

# PkCOs: Synchronization of Packet-Coupled Oscillators in Blast Wave Monitoring Networks

Yan Zong<sup>1</sup>, Member, IEEE, Xuewu Dai<sup>2</sup>, Member, IEEE, Shang Gao<sup>3</sup>, Member, IEEE,  
Pep Canyelles-Pericas<sup>4</sup>, Member, IEEE, and Shuxin Liu

**Abstract**—Blast waves with a large amount of energy, from the use of explosive weapons, is a major cause of traumatic brain injury in armed and security forces. The monitoring of blast waves is required for defence and civil applications. The utilization of wireless sensing technology to monitor blast waves has shown great advantages, such as easy deployment and flexibility. However, due to drifting embedded clock frequency, the establishment of a common timescale among distributed blast monitoring sensors has been a challenge, which may lead to a network failing to estimate the precise acoustic source location. This work adopts a packet-coupled oscillators (PkCOs) protocol to synchronize drifting clocks in a wireless blast wave monitoring network. In order to address packet collisions caused by the concurrent transmission, an anti-phase synchronization solution is utilized to maintain clock synchronization, and the corresponding superframe structure is developed to allow the hybrid transmission of the *Sync* packet and the blast wave monitoring data. As a network scales up and the hop distance grows, the packet exchange lag increases during a superframe. This, along with the drifting clock frequency, leads to the degradation of synchronization performance while the clock frequency is usually assumed to be zero and nondrifting. Thus, a compensation strategy is proposed to eliminate the joint impacts and to improve synchronization precision. The theoretical performance analysis of the PkCOs algorithm in the network is presented along with verification by simulation means. Finally, the performance of the PkCOs synchronization protocol is evaluated on an IEEE 802.15.4 hardware testbed. The experimental results show that the PkCOs algorithm provides an alternative clock synchronization solution for blast wave monitoring networks.

**Index Terms**—Anti-phase synchronization, desynchronization, packet-coupled oscillators (PkCOs), pulse-coupled oscillators (PCOs), time synchronization.

## I. INTRODUCTION

OVER the last two decades, with the increasing use of explosive devices, such as land mines, improvised explosive devices (IEDs), and rocket-propelled grenades (RPGs) in modern warfare, the mild traumatic brain injury (mTBI) has become prevalent in the armed forces [1]. Highly accurate blast wave monitoring is required in air explosions for defence and civil applications [2]. Compared with the optical fibre and radar-based approaches, distributed sensor networks have been widely used for monitoring blast waves due to their low-cost and easy-to-employ characteristics [3]. In order to guarantee that the meaningful results (e.g., a precise acoustic source location) are obtained via a distributed sensor network, all sensor nodes in a network need to share a common timescale. Thus, the key enabling technology, time synchronization (TS), is utilized to achieve such a goal.

Due to the importance of time synchronization, apart from the communication engineering research society, it attracts also interest from the mathematics and physics communities. Thus, we investigate the topic of synchronization under the framework of networked oscillators. A typical networked oscillators model, pulse-coupled oscillators (PCO) [4], is particularly suitable for distributed sensor networks due to its inherent simplicity, scalability [5], and discretization [6] properties.

In PCO, each oscillator behaves as an uncoupled oscillator running on the unit circle [Fig. 1(a)], with its state  $P$  increasing from zero to the threshold value  $\varphi$ . Once the state  $P$  reaches  $\varphi$  (arriving at the right), the oscillator fires (i.e., a *Pulse* with the firing information is transmitted) whilst  $P$  is reset. This firing-resetting behavior indeed is similar to the periodic resetting feature in the embedded clock, which is implemented by a crystal oscillator and a counter register [6]. However, this model still cannot be directly employed to a real network, because of the assumptions [e.g., all the oscillators sending *Pulses* simultaneously when the synchronization is achieved, see Fig. 1(b)] [7]. Thus, the PCO model needs to be improved, and the packet-coupled oscillators (PkCOs) framework was proposed in earlier work [21]. For the PkCOs algorithm, we model each clock as an oscillator moving clockwise on the unit circle, as indicated in Fig. 1(a). Upon arriving at the right

Manuscript received September 4, 2021; revised October 9, 2021; accepted November 1, 2021. Date of publication November 8, 2021; date of current version June 23, 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 61773111, Grant 61790574, and Grant U1834211; in part by the Fundamental Research Funds for the Central Universities under Grant 309181A8804 and Grant 30919011263; in part by the Natural Science Foundation of Jiangsu Province, China, under Grant BK20190464; in part by the Jiangsu Planned Projects for Postdoctoral Research Funds under Grant 1003-YBA20012; and in part by the Chinese Postdoctoral Science Foundation under Grant 2020M671481. (Corresponding authors: Xuewu Dai; Shang Gao.)

Yan Zong is with the State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, Shenyang 110819, China, and also with the School of Aerospace, Transport and Manufacturing, Cranfield University, Cranfield MK43 0AL, U.K. (e-mail: y.zong@cranfield.ac.uk).

Xuewu Dai and Shuxin Liu are with the State Key Laboratory of Synthetical Automation for Process Industries and the College of Information Science and Engineering, Northeastern University, Shenyang 110819, China (e-mail: daixuewu@mail.neu.edu.cn; 2102032@stu.neu.edu.cn).

Shang Gao is with the School of Mechanical Engineering, Nanjing University of Science and Technology, Nanjing 210016, China (e-mail: shang.gao@njust.edu.cn).

Pep Canyelles-Pericas is with the Department of Integrated Devices and Systems, MESA+ Institute for Nanotechnology, University of Twente, 7522 NH Enschede, The Netherlands (e-mail: j.canyellespericas@utwente.nl).

Digital Object Identifier 10.1109/IIOT.2021.3126059

TABLE I  
PERFORMANCE COMPARISON IN CLOCK SYNCHRONIZATION PROTOCOLS

Protocol	Hardware platform	Clock frequency (resolution)	Correction method	Network topology	TS precision
RBS [12]	Berkeley mote	500 kHz ( $2 \mu s$ )	Linear regression	Complex network with 5 nodes	$11.2 \mu s$ (6 ticks)
PISync [13]	MICAz	1 MHz ( $1 \mu s$ )	Proportional controller	20-hop linear network	$17 \mu s$ (17 ticks)
RMTS [14]	SSWS	32 MHz ( $31.25 ns$ )	Maximum-likelihood estimation	24-hop linear network	$15 \mu s$ (480 ticks)
R-Sync [15]	88MZ100	32 MHz ( $31.25 ns$ )	n/a	10-hop linear network	$450 \mu s$ (14400 ticks)
RFA [17]	MICAz	7.3728 MHz ( $135.63 ns$ )	n/a	Complex network consisting of 24 nodes	$100 \mu s$ (737 ticks)
PulseSS [20]	MICAz	7.3728 MHz ( $135.63 ns$ )	Dithered quantisation function	Complex network consisting of 45 nodes	$410 \mu s$ (3023 ticks)

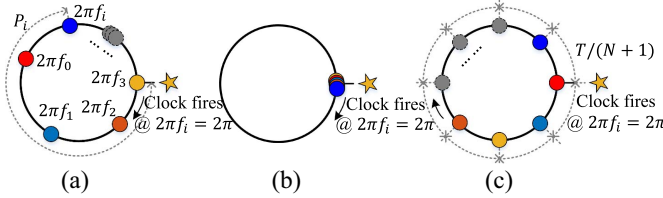


Fig. 1. Networked oscillators at (a) initial state, (b) synchronized state, and (c) anti-phase synchronized state.

(i.e.,  $P$  reaches  $\varphi$ ), the oscillator fires and sends a *Sync* packet for synchronization purposes. At the same time, the oscillator continuously moves clockwise, namely, the state  $P$  increases from zero again. Moreover, instead of using a constant amount  $\epsilon$  for clock adjustment (in PCO), this work adopts the dynamic adaptive correction solution to correct the drifting clock. The proposed clock adjustment method possesses the feature of compensating for the effects of the drifting clock frequency and delays. We also provide an anti-phase synchronization solution to avoid the occurrence of packet collisions, and the corresponding superframe structure allows the hybrid transmission of the control traffic and the data stream on the same channel.

#### A. Related Work

In order to determine the accurate position of an acoustic source, the solutions of time difference of arrival (TDoA) [8] and time of arrival (ToA) [9] have been the popular methods. Through processing the ToA data from different sensor nodes, the acoustic source's position, with respect to the wireless network, can be estimated [10]. Clearly, the performance of these methods are highly dependent on the time synchronization accuracy in the sensor network. In addition to the PCO model, the communication engineering community also has proposed many packet-exchange protocols for wireless networks.

1) *Packet-Exchange Synchronization Protocols*: These algorithms can measure clock offset through exchanging periodical timestamped packets among connected nodes. The employment of the estimated offsets lets a network achieve clock synchronization. In the literature, there exist two main solutions for improving synchronization performance. One

approach is to estimate a more accurate clock offset via the frequent packet exchange. For example, the precision time protocol (PTP) [11] improves accuracy by using the two-way message exchange mechanism, and reference broadcast synchronization (RBS) [12] removes the effects of send and access delays under the receiver-receiver scheme. For recent works, at synchronized state, the transmission of multiple packets (e.g., 100 in [14], and more than 32 in [15]) during each synchronization cycle  $T$  leads to better performance. However, the frequent radio frequency (RF) communication puts a strain in the node battery. The proposed method of this work only needs to send one packet in each cycle  $T$ . The other method is to utilize advanced processing techniques (see Table I). Even though the use of these technologies can achieve higher performance, the wireless nodes have to face challenges of considerable computational and memory overheads. The floating-point calculation in, for example, maximum-likelihood estimation, may also reduce its calculation accuracy. Thus, the controlling-strategy-based solution has been adopted in [13]. Even though [13] states that the proportional-integral (PI) controller is used, from (6) of this cited work, it actually is a proportional (P) controller.

From Table I, it can be seen that the TS protocols are implemented on different hardware platforms selecting the varying crystal oscillators as the clock sources. Since this is a hardware-based solution, it is unfair to directly compare their performance [16]. By measuring synchronization precision in clock ticks,<sup>1</sup> we can see RBS and PISync possess better performance.

2) *PCO-Like Protocols*: In the typical PCO-like protocol (e.g., [17]), once a network achieves synchronization, all the *Sync* packets from synchronized nodes interfere with each other (i.e., packet collisions), and no packets can be received successfully. The occurrence of packet collisions may also let the achieved synchronization be lost gradually, owing to the frequency difference among node clocks. To avoid packet collisions, the solution of anti-phase synchronization [18] [see Fig. 1(c)] is adopted, in order to guarantee all the *Sync* packets

<sup>1</sup>This metric relates to the maximum achievable precision on the underlying platform.

are scheduled to specified time slots for transmission, thereby reducing the probability of concurrent transmission.

In [19], once a network achieves clock synchronization,  $N + 1$  sensor nodes send the *Sync* packets (i.e., the control traffic) with interval space  $T/(N + 1)$  during each time synchronization cycle  $T$ . This means that the anti-phase synchronization is achieved in the network, as shown in Fig. 1(c). However, with this solution, it is difficult to send the blast wave monitoring data. The scheduling mechanism in [20] can solve this issue, where the complex dithered quantization function is still used, and the resource-constrained sensor nodes have to offer the extra resource to realize this. Hence, we provide a simple anti-phase synchronization solution, and the corresponding superframe design allows the hybrid transmission of both the control traffic and the data stream on the same wireless channel; meanwhile, the proposed scheme has a sleep period.

In the large-scale distributed network, with the increase of hop distance, the packet exchange lag (i.e., the time difference between scheduled slots) on wireless nodes grows. This and drifting clock frequency jointly have negative effects on the synchronization performance; while, the assumption of no drifting frequency is usually adopted in the literature (e.g., [19] and [20]). We study these two impacts, and propose a compensation strategy to improve the TS precision in the large-scale multihop wireless network.

Due to the limitation of RF communication, the packet exchange delay occurs between the transmission and reception of wireless packets. There exists a varying processing delay for calculation and register access, because of the processor's architecture and the impossibility of real-time computing [21]. However, the clock correction strategy in the PCO model fails to compensate for the effects of these two delays. Thus, this article utilizes the estimated clock offset, obtained from the timestamp, to correct the local clock. The PI controller is adopted to naturally eliminate the effects of varying processing delay.

### B. Contribution and Paper Organization

In this article, we adopt a PI-based PkCOs protocol to synchronize drifting clocks in blast wave monitoring networks. The packet-coupled synchronization algorithm utilizes a simple scheduling scheme to allow the hybrid transmission of the *Sync* packet and the blast wave monitoring data on the same wireless channel; meanwhile, it also reduce the possibility of *Sync* concurrent transmission. In addition to the PI controller, which can automatically remove the effects of varying processing delay, a compensation strategy is also proposed to eliminate the joint effects resulting from the drifting embedded clock frequency and *Sync* scheduling. Furthermore, the theoretical performance of the PkCOs algorithm in the blast wave monitoring network is studied and verified by simulation means. Finally, the PI-based PkCOs algorithm is evaluated in circular and triangle blast wave monitoring networks. The hardware experimental results show that the proportional-integral PkCOs protocol outperforms the precision of existing clock synchronization algorithms and provides an alternative TS solution for blast wave monitoring networks.

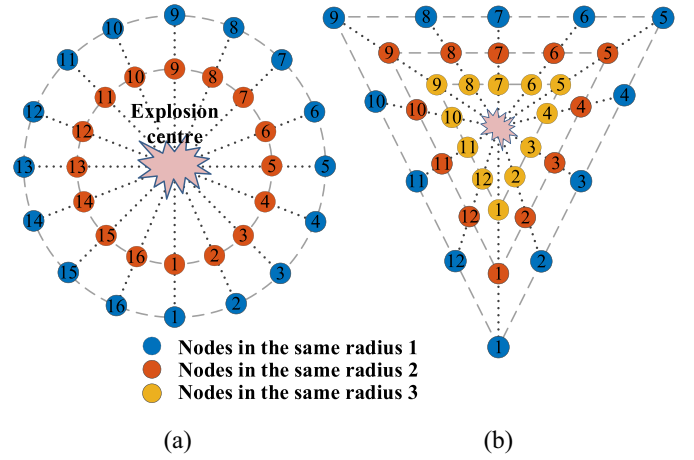


Fig. 2. (a) Circular and (b) triangle network topologies for blast wave monitoring.

The remainder of this article is organized as follows: Section II presents the mathematical problem formulation. Section III introduces the PI-based packet-coupled synchronization scheme and the theoretical analysis of the PkCOs algorithm in the blast wave monitoring network. The simulation and experimental results are shown in Sections IV and V, respectively. Finally, Section VI concludes this work.

## II. PROBLEM FORMULATION

Let the blast wave monitoring network in Fig. 2 be described by the directed and weighted graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$ , where  $\mathcal{V} = \{0, 1, \dots, N\}$  denotes the set of wireless nodes, and the set of edges  $\mathcal{E}$  induced by the adjacency matrix  $\mathcal{A} = [a_{ij}] \in \mathbb{R}^{(N+1) \times (N+1)}$ . The network consists of a single root node (i.e.,  $i = 0$ ) and a set of regular nodes represented by the set  $\mathcal{N} = \{i : i \in \mathcal{V}, i \geq 1\}$ . The root node is unique and equipped with a global positioning system (GPS) clock to provide the reference time to all the regular nodes. For the  $i$ th regular node, if it can receive the packet transmitted from a node  $j$ , the weight  $a_{ij}$ , which is an element of  $\mathcal{A}$ , is equal to 1 (i.e.,  $a_{ij} = 1$ , and  $a_{ij} \neq a_{ji}$ ); otherwise, it is zero. Due to the limitation of RF communication in wireless networks, the graph  $\mathcal{G}$  has no self-loops (i.e.,  $a_{ii} = 0$  for all  $i \in \mathcal{V}$ ). The in-degree of node  $i$  is defined as  $\deg_i = \sum_{j=0}^N a_{ij}$ . Thus, the Laplacian matrix  $\mathcal{L} = [l_{ij}] \in \mathbb{R}^{(N+1) \times (N+1)}$  of  $\mathcal{G}$  is represented by  $\mathcal{L} = \mathcal{D} - \mathcal{A}$ , where  $\mathcal{D} = \text{diag}(\deg_0, \deg_1, \dots, \deg_N)$  is the diagonal degree matrix. Moreover, the  $h$ th sensor node on radius  $c$  in Fig. 2 is represented by node  $i$  (i.e.,  $i = \sum_{q=2}^c M + h$ ), and each radius possesses  $M$  nodes. In the following, the reference clock is called the master clock, and the root node and regular node are referred to as the master node and sensor node, respectively.

As a result of its tradeoff between cost and signal accuracy, this work selects the crystal oscillator as a clock source [6]. The nonidentical model is utilized to mathematically describe the behavior of a crystal-oscillator-based clock. First, let us refer to the case of the master clock running at the nominal frequency  $f_0 = 1/\tau_0$ , where  $\tau_0$  is the nominal period. The linear monotonically increasing variable  $t[n]$ , following  $t[n] = n\tau_0$ , is usually used to denote the time reported by a perfect



clock at the  $n$ th clock update event. However, the time  $t[n]$  fails to model the clock's resetting feature. From [21], the clock state  $P_0[n]$  is adopted to describe the behavior of clock resetting, and  $P_0[n]$  at the  $n$ th event is given as

$$P_0[n] = t[n] - \sum_{q=1}^k \varphi_0[q] \quad (1)$$

where  $k$  means the total number of clock resetting has occurred from  $n = 0$  to the  $n$ th clock event, and it also indicates that the clock is at the  $k$ th synchronization cycle. The threshold of the master clock at the  $k$ th cycle is  $\varphi_0[k]$ . In practice,  $\varphi_0[k]$  may be constant, which is equal to the time synchronization cycle  $T$  (i.e.,  $\varphi_0[k] = T$ ). In embedded systems, the clock threshold is much greater than the clock period  $\tau_0$ ; then it is reasonable to assume that the clock updates  $m_0$  times during a single synchronization cycle, yielding  $T = m_0\tau_0$ .  $k$  can be calculated as  $k = \lfloor n/m_0 \rfloor$ , where the floor function  $\lfloor n/m_0 \rfloor$  represents the largest integer value not greater than  $n/m_0$ .

Owing to internal factors and environmental conditions, the  $i$ th clock state  $P_i[n]$  cannot be the same as the master clock state  $P_0[n]$ . According to [21], the state  $P_i[n]$  of the  $i$ th sensor node is modeled as

$$P_i[n] = t[n] + \frac{\sum_{q=0}^n \chi_i \tau_0}{f_0} + \frac{\phi_i[n]}{2\pi f_0} - \sum_{q=1}^k \varphi_i[q] \quad (2)$$

where  $\chi_i$  is the frequency deviation between the  $i$ th clock frequency  $f_i$  and  $f_0$  (i.e.,  $\chi_i = f_i - f_0$ ). The instantaneous phase variation  $\phi_i[n]$  is the Gaussian random process representing all instant phase fluctuations from  $t[0]$  to  $t[n]$  [22].  $\varphi_i[k]$  is the  $i$ th clock threshold.

Let the clock offset  $\theta_i[n]$  denote the difference between the  $i$ th clock state  $P_i[n]$  and  $P_0[n]$ , following:

$$\theta_i[n] = P_i[n] - P_0[n]. \quad (3)$$

By substituting (1) and (2) into (3), the  $i$ th offset  $\theta_i[n]$  is

$$\theta_i[n] = \frac{\sum_{q=0}^n \chi_i \tau_0}{f_0} + \frac{\phi_i[n]}{2\pi f_0} - \sum_{q=1}^k \Delta\varphi_i[q] \quad (4)$$

where  $\Delta\varphi_i[k]$  is the threshold difference between the  $i$ th sensor node clock and master clock.

Next, by calculating the difference of offsets in two consecutive clock events and transforming the clock offset from  $n$ -dimension (i.e., clock update period) to  $k$ -dimension (i.e., time synchronization cycle) [23], a concise nonidentical clock model, which is a state equation in the closed-loop time synchronization system (see Fig. 4), is obtained

$$\theta_i[k+1] = \theta_i[k] + \omega_i[k] \quad (5)$$

where  $\bar{\omega}_{\theta_i}[n] = (\phi_i[n+1] - \phi_i[n])/2\pi f_0$ , and  $\omega_i[k] = \gamma_i T - \Delta\varphi_i + [1, 1, \dots, 1][\bar{\omega}_{\theta_i}[km_0], \bar{\omega}_{\theta_i}[km_0+1], \dots, \bar{\omega}_{\theta_i}[(k+1)-1]]^T$  is the clock offset noise, which is considered as a Gaussian random noise process [22].  $\gamma_i$  is the  $i$ th clock skew,<sup>2</sup> representing the normalized difference between  $f_i$  and  $f_0$  (i.e.,  $\gamma_i = (f_i - f_0)/f_0$ ).

<sup>2</sup>In practice, it is represented by the clock performance with the unit of parts per million (ppm).

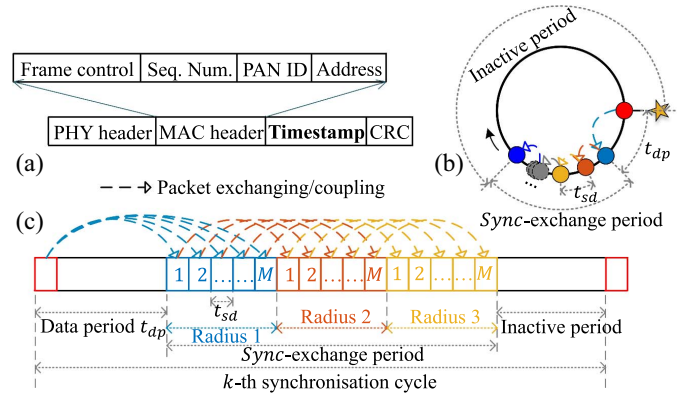


Fig. 3. (a) Structure of a Sync packet. (b) PkCOs at the proposed anti-phase synchronization state. (c) Proposed superframe.

In the PkCOs model, since all clocks are represented as oscillators spinning around the unit circle [see Fig. 1(a)], the order parameter can be utilized as a metric to macroscopically measure synchronization performance, following:

$$r[k]e^{j\psi[k]} = \frac{1}{N+1} \sum_{i=0}^N e^{j(\theta_i[k] + d_i)} \quad (6)$$

where the radius  $r[k]$  is the magnitude of the order parameter,  $\psi[k]$  is the average phase, and  $d_i$  is the  $i$ th sensor node anti-phase synchronization duration. This parameter is described in (9). For example, if all clocks are perfectly synchronized, then  $r[k] = 1$ ; and if all clocks (i.e., oscillators) are scattered around the unit circle, then  $r[k] \approx 0$ . The purpose of synchronization is to keep the order parameter  $r[k]$  as close to one as possible. That is

$$\lim_{k \rightarrow \infty} r[k] = 1. \quad (7)$$

In the next section, the PI-based packet-coupled synchronization scheme is introduced to explain the proposed algorithm's principle of achieving time synchronization. The theoretical analysis of PkCOs in a blast wave monitoring network is also presented.

### III. PROPORTIONAL-INTEGRAL PACKET-COUPLED SYNCHRONIZATION SCHEME

In the PkCOs protocol, the packet-coupled synchronization scheme provides a solution for estimating and correcting the  $i$ th sensor node clock offset. At the  $k$ th synchronization cycle, upon the reception of a  $\text{Sync}_j$  packet [see Fig. 3(a)] from the  $j$ th node after the packet coupling delay  $\kappa_j[k]$ , node  $i$  associates to it the timestamp  $\hat{P}_i[k]$ . Due to the impossibility of real-time computing and the architecture of a microprocessor, there exists a processing delay  $\eta_i[k]$  for the  $i$ th sensor node to determine its offset estimate  $\hat{\theta}_i[k]$  and to adjust the local clock by employing correction input  $\tilde{u}_i[k]$ .

#### A. PI-Based Packet-Coupled Synchronization Mechanism

For the  $k$ th synchronization cycle, the  $j$ th clock fires, and node  $j$  directly sends a  $\text{Sync}_j$  packet at  $t_{k_j}$ . After  $\kappa_j[k]$ , node  $i$  receives  $\text{Sync}_j$  at  $t_{k_j} + \kappa_j[k]$ . Once the  $i$ th node receives the

$\text{Sync}_j$  packet, a timestamp  $\hat{P}_i[k]$  is generated by reading the counter register, yielding

$$\hat{P}_i[k] = P_i^{t_{k_j} + \kappa_j[k]} \quad (8)$$

where the packet coupling delay  $\kappa_j[k]$  is regarded as a Gaussian random process with the nonzero mean of  $\bar{\kappa}_j$  and the finite variance of  $\sigma_{\kappa_j}^2$ . That is,  $\kappa_j[k] \sim (\bar{\kappa}_j, \sigma_{\kappa_j}^2)$  [6].

In order to avoid the occurrence of packet collisions, the anti-phase scheduling method is widely adopted. However, by doing this is difficult to send the blast wave monitoring data [see Fig. 1(c)]. This article proposes an anti-phase synchronization scheme to schedule node  $i$  to slot  $d_i$  for  $\text{Sync}$  packet transmission [see Fig. 3(b)]. Fig. 3(c) shows the superframe structure, where the data period  $t_{dp}$  is to transmit the blast wave monitoring data, and in the  $\text{Sync}$ -exchange period, the unit time  $t_{sd}$  is allocated to node  $i$  for  $\text{Sync}$  transmission. In addition, the superframe also has an inactive period, where the sensor node can turn off its transceiver circuit. From Fig. 3(b) and (c), it can be seen that once all nodes are scheduled to different time slots for packet transmission, all clocks (i.e., oscillators) are also anti-phase synchronized. The anti-phase synchronization duration of the  $i$ th sensor node is written as

$$d_i = \begin{cases} 0, & \text{if } i = 0 \text{ (master node)} \\ t_{dp} + (i - 1)t_{sd}, & \text{if } i \geq 1 \text{ (sensor node)} \end{cases} \quad (9)$$

where  $t_{dp}$  is the data period duration, and  $t_{sd}$  means the slot duration.

Using the local timestamp  $\hat{P}_i[k]$ , the offset estimate  $\hat{\theta}_i[k]$ , representing the offset measurement between node  $i$  and  $j$ , can be calculated

$$\hat{\theta}_i[k] = \theta_i^{t_{k_j} + \kappa_j[k]}. \quad (10)$$

The use of a PI controller not only avoids the occurrence of over-correction resulting from the timestamp inaccuracy but also eliminates the effects of varying processing delay  $\eta_i[k]$ . This article uses the PI controller for a multihop blast wave monitoring network, following:

$$\begin{cases} w_i[k + 1] = w_i[k] + \beta \sum_{j=0}^N l_{ij} (\Delta d_{ij} - \hat{\theta}_j[k] + \bar{\kappa}_j + \mu_i) \\ \tilde{u}_i[k] = w_i[k] + \alpha \sum_{j=0}^N l_{ij} (\Delta d_{ij} - \hat{\theta}_j[k] + \bar{\kappa}_j + \mu_i) \end{cases} \quad (11)$$

where  $w_i[k]$  is the integral controller.  $\alpha$  and  $\beta$  are the coefficients of the proportional and integral controllers, respectively.  $\Delta d_{ij} = d_j - d_i$  is the difference of anti-phase synchronization duration between nodes  $i$  and  $j$ . To address the issue of concurrent transmission, the comparison between  $\Delta d_{ij}$  and offset estimate  $\hat{\theta}_j[k]$  is fed to the PI controller. The packet exchange delay  $\kappa_i[k]$  is almost deterministic with little variance, which benefits from the collision-free packet transmission approach. By measuring the deterministic part of  $\kappa_i[k]$ , its impacts on the TS precision can be removed [6]. Note that  $\tilde{u}_i[k]$  of this work is equivalent to the coupling strength (which is a constant amount) in the PCO model. As a result of the use of the networked PI controller, the coupling strength in PkCOs is a dynamic adaptive correction input.

In the large-scale wireless network, with the increase in hop distance, the packet exchange lag (i.e., the time difference

between the  $i$ th node and the master) of exchanging the wireless  $\text{Sync}$  packet on a wireless node grows. Thus, the constant clock skew  $\gamma_i$  and  $d_i$  of (9) jointly have negative effects on the synchronization performance; while, the assumption of no drifting frequency is usually adopted [20]. This work removes these joint impacts via the term  $\mu_i$  of (11), following:

$$\mu_i = \rho \gamma_i d_i \quad (12)$$

where  $\rho$  is the coefficient of the compensation strategy.

Unavoidably, due to the existence of hardware interrupt latency, and the impossibility of real-time register access and real-time computing, there exists a varying processing delay  $\eta_i[k]$ . The correction input  $\tilde{u}_i[k]$  is applied to the local clock at the time  $t_{k_j} + \kappa_j[k] + \eta_i[k]$

$$P_i[k]^+ = P_i^{t_{k_j} + \kappa_j[k] + \eta_i[k]} + \tilde{u}_i[k] \quad (13)$$

where  $P_i[k]^+$  is the  $i$ th clock's state after it is corrected. The processing delay  $\eta_i[k]$  is also considered as a Gaussian random process with the nonzero mean  $\bar{\eta}_i$  and the finite variance of  $\sigma_{\eta_i}^2$ , which is  $\eta_i[k] \sim (\bar{\eta}_i, \sigma_{\eta_i}^2)$  [6].

From (2) and (4), it can be seen that the presence of clock offset leads to the clock state's inaccuracy, and thus the clock state adjustment (13) is equivalent to the correction of clock offset, following:

$$\theta_i[k]^+ = \theta_i^{t_{k_j} + \kappa_j[k] + \eta_i[k]} + \tilde{u}_i[k] \quad (14)$$

where  $\theta_i[k]^+$  is the clock offset after it is adjusted.

*Remark 1:* In the experiments,  $(\Delta d_{ij} - \hat{\theta}_j[k] + \bar{\kappa}_j + \mu_i)$  in (11) is obtained from

$$\begin{aligned} \Delta d_{ij} - \hat{\theta}_j[k] + \bar{\kappa}_j + \mu_i \\ = \begin{cases} \hat{P}_i[k] - \bar{\kappa}_j - \Delta d_{ij} - \mu_i & \text{if } \hat{P}_i[k] - \bar{\kappa}_j - \Delta d_{ij} < \frac{\varphi_i[k]}{2} \\ \hat{P}_i[k] - \bar{\kappa}_j - \Delta d_{ij} - \varphi_i[k] - \mu_i & \text{if } \hat{P}_i[k] - \bar{\kappa}_j - \Delta d_{ij} \geq \frac{\varphi_i[k]}{2} \end{cases} \end{aligned} \quad (15)$$

*Remark 2:* For an arbitrary graph, we have  $\mathcal{L}\mathbf{1} = \mathbf{0}$ , where  $\mathbf{1}$  and  $\mathbf{0}$  are  $(N + 1) \times 1$  vectors. Thus, the following equation holds:

$$\sum_{j=0}^N l_{ij} \hat{\theta}_j[k] = \sum_{j=0, j \neq i}^N l_{ij} (\hat{\theta}_j[k] - \hat{\theta}_i[k]). \quad (16)$$

This shows that the employment of clock offset differences between connected nodes is equivalent to the application of offsets from preceding hop nodes. For simplicity of explanation, we keep the notation in (11).

### B. Theoretical Performance Analysis

In the following, the theoretical performance of the proposed clock correction method (11) is studied in a blast wave monitoring network. This begins with constructing a closed-loop synchronization system for node  $i$ . Then, a single closed-loop system is extended to the networked system. Finally, the stability and convergence performance of the proposed synchronization method in the network is analyzed.

Based on [21], the value of clock state increment within the delay duration  $\kappa_i[k]$  is the summation of the value of delay

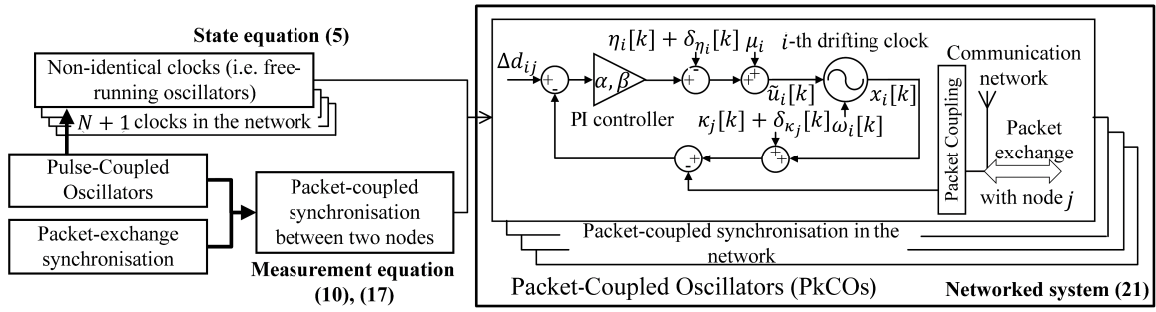


Fig. 4. Framework of PI-based PkCOs.

$\kappa_i[k]$  and the extra value  $\delta_{\kappa_i}[k]$  dependent on the length of packet exchange delay. Thus, the measurement equation (10) [see Fig. 4] is modified as

$$\hat{\theta}_i[k] = \theta_i[k] + (\kappa_j[k] + \delta_{\kappa_j}[k]). \quad (17)$$

Likewise, (14) is rewritten as

$$\theta_i[k]^+ = \theta_i[k]^- + (\tilde{u}_i[k] - (\eta_i[k] + \delta_{\eta_i}[k])) \quad (18)$$

which means that the extra value of  $\eta_i[k] + \delta_{\eta_i}[k]$  is unintentionally employed to correct the local clock.

Hence, the proportional-integral closed-loop synchronization system (see Fig. 4) for the *i*th node satisfies the following form:

$$\begin{cases} \theta_i[k+1] = \theta_i[k] + \tilde{u}_i[k] + \omega_i[k] \\ \hat{\theta}_i[k] = \theta_i[k] + (\kappa_i[k] + \delta_{\kappa_i}[k]) \\ w_i[k+1] = w_i[k] + \beta \sum_{j=0}^N l_{ij} (\Delta d_{ij} - \hat{\theta}_j[k] + \bar{\kappa}_j + \mu_i) \\ \tilde{u}_i[k] = w_i[k] + \alpha \sum_{j=0}^N l_{ij} (\Delta d_{ij} - \hat{\theta}_j[k] + \bar{\kappa}_j + \mu_i) - (\eta_i[k] + \delta_{\eta_i}[k]). \end{cases} \quad (19)$$

Letting  $x_i[k] = [\theta_i[k], w_i[k]]^T$ ,  $y_i[k] = \hat{\theta}_i[k]$ ,  $u_i[k] = [u_{\theta_i}[k], u_{w_i}[k]]^T$ ,  $o_i[k] = [\omega_i[k], 0]^T$ ,  $F_i[k] = -[\eta_i[k] + \delta_{\eta_i}[k], 0]^T$ , and  $v_i[k] = (\kappa_i[k] + \delta_{\kappa_i}[k]) - \bar{\kappa}_i - \mu_i$ , system (19) is rewritten in the matrix-vector form

$$\begin{cases} x_i[k+1] = Ax_i[k] + Bu_i[k] + o_i[k] \\ y_i[k] = Cx_i[k] + v_i[k] \\ u_i[k] = K \sum_{j=0}^N l_{ij} (\Delta d_{ij} - y_j[k]) + F_i[k] \end{cases} \quad (20)$$

where  $A$ ,  $B$ ,  $C$ , and  $K$  are equal to

$$A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}, B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, C = [1 \quad 0], K = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}.$$

By defining  $\mathcal{X}[k] = [x_0^T[k], x_1^T[k], \dots, x_N^T[k]]^T$ ,  $\mathcal{Y}[k] = [y_0[k], y_1[k], \dots, y_N[k]]^T$ ,  $\mathcal{U}[k] = [u_0^T[k], u_1^T[k], \dots, u_N^T[k]]^T$ ,  $\mathcal{O}[k] = [o_0^T[k], o_1^T[k], \dots, o_N^T[k]]^T$ ,  $\mathcal{F}[k] = [F_0^T[k], F_1^T[k], \dots, F_N^T[k]]^T$ ,  $\mathcal{V}[k] = [v_0[k], v_1[k], \dots, v_N[k]]^T$ , and  $\mathbf{d} = [d_0, d_1, \dots, d_N]^T$ , a blast wave monitoring network system is obtained

$$\begin{cases} \mathcal{X}[k+1] = (\mathcal{I} \otimes A)\mathcal{X}[k] + (\mathcal{I} \otimes B)\mathcal{U}[k] + \mathcal{O}[k] \\ \mathcal{Y}[k] = (\mathcal{I} \otimes C)\mathcal{X}[k] + \mathcal{V}[k] \\ \mathcal{U}[k] = (\mathcal{I} \otimes K)\mathcal{L}(-\mathbf{d} - \mathcal{Y}[k]) + \mathcal{F}[k] \end{cases} \quad (21)$$

where  $\otimes$  is the Kronecker product, and  $\mathcal{I}$  is the  $(N+1) \times (N+1)$  identity matrix.

**Theorem 1:** Given a blast wave monitoring network denoted by  $\mathcal{G}$ , consisting of an ideal master clock and  $N$  sensor node clocks with nonidentical frequencies  $f_i \in \{f_i : f_i \neq f_0 \text{ and } i \in \mathcal{N}\}$ , and a PI controller gain  $K$ , if all the eigenvalues of the following matrix:

$$\begin{bmatrix} A - BK(l_{11} - l_{01})C & \dots & BK(l_{1N} - l_{0N})C \\ \vdots & \ddots & \vdots \\ BK(l_{N1} - l_{01})C & \dots & A - BK(l_{NN} - l_{0N})C \end{bmatrix} \quad (22)$$

are inside the unit circle, then the networked system is stable. This means that, at steady synchronization state, the *i*th sensor node clock offset  $\theta_i[k]$  approaches to  $-d_i$ , and the order parameter is close to 1. That is

$$\lim_{k \rightarrow \infty} r[k] = 1. \quad (23)$$

**Proof:** For simplifying the analysis of a networked system (21), by letting  $\tilde{x}_i[k] = x_i[k] - x_0[k]$ ,  $\tilde{y}_i[k] = y_i[k] - y_0[k]$ ,  $\tilde{u}_i[k] = u_i[k] - u_0[k]$ ,  $\tilde{o}_i[k] = o_i[k] - o_0[k]$ ,  $\tilde{F}_i[k] = F_i[k] - F_0[k]$ ,  $\tilde{v}_i[k] = v_i[k] - v_0[k]$ , and  $\tilde{d}_i = d_i - d_0$  [24], [25], a reduced system is given

$$\begin{cases} \tilde{\mathcal{X}}[k+1] = (\tilde{\mathcal{I}} \otimes A)\tilde{\mathcal{X}}[k] + (\tilde{\mathcal{I}} \otimes B)\tilde{\mathcal{U}}[k] + \tilde{\mathcal{O}}[k] \\ \tilde{\mathcal{Y}}[k] = (\tilde{\mathcal{I}} \otimes C)\tilde{\mathcal{X}}[k] + \tilde{\mathcal{V}}[k] \\ \tilde{\mathcal{U}}[k] = (\tilde{\mathcal{I}} \otimes K)\tilde{\mathcal{L}}(-\tilde{\mathbf{d}} - \tilde{\mathcal{Y}}[k]) + \tilde{\mathcal{F}}[k] \end{cases} \quad (24)$$

where  $\tilde{\mathcal{X}}[k] = [\tilde{x}_1^T[k], \tilde{x}_2^T[k], \dots, \tilde{x}_N^T[k]]^T$ ,  $\tilde{\mathcal{Y}}[k] = [\tilde{y}_1[k], \tilde{y}_2[k], \dots, \tilde{y}_N[k]]^T$ ,  $\tilde{\mathcal{U}}[k] = [\tilde{u}_1^T[k], \tilde{u}_2^T[k], \dots, \tilde{u}_N^T[k]]^T$ ,  $\tilde{\mathcal{O}}[k] = [\tilde{o}_1^T[k], \tilde{o}_2^T[k], \dots, \tilde{o}_N^T[k]]^T$ ,  $\tilde{\mathcal{F}}[k] = [\tilde{F}_1^T[k], \tilde{F}_2^T[k], \dots, \tilde{F}_N^T[k]]^T$ ,  $\tilde{\mathcal{V}}[k] = [\tilde{v}_1[k], \tilde{v}_2[k], \dots, \tilde{v}_N[k]]^T$ ,  $\tilde{\mathbf{d}} = [\tilde{d}_1, \tilde{d}_2, \dots, \tilde{d}_N]^T$ ,  $\tilde{\mathcal{I}}$  is the  $N \times N$  identity matrix, and  $\tilde{\mathcal{L}}$  is equal to

$$\begin{bmatrix} l_{11} - l_{01} & l_{12} - l_{02} & \dots & l_{1N} - l_{0N} \\ l_{21} - l_{01} & l_{22} - l_{02} & \dots & l_{2N} - l_{0N} \\ \vdots & \vdots & \ddots & \vdots \\ l_{N1} - l_{01} & l_{N2} - l_{02} & \dots & l_{NN} - l_{0N} \end{bmatrix}.$$

The reduced network system (24) can be rewritten in the following form:

$$\begin{aligned} \tilde{\mathcal{X}}[k+1] &= ((\tilde{\mathcal{I}} \otimes A) - (\tilde{\mathcal{I}} \otimes B)(\tilde{\mathcal{I}} \otimes K)\tilde{\mathcal{L}}(\tilde{\mathcal{I}} \otimes C))\tilde{\mathcal{X}}[k] \\ &\quad - (\tilde{\mathcal{I}} \otimes B)(\tilde{\mathcal{I}} \otimes K)\tilde{\mathcal{L}}\tilde{\mathbf{d}} + (\tilde{\mathcal{I}} \otimes B)\tilde{\mathcal{F}}[k] \\ &\quad + (\tilde{\mathcal{I}} \otimes B)(\tilde{\mathcal{I}} \otimes K)\tilde{\mathcal{L}}\tilde{\mathcal{V}}[k] + \tilde{\mathcal{O}}[k]. \end{aligned} \quad (25)$$

Using the  $z$ -transformation, it is easy to find that the characteristic polynomial of this system is

$$\begin{vmatrix} zI - A + BK(l_{11} - l_{01})C & \dots & -BK(l_{1N} - l_{0N})C \\ \vdots & \ddots & \vdots \\ -BK(l_{N1} - l_{01})C & \dots & zI - A + BK(l_{NN} - l_{0N})C \end{vmatrix}. \quad (26)$$

Suppose (22) holds, all the roots of the above polynomial are in the unit circle. Thus, the blast wave monitoring network system  $\mathcal{G}$  is stable.

The  $z$ -transformation of (25) is given by

$$\begin{aligned} \bar{\mathcal{X}}[z] = & -(z\bar{\mathcal{I}} - ((\bar{\mathcal{I}} \otimes A) - (\bar{\mathcal{I}} \otimes B)(\bar{\mathcal{I}} \otimes K)\bar{\mathcal{L}}(\bar{\mathcal{I}} \otimes C)))^{-1} \\ & \times \frac{z(\bar{\mathcal{I}} \otimes B)(\bar{\mathcal{I}} \otimes K)\bar{\mathcal{L}}\bar{\mathbf{d}}}{z - 1} \\ & + (z\bar{\mathcal{I}} - ((\bar{\mathcal{I}} \otimes A) - (\bar{\mathcal{I}} \otimes B)(\bar{\mathcal{I}} \otimes K)\bar{\mathcal{L}}(\bar{\mathcal{I}} \otimes C)))^{-1} \\ & \times \frac{z(\bar{\mathcal{I}} \otimes B)\bar{\mathcal{F}}[z]}{z - 1} \\ & + (z\bar{\mathcal{I}} - ((\bar{\mathcal{I}} \otimes A) - (\bar{\mathcal{I}} \otimes B)(\bar{\mathcal{I}} \otimes K)\bar{\mathcal{L}}(\bar{\mathcal{I}} \otimes C)))^{-1} \\ & \times ((\bar{\mathcal{I}} \otimes B)(\bar{\mathcal{I}} \otimes K)\bar{\mathcal{L}}\bar{\mathbf{v}}[z] + \bar{\mathcal{O}}[z]). \end{aligned} \quad (27)$$

Let  $E(\bar{\mathcal{X}}[k])$  denote the expected value of  $\bar{\mathcal{X}}[k]$  at steady synchronized state. That is,  $E(\bar{\mathcal{X}}[k]) = \lim_{k \rightarrow \infty} 1/k \sum_{q=k}^{2k} \bar{\mathcal{X}}[q]$ . Based on the generalized final value theorem [26], [27], we find that, at steady state,  $E(\bar{\mathcal{X}}[k])$  approaches to a certain value satisfying

$$\begin{aligned} E(\bar{\mathcal{X}}[k]) = & \lim_{z \rightarrow 1} (z - 1)\bar{\mathcal{X}}[z] \\ = & -(\bar{\mathcal{I}} - ((\bar{\mathcal{I}} \otimes A) - (\bar{\mathcal{I}} \otimes B)(\bar{\mathcal{I}} \otimes K)\bar{\mathcal{L}}(\bar{\mathcal{I}} \otimes C)))^{-1} \\ & \times (\bar{\mathcal{I}} \otimes B)(\bar{\mathcal{I}} \otimes K)\bar{\mathcal{L}}\bar{\mathbf{d}} \\ & + (\bar{\mathcal{I}} - ((\bar{\mathcal{I}} \otimes A) - (\bar{\mathcal{I}} \otimes B)(\bar{\mathcal{I}} \otimes K)\bar{\mathcal{L}}(\bar{\mathcal{I}} \otimes C)))^{-1} \\ & \times (\bar{\mathcal{I}} \otimes B)\bar{\mathcal{F}}[z] \\ = & - \begin{bmatrix} d_1 - d_0 \\ \bar{\eta}_1 + \bar{\delta}_{\eta_1} \\ \vdots \\ d_N - d_0 \\ \bar{\eta}_N + \bar{\delta}_{\eta_N} \end{bmatrix} \end{aligned} \quad (28)$$

where  $\bar{\eta}_i + \bar{\delta}_{\eta_i}$  is the mean value of processing delay.

Thus, it can be seen that by using the proposed controlling strategy (11), the effects of processing delay can be eliminated, and the expected value the clock offset (i.e.,  $E(\theta_i[k]) = \lim_{k \rightarrow \infty} 1/k \sum_{q=k}^{2k} \theta_i[q]$ ) approaches to  $-d_i$  at synchronized state. That is

$$E(\theta_i[k]) = -d_i. \quad (29)$$

Finally, at steady synchronized state, the order parameter approaches to

$$\lim_{k \rightarrow \infty} r[k]e^{j\psi[k]} = \frac{1}{N+1} \sum_{i=0}^N e^{j(E(\theta_i[k]) + d_i)} = 1. \quad (30)$$

This means that  $r[\infty] = 1$  and  $\psi[\infty] = 0$ . At synchronized state, all clocks (i.e., oscillators) fire at the specified time  $d_i$ ,

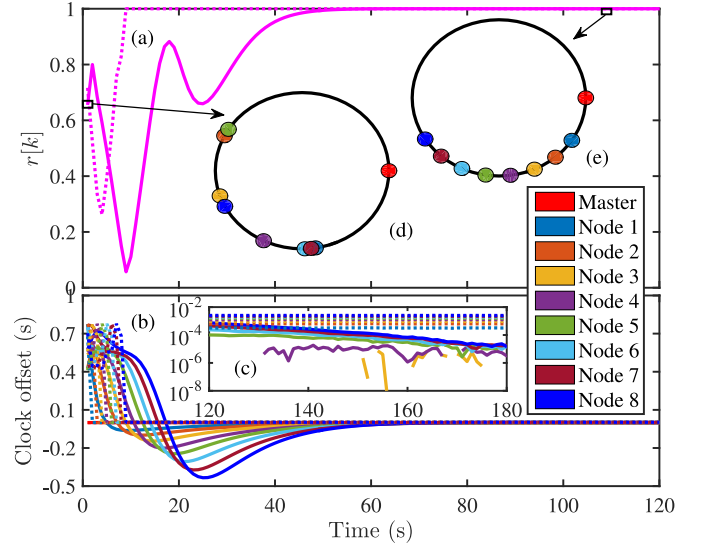


Fig. 5. Evolution of (a) the order parameter  $r[k]$  and (b), (c) the clock offset in the linear network (solid line: PI-based PkCOs, dotted line: PISync). PkCOs on the unit circle at (d) initial state, and (e) anti-phase synchronized state via the PI-based PkCOs protocol.

and all nodes transmit *Sync* packets at the allocated time slot  $d_i$  of each synchronization cycle. ■

**Remark 3:** Theorem 1 presents the results for the PI controller design in the blast wave monitoring network  $\mathcal{G}$ . It is necessary to mention that the proposed design condition (22) can be effectively solved via several optimization solutions (e.g., *fminsearch*).

#### IV. SIMULATION RESULTS

For the purpose of validating the theoretical results presented in the preceding section, we conduct simulations in a directed eight-hop linear network (consisting of one master and eight sensor nodes). In the simulations, the offset  $\theta_i[0]$  and skew  $\gamma_i$  of a drifting clock are chosen randomly following the uniform distribution in the corresponding interval (0.4 s, 0.8 s) and (0 ppm, 10 ppm). The clock offset is subject to a random perturbation with the standard deviation  $\sigma_{\theta_i} = 1 \mu\text{s}$  [22]. The synchronization cycle is configured to 1 s. The mean values of packet exchange delay and processing delay are, respectively, set to 513.873 and 311.475  $\mu\text{s}$ ; and the standard deviations of their noises are  $\sigma_{\kappa_i} = 0.296 \mu\text{s}$  and  $\sigma_{\eta_i} = 3.899 \mu\text{s}$ , respectively, [21]. The PISync protocol is chosen for performance evaluation.

Fig. 5 shows the evolution of the order parameter and clock offset under the PI-based PkCOs and PISync protocols. Both two methods let the order parameter  $r[k]$  grow from the initial value to 1, and the clock offset approach to zero. This means that two approaches guarantee all sensor node clocks synchronize with the master clock (i.e., the synchronized state is reached). In the PISync protocol, since the complete offset estimate is used for clock correction ( $\alpha = 1$ ), its synchronization speed is faster than that of PI-based PkCOs [see Fig. 5(a) and (b)]. However, the PI-based PkCOs solution achieves better performance [see Fig. 5(c)], due to the utilization of an



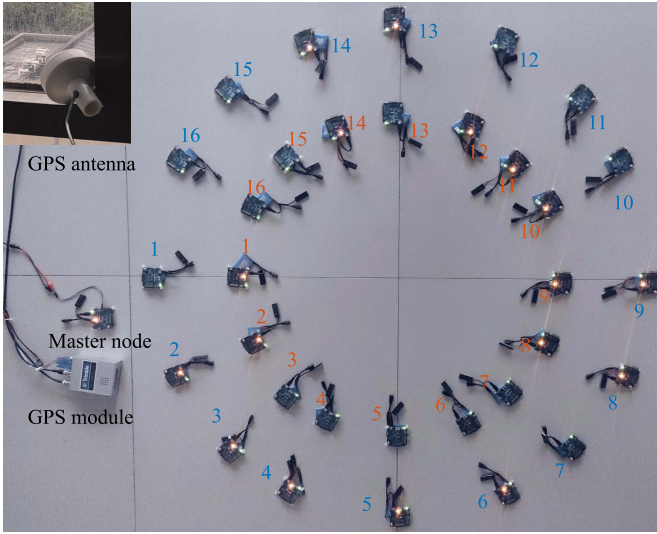


Fig. 6. Hardware testbed of a circular network by using Atmel SAM R21 wireless boards [28].

integral controller. In addition, at steady synchronization state, the PkCOs protocol allocates a time slot for each node to send the *Sync* packet [see Fig. 5(e)].

## V. EXPERIMENTAL EVALUATION

This section presents the performance evaluation of the PI-based PkCOs protocol in two blast wave monitoring networks of Fig. 2 (also see Fig 6). In the implementation, the clock state is represented by a 32-bit counter register of the RTC module, which uses a 32.768 kHz crystal oscillator as the clock source. The RTC's threshold register is configured to 32767 to let the clock reset each second. Once counter matches the threshold register, it is reset to zero. Meanwhile, the processor triggers a hardware interrupt, where the *Sync* packet is directly sent to the wireless channel. The parameters  $t_{dp}$  and  $t_{sd}$  are 9.15 ms and 3.66 ms,<sup>3</sup> respectively, for demonstration purposes. Based on Theorem 1, the coefficients of PI controller are  $\alpha = 0.5$  and  $\beta = 0.025$ , respectively, by using *fminsearch* in MATLAB.

Whenever the addressing fields of the received *Sync* packet match the local addresses, the hardware interrupt is issued to generate a timestamp via reading the counter register. In the interrupt, the processor computes the correction value  $\tilde{u}_i[k]$  according to (11), and then  $\tilde{u}_i[k]$  is employed to the counter register for clock correction. The GPS Clock [29], providing the pulse per second (PPS) signal, is connected to the master node. Once the master receives the PPS signal, it issues a hardware interrupt for transmitting the *Sync* packet. The logic analyzer [30] is adopted to collect the data for analysis.

In addition to the macroscopic metric order parameter  $r[k]$ , this work also uses root synchronization precision, local synchronization precision, and global synchronization precision to microscopically measure achieved performance. The root precision is defined as the time difference between the sensor node clock and master clock. The local synchronization

<sup>3</sup>The minimum value of  $t_{sd}$  is 1 ms, as the packet exchange delay is around 500  $\mu$ s in this work [21].

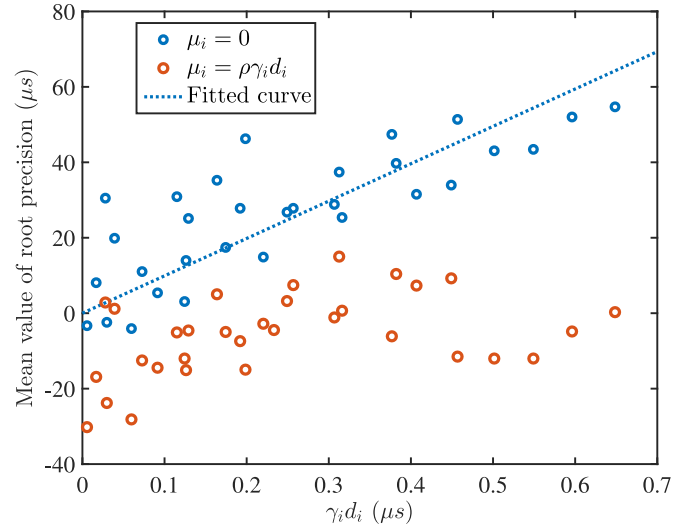


Fig. 7. Mean values of the root synchronization precision in the circular network by setting  $\mu_i$  to zero, and to the value obtained from (12).

precision is the clock difference between two adjacent nodes, and global synchronization precision is the difference of the local clock between two arbitrary nodes in a network. The RBS, PCO, and PISync protocols are selected for performance comparison.

Fig. 7 presents the root precision's mean value in the circular network under PI-based PkCOs. If the term  $\mu_i$  of (11) is set to zero (i.e.,  $\mu_i = 0$ ), the mean value increases linearly (see blue circles), and it is also dependent on the clock skew  $\gamma_i$  and anti-phase synchronization duration  $d_i$  [see (12)]. The linear fitting indicates that  $\rho$  is 99.07 in the circular network. By employing  $\mu_i$ , which is calculated based on (12), to the correction input  $\tilde{u}_i[k]$ , it can be seen that the mean value of root precision is around zero (see orange circles).  $\rho$  is 90.09 for the triangle network. In the following experiments,  $\mu_i$  is set to  $\rho\gamma_id_i$  to further improve PI-based PkCOs performance.

Fig. 8 shows the synchronization performance of four protocols in the circular network of Fig. 2 (also see Fig. 6 for a hardware testbed). Due to the intrinsic characteristic of PCO and PISync, both fail to remove the impacts of packet exchange or processing delays, and the achieved performance (i.e., average local precision, average global precision, and root precision) is much worse than the PkCOs accuracy. From Fig. 8(a), it can be seen that the maximal local and global precisions of PI-based PkCOs are slightly worse than the other three protocols' performance. This results from the saturation effects on the PI controller and can be resolved by resetting the integral controller at steady state. Nevertheless, the PI-based PkCOs protocol is capable of achieving much better performance (of around 50  $\mu$ s), in terms of the root precision [see Fig. 8(b)].

The performance of four synchronization protocols in a triangle network is presented in Fig. 9. Clearly, PI-based PkCO outperforms the other three algorithms (i.e., RBS, PCO, and PISync). Overall, the PI-based PkCOs protocol provides an alternative time synchronization solution for the blast wave monitoring network.



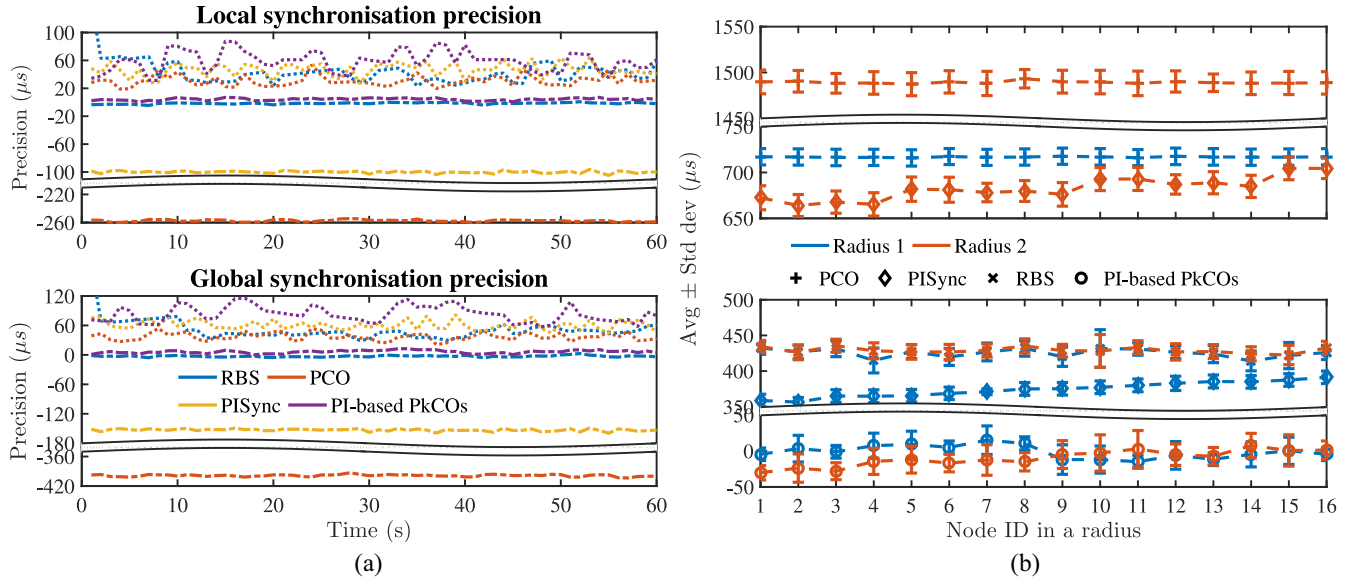


Fig. 8. (a) Evolution of the local and global precisions [dotted line: instantaneous maximal local/global precision, dash-dot line: instantaneous average local/global precision] and (b) mean values and standard deviations of the root precision in the circular network.

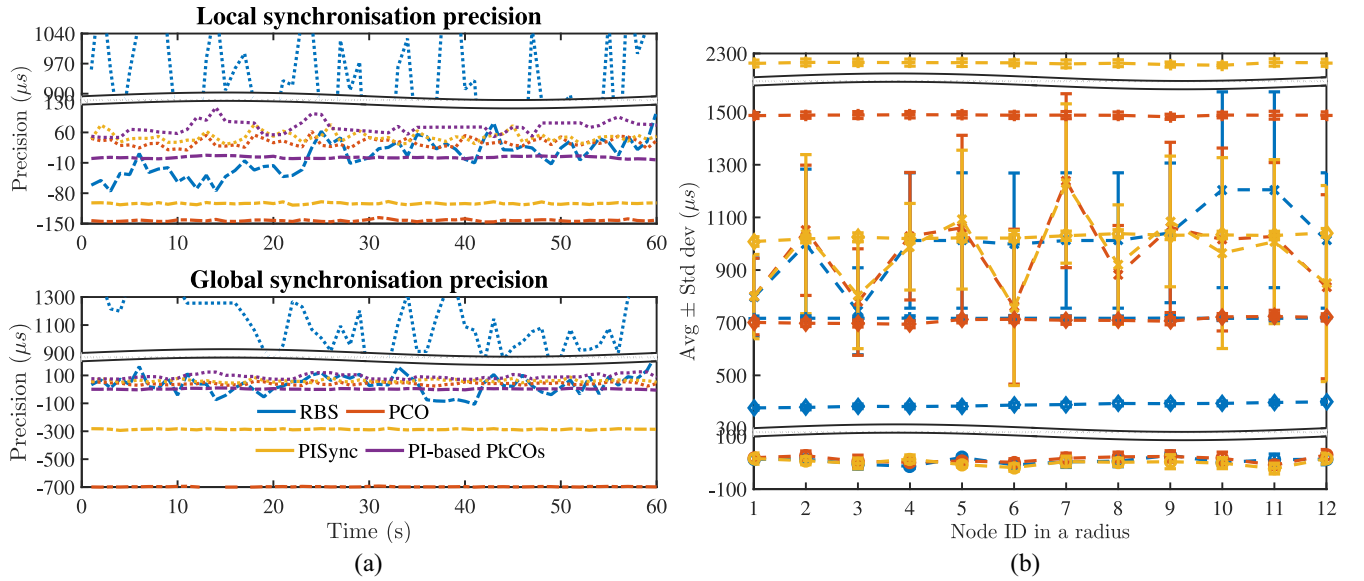


Fig. 9. (a) Evolution of the local and global precisions [dotted line: instantaneous maximal local/global precision, dash-dot line: instantaneous average local/global precision] and (b) mean values and standard deviations of the root precision in the triangle network [color is the same as that in Fig. 2(b), and markers are the same as that of Fig. 8(b)].

## VI. CONCLUSION

In this article, we adopted a PI-based PkCOs protocol to synchronize drifting clocks in blast wave monitoring networks. To avoid the occurrence of *Sync* packet concurrent transmission at synchronized state, the packet-coupled synchronization algorithm uses a scheduling scheme to reduce the possibility of packet collisions. The proposed algorithm also possesses the feature of allowing the hybrid transmission of the *Sync* packet and the blast wave monitoring data on the same wireless channel. A compensation strategy based on the PI controller was proposed to eliminate for the impacts subject to the drifting clock frequency and anti-phase synchronization duration. We also studied the theoretical performance of the PkCOs synchronization scheme in the network. The experimental results showed

that the PI-based PkCOs protocol provides an alternative clock synchronization solution for the blast wave monitoring network.

## REFERENCES

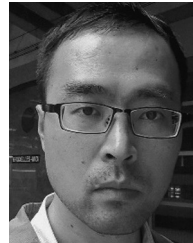
- [1] A. Azar *et al.*, "An optical fiber transducer for measuring kinetics of skull-brain interaction in a surrogate model of the human head subjected to blast overpressure," *IEEE Sensor J.*, vol. 19, no. 2, pp. 548–559, Jan. 2019.
- [2] S. Gao *et al.*, "A lightweight wireless overpressure node based efficient monitoring for shock waves," *IEEE/ASME Trans. Mechatronics*, vol. 26, no. 1, pp. 448–457, Feb. 2021.
- [3] S. Gao *et al.*, "A B-spline method with AIS optimization for 2-D IoT-based overpressure reconstruction," *IEEE Internet Things J.*, vol. 7, no. 3, pp. 2005–2013, Mar. 2020.
- [4] R. E. Mirollo and S. H. Strogatz, "Synchronization of pulse-coupled biological oscillators," *SIAM J. Appl. Math.*, vol. 50, no. 6, pp. 1645–1662, 1990.

- [5] R. Pagliari, Y.-W. P. Hong, and A. Scaglione, "Bio-inspired algorithms for decentralized round-robin and proportional fair scheduling," *IEEE J. Sel. Areas Commun.*, vol. 28, no. 4, pp. 564–575, May 2010.
- [6] Y. Zong, X. Dai, P. Canyelles-Pericas, K. Busawon, R. Binns, and Z. Gao, "Modelling and synchronisation of delayed packet-coupled oscillators in industrial wireless sensor networks," in *Proc. 21st IFAC World Congr. (Virtual)*, 2020, pp. 8381–8387.
- [7] I. Bojic and K. Nymoen, "Survey on synchronization mechanisms in machine-to-machine systems," *Eng. Appl. Artif. Intell.*, vol. 45, pp. 361–375, Oct. 2015.
- [8] L. Qiu, B. Liu, S. Yuan, and Z. Su, "Impact imaging of aircraft composite structure based on a model-independent spatial-wavenumber filter," *Ultrasonics*, vol. 64, pp. 10–24, Jan. 2016.
- [9] J. Cai, X. Wang, and Z. Zhou, "A signal domain transform method for spatial resolution improvement of lamb wave signals with synthetically measured relative wavenumber curves," *Struct. Health Monitor.*, vol. 18, nos. 5–6, pp. 1633–1651, 2019.
- [10] F. Nunez, Y. Wang, D. Grasing, S. Desai, G. Cakiades, and F. J. Doyle, III, "Pulse-coupled time synchronization for distributed acoustic event detection using wireless sensor networks," *Control Eng. Pract.*, vol. 60, pp. 106–117, Mar. 2017.
- [11] *IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems*, IEEE Standard 1588-2008, 2008.
- [12] J. Elson, L. Girod, and D. Estrin, "Fine-grained network time synchronization using reference broadcasts," in *Proc. 5th Symp. Oper. Syst. Design Implement.*, 2002, pp. 147–163.
- [13] K. S. Yildirim, R. Carli, and L. Schenato, "Adaptive proportional—Integral clock synchronization in wireless sensor networks," *IEEE Trans. Control Syst. Technol.*, vol. 26, no. 2, pp. 610–623, Mar. 2018.
- [14] F. Shi, X. Tuo, S. X. Yang, J. Lu, and H. Li, "Rapid-flooding time synchronization for large-scale wireless sensor networks," *IEEE Trans. Ind. Informat.*, vol. 16, no. 3, pp. 1581–1590, Mar. 2020.
- [15] T. Qiu, Y. Zhang, D. Qiao, X. Zhang, M. L. Wymore, and A. K. Sangaiah, "A robust time synchronization scheme for Industrial Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 14, no. 8, pp. 3570–3580, Aug. 2018.
- [16] W. Masood, J. F. Schmidt, G. Brandner, and C. Bettstetter, "DISTY: Dynamic stochastic time synchronization for wireless sensor networks," *IEEE Trans. Ind. Informat.*, vol. 13, no. 3, pp. 1421–1429, Jun. 2017.
- [17] G. Werner-Allen, G. Tewari, A. Patel, M. Welsh, and R. Nagpal, "Firefly-inspired sensor network synchronicity with realistic radio effects," in *Proc. 3rd Int. Conf. Embedded Netw. Sens. Syst.*, 2005, pp. 142–153.
- [18] G. V. Joseph and V. Pakrashi, "Limits on anti-phase synchronization in oscillator networks," *Sci. Rep.*, vol. 10, pp. 1–9, Jun. 2020.
- [19] J. Degesys, I. Rose, A. Patel, and R. Nagpal, "DESYNC: Self-organizing desynchronization and TDMA on wireless sensor networks," in *Proc. 6th Int. Symp. Inf. Process.*, Cambridge, MA, USA, 2007, pp. 11–20.
- [20] R. Gentz, A. Scaglione, L. Ferrari, and Y.-W. P. Hong, "PulseSS: A pulse-coupled synchronization and scheduling protocol for clustered wireless sensor networks," *IEEE Internet Things J.*, vol. 3, no. 6, pp. 1222–1234, Dec. 2016.
- [21] Y. Zong, X. Dai, and Z. Gao, "Proportional—Integral synchronization for nonidentical wireless packet-coupled oscillators with delays," *IEEE Trans. Ind. Electron.*, vol. 68, no. 11, pp. 11598–11608, Nov. 2021.
- [22] G. Giorgi and C. Narduzzi, "Performance analysis of Kalman-filter-based clock synchronization in IEEE 1588 networks," *IEEE Trans. Instrum. Meas.*, vol. 60, no. 8, pp. 2902–2909, Aug. 2011.
- [23] Y. Zong, X. Dai, Z. Gao, K. Busawon, and J. Zhu, "Modelling and synchronization of pulse-coupled non-identical oscillators for wireless sensor networks," in *Proc. 16th Int. Conf. Ind. Informat. (INDIN)*, Porto, Portugal, 2018, pp. 101–107.
- [24] D. Xie and S. Wang, "Consensus of second-order discrete-time multi-agent systems with fixed topology," *J. Math. Anal. Appl.*, vol. 387, no. 1, pp. 8–16, Mar. 2012.
- [25] Y. Zhang and Y.-P. Tian, "Consentability and protocol design of multi-agent systems with stochastic switching topology," *Automatica*, vol. 45, no. 5, pp. 1195–1201, May 2009.
- [26] E. Gluskin, "Let us teach this generalization of the final-value theorem," *Eur. J. Phys.*, vol. 24, pp. 591–597, Sep. 2003.
- [27] E. Gluskin and J. Walrave, "On two generalisations of the final value theorem: Scientific relevance, first applications, and physical foundations," *Int. J. Syst. Sci.*, vol. 42, no. 12, pp. 2045–2055, Sep. 2010.
- [28] "Atmel SAM R21 Xplained Pro Evaluation Kit." [Online]. Available: <https://www.microchip.com/en-us/development-tool/ATSAMR21-XPRO> (Accessed: Nov 13, 2021).
- [29] "Thunderbolt A E GPS Disciplined Clock." [Online]. Available: <https://timing.trimble.com/products/disciplined-clocks/thunderbolt-a-gps-disciplined-clock/> (Accessed: Nov 13, 2021).
- [30] "Kingst LA5016 LA5016 Logic Analyser." [Online]. Available: <http://www.qdkingst.com/en/products> (Accessed: Nov 13, 2021).



**Yan Zong** (Member, IEEE) received the B.Eng. degree in electrical and electronic engineering from Nanjing Normal University, Nanjing, China, in 2016, and the Ph.D. degree in electrical engineering from Northumbria University, Newcastle upon Tyne, U.K., in 2020.

He is currently with Cranfield University, Cranfield, U.K., and also with the State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, Shenyang, China. Prior to that, he was an Embedded Software Engineer with Keiky Ltd., Earley, U.K. His research interests cut across several disciplines, which include machine learning, networked control system, and clock synchronization, and their applications to the Industrial Internet of Things.



**Xuewu Dai** (Member, IEEE) received the B.Eng. degree in electronic engineering and the M.Sc. degree in computer science from Southwest University, Chongqing, China, in 1999 and 2003, respectively, and the Ph.D. degree from the University of Manchester, Manchester, U.K., in 2008.

He is currently with the State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, Shenyang, China. From 2009 to 2012, he held research positions with the University of College, London, U.K., and the University of Oxford, Oxford, U.K. His main research interests include robust adaptive signal processing, state estimation, intelligent transportation systems, automatic train control, and rescheduling.



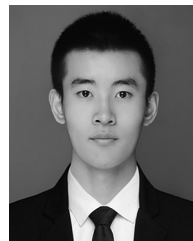
**Shang Gao** (Member, IEEE) received the bachelor's, master's, and Doctoral degrees from the School of Measuring and Testing Technology and Instrument Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2007, 2010, and 2017, respectively.

He is currently an Assistant Professor with the School of Mechanical Engineering, Nanjing University of Science and Technology, Nanjing. He has also paid a visit to Newcastle University, Newcastle upon Tyne, U.K., for the academic research on wireless piezoelectric network from 2015 to 2016, and 2019. His main research areas are industrial wireless sensor network, structural health monitoring, smart sensor, and measurement system development.



**Pep Canyelles-Pericas** (Member, IEEE) received the double B.Eng. degrees in electrical and electronic engineering from the Technical University of Catalonia, Barcelona, Spain, and the University of Northumbria, Newcastle upon Tyne, U.K., in 2006, the M.Sc. degree in engineering management from the University of Sunderland, Sunderland, U.K., in 2008, and the Ph.D. degree in control systems from the University of Northumbria in 2016.

He is currently a Researcher with the MESA+ Institute of Nanotechnology, University of Twente, Enschede, The Netherlands. From 2009 to 2012, he worked as an Optoelectronics Research Engineer with the University of the Balearic Islands, Palma, Spain. After Ph.D. graduation, he engaged with Innovate U.K. projects in the Knowledge Transfer Partnership and the Innovation to Commercialization of University Research schemes. His research interests are acoustofluidics, optical sensing, instrumentation, and control.



**Shuxin Liu** received the B.Eng. degree in communication engineering from Dalian Jiaotong University, Dalian, China, in 2021. He is currently pursuing the M.Sc. degree in control science and engineering with the State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, Shenyang, China.

His research interests include time synchronization in industrial wireless sensor networks and industrial Internet of Things.