PDF-based tuning of stochastic optimal controller design for cyber-physical systems with uncertain delay dynamics

Shanshan Bi, Maciej Zawodniok

Abstract: Uncertain dynamics in communication network, including random delays and packet losses make it difficult to guarantee stability of cyber-physical systems (CPSs). Many existing works consider the uncertainties of network channel with strong assumptions that network delay bounds and its distribution are known a priori and time-invariant. However, these assumptions could be invalidated in realistic CPSs by malicious attacks, system hardware faults, topology changes etc. A probability density function (PDF)-based tuning of stochastic optimal control (PTSOC) is proposed to manage the unknown dynamics in the embedded network. The update law of the proposed controller is derived and updated based on the PDF estimation of network delays that explicitly consider delays and its time-varying distribution. The results illustrate that the proposed PTSOC has a better performance in terms of the overshoot, convergence time, and cost when compared with the conventional stochastic optimal control.

1 Introduction

Modern industrial systems, such as smart grid, healthcare and automotive control systems, are implemented as distributed event-triggered control systems wherein the control loops are connected by a real-time communication network with limited resources. Such systems are referred to cyber-physical systems (CPSs) which offer many advantages: the ease of maintenance and installation, flexibility, and low cost. However, limited resources and constraints of the embedded communication network create challenges for a control system stability. For example, changing topology or background traffic result in varied delays, packet losses, and quantisation over time. In addition, cyber-attacks including denial-of-service, spoofing, and eavesdropping can degrade the system performance. For example, delays exceeding the range assumed for the controller due to jamming cyber-attack lead to the control signal arriving too late for the actuator to take appropriate actions. Consequently, system outputs deviate from the desired trajectory and the entire CPS becomes unstable. Therefore, an optimal networked resilience controller design for CPSs is needed. It has to mitigate the negative effects of the network uncertainties and dynamics on the CPS performance.

Many existing works on cyber security [1–4] proposed network schemes to defend against the cyber-attacks without modification to the controller design. In contrast, other researchers focused on fault tolerant controller design to improve resilience against physical component faults [5–10]. In these works, the network dynamics and uncertainties are often oversimplified under strong assumptions: (i) the delays are bounded within a specific range and (ii) the distribution of delays and package losses are known and time-invariant. However, in realistic CPSs, the delay can easily exceed such restrictive bounds and lead to unstable system. The unexpected variation in delay leads to the malfunction or overreaction on the controller and actuator, and eventually instability of the entire CPSs. Therefore, relaxing these assumptions should be addressed to improve the resilience of CPSs.

In this paper, a PDF-based tuning of stochastic optimal controller (PTSOC) is developed to address the degraded system performance induced by network uncertainties including attacks, transmission faults, and channel dynamics. This relaxes the earlier mentioned strong assumptions (i) and (ii) made in existing works. A stochastic system model, which includes stochastic parameters that represents network dynamics, is employed. A kernel density estimation (KDE)-based online PDF identifier is proposed to capture the variation of network dynamics. The probabilities of delays provided by the PDF identifier are used to tune the control law. In addition, the system stability is mathematically analysed using Lyapunov approach. Overall, the proposed approach improves the robustness, optimises cost of regulation, prevents the physical components from unrepairable damages, and keeps the CPS working within the desired operating condition.

This paper is organised as follows: In Section 2, the related works and motivation are briefly discussed. The proposed controller design in terms of PDF identifier and stability analysis are presented in Section 4. Section 6 illustrates the effectiveness of the proposed controller through simulations in MATLAB. Section 7 gives the conclusion.

2 Motivation example

The embedded cyberspace imposes restrictions on the exchange of information in CPSs, such a limited channel capacity and traffic congestions. Malicious cyber-attacks further restrict the information delivery. The delay and packet loss caused by the above restrictions are stochastic with unknown bounds and difficult to predict. Importantly, they have a potentially negative effect on the performance and stability of CPSs. For example, inappropriate control actions caused by increasing delay lead to a big overshoot as well as more actuation cost to manage such overshoot.

A following example shows that the network dynamics significantly affect the performance of CPS and the existing control approaches may fail to guarantee its stability.

This scenario emulates a route hijacking by an attacker to eavesdrop control information (data integrity). Such a route hijacking increases delay and delay variation on the longer path. A network is simulated using network simulator 2 with a random topology of 11 nodes. Ad hoc on-demand distance vector routing scheme is adopted. The route through the topology is altered during the simulation such that the packet delays vary for the...
controller loop, as shown in Fig. 1a. Note that similar network performance could be a result of topology or traffic pattern changes. The results show the disturbance in CPS introduced by the network dynamics. With these delays, a proportional–integral–derivative controller (PID) controller is simulated for a simple system. The controller design of physical systems is not uncontrollable and unstable. Therefore, it is necessary to include disturbance increases the probability that the system becomes uncontrollable and unstable. Consequently, the CPS becomes unstable due to such dynamics of delay.

The attack changes the delay distribution. This stochastic disturbance increases the probability that the system becomes uncontrollable and unstable. Therefore, it is necessary to include such dynamics both in system model and controller design.

3 Related works

In this section, first, some existing works on control schemes for CPSSs are introduced. Then, some methodologies for capturing network dynamics are discussed to illustrate that KDE is suitable to be used in the proposed control scheme. At last, the main contributions of our work are highlighted and demonstrated.

3.1 Control design for CPS

The following literature review discusses the existing approaches that address the uncertain dynamics of network, especially long-time delays and packet losses, either in network or physical system.

Many researchers developed network protocols and tools [1–4] to keep delays and packet losses within traditional controller constraints. The controller design of physical systems is not modified in these works. Such approaches can maintain system stability if the network configuration is simple and fixed. Designing more complicated networks and systems under cyber-attacks becomes challenging if not impossible due to restrictive constraints dictated by the physical controller. Xie [1] proposed a channel estimation approach to calculate the packet success rate, the worst-case packet delay and average energy consumption based on acknowledgement information. However, the worst-case delay has a upper bound known a priori while the system model is linear time-invariant and not stochastic. Lee et al. [2] proposed a quality-of-service-based remote control scheme for CPSSs via the ProfiBus token passing protocol. They used the time delay data provided by transmitting the real-time messages to approximate the delays for high- and low-priority messages. They assumed a strong condition that the network delay has time-invariant lower and upper bounds. The control schemes in above literature are valid if the delays are bounded within the acceptable range. However, if the configuration of the embedded network becomes complicated, dynamic, and stochastic, the bound constraints cannot be met. For instant, in intelligent transportation systems, moving vehicles exchange their information with other vehicles, transportation management system, and users. Such a complicated and dynamic network results in delays and packet losses exceeding their bounds because of interference from the environment and network topology changes. Thus the controller cannot guarantee the stability of CPSSs. Overall, a more robust approach should consider network dynamics in controller design to relax the constraints.

Simultaneously, other researchers had designed several controllers to address stochastic network dynamics under strong constraints. Gao et al. [5, 6] modelled a CPS as a sampled-data system and solved a set of linear matrix inequalities to derive the feedback gain of a memory-less controller. The dynamics in the cyberspace are simplified as known bounded delays. Similarly, Hao et al. [7], Tian et al. [8] and Liu [9] proposed networked controllers by using Lyapunov stability analysis with a priori knowledge of the delay bounds. Moreover, Tiberi [10] proposed a self-triggered sampling for achieving substantial reduction of communication traffic. The system stability can be guaranteed under certain assumptions: (i) the measurements are sent to the central node within a bounded time delay and (ii) the system states have to converge initially. All these works considered the network delay issues in various perspective. However, they commonly assumed the network delay had a known bound and the distribution of delay was fixed. Such schemes would fail in realistic CPSSs where delay bound is unknown beforehand. For instance, new communication nodes generating new traffic will increase the upper bound which cannot be known ahead of time. Furthermore, dynamics of the delay distribution make the entire system unstable.

Xu et al. [11] made a significant progress when they included the random parameters representing network dynamics in system model. All of system matrices and control input calculation concurrently change as network delays and packet losses randomly change. This model is the prototype we used in the proposed PTSOC design.

The conventional discrete time model was described as [12]

\[
x_{k+1} = A_x x_k + B_0 u_k + B_1 u_{k-1} + \cdots + B_{d-1} u_{k-d}
\]

where \( x_k = x(kT) \) denotes system states; \( A_x = e^{iT} \), \( B_0 = \int_0^T e^{iT} dT \), and \( B_i = \int_0^T e^{-i(1+1)T} e^{-i(1+i+2)T} \) \( B_{d-1} \) \( i = 1, 2, \ldots, d \) are the system matrices. \( T \) is sampling interval. \( A_x \) is the system matrix of the continuous-time model.

Xu [11] derived the stochastic model expressed as

\[
z_{k+1} = A_x z_k + B_x u_k
\]

\[\text{Fig. 1} \quad \text{Effects of hijacking attack on system performance}\]

\( a \) Delays

\( b \) System performance with a PID controller

IET Cyber-Phys. Syst., Theory Appl., 2017, Vol. 2, Iss. 1, pp. 1–9

This is an open access article published by the IET under the Creative Commons Attribution License (http://creativecommons.org/licenses/by/3.0/)
where \( z_k = [x_k^T \ u_{k-1}^T \ \ldots \ u_{k-d}^T]^T \) is the state variables vector

\[
A_{dk} = \begin{bmatrix}
A & \gamma_k^f B_2 & \ldots & \gamma_k^f B_d \\
0 & 0 & \ldots & 0 \\
I_m & 0 & \ldots & 0 \\
0 & 0 & \ldots & I_m
\end{bmatrix}, \quad B_{dk} = \begin{bmatrix}
\gamma_k^f B_0 \\
0 \\
0 \\
0
\end{bmatrix}
\]

and are the time-varying system matrices [11]; \( A \) and \( B^f_i \) are the system matrices calculated by (1). \( u_i \) is the control input; \( \gamma_k^f \) and \( \gamma_k^{f-} \) are binary random variables representing the package reception status (if the package is received, \( \gamma = 1 \); otherwise, \( \gamma = 0 \)). \( I_m \) is an identity matrix.

Then, the stochastic optimal control (SOC) law can be obtained by optimised the following cost function.

\[
J = E\left[ \sum_{m=0}^{\infty} e^T_m Q_m z_m + u_m^T R_m u_m \right], \quad k = 0, 1, 2, \ldots \quad (3)
\]

where \( Q_m = \text{diag}\{Q, R/d, \ldots \} \), and \( R = R/d \) are symmetric positive semi-definite and symmetric positive definite, respectively. \( z_m \) is the state variables vector, and \( u_m \) is the control inputs vector. \( E(\cdot) \) is the expected operator of \( \sum_{m=0}^{\infty} (z_m^T Q_m z_m + u_m^T R_m u_m) \).

Although the above model included the terms of network dynamics, the SOC only indirectly considers such dynamics through the stochastic model. It also has a strong assumption that the delay distribution is fixed. System performance becomes suboptimal when the above assumptions are invalidated by changes in bounds or distribution as observed in simulations for cases B and C in Section 6.

### 3.2 Capturing network dynamics

To relax the bounds and distribution constraints, the system model and the control law should be tuned based on current probabilities of delays. Therefore, a PDF estimation is needed.

For estimation of delay probabilities, several PDF identification methods exist in literature, such as histogram, KDE, and maximum likelihood estimation [13]. The advantage of KDE is more accurate estimation with fewer samples than other methods. Offline KDE has been used in various applications, including computer graphics [14], image processing [12, 15–17], and industrial process [18, 19]. He et al. [18] introduced a novel KDE-based framework for non-linear metric learning, Kernel density metric learning. This method has been successfully applied in face recognition. Elgammal [17] investigated the use of fast Gaussian transform for efficient computation of KDE techniques for computer vision applications. Chen [19] proposed a KDE-based method to estimate the spatial intensity of false alarms for multi-target tracking system. Additionally, KDE is suitable [13] for PDF identification in CPS because it can handle different types of distributions, such as mixture distribution and Poisson distribution. In contrast, the other approaches only work for normal distributions. In Section 4.2, an online KDE estimator is introduced. It can estimate unknown PDF with a good accuracy and adapt to the dynamic changes of delay distribution. Then, the provided PDF information is used to tune the PTSOC control law. Hence, the controller adapts to the given network dynamics.

In this paper, we proposed PTSOC to relax the bounds and distribution constraints in the existing works. The PDF variation of network delays can be captured by the KDE-based PDF identifier. Also, the PDF information is used to tuning the optimal control law. Thus, the resilience of the entire CPS is improved. Next, the details of the proposed PTSOC are presented.

### 4 Proposed PTSOC design

In this section, the overview of the proposed control scheme is given. The PTSOC scheme with an online PDF estimator is introduced. The online KDE-based PDF identifier is introduced in Section 4.2. Then, PTSOC is derived from the proposed cost function 4 in Section 4.3.

### 4.1 Overview

PTSOC takes into account uncertain network dynamics by applying online KDE to capture PDF variation of delays and tuning its control law based on the PDF information. The overall architecture of the proposed control system is shown in Fig. 2. The proposed PTSOC includes three main steps that are continuously repeated:

(a) **Data collection of delays**: \( n \) delays \( (d_{k-n+1}, \ldots, d_k) \) in the sliding window are used to do the PDF estimation at time \( k \) (PDF(t)). When new delay is measured, the data in the sliding window is updated.

(b) **PDF estimation**: The PDF of these \( n \) delays is obtained. The probability for each delay interval is calculated.

(c) **PTSOC law calculation**: For each delay interval, the cost function (5) and the optimal control law (8) are derived from (3). The cost function of PTSOC defined by (4) is a probability weighted sum of the cost functions for all delay intervals. Similarly, the PTSOC law derived in Section 4.3 is the summation of the weighted SOC laws of delay intervals. These weights are probabilities from the delay PDF. The final cost function is

\[
J^k = \sum_{m=1}^{n} P_m J^k \quad (4)
\]

where \( d_{int} \) represents the delay interval that we take 0.1 s in the simulation section. If \( d_{int} < d^* < d_{int}(i+1) \), \( d^* \) is classified in \( i \)th delay case; \( n \) is the total number of delay cases; \( k \) represents sampling interval; \( P_i \) is probability of delay within \( d_{int} \) to \( d_{int}(i+1) \) provided by the KDE-based PDF identifier; \( x \) is states vector; \( u_i \) is control inputs vector; \( Q, R_i \) are weighted constants of states and control inputs, respectively.

![Fig. 2 Overall architecture of stochastic CPS with PDF identifier](image)

**Remark 1**: The cost function of PTSOC is more generalised than that of SOC. SOC only considers the worst delay case that is \( d_{int} < d^* < d_{int}(i+1) \), \( P_i = 1 \), and \( P_j = 0, j \neq i \). In such a case, the PTSOC cost function (7) becomes SOC (3). As the probabilities of each possible delay changes, PTSOC continuously tracks the network dynamics with a PDF identifier.
and updates its parameters based on PDF information of delay to adapt to the given system situation.

4.2 PDF identifier

To capture the dynamics of network delays, an online KDE-based PDF identifier is proposed to iteratively estimate the distribution. The data used to do identification is updated every sampling interval for a window of $n$ last packet delays. The main steps of online PDF identification are shown in Table 1. $n$ delays are loaded in the sliding window first. Then, KDE algorithm is applied to estimate the probability of each delay. The overall PDF of this finite data set is obtained with selecting a normal kernel smoother here. Finally, the estimated PDF is used to tune control law and the new delay is loaded in the sliding window. The PDF is updated in an online manner to capture the variation of delay distribution.

Remark 2: With a large window size, KDE provides an accurate estimation of PDF with neglectable bias [20]. In this paper, we estimate PDF of $n$ samples in the sliding window. Such that the estimation error is guaranteed to converge to a small value denoted as $d_{KDE}$. The effect of $d_{KDE}$ on the regulation error convergence is shown in Theorem 3.

4.3 Optimal controller design with consideration of dynamics of delay distribution

In this section, PDF identification is employed to develop a novel SOC for CPSs while considering stochastic dynamics of network. First, we derive time-varying system matrices that explicitly include probability information of delays. Each element of the matrices is derived. Then, PTSOC is proposed based on the time-driven sensing and event-driven actuating controller framework. The maximum number of control input that affects system matrices is $d$. Only the latest control input is allowed to act on the controlled plant when several control inputs are received at the same time [11].

In the stochastic model (2), the candidate models (dynamic matrices $A_{k,i}$ and $B_{k,i}$) and their corresponding probabilities depend on the $\gamma_k$ values and their probabilities. $\gamma_k$ corresponds to control input with the delay less than sampling period ($T_s$). $\gamma_k$ ($k = 1, 2, d$) is determined by $\gamma_{k-1}$. Therefore, $\gamma_k$ depends on $\gamma_{k-1}$ and $\gamma_{k-2}$. The probability $P_{\gamma_k}$ can be determined by $P_{d^k > T_s}$. Equation (5) denotes the probability for each $\gamma_k$: (see (5))

where $P_{d^k < d_{min}(d_{max} + \Delta)} = \int_{d_{min}}^{d_{max} + \Delta} f(x) \, dx$, $d_{min}$ is delay bandwidth.

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 $\tau$ with a bandwidth $h$
   (b) Each observation $\tau$ receives a specific weight proportional to the scaled distance from the observation $\tau$ to $\tau$, which is $u = (\tau - \tau_i)/h$
   (c) At a given $\tau$, the estimate is found by vertically summing up over the $k$ shapes.

This can be synthesised as

\[ \gamma(\tau) = \frac{1}{h} \sum_{\tau_i} K(\tau - \tau_i) \]

The general formula for KDE will be given by

\[ \gamma_k = \frac{1}{nh} \sum_{\tau_i} K(\tau - \tau_i) \]

where the dependence of the estimate on the kernel function $K(\cdot)$ is denoted as $\gamma_k$.

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

All of the probabilities in (5) can be obtained from the PDF identifier. The data in the sliding window maps the PDF and provide the probabilities of delays in different ranges $P(d_k^0, P(d_k^1), \ldots, P(d_k^{d-1})$. The probability of delays can be denoted as $P(\gamma_k = i)$ and $P(\gamma_k = i)$. Then, it is easy to obtain the probabilities of each possible $\mathbf{B}_k^i$ based on the above information. Finally, a set of candidate stochastic models $A_{k,i}$ and $B_{k,i}$ are defined. The corresponding control law (candidate control law) for each candidate model (6) is obtained by optimising its corresponding cost function (7). Each candidate cost function has its optimal parameters $Q_{k,i}$ and $R_{k,i}$ for the corresponding control law derivation. The final proposed control law (8) is a weighted sum of these candidate control laws, with weights that are equal to the corresponding probabilities from the PDF identifier. (see (6))

\[
J_1 = E \left[ \sum_{m=0}^{\infty} z_{1m}^T (Q_{1m} - K_{1m}^T R_{1m} K_{1m}) z_{1m} \right]
\]

\[
J_2 = E \left[ \sum_{m=0}^{\infty} z_{2m}^T (Q_{2m} - K_{2m}^T R_{2m} K_{2m}) z_{2m} \right]
\]

\[
J_3 = E \left[ \sum_{m=0}^{\infty} z_{3m}^T (Q_{3m} - K_{3m}^T R_{3m} K_{3m}) z_{3m} \right]
\]

\[
J_{\text{opt}} = E \left[ \sum_{m=0}^{\infty} z_{\text{opt}}^T (Q_{\text{opt},m} - K_{\text{opt},m}^T R_{\text{opt},m} K_{\text{opt},m}) z_{\text{opt},m} \right]
\]

\[
\gamma_0 = \begin{cases} 
0 & \text{if } d^k > T_s, \text{ the corresponding probability } P_{\gamma_k = 0} = P_{d^k > T_s} \\
1 & \text{if } d^k < T_s, \text{ the corresponding probability } P_{\gamma_k = 1} = P_{d^k < T_s}
\end{cases}
\]

\[
\gamma_1 = \begin{cases} 
0 & \text{if } d^k < T_s, \quad d^k < d^{k-1} - T_s, \quad P_{\gamma_k = 0} = P_{d^k > T_s} P_{d^{k-1} < T_s} P_{d^k < d^{k-1} - T_s} \\
1 & \text{if } d^k > T_s, \quad d^{k-1} > T_s, \quad P_{\gamma_k = 1} = P_{d^k > T_s} P_{d^{k-1} < T_s} P_{d^k > d^{k-1} - T_s}
\end{cases}
\]

\[
\gamma_2 = \begin{cases} 
0 & \text{if } d^k < T_s, \quad d^{k-1} > T_s, \quad P_{\gamma_k = 0} = P_{d^k < T_s} P_{d^{k-1} > T_s} P_{d^k > d^{k-1} - T_s} \\
1 & \text{if } d^k > T_s, \quad d^{k-1} < T_s, \quad P_{\gamma_k = 1} = P_{d^k > T_s}
\end{cases}
\]

IET Cyber-Phys. Syst., Theory Appl., 2017, Vol. 2, Iss. 1, pp. 1–9
This is an open access article published by the IET under the Creative Commons Attribution License (http://creativecommons.org/licenses/by/3.0/)
where $K_{ik}$ is the optimal gain when $d_{ul} < d^k < d_{ul}(i + 1)$ at $k$, $J_i$ is the corresponding cost function.

Finally, the control law of the proposed PTSOC is derived as

$$\begin{align*}
u_k &= -K_{ik}z_k \\
K_k &= \sum_{j=1}^{n_d} P_{j}(B_d z)^T (A_d z + S_d z_{i}) = \sum_{j=1}^{n_d} P_{j} K_{ij}
\end{align*}$$

(8)

where $K_k$ is the optimal gain and $u_k$ is the control input. $S_d z_{i} \geq 0$ is the solution to the ARE. $n_d = d_{upper}/d_{un}$. $d_{upper}$ is the maximum delay in the sliding window; $P_{j} K_{ij}$ is the probability of $d_{ul}^k < d^k < d_{ul}(i + 1)$.

PTSOC, which considers PDF of delays, is able to optimise the trade-off between system state error and cost of regulation. Its control law is more accurate for the given network situation than that of SOC because the weights of each candidate control law are continuously updated to reflect the actual PDF of delays. In contrast, SOC considers only one of delay cases (7) in optimisation.

5 Stability analysis

Three theorems and their corresponding proofs are presented to demonstrate stability of the proposed PTSOC. Lyapunov-based stability analysis is used. Theorems 1 and 2 demonstrate asymptotic convergence of regulation error and estimation error of the control gain. Theorem 3 shows uniformly ultimately bounded (UUB) stability of the regulation error when the irremovable bias of PDF estimation exists.

In Theorem 1, the control law tuned by probability information guarantees the asymptotic convergence for regulation error with an assumption that delay PDFs are accurately estimated without bias. Theorem 2 relaxes the assumption in Theorem 1. It shows the control gain estimation asymptotically converges even if PDF estimation has an error provided it asymptotically converges to zero. Theorem 3 considers the irremovable bias of PDF estimation as a bounded disturbance. However, an UUB stability is guaranteed.

Theorem 1: (Asymptotic stability of regulation error with a perfect PDF estimation). Given the initial conditions as the system state $z_k$ and system matrices $A_{d(i)}$ and $B_{d(i)}$, let $u_k(z_k)$ be an initial admissible control policy for the CPS (1). Let the control update law be given by (8) with properly selected $Q_{zw}$ and $R_{zw}$. Let the PDF estimation be a no error estimation. Then, there exists a constant $K_{max}$ satisfying $K_{max} \leq (1 - a)/b$ such that the regulation error of system states converges to zero asymptotically in the mean.

Proof: Consider the following positive definite Lyapunov function candidate: $V_{z_k} = z_k^T K_z z_k$ is the state vector of $k$.

The system matrices are time-varying and stochastic; therefore, we consider $\Delta V_{z_k} = V_{z_k,i+1} - V_{z_k,i}$ for each possible system matrices $A_{d(i)}$ and $B_{d(i)}$, $m$ represents the number of the candidate systems. If the maximum value of $\Delta V_{z_k,i}$ is negative definite, the convergence of system states is proved.

$$\begin{align*}
\Delta V_{z_k,i} &= V_{z_k,i+1} - V_{z_k,i} \\
&= (A_{d(i)} z_k + B_{d(i) u_k})^T (A_{d(i)} z_k + B_{d(i) u_k}) - z_k^T z_k \\
&= \|A_{d(i) z_k} + B_{d(i) u_k}\|^2 - \|z_k\|^2 \\
&= \|A_{d(i) z_k} - B_{d(i) zw} \sum_{j=1}^{n_d} P_{j} K_{ij} z_k\|^2 - \|z_k\|^2 \\
&= \left(\|A_{d(i) z_k} - B_{d(i) zw} \sum_{j=1}^{n_d} P_{j} K_{ij} z_k\|^2 - 1\right) \|z_k\|^2
\end{align*}$$

With Cauchy-Schwarz inequality,

$$\Delta V_{z_k,i} \leq \left((\|A_{d(i) z_k}\| + \|B_{d(i) zw}\| \|K_{ij}\|) - 1\right) \|z_k\|^2 \leq \left((\|A_{d(i) z_k}\| + \|B_{d(i) zw}\| K_{max}) - 1\right) \|z_k\|^2 \leq \left((1 + b_{max} K_{max}) - 1\right) \|z_k\|^2 \forall k = 1, 2, \ldots, n_d$$

(9)

Next, the assumption that PDF estimation has no bias is relaxed with Theorem 2. The control gain estimation error still has an asymptotic convergence.

Theorem 2 (Control gain estimation error convergence): As the delay data keeps updating PDF identification and $\sum_{j=1}^{n_d} P_{j(i+1)} - \sum_{j=1}^{n_d} P_{j(i)} < 0$ is satisfied, then the estimation error for control gain $\hat{K}_{i+1} z_k$ asymptotically converges to zero.

Proof: First, we define the estimation error of control gain $K$ as $\hat{K}_i = K_k - \hat{K}_i = \sum_{j=1}^{n_d} P_{j} K_{ij} - \sum_{j=1}^{n_d} \hat{P}_{j} K_{ij}$, $\hat{P}_{j}$ is the actual probability at $k$. Then, Lyapunov function candidate is $V_{\hat{K}_i} = \|\hat{K}_{i+1} z_k\|$. (see equation at the bottom of next page)

Since $V_{\hat{K}_i}$ is the positive definite and $\Delta V_{\hat{K}_i,i}$ is negative definite provided $K_{max}$ is selected as above. Therefore, the regulation error converges to zero asymptotically.

Remark 3: Monotonically decreased estimation error for each delay range ($\|\hat{P}_{j(i+1)} - \hat{P}_{j(i)}\| < 0$) is not a necessary condition. The convergence only requires the estimation error of the entire PDF
((\|\sum_{j=1}^{n} \tilde{P}_{(k+1)j} - \|\sum_{j=1}^{n} \tilde{P}_{kj}\| < 0) monotonically decreasing over time. 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.

Theorem 2 considers an ideal case where the PDF estimation error converges to zero. In realistic case, there is an irremovable bias due to the finite sliding window size. Theorem 3 shows the UUB convergence of the regulation error in such a case.

**Theorem 3 (UUB stability of the regulation error):** Given the initial conditions as the system state \( z_0 \) and system matrices \( A_0 \) and \( B_0 \), let \( u_0(z_0) \) be an initial admissible control policy for the CPS (1). Let the control update law be given by (8) and if the disturbance induced by the irremovable bias of PDF estimation has a bound \( \|d_{\text{KDE}}\| \) and \( K_{\text{min}} < 1/b_{\text{min}} \) such that the regulation error of system states has an uniformly ultimate bounded convergence in the mean.

**Proof:** Consider the following positive definite Lyapunov function candidate: \( V_{z_k} = \frac{1}{2}z_k^Tz_k \). The corresponding estimated Lyapunov is \( \tilde{V}_{z_k} \). Therefore, \( \Delta \tilde{V}_{z_k} = \tilde{V}_{z_{k+1}} - \tilde{V}_{z_k} \). Similar to proof of Theorem 1, we consider \( \Delta \tilde{V}_{z_k} = \tilde{V}_{z_{k+1}} - \tilde{V}_{z_k} \) for each possible system matrices (\( A_{\text{dm}} \) and \( B_{\text{dm}} \)). \( m \) represents one of the possible cases. If the maximum value of \( \Delta \tilde{V}_{z_k} \) is negative definite, the system convergence is proved. The irremovable bias of PDF estimation is considered the system state disturbance \( d_k \) bounded by \( d_{\text{st}} \). (see equation at the bottom of this page)

where \( \Delta \) is positive definite, \( b_{\text{min}} = \min(\|B_{\text{dm}}\|, \|B_{\text{dm}}\|, \ldots, \|B_{\text{dm}}\|) \). \( K_{\text{min}} = \min(\|K_1\|, \|K_2\|, \ldots, \|K_n\|) \).

\[
\Delta V_{z_k} = V_{z_{k+1}} - V_{z_k} \\
= \tilde{V}_{z_{k+1}} - \tilde{V}_{z_k} \\
= (\tilde{K}_{k+1} - \tilde{K}_k)(\tilde{K}_{k+1} - \tilde{K}_k)^T \\
= (\sum_{j=1}^{n} \tilde{P}_{(k+1)j} - \sum_{j=1}^{n} \tilde{P}_{kj})^T (\sum_{j=1}^{n} \tilde{P}_{(k+1)j} - \sum_{j=1}^{n} \tilde{P}_{kj}) \\
= \sum_{j=1}^{n} (\sum_{j=1}^{n} \tilde{P}_{(k+1)j} - \sum_{j=1}^{n} \tilde{P}_{kj})^2 \\
= \sum_{j=1}^{n} P_{(k+1)j}^2 + \sum_{j=1}^{n} \tilde{P}_{kj}^2 - 2 \sum_{j=1}^{n} P_{(k+1)j} \tilde{P}_{kj} \\
= \Delta \left( \sum_{j=1}^{n} \tilde{P}_{(k+1)j} - \sum_{j=1}^{n} \tilde{P}_{kj} \right) \\
\leq \Delta \left( \sum_{j=1}^{n} \tilde{P}_{(k+1)j} - \sum_{j=1}^{n} \tilde{P}_{kj} \right) \|K_{\text{max}}\| \Delta > 0, \ K_{\text{max}} = \max(\|K_1\|, \|K_2\|, \ldots, \|K_n\|) \\

\[
\Delta \tilde{V}_{z_{km}} = \tilde{V}_{z_{km+1}} - \tilde{V}_{z_{km}} \\
= \|A_{\text{dm}} - B_{\text{dm}}K_{\text{dm}}z_k + d_k\|^2 - \|z_k\|^2 \\
= \left( \|A_{\text{dm}} - B_{\text{dm}}K_{\text{dm}}z_k + d_k\| + \|z_k\| \right) \left( \|A_{\text{dm}} - B_{\text{dm}}K_{\text{dm}}z_k + d_k\| - \|z_k\| \right) \\
= \Delta \left( \|A_{\text{dm}} - B_{\text{dm}}K_{\text{dm}}z_k + d_k\| - \|z_k\| \right) \\
\leq \Delta \left( \|A_{\text{max}} - b_{\text{max}}K_{\text{min}}z_k + d_{\text{st}}\| - \|z_k\| \right) \\
\leq \Delta \left( \|A_{\text{max}} + b_{\text{min}}K_{\text{min}}z_k + d_{\text{st}}\| - \|z_k\| \right) \\
\forall k = 1, 2, \ldots \\
\forall m = 1, 2, \ldots, n_d
\]
improved by employing PTSOC, even though the bound changes over time due to network changes or cyber-attacks. In realistic CPSs, topology variation and cyber-attacks bring more dynamics and uncertainties to the embedded network and entire CPS. Therefore, the bounds and constrains of delays made in ideal case are relaxed in Case C. This illustrates that PTSOC has a better performance in terms of overshoot, convergence time, and cost of regulation than that of the conventional SOC when the delay bound and PDF are time-varying.

Simulated benchmark example:
The simulations employ the continuous-time model of a batch reactor system [11, 13, 14] whose dynamic are given by

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

(9)

where \( x \in \mathbb{R}^{4 \times 1} \) and \( u \in \mathbb{R}^{2 \times 1} \).

The parameters of this CPS are selected as (i) the sampling time \( T_s = 100 \) ms and (ii) \( d = 2 \), while the delays are not bounded. The continuous-time model is converted to a discrete time model with stochastic parameters representing network dynamics [11].

### 6.1 Case A: bounded random delays with a fixed PDF

PTSOC is evaluated for an ideal case with delays that satisfy the a priori set bound constrains of SOC [11]. The delays follow a normal distribution \( d \sim N(0.15, 0.05^2) \). The upper bound is 0.2 s. Overshoot, convergence time, and cost of regulation are the primary metrics to evaluate SOC and PTSOC. The tests are repeated 50 times for the statistical validation.

In Figs. 3, 4 and Table 2, the performance of SOC and PTSOC are compared. Both of these controllers make the errors converge. PTSOC presents an opposite improvement with high percentages but closed values. This illustrates that SOC is optimal when the bounds and distribution of delays are satisfied the assumed constraints. Similarly, a longer convergence time of PTSOC is induced because the sliding window needs several iterations to converge the PDF estimation. Importantly, a significantly reduced cost by 92.1% is conspicuous. It demonstrates that PTSOC tuned by PDF of delays can optimise the trade-off between regulation error and cost. Overall, SOC and PTSOC have good and comparable performance when the delays are less than 0.2 s.

### 6.2 Case B: increased random delays with a fixed PDF

In this case, the delays increase beyond the initially set bounds of SOC while maintaining a fixed PDF distribution \( d \sim (0.3, 0.1^2) \). Such a case aims to demonstrate PTSOC still guarantee a good performance with delays with an unknown bound, while SOC performance degrades in terms of overshoot, convergence time, and the cost of regulation.

The performance of SOC and PTSOC are presented in Figs. 5 and 6. PTSOC has a significant improved performance when compared with SOC, as shown in Table 3. Its overshoot is reduced at least by 36.4%. The convergence time is reduced by 7.1%. Moreover, the cost of regulation is reduced by 94.9% thus the control inputs implemented by the actuators are within their preferred ranges. Such that the physical components are prevented from excessive wear and damages. Overall, SOC no longer guarantees the stability if the delay is over its upper bound. In contrast, PTSOC uses PDF information to tune the control law such that the performance and stability of the entire CPS is guaranteed.

Remark 5: In this case, the initial conditions are same as that in Case A. The initial control law of SOC is pre-tuned for the smaller delay bound. Consequently, SOC control law is selected inappropriately for the case with longer delays. Hence, for the first second, the system states have large deviations and overshoots as shown in Fig. 5. Over time, SOC updates the system model and allows it to converge. In contrast, PTSOC can tune its control laws based on

---

**Table 2** Performance comparison (statistical average values for 50 times tests)

| Performance metrics | SOC | PTSOC | Improvements (Ave (min, max)) (SOC-PTSOC)/SOC |
|---------------------|-----|-------|--------------------------------------------|
| overshoot           |     |       |                                            |
| fluid level, cm     | 9.1 | 9.3   | −1.5% (−3.6%, −0.2%)                      |
| inside temperature, K | 8.6 | 9.7   | −14.9% (−41.7%, −0.1%)                    |
| product outlet flow rate, g/s | 9.8 | 9.9   | −3.3% (−20.2%, 7.6%)                      |
| coolant outlet flow temperature, K | 7.3 | 8.2   | −12.0% (−42.9%, 1.4%)                     |
| convergence time, s | 3.7 | 4.0   | −40.4% (−55.6%, 7.8%)                     |
| cost of the first 20 s | 28,531.3 | 2248.4 | 92.1% (91.4%, 93.2%)                      |

Remark 4: Tuning the \( Q \) and \( R \) of PTSOC cost function can improve its performance in terms of overshoot and convergence time at the expense of a higher actuation cost. However, increasing the cost can lead to inefficient actuation and operating the actuators beyond its preferred range which leads to excessive wear and damages to actuators and the system.
both dynamic model changes and the drift of PDF of delays. Such that PTSOC quicker adapts to the uncertainties on system model and network delay.

6.3 Case C: high variation delay with time-varying PDF

Next, the distribution of delays is allowed to change over time. First, it follows a normal distribution with 0.45 s mean value. Then, the mean value jumps to 0.2 s at randomly selected time within [40 s, 50 s]. The simulations are repeated to obtain statistically valid comparisons. The results firmly indicate that tracking of PDF changes is critical in both system modelling and optimal controller design.

Fig. 7 shows one specific case when the distribution changed at time 47 s. The parameters $Q_z = 0.15 \times I_8 \times I_8$ and $R_z = I$ are set for the SOC. The simulations show that the controller can make the regulation error converge. However, the errors increase right after the change in PDF of delays. Additionally, the input command to actuator increases. Such large inputs often exceed the preferred operating range for actuators thus increasing wear and tear or saturating actuator response.

In Fig. 8, the performance of PTSOC is presented. PTSOC convergence is sped up by using the updated PDF information. Hence, a shorter convergence time and a lower cost of regulation (Table 4) are observed.

Overall, SOC makes the error converge with a significant overshoot and a high actuation cost. Instead, the overshoot and cost of PTSOC are significantly reduced by several orders of magnitude. Table 4 shows the average value for 50 simulations. As the weights of each control input are updated continuously in terms of cost function, PTSOC has a better performance on overshoot and cost of regulation (reduced by 96.1%). A cost function with the optimal parameters ($Q$ and $R$) selection not only guarantees that algebraic Riccati equation has a finite solution but also reduces the control input cost. The convergence time of PTSOC is improved by 17.3%. Overall, PTSOC significantly improves the system performance in terms of overshoot, convergence time, and cost for every simulated scenario. Therefore, PTSOC indeed strengthens the resilience of the entire CPS.

Remark 6: In this case, the delay PDF is kept constant for at least 40 s to allow both PTSOC and SOC to achieve initial convergence. With such ‘pre-training’ process, the physical system model (1) is accurately tuned. When the change of PDF occurs, only the network dynamic terms in the model (2) are updated. Thus, both SOC and PTSOC perform better on overshoots than that for Case B (Table 3).

### Table 3 Performance comparison (statistical average values for 50 times tests)

| Performance metrics       | SOC       | PTSOC     | Improvements (Ave (min, max)) ((SOC-PTSOC)/SOC) (%) |
|---------------------------|-----------|-----------|-----------------------------------------------------|
| overshoot                 | 101.6     | 42.6      | 55.6% (10.5%, 83.0%)                                 |
| fluid level, cm           | 54        | 32        | 36.4 (~26.1%, 67.0%)                                 |
| inside temperature, K     | 133.4     | 53.7      | 52.4% (15.2%, 86.3%)                                 |
| product outlet flow rate, g/s | 48.9    | 25.1      | 42.7% (~6.6%, 72.1%)                                 |
| coolant outlet flow temperature, K | 4.2   | 3.9       | 7.1% (~34.4%, 31.3%)                                 |
| convergence time, s       | 154,530.1 | 7,891.4   | 94.9% (94.2%, 95.7%)                                 |
| cost of the first 20 s     |           |           |                                                     |

### Table 4 Performance comparison (statistical average values for 50 times tests)

| Performance metrics       | SOC       | PTSOC     | Improvements (Ave (min, max)) ((SOC-PTSOC)/SOC) (%) |
|---------------------------|-----------|-----------|-----------------------------------------------------|
| overshoot                 | 187.7     | 34.0      | 82.0% (80.6%, 85.7%)                                 |
| fluid level, cm           | 26.1      | 4.8       | 81.7% (80.5%, 85.6%)                                 |
| inside temperature, K     | 111.9     | 37.6      | 66.4% (62.6%, 73.4%)                                 |
| product outlet flow rate, g/s | 36.3    | 6.7       | 81.4% (80.1%, 85.3%)                                 |
| coolant outlet flow temperature, K | 9.4    | 7.8       | 17.3% (5.3%, 22.8%)                                 |
| convergence time, s       | 208,299.5 | 8191.4    | 96.1% (95.9%, 96.2%)                                 |
| cost after distribution changes |           |           |                                                     |
7 Conclusions

In this work, PTSOC is proposed to address uncertainties of the embedded cyberspace, particularly delays with an unknown bound and time-varying distribution. The simulation results show that PTSOC has a reduced overshoot (by 80%), convergence time (by 17.3%), and cost of regulation (by 96%) over that of the SOC. Continuous update of the probability weights speeds up convergence, optimises control input selection, and make control law adapt to delays with unknown bound and time-varying distribution. It facilitates the trade-off between system states and cost of regulation. Additionally, the constraints of known bound of delays and fixed distribution in existing works are relaxed. Overall, reliability of CPSs in terms of resilience is improved.

8 References

1 Xie, S., Low, K., Gunawan, E.: ‘An adaptive tuning algorithm for IEEE 802.15.4-based network control system’, IEEE Ninth Int. Conf. on Intelligent Sensors, Sensor Networks and Information Processing, 2014, pp. 1–6
2 Lee, K., Lee, S., Lee, M.: ‘QoS-based remote control of networked control systems via profibus token passing protocol’, IEEE Trans. Ind. Inf., 2005, 1, (3), pp. 183–191
3 Ji, X., Yu, H., Fan, G., et al.: ‘Attack-defense trees based cyber security analysis for CPSs’. 2016 17th IEEE/ACIS Int. Conf. on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), Shanghai, 2016, pp. 693–698
4 Kashell, R., Chen, I.R.: ‘Modeling and analysis of attacks and counter defense mechanisms for cyber physical systems’, IEEE Trans. Reliab., 2016, 65, (1), pp. 350–358
5 Gao, H., Chen, T.: ‘Network-based H∞ output tracking control’, IEEE Trans. Autom. Control, 2008, 53, (3), pp. 655–667
6 Gao, H., Meng, X., Chen, T.: ‘Stabilization of networked control systems with a new delay characterization’, IEEE Trans. Autom. Control, 2008, 53, (9), pp. 2142–2148
7 Hao, F., Zhao, X.: ‘Linear matrix inequality approach to static output feedback stabilisation of discrete-time networked control systems’, IET Control Theory Appl., 2010, 4, (7), pp. 1211–1221
8 Tian, E., Yue, D., Peng, C.: ‘Brief paper: reliable control for networked control systems with probabilistic sensors and actuators faults’, Inst. Inf. Control Eng., 2010, 4, (8), pp. 1478–1488
9 Liu, G., Xia, Y., Hu, W.: ‘Design and stability criteria of networked predictive control systems with random network delay in the feedback channel’, IEEE Trans. Syst. Man Cybern. C Appl. Rev., 2007, 37, (2), pp. 173–184
10 Thai, U., Fischione, C., Johansson, K.H., et al.: ‘Energy-efficient sampling of networked control systems over IEEE 802.15.4 wireless networks’, Automatica, 2013, 49, pp. 712–724
11 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
12 Blandell, R., Duncan, A.: ‘Kernel regression in empirical microeconomics’, J. Hum. Res. 1998, 33, (1), pp. 62–87
13 Silverman, B.W.: ‘Density estimation for statistics and data analysis’ (CRC Press, 1986, vol. 26)
14 Hutter, C., Ensen, O., Teela, A.: ‘Graph bundling by kernel density estimation’, Comput. Graph. Forum. 2012, 31, (3 PART1), pp. 865–874
15 Bors, A.G., Nasis, N.: ‘Kernel bandwidth estimation for nonparametric modelling’, IEEE Trans. Syst. Man Cybern. B Cybern., 2009, 39, (6), pp. 1543–1555
16 Calabrese, R., Zenga, M.: ‘Bank loan recovery rates: measuring and nonparametric density estimation’, J. Bank. Fin., 2010, 34, (5), pp. 903–911
17 Elgammal, A., Duraiswami, R., Davis, L.S.: ‘Efficient kernel density estimation using the fast gauss transform with applications to color modeling and tracking’, IEEE Trans. Pattern Anal. Mach. Intell., 2003, 25, (11), pp. 1499–1504
18 He, Y., Mao, Y., Chen, W., et al.: ‘Nonlinear metric learning with kernel density estimation’, IEEE Trans. Knowl. Data Eng., 2015, 27, (6), pp. 1602–1614
19 Chen, X., Tharmarasa, R., Kirubarajan, T., et al.: ‘Online clutter estimation using a Gaussian kernel density estimator for multitarget tracking’, IET Radar Sonar Navig., 2015, 9, (1), pp. 1–9
20 Gisbert, F.: ‘Weighted samples, kernel density estimators and convergence’, Empir. Econ., 2003, 28, pp. 335–351
21 Carnevale, D., Teel, A.R., Nesci, D.: ‘A lyapunov proof of an improved maximum allowable transfer interval for networked control systems’, IEEE Trans. Autom. Control, 2007, 52, (5), pp. 892–897

IET Cyber-Phys. Syst., Theory Appl., 2017, Vol. 2, Iss. 1, pp. 1–9
This is an open access article published by the IET under the Creative Commons Attribution License (http://creativecommons.org/licenses/by/3.0/)