scheme effectively relaxes those assumptions in the existing works that network delays are bounded in a specific range or its distribution is time invariant. Most of the existing works on fault diagnosis and prognosis addressed the physical system fault detection and isolation, and ignore cyber network faults. A novel cyber network fault prognosis scheme is proposed to deal with both of cyber and physical system fault. It can identify when a cyber fault has occurred, and pinpoint the type of fault based on CPS system performance prediction, then, trigger resilience controller at an appropriate time to minimise the computational overhead. Thus, it can guarantee the stability of the entire CPS and substantially reduce computational overhead of the resilience control by triggering it if necessary.

1 Introduction

Cyber–physical systems (CPSs) refer to systems with integrated computational network and physical components. With the increasing connectivity among the cyberspace and physical systems, capturing the interactions between the cyber and the physical systems becomes increasingly important [1]. In particularly, network imperfections and dynamics – such as limited channel capacity, traffic congestions, and malicious attacks – can degrade the performance or even destabilise the control system. This makes controller design more challenging and complex. In the existing literature, this issue is often oversimplified when designing CPS controllers and could result in severe system failure. For example, hackers can remotely take control of a vehicle and cut its transmission on the highway [2]. The threat of automotive cyberattacks also threatens people's life. Therefore, detection, estimation, isolation, and mitigation scheme of cyberattacks/faults has to be investigated to improve the resilience of the entire CPSs.

In the past few years, many control and system researchers have pioneered the development of approaches and tools to model and control CPSs. [3–7] addressed the fault detection, isolation, and mitigation in the physical subsystem alone (e.g. actuators, sensors, and controlled components). At the same time, communication and signal process researchers have made major breakthroughs in monitoring, identification, and defence of cyberattacks and other security issues on the cyber side [8–17]. Such existing approaches focus on either cyber or physical control aspects while ignoring or oversimplifying the other aspect. However, such decoupled designs will often fail in practical CPS. Therefore, to address the aforementioned issues, a control and fault prognosis system redesign which takes into account the interaction between cyberspace and physical systems is necessary.

Inspired by this motivation, we proposed a novel prognosis scheme in this work. The main contributions are:

(a) Proposed a novel prognosis scheme for cyber network fault detection and prediction.
(b) Derived the estimation of network delay distribution based on time-series analysis. The convergence of the estimation error is presented in Lemma 1.
(c) Proposed an isolation scheme to distinguish soft and hard faults based on the prediction of potential failures on system states. Theorem 1 shows the convergence of such prediction.
(d) Developed a decision-making scheme for resilience control triggering. The simulation results in Section 6 illustrate that this scheme proactively trigger the resilience control and effectively avoid physical system failure.

The rest paper is organised as following. In Section 2, a motivation example is given to illustrate the relationship of cyber condition and system behaviour. Next, the related works on fault diagnosis and prognosis are presented in Section 3. In Sections 4 and 5, the proposed prognosis scheme is demonstrated. The simulation results are shown in Section 6 and the conclusions are given in Section 7.

2 Motivation

In this section, the design challenge introduced by the interaction between cyber network and physical system is discussed in a simple scenario. This example emulates a route hijacking by an attacker who is eavesdropping the control information which could be later used for taking over the control. The network delay and delay variation will be increased when the attacker secretly relays and possibly alters the communication between the controller and actuators. In this section, the network is simulated using Network Simulator 2 (NS2) with a random topology of 11 nodes. *Ad hoc* on-demand distance Vector (AODV) routing scheme is adopted. In Fig. 1*a*, an attacker relayed the transmission. The change of the network topology introduced sudden packet delays and variation for the control loop. Similar network performance could be a result of topology or traffic pattern changes. Then, an optimal controller to regulate a two input four output (2I4O) system [18].

The network topology and traffic changes will also lead to the disturbance in CPS. In Fig. 1*b*, a route hijacking attack is launched at $t = 10.5$ s. The sudden changes of the network delay make the optimal controller controlled system state start to oscillate. The network dynamic leads to unstable CPS.

Therefore, cyber network attacks indeed affect the system performance. Cyberspace is particularly difficult to secure due to its vulnerabilities of connections between cyber and physical

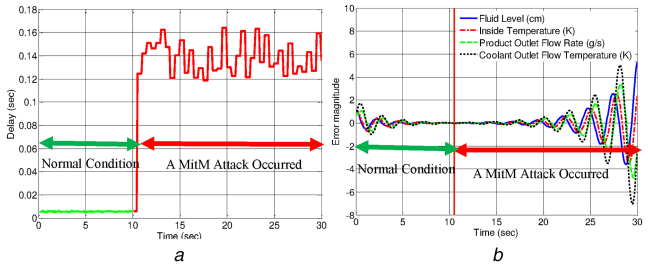

Fig. 1 *The effect of delay variation on system stability*
(a) Delays, ***(b)*** Tracking errors of optimal controller

systems. Of growing concern is the cyber threat to critical hardware devices. Cyber network faults could cause harm or disrupt services upon which our economy and the daily lives depend on. In light of the risk and potential consequences of cyber events, strengthening the risk awareness and resilience of CPSs has become an important mission.

To address the cyber network fault issues on the physical system side, a full knowledge of the relation between cyber condition and system performance is essential. Hence, we propose a scheme for detecting and isolating cyber and physical system faults. Also, a resilience control triggering strategy is proposed to proactively trigger the controller and accommodate the potential failures ahead of time.

3 Related works

In this section, the existing works on cyber security is briefly discussed first. Then, the works on fault awareness of the physical system are discussed.

The overall goals of cyber security include integrity (the trustworthiness of data or resources), availability (accessibility upon demand), and confidentiality (keeping information secret from unauthorised users). Many researchers addressed these issues with different technologies, such as authentication schemes, access control, and other defence scheme [9–17]. An assumption that the adversary/attack model is fully known is often required; however, it is challenging to obtain. In [10], deception and denial of service attacks against a networked control system (NCSs) are addressed. They proposed a countermeasure based on semi-definite programming. This work and the following literature are only valid for a specific attack model, which cannot be known in priori. A defence scheme without requiring the knowledge about the attack model is needed.

In [11], false data injection attacks against static state estimators are introduced. Undetectable false data injection attacks can be designed even when the attacker has limited resources. Also, stealthy deception attacks against the supervisory control and data acquisition system are studied in [12]. In [13], the effect of replay attacks on a control system is studied. In [14], the effect of covert attacks against control systems is investigated. A parameterised decoupling structure alter the behaviour of the physical plant while remaining undetected from the original controller. Then, [15] proposed a general theory of event compensation as an information flow security enforcement mechanism for CPSs. Message scheduling methods were given to improve the security quality of wireless networks for mission-critical CPSs in [16].

With respect to the above works, [17] proposed a mathematical framework for CPSs, attacks, and monitors, and given the fundamental limitations of monitors from system-theoretic and graph-theoretic perspectives. Finally, centralised and distributed attack detection and identification monitors were designed. Overall, many cyberattacks can be addressed on the cyber side. However, the effects of cyberattacks/faults on the physical system behaviour are oversimplified in the above-mentioned existing works. Moreover, the injection time and model of the attacks/faults are difficult to learn ahead of time in practical CPSs.

The control researchers focused on the conventional fault detection techniques that have successfully applied to industrial NCSs. They indeed considered the network delay and packet loss in various ways. In [19], a resilient control problem is studied, in which control packets transmitted over a network are corrupted by a human adversary. They proposed a receding-horizon Stackelberg control law to stabilise the control system despite the attack. However, the proposed approach required a priori knowledge on attack model and type. In [3], network delays were modelled as a constant delay (time buffer), an independent random delay, and a delay with known probability distribution governed by the Markov chain model. In [4], a networked predictive controller in the presence of random delay in both forward and feedback channels was proposed to minimise the effects of network failures. A robust H_∞ control for a non-linear T-S fuzzy model system was proposed to address the network delays and packet drop in [5]. However, they assumed the upper bound of delays is known. This is challenging to be satisfied in reality. Then, [6, 7] employed a state observer–based fault detection method on the uncertain long time delay. Although, the network delays and packet drop caused by network faults/failures were considered in above works, the assumptions, such as known bounds and time-invariant distribution of delays and packet loss, are always made. In addition, most of the above works aimed to detect the faults of physical components (sensors, actuators, and system plant), not the faults in the cyberspace.

This work is motivated to address cyber network faults detection, isolation, and prediction. Meanwhile, the tolerant control scheme and it triggering strategy are proposed to stabilise the CPS despite cyber network faults and optimise the computational overhead.

4 Proposed prognosis scheme

In this section, the overview of the proposed prognosis scheme is given in Section 4.1. An online kernel density estimation (KDE)–based probability density function (PDF) identifier is introduced in Section 4.2. Then, the details of the proposed prognosis scheme are presented in Section 5. At last, the resilience controller is designed in Section 5.3.

4.1 Overview

In this work, the uncertainties in the cyberspace, including traffic congestions, topology changes, and attacks, are causing abnormal delays and packet losses on the physical system side. Monitoring such delays and packet losses online is required for detection of cyber network faults. Moreover, an observer is needed to detect physical system faults and isolate them from cyber network faults [7].

The proposed prognosis scheme is shown in Fig. 2. It includes four main steps that are continuously repeated:

(a) Sensing of network delays. In this work, we assume a transmission control protocol (TCP)-based network, such that the delay is provided by the acknowledgement for each data packets.
(b) Delay update. n delays ($[d_{k-n+1}, \ldots, d_k]$) in the sliding window are used to do the PDF estimation at time k (PDF_k). When the new delay is measured, the data in the sliding window are updated.
(c) Cyber Network Fault Detection. The PDF of these n delays is obtained by using the online KDE-based PDF identifier. The probability for each delay interval P_i^k is calculated. Compare P_i^k to P_i^{k-1} to compute the variation of probabilities ΔP_i^k. If ΔP_i^k exceeds the set threshold R_{p_i}, the PDF variation is captured and a cyber network fault is detected. Meanwhile, if there is no abnormal behaviour presented in the observer, it can be confirmed that only cyber network fault happens.
(d) Potential degradation prediction. If a cyber network fault is detected, the PDF of new delay distribution is predicted by using time-series analysis. Then, the delays following the new distribution are resampled. Finally, the prediction of the future physical system outputs is obtained. If the system states deviate out of the acceptable range, the hard fault is detected. Otherwise, it is a soft fault which is not severe enough to trigger the resilience controller. More details about fault isolation are presented in Section 5.

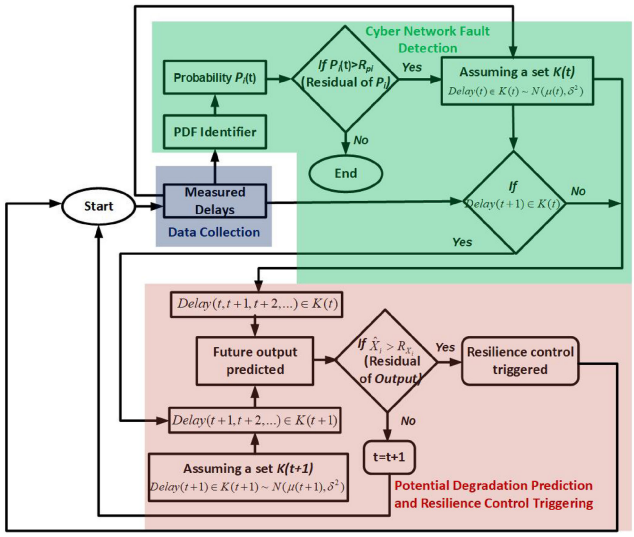

Fig. 2 *Flow chart of cyber network fault prognosis scheme*

(e) Resilience controller triggering. If a hard fault is detected, the resilience controller is triggered and its parameters are tuned online by the probabilities of delays computed in b.

Such a scheme can detect stochastic cyber failures/attacks without requiring the knowledge of attack model and its injection time a priori. Only monitoring the PDF of delays in real-time is required to do PDF and system-state prediction. Moreover, the resilience control law tuned by the probabilities of delays is derived accurately for the given cyber performance. Its details are introduced in Section 5.

4.2 PDF identifier

To obtain the probability information mentioned in Step (b), a PDF identifier is employed [20]. It uses kernel density estimation (KDE) to estimate the distribution of delays iteratively. The data used to make the identification are updated at every sampling interval for a window of n last packet delays. The main steps of online PDF identification are shown in Table 1 (Appendix 9.1). Here, a normal kernel smoother is selected for PDF estimation.

Such a sliding window-based PDF identifier provides the PDF profile of delays in real-time such that the variation of PDF can be captured and observed easily.

5 Cyber network fault detection and isolation

Cyber network fault is detected by monitoring probability residual (PR). The other residuals – modelled system output residual and system performance residual – are used to isolate cyber network and physical components fault. Then, the prediction of the future new delay distribution and system-state prediction is used to isolate soft and hard cyber network faults. Finally, the decision of resilience control triggering is made.

5.1 Cyber network fault detection

For cyber network fault detection, three residuals have to be monitored in an online manner:

(a) Probability residual (*PR*): It is the difference of the probability at k and the last interval $k-1$. Such information is provided by the proposed PDF identifier. The PR at time k for each delay interval is denoted as $\Delta P_i^k = P_i^k - P_i^{k-1}$. The corresponding threshold R_{p_i} is customisable by the users for satisfying the requirement of fault awareness capability. If $\Delta P_i^k > R_{p_i}$, the cyber network fault is detected.

(b) Modelled system output residual (*MSOR*): It is provided by the observer, which is the difference of outputs of modelled and that of actual systems: $\text{MSOR} = x(k) - \hat{x}(k)$. The corresponding threshold

Table 1 Online PDF identification algorithm

1. Determining the data in the sliding window for time k:

a) Choosing a kernel function K centred on x with a bandwidth h;

b) Each observation x_i receives a specific weight proportional to the scaled distance from the observation x_i to x, which is $u = (x - x_i)/h$;

c) At a given x, the estimate is found by vertically summing up over the k shapes.

This can be synthesised as:

$$\hat{f}(x) = \frac{1}{nh} x_i \in \left[x - \frac{h}{2}, x + \frac{h}{2}\right]$$

The general formula for KDE will be given by

$$\hat{f}_k(x) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right)$$

where the dependence of the estimate on the kernel function $K(.)$ is denoted as $\hat{f}_k$.

2. Updating the new data for time $k+1$ in the sliding window and go back to Step 1;

R_{MSOR} is selected to detect physical system faults. If $\text{MSOR} > R_{\text{MSOR}}$, the physical system fault is detected.

(c) System performance residual (*SPR*): It is the difference between actual and desired system outputs: $\text{SPR} = x_d(k) - x(k)$. The threshold for each system output variable R_{SPR} is determined by the acceptable error magnitudes of system states. This residual is used to evaluate the system performance and determine when the system should shut down.

If *PR* exceeds its threshold, a cyber network fault is detected. Meanwhile, *MSOR* and *SPR* should be supervised to do the root-cause analysis of the degradation of system performance. If a cyber network fault is detected, the type of this fault (soft or hard fault) should be learned before triggering the resilience controller. That is because not all types of cyber network fault need to be mitigated by the resilience controller. Unnecessary triggering will result in additional computational resource wasting. When soft faults happen, the adverse effects on system performance can be handled by the existing controller. Therefore, there is no need to take other control actions. Typically, an alarm or warning is sufficient. On the contrary, hard faults potentially threaten the system performance in terms of overshoot, time-to-recover (TTR), and cost of regulation, even stability. Moreover, wear and tear or severe damages of the system components might be induced by such faults. Hence, timely detecting hard faults and triggering the resilience controller are vital for guaranteeing system stability. Moreover, with isolating of soft and hard faults, inefficient triggering of the resilience controller is avoided and the overall computational cost is reduced.

5.2 Soft and hard cyber network fault isolation

To recognise hard cyber network fault, we proposed an approach to trend the delay distribution into the future. Next, a system-state prediction scheme evaluates system performance for the estimated future delay distribution. In most cases, there is no need to perform the computation expensive prediction. Hence, it is triggered based on a user-defined threshold. As shown in Fig. 2, R_{p_i} is a user-defined threshold for each probability variation $P_i(t)$. If $P_i(t) > R_{p_i}$ is true and the new delay Delay(t) follows the distribution of time $t-1$, there is no cyber network fault. The distribution and system-state predictors will not be activated. Otherwise, a delay distribution change will be observed in PDF identifier and the predictors are activated.

The predictions include four main steps that are repeated until the resilience controller is triggered:

Step 1: new distribution estimation;
Step 2: resampling;
Step 3: system output prediction;

Step 4: soft and hard fault isolation and resilience control triggering.

These steps are discussed in details next.

5.2.1 Step 1: New distribution estimation: The expectation and standard deviation of the new distribution are estimated based on the delays which induce a new distribution.

Time series analysis is utilised to estimate the autoregressive (AR) model for the expectation E and standard deviation D of the new distribution. The hypothesis of the model is given by:

$$\begin{bmatrix} E(k+1|k) \\ D(k+1|k) \end{bmatrix} = \begin{bmatrix} \beta_{E0} \\ \beta_{D0} \end{bmatrix} + \begin{bmatrix} \beta_{E1} & 0 \\ 0 & \beta_{D1} \end{bmatrix} \begin{bmatrix} E(k) \\ D(k) \end{bmatrix} \tag{1}$$

$\hat{E}(k+1|k)$ is the forecast of $E(k+1|k)$ and $D(k+1|k)$ based on $E(k)$ and $D(k)$, using the estimated coefficients $\hat{\beta}_{E0}$, $\hat{\beta}_{D0}$, $\hat{\beta}_{E1}$, and $\hat{\beta}_{D1}$.

$$\begin{aligned} \begin{bmatrix} \hat{E}(k+1|k) \\ \hat{D}(k+1|k) \end{bmatrix} &= \begin{bmatrix} \hat{\beta}_{E0} \\ \hat{\beta}_{D0} \end{bmatrix} + \begin{bmatrix} \hat{\beta}_{E1}(k) & 0 \\ 0 & \hat{\beta}_{D1}(k) \end{bmatrix} \begin{bmatrix} E(k) \\ D(k) \end{bmatrix} \\ &= \hat{\theta}(k)\varphi(k) + C_0(k) \end{aligned} \tag{2}$$

where $\varphi(k) = [E(k)\&D(k)]$, and $\hat{\theta}(k) = \begin{bmatrix} \hat{\beta}_{E1}(k) & 0 \\ 0 & \hat{\beta}_{D1}(k) \end{bmatrix}$.

The one-period ahead prediction error is:

$$\begin{bmatrix} e_E(k+1) \\ e_D(k+1) \end{bmatrix} = \begin{bmatrix} E(k+1|k) \\ D(k+1|k) \end{bmatrix} - \begin{bmatrix} \hat{E}(k+1|k) \\ \hat{D}(k+1|k) \end{bmatrix} \tag{3}$$

The prediction errors can converge overtime by minimising the following cost function:

$$J = \begin{bmatrix} e_E(k+1) \\ e_D(k+1) \end{bmatrix}^T \begin{bmatrix} e_E(k+1) \\ e_D(k+1) \end{bmatrix} \tag{4}$$

Such that the update law of $\hat{\theta}(k)$, $L(k)$, and $O(k)$ can be obtained.

$$\begin{aligned} \hat{\theta}(k) &= \hat{\theta}(k-1) + L(k)e(k) \\ L(k) &= \frac{O(k-1)\varphi(k)}{\varphi(k)^T O(k-1)\varphi(k)} \\ O(k) &= (I - L(k)\varphi(k)^T)O(k-1) \end{aligned} \tag{5}$$

where $L(k)$ and $O(k)$ denote estimator gain and estimation of error variance, respectively. Their initial values are randomly set.

Remark 1: The parameters $\hat{\theta}(k)$, $L(k)$, and $O(k)$ are continuously updated by the training data in the sliding window at time K. We assume that the distribution of time $k+1$ does not change so that the one-step prediction for time $k+1$ is valid to predict the system behaviour.

Lemma 1: With the update law (5) and more new delays loaded in the sliding window, the objective index (4) is continuously minimised. Then, the following statements are true:

(a) the estimation errors of the expected value and standard deviation of new delays converge.

(b) $(\| \sum_{j=1}^{n} \tilde{P}_{(k+1)j} \| - \| \sum_{j=1}^{n} \tilde{P}_{kj} \|) < 0$ holds.

Remark 2: Even if the network condition is perfect, unexpected delays, which are out of the healthy range, occasionally occurs in a long period. That can lead to the inefficient triggering of resilience control. Using the above time-series analysis, not only the distribution change can be tracked in real time but also the trend of distribution change is identified and predicted. Such that the occasional event can be filtered without resilience control triggering.

5.2.2 Step 2: resampling: Based on the future delay distribution provided by step 1, a series of random delays is generated, which follows the new distribution. The number of generated delays should be determined by the window size for PDF identification.

5.2.3 Step 3: system output prediction: The resampled delays are fed to the system model which takes into account dynamic delays and packet losses. Such a time-varying system is given by:

$$z(k+1) = A_z(k)z(k) + B_z(k)u(k) \tag{6}$$

where $z = \begin{bmatrix} x(k)^T & u(k-1)^T & \dots & u(k-d)^T \end{bmatrix}^T$ is the state variables vector; u_k is the control input; $A_z(k)$ and $B_z(k)$ are the system dynamic matrices and given by

$$A_z(k) = \begin{bmatrix} A & \gamma(k-1)B_1(k) & \dots & \gamma(k-i)B_i(k) & \dots & \gamma(k-d)B_d(k) \\ 0 & 0 & \dots & \dots & \dots & 0 \\ 0 & I_m & \dots & \dots & 0 & 0 \\ \vdots & 0 & I_m & \dots & \dots & 0 \\ \vdots & \vdots & & \ddots & & \vdots \\ 0 & 0 & \dots & \dots & I_m & 0 \end{bmatrix},$$

$$B_z(k) = \begin{bmatrix} \gamma(k)B_0(k) & I_m & 0 & 0 & \vdots & 0 \end{bmatrix}^T,$$

$$\gamma(k) = \begin{cases} I^{n\times n} & \text{if the control input is received at time } k \\ 0^{n\times n} & \text{if the control input is lost at time } k \end{cases}$$

Finally, the possible system behaviour induced by the new distribution of delays are estimated and denoted as $\hat{z}_k$.

Prediction convergence analysis: The prediction error $\tilde{z}_k$ convergence is demonstrated in Theorem 1. The dynamic matrices A_{zj} and B_{zj} for each delay interval are deterministic and their calculation can be found in [18].

Theorem 1: (Error of system-state prediction convergence): As the delay data keeps updating PDF identifier and $(\| \sum_{j=1}^{n} \tilde{P}_{(k+1)j} \| - \| \sum_{j=1}^{n} \tilde{P}_{kj} \|) < 0$ is satisfied, then the prediction error for system output $\| \tilde{z}(k) \|$ asymptotically converges to zero.

Proof: The prediction error is given by

$$\begin{aligned} \tilde{z}_k &= z_k - \hat{z}_k \\ &= (A_z(k) - B_z(k)K(k))z(k) - (\hat{A}_z(k) - \hat{B}_z(k)K(k))z(k) \\ &= (A_z(k) - \hat{A}_z(k))z(k) - (B_z(k) - \hat{B}_z(k))K(k)z(k) \\ &= (\tilde{A}_z(k) - \tilde{B}_z(k)K(k))z(k) \end{aligned}$$

Therefore, the convergence of $\tilde{z}_i$ can be proven by proving the convergence of $\tilde{A}_z(k)$ and $\tilde{B}_z(k)$

We define the prediction error of $A_z(k)$ as $\tilde{A}_z(k) = A_z(k) - \hat{A}_z(k)$. $A_z(k)$ can be expressed as $\sum_{j=1}^{n} P_j(k)A_{zj}$ $P_j(k)$ is the actual probability at k. Similarly, we denote $\widehat{A_z(k)} = \sum_{j=1}^{n} \hat{P}_j(k)A_{zj}$. $\hat{P}_j(k)$ is the estimate probability provided by the PDF profile. The

estimation error of the probability is $\tilde{P}_j(k) = P_j(k) - \hat{P}_j(k)$. Then, Lyapunov function candidate is $V_{A_z(k)} = \tilde{A}_z(k)^T \tilde{A}_z(k)$.

$$
\begin{aligned}
\Delta V_{A_z(k)} &= \tilde{A}_z(k+1)^T \tilde{A}_z(k+1) - \tilde{A}_z(k)^T \tilde{A}_z(k) \\
&= \left(\sum_{j=1}^{n} P_j(k+1) A_{zj} - \sum_{j=1}^{n} \hat{P}_j(k+1) A_{zj} \right)^T \\
&\quad \times \left(\sum_{j=1}^{n} P_j(k+1) A_{zj} - \sum_{j=1}^{n} \hat{P}_j(k+1) A_{zj} \right) \\
&\quad - \left(\sum_{j=1}^{n} P_j(k) A_{zj} - \sum_{j=1}^{n} \hat{P}_j(k) A_{zj} \right)^T \\
&\quad \times \left(\sum_{j=1}^{n} P_j(k) A_{zj} - \sum_{j=1}^{n} \hat{P}_j(k) A_{zj} \right) \\
&= \left(\| \sum_{j=1}^{n} \tilde{P}_j(k+1) \|^2 - \| \sum_{j=1}^{n} \tilde{P}_j(k) \|^2 \right) \| A_{zj} \|^2 \\
&= \underbrace{\left(\| \sum_{j=1}^{n} \tilde{P}_j(k+1) \| + \| \sum_{j=1}^{n} \tilde{P}_j(k) \| \right)}_{\Delta_1} \\
&\quad \times \left(\| \sum_{j=1}^{n} \tilde{P}_j(k+1) \| - \| \sum_{j=1}^{n} \tilde{P}_j(k) \| \right) \| A_{zj} \|^2 \\
&= \Delta_1 \left(\| \sum_{j=1}^{n} \tilde{P}_j(k+1) \| - \| \sum_{j=1}^{n} \tilde{P}_j(k) \| \right) \| A_{zj} \|^2 \\
\Delta_1 &> 0
\end{aligned}
$$

Since $V_{A_z(k)}$ is positive definite and $\Delta V_{A_z(k)}$ is negative definite provided $(\| \sum_{j=1}^{n} \tilde{P}_j(k+1) \| - \| \sum_{j=1}^{n} \tilde{P}_j(k) \|) < 0$ (Lemma 1). Therefore, the prediction error of $A_z(k)$ asymptotically converge to zero. Similarly, the prediction error of $B_z(k)$ can be proven with the same procedure. Such that $\tilde{z}(k)$ asymptotically converge to zero. □

Remark 3: The maximum error occurs when the first sample of the new distribution comes in the sliding window. Then, the accuracy of PDF estimation improves as the sliding window includes more and more new samples from the new distribution after the PDF change occurs. Therefore, $(\| \sum_{j=1}^{n} \tilde{P}_{(k+1)j} \| - \| \sum_{j=1}^{n} \tilde{P}_{kj} \|) < 0$ holds.

5.2.4 Step 4: soft and hard fault isolation and resilience control triggering strategy: The acceptable error magnitude of state i is defined as $R_{\tilde{z}_i}$. If $\hat{z}_i > R_{\tilde{z}_i}$, this fault is marked as a hard cyber network fault. A warning is triggered as well as the resilience controller. Otherwise, this is a soft fault that can be handled with the original controller operating normally.

In summary, the proposed prognosis scheme can timely detect cyber network faults and isolate soft and hard faults because the dynamics of the network is continuously monitored. Accurately isolating soft and hard fault optimise the decision of resilience controller triggering as well as the computational resources allocation. When hard faults occur, the resilience controller can be timely triggered before adverse effects on system performance happening.

5.3 Resilience control strategy

In this section, the employed resilience controller is presented for completeness. PDF-based tuning of stochastic optimal controller (PTSOC) [20] mitigates the adverse effects induced by the uncertainties of cyberspace and adapt to the random occurrence of cyber network faults.

Remark 4: PTSOC has a good adaptability to a time-varying distribution of delays, but lead to more computation overhead than the traditional resilience controller. Therefore, the above strategy Section 5.2.4 aims to determine an appropriate time to trigger the resilience controller without consuming the computational overhead. Meanwhile, the proposed strategy based on fault isolation proactively trigger the controller, rather than triggering it when a failure or damage has occurred. Such that, the system performance and stability are guaranteed.

The PTSOC control law considers the PDF of delays by optimising a weighted summation of cost functions of different delay ranges (7). Each weight is the probability of its corresponding delay intervals from the PDF identifier.

$$
J^k = E\left[\sum_{i=1}^{n} P_i J_i^k \right] = E\left[\sum_{i=1}^{n} P_i (x_i^{kT} Q_{zi} x_i^k + u_i^{kT} R_{zi} u_i^k) \right] \tag{7}
$$

where i presents the delay interval ($d_{\text{int}} i < d^k < d_{\text{int}}(i+1)$); n is the total number of delay cases; k represents sampling interval; P_i is probability of delay within $d_{\text{int}} i$ to $d_{int}(i+1)$ provided by the PDF identifier; x_i is the states vector; u_i is the control inputs vector; $Q_{zi} = \text{diag}[Q_i, R_i/d, \ldots]$ and $R_{zi} = R_i/d$ are symmetric positive semi-definite and symmetric positive definite, respectively. $E[\bullet]$ is the expectation operator.

By optimising (7), the control input is given by:

$$
u(k) = -K(k) Z(k) \tag{8}
$$

$$
\begin{aligned}
K(k) = &\sum_{i=1}^{n_d} P_i(k) (B_{zi}(k)^T Z_i(k) B_{zi}(k) + R_z(k))^{-1} \\
&\times (B_{zi}(k)^T Z_i(k) A_{zi}(k) + S_{zi}(k))
\end{aligned} \tag{9}
$$

where $K(k)$ is the optimal gain and $u(k)$ is the control input; $S_{zi}(k) \geq 0$ is the solution of the algebraic riccati equation (ARE) equation; $n_d = d_{\text{upper}}/d_{\text{int}}$, d_{upper} is the maximum delay in the sliding window; $P_i(k)$ is the probability of $d_{\text{int}} i < d(k) < d_{\text{int}}(i+1)$.

Remark 5: The Q_i and R_i in the cost function for each delay range should be different because each pair of Q_i and R_i should be the optimal values for the delays bounded in a specific range. They cannot guarantee a high level with the delays out of such boundaries.

Stability Analysis [20]:

Two theorems and their corresponding proofs are presented to demonstrate the stability of the proposed PTSOC. Lyapunov-based stability analysis is used. Theorem 2 (Appendix 9.2) shows the control gain estimation asymptotically converges even if PDF estimation has an error provided it asymptotically converges to zero. Theorem 3 (Appendix 9.3) considers the irremovable bias of PDF estimation as a bounded disturbance. However, a UUB stability is guaranteed. The proofs for these theorems can be found in [20].

6 Simulation and discussion

In this section, the proposed prognosis scheme is evaluated by simulations in MATLAB. Section 6.1 demonstrates the convergence of the system-state prediction. In this case, the resilience controller triggering is disabled to observe the prediction performance alone. Then, both soft and hard cyber network fault scenarios are presented separately to demonstrate the cyber network fault detection and isolation performance in Sections 6.2 and 6.3. The resilience controller in Section 5.3 is applied. A conventional stochastic optimal control [18] is employed as a reference.

A continuous-time batch reactor system is taken as a case study. Its dynamics are given by [18].

$$\dot{x} = \begin{bmatrix} 1.38 & -0.2077 & 6.715 & -5.676 \\ -0.5814 & -4.29 & 0 & 0.675 \\ 1.067 & 4.273 & -6.654 & 5.893 \\ 0.048 & 4.273 & 1.343 & -2.104 \end{bmatrix} x + \begin{bmatrix} 0 & 0 \\ 5.679 & 0 \\ 1.136 & -3.146 \\ 1.136 & 0 \end{bmatrix} u \quad (10)$$

The parameters of this CPS are selected as:

(a) The sampling time is 100 ms;
(b) The considered delays in the system model is <2 sampling interval, $d = 2$;
(c) $d_{int} = 0.1$ s;
(d) The threshold of the probability variation R_{pi} is 0.03 s, unless otherwise states;
(e) The sliding window size M is 30.

6.1 System state prediction evaluation

This scenario demonstrates that the accuracy of the system-state prediction improves as more new delay measurements update the distribution estimation.

Here, the PTSOC triggering is disabled to allow continuous, uninterrupted predictions of system states. The fault is injected at 47 s. Fig. 3 shows that the prediction at 47.1 s significantly diverges from the actual system behaviour. As more new delays are loaded in the sliding window, the PDF estimation of the new distribution improves. Such that the predicted system states become more accurate. The predictions at 47.5 s is more accurate than that at 47.1 s. Other results are shown in Appendix 9.4.

6.2 Soft cyber network fault

In this scenario, a network congestion fault is simulated, which occurs when a network node is relaying more data than it can handle. It usually causes a gradual increase of delays. R_{pi} and M are user-defined parameters. These simulations are repeated 50 times for the statistical validation. With 50 repeated simulations for different soft faults, the proposed scheme only needs 0.42 s in average to detect the fault. With $R_{pi} = 0.03$ s, the faults are 100% detected. The results in Fig. 4 are only for one case to illustrate the performance of the proposed prognosis scheme.

Before the first 50 s, the delays follow a normal distribution $N(0.3, 0.05^2)$. Then, a network congestion attack (e.g. denial-of-service) is launched at 50 s and the delays after 50 s follow a new normal distribution $N(0.5, 0.1^2)$. Fig. 4 presents the result for fluid level. As shown in Fig. 4*a*, the probability variation exceeds the threshold at 50.2 s. A cyber network fault is detected. Simultaneously, the awareness of the cyber network fault triggers the system-state prediction shown in Fig. 4*b* and Appendix 9.5. The oscillation are observed, but are small enough for the basic controller to handle. Therefore, this fault is a soft fault. The resilience controller does not have to be triggered.

The traditional diagnosis scheme [18] usually preset a threshold, which is a constant, for the network delay to capture the delay variation. When the delay exceeds the bound, the resilience controller will be activated. Such that some unnecessary triggering might occur resulting in increased computational overhead and false reactions of resilience controller. According to Fig. 4*a*, the resilience controller should be activated 12 times if the traditional fault diagnosis is applied. However, applying the resilience control is not necessary and induce more resource waste and wear and tear of system hardware. On the contrary, such negative consequence can be avoided with applying our proposed scheme.

Remark 6: There are several cyber network fault detection before 50 s because R_{pi} for this scenario is selected at low level. Hence, false detection occur. However, they would only cause more computational overhead and have no input on system stability. Overall, this trade-off should be considered when selecting R_{pi}.

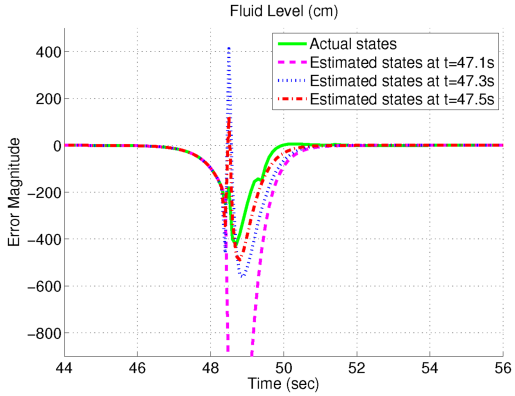

Fig. 3 *Case A: Actual and predicted system behaviour*

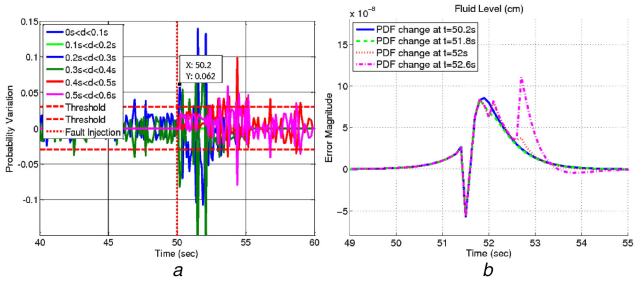

Fig. 4 *Case B*
(a) Selected probability variation, ***(b)*** Predicted and actual system output

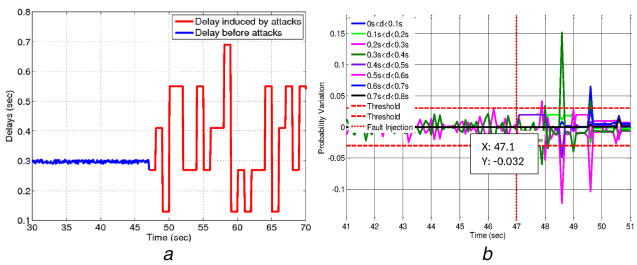

Fig. 5 *Case C*
(a) The simulated delays, ***(b)*** Selected probability variation

Remark 7: After the first soft fault detection, the proposed scheme should continuously supervise the cyber condition. That is because a soft fault possibly becomes a hard fault in the near future. Also, a warning should be issued to human supervisor to take additional precautions (e.g. investigate attack or update firewall).

6.3 Hard cyber network fault

In this scenario, a man-in-the-middle attack (MitM) is simulated. The attacker secretly relays and possibly alters the communication between two parties who believe they are directly communicating with each other. The transmitted information, such as control commands and feedback measurements, can be eavesdropped and delayed. Here, the delays before 47 s follows a normal distribution $(0.3, 0.05^2)$. Then, the attacker injects MitM attacks intermittently. As the results, the distribution of delays is varied overtime (Fig. 5*a*). The acceptable error magnitudes are set for four system states: 400 cm for the fluid level; 50k for the inside temperature; 40 g/s for the product outlet flow rate; and 50k for the coolant outlet temperature.

As Fig. 5*b* showing, the sudden change of delay at 47 s is detected at 47.1 s because the probability of delays within [0.2, 0.3] suddenly decreases. In Fig. 6*a* and Appendix 9.6, all the predicted system outputs exceed their acceptable range. The estimated and

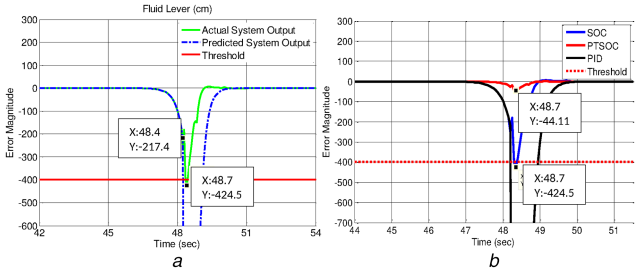

Fig. 6 *Case C*
(a) Predicted and actual system output, ***(b)*** Fault mitigation performance

Table 2 Crossing points

Variables	Estimated point, s	Actual point, s	Estimation error, %
fluid level	48.4	48.7	0.6
inside temperature	48.4	48.6	0.4
product outlet flow rate	48.2	48.2	0
coolant outlet temperature	48.2	48.3	0.2

Table 3 Comparison of overshoot and TTR

Variables	Overshoot/TTR					TTR				
	SOC	PTSOC	Improve, %	PID	Improve, %	SOC, s	PTSOC, s	Improve, %	PID, s	Improve, %
fluid level	425.5 cm	44.11 cm	89.6	6792 cm	99.4	6.4	4.4	31.2	6.9	36.2
inside temperature	57.5 K	4.71 K	91.8	1573 K	99.7	4.8	3.1	35.4	5.3	41.5
product outlet flow rate	460.7 g/s	47.51 g/s	89.7	6599 g/s	99.3	4.4	2.4	45.5	4.5	46.7
coolant outlet temperature	80.75 K	8.61 K	89.3	4604 K	99.8	7.7	4.3	44.2	8.1	46.9

actual points that the system states pass through the acceptable error are shown in Table 2. This prediction can achieve at least 99.6% accuracy. It is concluded that this fault is a hard cyber network fault and its adverse effects on the system performance is predicted. The resilience control is triggered at 47.1 s to mitigate such effects. Comparing with the original SOC, the overshoots are reduced by at least 89.6%, the TTRs are shortened by 31.2%. The summary of improvements can be found in Table 3.

When applying the proposed scheme, the fault is quickly detected and the resilience controller is timely triggered ahead of the serious degradation of system performance. Also, the overshoot of each system output is significantly reduced in term of its corresponding TTR. In contrast, without applying the proposed scheme, the fault still can be detected when the system states exceed the acceptable error magnitude at 48.5 s. The fault tolerant controller, which is a tuned PID controller, is triggered. However, it is too late to recover the system performance with such a late activation of the resilience controller. In such case, the basic controller will try to apply excessive actuation (Table 3) to stabilise. This might lead to significant damage of the components or cause an unscheduled downtime. Even worse, the system could be compelled to stop. The above simulation is repeated for 50 times. All the faults are accurately detected.

6.4 Discussion

We conducted 100 simulations with 50 soft and 50 hard fault cases and, to evaluate the isolation accuracy of the proposed scheme. All of the faults are detected. However, 58 hard faults are identified, that is eight soft faults are incorrectly recognised as hard faults. The threshold for fault isolation is set ensure 100% correct isolation of hard faults. Those false hard fault identification have no negative impact on system stability and performance, only increase the computational overhead.

It is important to note that the traditional physical system fault detection, which is a model-based observer, cannot detect any abnormalities in cyberspace. The network dynamics concurrently change the mathematical model of the physical system and the model used for observer design. Such that the outputs from the observer and physical system are same. Therefore, the model-based observer can only be used for physical component fault detection, not cyber network fault. In addition, designing a traditional observer for cyber network fault detection is impossible because, in realistic CPS, cyber network fault model cannot be obtained ahead of time.

7 Conclusions

The proposed novel prognosis scheme is shown to quickly detect and predict cyber network faults using PDF monitoring and estimation. Moreover, soft and hard faults are isolated to optimise the computational cost of resilience control. The convergence of the future delay distribution estimation and the system-state prediction are theoretically proven. With the proposed resilience controller, the adverse effects caused by cyber network faults are efficiently mitigated.

The simulation results show that the proposed scheme accurately detect the cyber network faults before the performance degrades beyond the acceptable range. Moreover, the PTSOC is timely triggered to mitigate the negative effects on the CPSs performance. The overshoot is significantly reduced by 90% and TTR is shortened by 30%. Although the accuracy of the soft and hard fault isolation can only achieve 84%, the hard faults are 100% detected. Those soft faults which are misclassified to hard faults only consume the resources for triggering resilience controller. However, the stability of the entire CPS is always guaranteed.

8 References

[1] Fisher, A., Jacobson, C.A., Lee, E.A.: 'Industrial cyber-physical systems – iCyPhy', in Aiguier, M., Boulanger, F., Krob, D., *et al.* (Eds.): '*Complex systems design and management*' (Springer International Publishing, Dordrecht, Switzerland, 2014), pp. 21–37

[2] Yagdereli, E., Gemci, C., Aktas, A.Z.: 'A study on cyber-security of autonomous and unmanned vehicle', *J. Def. Model. Simul.*, 2015, **12**, (4), pp. 369–381

[3] Liu, F.C., Yao, Y.: 'Modeling and analysis of networked control systems using hidden Markov models'. Proc. Int. Conf. Machine Learning Cybernetics, Guangzhou, China, 2005, pp. 928–931

[4] Liu, G.P., Xia, Y., Chen, J., *et al.*: 'Networked predictive control of systems with random network delays in both forward and feedback channels', *IEEE Trans. Ind. Electron.*, 2007, **54**, (3), pp. 1282–1297

[5] Zhang, H., Yang, J., Su, C.-Y.: 'T-S fuzzy-model-based robust H_∞ design for networked control systems with uncertainties', *IEEE Trans. Ind. Inf.*, 2007, **3**, (4), pp. 289–301

[6] Wang, Y., Ye, H., Wang, G.: 'A new method for fault detection of networked control systems'. Proc. IEEE Conf. Industrial Electronics Applications, Singapore, 24–26 May 2006, pp. 1–4
[7] Zhu, Z.Q., Zhou, X.Z.: 'Fault detection based on the states observer for networked control systems with uncertain long time-delay'. Proc. IEEE Int. Conf. Automation Logistics, Jinan, China, August 2007, pp. 2320–2324
[8] Rawat, D., Rodrigues, J., Stojmenovic, I.: 'Cyber-physical systems: from theory to practice', 2016
[9] Cardenas, A., Amin, S., Sastry, S.: 'Secure control: towards survivable cyber-physical systems'. 2008 The 28th Int. Conf. on Distributed Computing Systems Workshops, Beijing, China, 2008, pp. 495–500
[10] Amin, S., Cárdenas, A., Sastry, S.: 'Safe and secure networked control systems under denial-of-service attacks', *Hybrid Syst.: Comput. Control*, 2009, **5469**, pp. 31–45
[11] Liu, Y., Reiter, M.K., Ning, P.: 'False data injection attacks against state estimation in electric power grids'. Proc. ACM Conf. Computer Communications Security, Chicago, IL, USA, November 2009, pp. 21–32
[12] Teixeira, A., Amin, S., Sandberg, H., *et al.*: 'Cyber security analysis of state estimators in electric power systems'. Proc. IEEE Conf. Decision Control, Atlanta, GA, USA, December 2010, pp. 5991–5998
[13] Mo, Y., Sinopoli, B.: 'Secure control against replay attacks'. Proc. Allerton Conf. Communication, Control, Computing, Monticello, IL, USA, September 2010, pp. 911–918
[14] Smith, R.: 'A decoupled feedback structure for covertly appropriating network control systems'. Proc. IFAC World Congress, Milan, Italy, August 2011, pp. 90–95
[15] Gamage, T., McMillin, B.M., Roth, T.P.: 'Enforcing information flow security properties in cyber-physical systems: a generalized framework based on compensation'. 2010 IEEE 34th Annual Computer Software and Applications Conf. Workshops (COMPSACW), Seoul, South Korea, 2010, pp. 158–163
[16] Jiang, W., Guo, W.H., Sang, N.: 'Periodic real-time message scheduling for confidentiality-aware cyber-physical system in wireless networks'. Proc. of Fifth Int. Conf. on Frontier of Computer Science and Technology, Changchun, China, 2010
[17] Pasqualetti, F., Dorfler, F., Bullo, F.: 'Attack detection and identification in cyber-physical systems', *IEEE Trans. Autom. Control*, 2013, **58**, (11), pp. 2715–2729
[18] Xu, H., Jagannathan, S.: 'Stochastic optimal control of unknown linear networked control system in the presence of random delays and packet losses', *Automatica*, 2012, **48**, pp. 1017–1029
[19] Zhu, M., Martinez, S.: 'Stackelberg-game analysis of correlated attacks in cyber-physical systems'. Proc. American Control Conf., San Francisco, CA, USA, July 2011, pp. 4063–4068
[20] Bi, S., Zawodniok, M.: 'PDF-based tuning of stochastic optimal controller design for cyber-physical systems with uncertain delay dynamics', *IET Cyber-Phys. Syst., Theory Appl.*, 2017, **2**, (1), pp. 1–9
[21] Bi, S., Zawodniok, M.: 'A novel cyber network fault diagnosis scheme for cyber-physical systems'. 2017 IEEE Int. Conf. on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), Exeter, UK, 2017, pp. 30–36

9 Appendix

9.1 Identification Algorithm

[21]
See Table 1.

9.2 Theorems 2 and Proof (To be included in paper as the approach)

Theorem 2: (Control gain estimation error convergence): As the delay data keeps updating PDF identifier and $(\parallel \sum_{j=1}^{n} \widetilde{P}_{(k+1)j} \parallel - \parallel \sum_{j=1}^{n} \widetilde{P}_{kj} \parallel) < 0$ is satisfied, then the estimation error for control gain $\parallel \widetilde{K}_k \parallel$ asymptotically converges to zero.

Proof: First, we define the estimation error of control gain K as $\widetilde{K}_k = K_k - \widehat{K}_k = \sum_{j=1}^{n} P_{kj}K_j - \sum_{j=1}^{n} \hat{P}_{kj}K_j$. P_{ij} is the actual probability at k. Then, Lyapunov function candidate is $V_{K_k} = \widetilde{K}_k^T \widetilde{K}_k$.

$$
\begin{aligned}
\Delta V_{K_k} &= V_{K_{k+1}} - V_{K_k} \\
&= \widetilde{K}_{k+1}^T \widetilde{K}_{k+1} - \widetilde{K}_k^T \widetilde{K}_k \\
&= (K_{k+1} - \widehat{K}_{k+1})^T (K_{k+1} - \widehat{K}_{k+1}) \\
&\quad -(K_k - \widehat{K}_k)^T (K_k - \widehat{K}_k) \\
&= \left(\sum_{j=1}^{n_d} P_{(k+1)j}K_j - \sum_{j=1}^{n_d} \hat{P}_{(k+1)j}K_j \right)^T \left(\sum_{j=1}^{n_d} P_{(k+1)j}K_j \right. \\
&\quad \left. - \sum_{j=1}^{n_d} \hat{P}_{(k+1)j}K_j \right) - \left(\sum_{j=1}^{n_d} P_{kj}K_j - \sum_{j=1}^{n_d} \hat{P}_{kj}K_j \right)^T \\
&\quad \times \left(\sum_{j=1}^{n_d} P_{kj}K_j - \sum_{j=1}^{n_d} \hat{P}_{kj}K_j \right) \\
&= \parallel \sum_{j=1}^{n_d} (P_{(k+1)j} - \hat{P}_{(k+1)j})K_j \parallel^2 \\
&\quad - \parallel \sum_{j=1}^{n_d} (P_{kj} - \hat{P}_{kj})K_j \parallel^2 \\
&= \parallel \sum_{j=1}^{n_d} \tilde{P}_{(k+1)j}K_j \parallel^2 - \parallel \sum_{j=1}^{n_d} \tilde{P}_{kj}K_j \parallel^2 \\
&= \underbrace{\left(\parallel \sum_{j=1}^{n_d} \tilde{P}_{(k+1)j}K_j \parallel + \parallel \sum_{j=1}^{n_d} \tilde{P}_{kj}K_j \parallel \right)}_{\Delta_2} \\
&\quad \times \left(\parallel \sum_{j=1}^{n_d} \tilde{P}_{(k+1)j}K_j \parallel - \parallel \sum_{j=1}^{n_d} \tilde{P}_{kj}K_j \parallel \right) \\
&= \Delta_2 \left(\parallel \sum_{j=1}^{n_d} \tilde{P}_{(k+1)j}K_j \parallel - \parallel \sum_{j=1}^{n_d} \tilde{P}_{kj}K_j \parallel \right) \\
&= \Delta_2 [(\parallel \tilde{P}_{(k+1)1} \parallel - \parallel \tilde{P}_{k_1} \parallel) \parallel K_1 \parallel \\
&\quad + \parallel \tilde{P}_{(k+1)2} \parallel - \parallel \tilde{P}_{k_2} \parallel) \parallel K_2 \parallel + \cdots \\
&\quad + \parallel \tilde{P}_{(k+1)n_d} \parallel - \parallel \tilde{P}_{k_{n_d}} \parallel) \parallel K_{n_d} \parallel] \\
&\leq \Delta_2 \left(\parallel \sum_{j=1}^{n} \tilde{P}_{(k+1)j} \parallel - \parallel \sum_{j=1}^{n} \tilde{P}_{kj} \parallel \right) \parallel K_{max} \parallel
\end{aligned}
$$

Since V_{K_k} is positive definite and ΔV_{K_k} is negative definite provided $\widetilde{K}_k = K_k - \widehat{K}_k = \sum_{j=1}^{n} P_{kj}K_j - \sum_{j=1}^{n} \hat{P}_{kj}K_j$. Therefore, the estimation error of control gain asymptotically converge to zero. □

9.3 Theorem 3 and Proof (To be included in paper as the approach)

Theorem 3: (UUB Stability of the Regulation Error): Given the initial conditions as the system state z_0 and system matrices A_{z0}, and B_{z0}, let $u_0(z_k)$ be an initially admissible control policy for the CPS (6). Let the control update law be given by (8) and (9) and if the disturbance induced by the irremovable bias of PDF estimation has a bound $\parallel d_{KDE} \parallel$ and $K_{min} < 1/b_{min}$ such that the regulation error of system states has a uniformly ultimate bounded convergence in the mean.

Proof: Consider the following positive definite Lyapunov function candidate: $V_{z_k} = z_K^T z_k$. z_k is the state vector of k. The corresponding estimated Lyapunov is $\hat{V}_{z_k}$, therefore, $\Delta \hat{V}_{z_k} = \hat{V}_{z_{k+1}} - \hat{V}_{z_k}$. We consider $\Delta \hat{V}_{z_{km}} = \hat{V}_{z_{(k+1)m}} - \hat{V}_{z_k}$ for each possible system matrices (A_{zkm} and B_{zkm}). m represents one of the possible cases. If the maximum value of $\Delta \hat{V}_{z_k m}$ is negative definite, the system convergence is proven. The irremovable bias of PDF

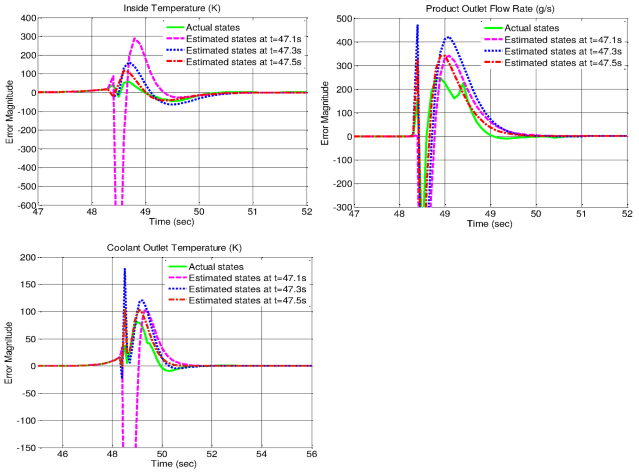

Fig. 7 *Case A: actual and predicted system behaviour*

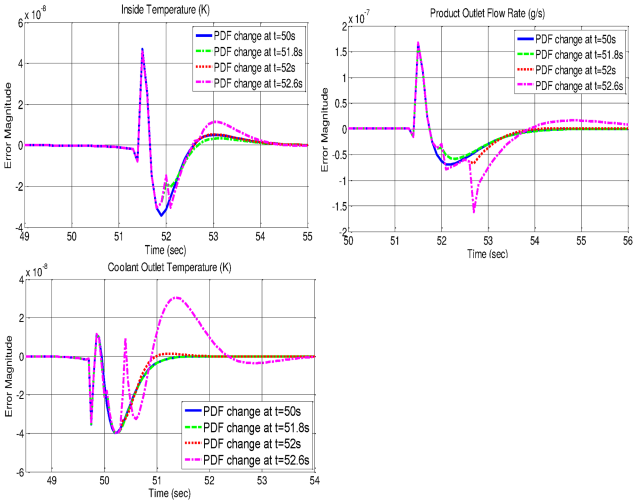

Fig. 8 *Case B: predicted and actual system behaviour*

estimation is considered the system-state disturbance d_k bounded by d_M.

$$
\begin{aligned}
\Delta \widehat{V}_{z_{km}} &= \widehat{V}_{z_{k+1}m} - \widehat{V}_{z_{k}m} \\
&= \| A_{zkm} - B_{zkm}K_k z_k + d_k \|^2 - \| z_k \|^2 \\
&= \underbrace{(\| A_{zkm} - B_{zkm}K_k z_k + d_k \| + \| z_k \|)}_{\Delta_3} \\
&\quad \times (\| A_{zkm} - B_{zkm}K_k z_k + d_k \| - \| z_k \|) \\
&= \Delta_3 (\| A_{zkm} - B_{zkm}K_k z_k + d_k \| - \| z_k \|) \\
&\leq \Delta_3 (\| a_{max} - b_{min}K_{min} z_k + d_M \| - \| z_k \|) \\
&\leq \Delta_3 (a_{max} + b_{min}K_{min} \| z_k \| + \| d_M \| - \| z_k \|) \\
\forall k &= 1, 2, \ldots \\
\forall m &= 1, 2, \ldots, n_d
\end{aligned}
$$

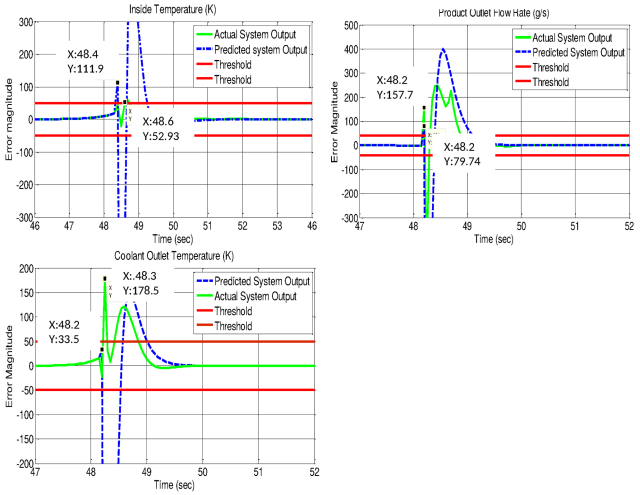

Fig. 9 *Case C: predicted and actual system behaviour*

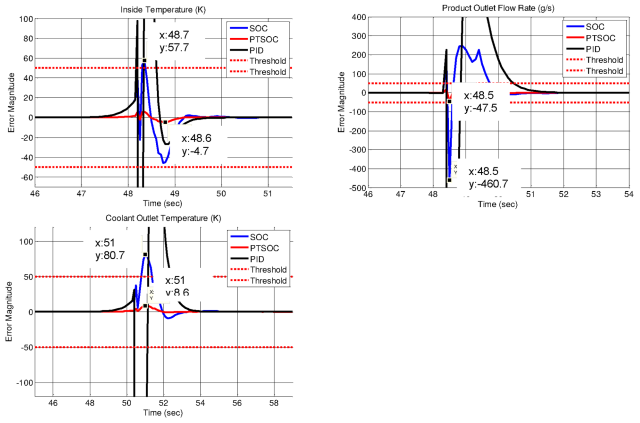

Fig. 10 *Case C: fault mitigation performance*

where Δ_3 is positive definite, $b_{min} = \min\{ \| B_{zk1} \|, \| B_{zk2} \|, \ldots, \| B_{zkm} \| \}$, $K_{min} = \min\{ \| K_1 \|, \| K_2 \|, \ldots, \| K_{n_d} \| \}$.

Since $\widehat{V}_{z_k}$ is positive definite and $\Delta \widehat{V}_{K_k}$ is negative definite provided the system state $\| z_k \| \geq ((\| d_M \| + A_{max})/(1 - b_{min}K_{min}))$ and $K_{min} < 1/b_{min}$. Therefore, UUB stability of the regulation error is proven. □

9.4 Other results for Case A

See Fig. 7.

9.5 Other results for Case B

See Fig. 8.

9.6 Other results for Case C

See Figs. 9 and 10.