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 utilisation 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 synchronise drifting clocks in a wireless blast wave monitoring network. In order to address packet collisions caused by the concurrent transmission, an anti-phase synchronisation solution is utilised to maintain clock synchronisation, 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 synchronisation performance while the clock frequency is usually assumed to be zero and non-drifting. Thus, a compensation strategy is proposed to eliminate the joint impacts and to improve the synchronisation 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 synchronisation protocol is evaluated on an IEEE 802.15.4 hardware testbed. The experimental results show that the PkCOs algorithm provides an alternative clock synchronisation solution for blast wave monitoring networks.

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 the sensor nodes in a network need to share a common timescale. Thus, the key enabling technology, Time Synchronisation (TS), is utilised to achieve such a goal.

Due to the importance of time synchronisation, apart from the communication engineering research society, it attracts also interest from the mathematics and physics communities. Thus we investigate the topic of synchronisation 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 discretisation properties [6].

In PCO, each oscillator behaves as an uncoupled oscillator running on the unit circle (Fig. 1a), 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 firing information is transmitted) whilst $P$ is reset. This firing-resetting behaviour 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 Pulse simultaneously when the synchronisation is achieved, see Fig. 1b) [7]. Thus, the PCO model needs to be improved, and the 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. 1a. Upon arriving at the right (i.e. \( P \) reaches \( \phi \) ), the oscillator fires and sends a Sync packet for synchronisation 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 synchronisation 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 a popular method. Through processing the ToA data from different sensor nodes, the acoustic source’s position, in respect to the wireless network, can be estimated [10]. Clearly, the performance of these methods are highly dependent on the time synchronisation 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 Synchronisation Protocols:** These algorithms can measure the clock offset through exchanging the periodical timestamped packets among connected nodes. The employment of the estimated offset lets a network achieve clock synchronisation. In the literature, there exist two main solutions for improving synchronisation performance. One approach is to estimate a more accurate clock offset via the frequent packet exchange. For example, Precision Time Protocol (PTP) [11] improves the accuracy by using the two-way message exchange mechanism, and Reference Broadcast Synchronisation (RBS) [12] removes the effects of send and access delays under the receiver-receiver scheme. For recent works, at synchronised state, the transmission of multiple packets (e.g. 100 in [14], more than 32 in [15]) during each synchronisation 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 utilise advanced processing techniques (see Table 1). Even though the use of these technologies can achieve higher performance, the wireless nodes have to face the 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 1, it can be seen that the TS protocols are implemented on different hardware platforms selecting the varying crystal oscillator as the clock source. Since this is a hardware-based solution, it is unfair to directly compare their performance [16]. By measuring the synchronisation precision in clock ticks\(^1\), 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 synchronisation, all the Sync packets from synchronised 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 synchronisation be lost gradually, owing to the frequency difference among node clocks. To avoid packet collisions, the solution of anti-phase synchronisation [18] (see Fig. 1c) is adopted, in order to guarantee all the Sync packets are scheduled to specified time slots for transmission, thereby reducing the probability of concurrent transmission.

In [19], once a network achieves clock synchronisation, \( N+1 \) sensor nodes send the Sync packets (i.e. the control traffic) with interval space \( T/(N+1) \) during each time synchronisation cycle.

![Table 1: Performance comparison in the clock synchronisation protocols](image)

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

\(^1\)This metric relates to the maximum achievable precision on the underlying platform.
sation cycle $T$. This means that the anti-phase synchronisation is achieved in the network, as shown in Fig. 1c. 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 quantisation function is still used, and the resource-constrained sensor nodes have to offer the extra resource to realise this. Hence, we provide a simple anti-phase synchronisation 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 synchronisation performance; while, the assumption of no drifting frequency is usually adopted in the literature (e.g. [19], [20]). We study these two impacts, and propose a compensation strategy to improve the TS precision in the large-scale multi-hop 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 paper utilises 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 Organisation

In this paper, we adopt a PI-based PkCOs protocol to synchronise drifting clocks in blast wave monitoring networks. The packet-coupled synchronisation algorithm utilises 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 the existing clock synchronisation algorithms, and provides an alternative TS solution for blast wave monitoring networks.

The rest of this paper is organised as follows: Section 2 presents the mathematical problem formulation. Section 3 introduces the PI-based packet-coupled synchronisation scheme and the theoretical analysis of the PkCOs algorithm in the blast wave monitoring network. The simulation and experimental results are shown in Section 4 and Section 5, respectively. Finally, Section 6 presents the conclusions.

II. PROBLEM FORMULATION

Let the blast wave monitoring network in Fig. 2 be described by the directed and weighted graph $G = (V, E, A)$, where $V = \{0, 1, ..., N\}$ denotes the set of wireless nodes, and the set of edges $E$ induced by the adjacency matrix $A = [a_{ij}] \in \mathbb{R}^{(N+1) \times (N+1)}$. The network consists of a single root node (i.e. $i = 0$) and a set of regular nodes represented by the set $\mathcal{N} = \{i : i \in V, i \geq 1\}$. 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 $A$, 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 $G$ has no self-loops (i.e. $a_{ii} = 0$ for all $i \in V$). The in-degree of node $i$ is defined as $deg_i = \sum_{j=0}^{N} a_{ij}$. Thus, the Laplacian matrix $L = [l_{ij}] \in \mathbb{R}^{(N+1) \times (N+1)}$ of $G$ is represented by $L = D - A$, where $D = diag(deg_0, deg_1, ..., deg_N)$ is the diagonal degree matrix. Moreover, the $h$-th sensor node on the radius $c$ in Fig. 2 is represented by node $i$ (i.e. $i = (\sum_{q=2}^{N} M + h)$, 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 trade-off between cost and signal accuracy, this work selects the crystal oscillator as a clock source [6]. The non-identical model is utilised to mathematically describe the behaviour of a crystal oscillator-based clock. First, let’s 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 behaviour of clock
resetting, and \( P_0[n] \) at the \( n \)-th event is given as

\[
P_0[n] = t[n] - \sum_{q=1}^{k} \varphi_0[q],
\]

where \( k \) means the total number of clock resets has occurred from \( n = 0 \) to the \( n \)-th clock event, and it also indicates that the clock is at the \( k \)-th synchronisation 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 time synchronisation cycle \( T \) (i.e. \( \varphi_0[k] = T \)). In embedded systems, the clock threshold is much greater than the clock period \( T_0 \); then it is reasonable to assume that the clock updates \( m_0 \) times during a single synchronisation cycle, yielding \( T = m_0 T_0 \). \( k \) 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 modelled as

\[
P_i[n] = t[n] + \sum_{q=0}^{n} \chi_i[q] T_0 + \frac{\phi_i[n]}{2 \pi f_0} - \sum_{q=1}^{k} \varphi_i[q],
\]

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

\[
\theta_i[n] = P_i[n] - P_0[n].
\]

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

\[
\theta_i[n] = \sum_{q=0}^{n} \frac{\chi_i[q] T_0}{f_0} + \frac{\phi_i[n]}{2 \pi f_0} - \sum_{q=1}^{k} \Delta \varphi_i[q],
\]

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

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

\[
\theta_i[k + 1] = \theta_i[k] + \omega_i[k],
\]

where \( \omega_i[n] = (\phi_i[n + 1] - \phi_i[n])/2 \pi f_0 \) and \( \omega_i[k] = \gamma_i T - \Delta \varphi_i[1, 1, \ldots, 1][\varphi_0[km_0], \varphi_0[km_0 + 1], \ldots, \varphi_0[(k + 1) - 1]]^T \) is the clock offset noise, which is considered as a Gaussian random noise process [22]. \( \gamma_i \) is the \( i \)-th clock skew, representing the normalised difference between \( f_i \) and \( f_0 \) (i.e. \( \gamma_i = (f_i - f_0)/f_0 \)).

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

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

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

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 synchronisation duration. This parameter is described in (9). For example, if all the clocks are perfectly synchronised, then \( r[k] = 1 \); and if all the clocks (i.e. oscillators) are scattered around the unit circle, then \( r[k] \approx 0 \). The purpose of synchronisation is to keep the order parameter \( r[k] \) as close to one as possible. That is

\[
\lim_{k \to \infty} r[k] = 1.
\]

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

### III. Proportional-Integral Packet-Coupled Synchronisation Scheme

In the PkCOs protocol, the packet-coupled synchronisation scheme provides a solution for estimating and correcting the \( i \)-th sensor node clock offset. At the \( k \)-th synchronisation cycle, upon the reception of a Sync\(_j\) packet (see Fig. 3a) from the \( j \)-th node after the packet coupling delay \( \kappa_j[k] \), node \( i \) associates to it the timestamp \( P_i[k] \). 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 the correction input \( \hat{u}_i[k] \).

#### A. PI-based Packet-Coupled Synchronisation Mechanism

For the \( k \)-th synchronisation cycle, the \( j \)-th clock fires, and node \( j \) directly sends a Sync\(_j\) packet at \( t_{kj} \). After \( \kappa_j[k] \), node
i receives $\text{Sync}_j$ at $t_{kj} + \kappa_i[k]$. 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:

$$\hat{P}_i[k] = P_i^{t_{kj} + \kappa_i[k]},$$

(8)

where the packet coupling delay $\kappa_i[k]$ is regarded as a Gaussian random process with the nonzero mean of $\bar{\kappa}_i$ and the finite variance of $\sigma_{\kappa_i}^2$. That is $\kappa_i[k] \sim (\bar{\kappa}_i, \sigma_{\kappa_i}^2)$ [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 transmission (see Fig. 3b). Fig. 3c shows the superframe structure, where the data period $t_{sd}$ 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 Figs. 3b and 3c, it can be seen that once all the nodes are scheduled to different time slots for packet transmission, all the clocks (i.e. oscillators) are also anti-phase synchronised.

Different time slots for packet transmission, all the clocks (i.e. oscillators) are also anti-phase synchronised. The anti-phase synchronisation method in the network is analysed.

To address the issue of correcting a varying processing delay $\eta_i[k]$, the correction input $\bar{u}_i[k]$ is applied to the local clock at the time $t_{kj} + \kappa_i[k] + \eta_i[k]$:

$$P_i[k]^{\pm} = P_i^{t_{kj} + \kappa_i[k] + \eta_i[k]} + \bar{u}_i[k],$$

(13)

where $P_i[k]^{\pm}$ 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].

Fig. 2 shows that the occurrence of packet collisions can be avoided by the proposed clock correction method (11) is studied in a blast wave monitoring network, following

$$\theta_i[k]^{\pm} = \theta_i^{t_{kj} + \kappa_i[k] + \eta_i[k]} + \bar{u}_i[k],$$

(14)

where $\theta_i[k]^{\pm}$ 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

$$\Delta d_{ij} - \hat{\theta}_j[k] + \bar{\kappa}_j + \mu_i = \begin{cases} P_i[k] - \bar{\kappa}_j - \Delta d_{ij} - \mu_i, & \text{if } P_i[k] - \bar{\kappa}_j - \Delta d_{ij} < \frac{\Delta d_{ij}}{2} \\
\text{if } P_i[k] - \bar{\kappa}_j - \Delta d_{ij} - \varphi_i[k] - \mu_i, & \text{if } P_i[k] - \bar{\kappa}_j - \Delta d_{ij} \geq \frac{\Delta d_{ij}}{2} \end{cases}$$

(15)

Remark 2. For an arbitrary graph, we have $C_1 = 0$, where 1 and 0 are $(N + 1) \times 1$ vectors. Thus, the following equation holds:

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

(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 synchronisation 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 synchronisation method in the network is analysed.

Based on [21], the value of clock state increment within the delay duration $\kappa_i[k]$ is the summation of the value of delay $\kappa_i[k]$ and the extra value $\delta_i[k]$ dependent on the length of
packet exchange delay. Thus, the measurement equation (10) [see Fig. 4] is modified to
\[ \hat{\theta}_i[k] = \theta_i[k] + (\kappa_j[k] + \delta_{\eta_i}[k]). \]  
(17)

Likewise, (14) is re-written as
\[ \theta_i[k]^+ = \theta_i[k]^- + (\tilde{u}_i[k] - (\eta_i[k] + \delta_{\eta_i}[k])), \]  
(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 synchronisation system (see Fig. 4) for the \( i \)-th node satisfies the following form
\[
\begin{aligned}
\dot{\theta}_i[k] &= \theta_i[k] + \kappa_j[k] + \delta_{\eta_i}[k] \\
\dot{\eta}_i[k] &= \theta_i[k] + \kappa_j[k] + \delta_{\eta_i}[k] \\
\dot{u}_i[k] &= \theta_i[k] + \eta_i[k] + \delta_{\eta_i}[k] - \kappa_j[k] - \mu_i(n) - \delta_{\eta_i}[k] \\
\end{aligned}
\]

which is given:
\[ X[k+1] = [\theta_i[k], w_i[k]]^T, \quad Y_i[k] = \dot{\theta}_i[k], \]
\[ u_i[k] = [u_{\theta_i}[k], u_{\tilde{u}_i}[k]]^T, \quad o_i[k] = [\omega_i[k], 0]^T, \quad F_i[k] = -[\eta_i[k] + \delta_{\eta_i}[k], 0]^T, \nu_i[k] = (\kappa_j[k] + \delta_{\eta_i}[k]) - \kappa_j[k] - \mu_i(n), \]
\[
\begin{aligned}
\dot{x}_i[k] &= A x_i[k] + B u_i[k] + o_i[k] \\
\dot{y}_i[k] &= C x_i[k] + \nu_i[k] \\
\dot{u}_i[k] &= K \sum_{j=0}^{N} l_{ij} (\Delta d_{ij} - \dot{y}_j[k]) + F_i[k] \\
\end{aligned}
\]

where \( A, B, C, K \) are equal to
\[
A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 \end{bmatrix}, \quad K = \begin{bmatrix} \alpha & \beta \end{bmatrix}. \]

Through defining \( \mathcal{X}[k] = [x_0^T[k], x_1^T[k], ..., x_N^T[k]]^T, \mathcal{Y}[k] = [y_0[k], y_1[k], ..., y_N[k]]^T, \mathcal{U}[k] = [u_0^T[k], u_1^T[k], ..., u_N^T[k]]^T, \mathcal{O}[k] = [o_0^T[k], o_1^T[k], ..., o_N^T[k]]^T, \mathcal{F}[k] = [F_0[k], F_1[k], ..., F_N[k]]^T, \mathcal{V}[k] = [v_0[k], v_1[k], ..., v_N[k]]^T, \mathcal{d} = [d_0, d_1, ..., d_N]^T, \mathcal{I} = \text{the } N \times N \text{ identity matrix}, \text{ and } \mathcal{L} \text{ is equal to}
\]
\[
\begin{bmatrix}
1 & 0 & 0 & 0 & ... & 0 \\
0 & 1 & 0 & 0 & ... & 0 \\
0 & 0 & 1 & 0 & ... & 0 \\
0 & 0 & 0 & 1 & ... & 0 \\
... & ... & ... & ... & ... & ... \\
0 & 0 & 0 & 0 & ... & 1
\end{bmatrix}
\]

then the system (19) is re-written in the matrix-vector form
\[
X[k+1] = (\mathcal{I} \otimes A) \mathcal{X}[k] + (\mathcal{I} \otimes B) \mathcal{U}[k] + \mathcal{O}[k] \]
\[
\mathcal{Y}[k] = (\mathcal{I} \otimes C) \mathcal{X}[k] + \mathcal{V}[k] \]
\[
\mathcal{U}[k] = (\mathcal{I} \otimes K) \mathcal{L} (-\mathcal{d} - \mathcal{Y}[k]) + \mathcal{F}[k]
\]

where \( \mathcal{X}[k] = [x_0^T[k], x_1^T[k], ..., x_N^T[k]]^T, \mathcal{Y}[k] = [y_0[k], y_1[k], ..., y_N[k]]^T, \mathcal{U}[k] = [u_0^T[k], u_1^T[k], ..., u_N^T[k]]^T, \mathcal{O}[k] = [o_0^T[k], o_1^T[k], ..., o_N^T[k]]^T, \mathcal{F}[k] = [F_0[k], F_1[k], ..., F_N[k]]^T, \mathcal{V}[k] = [v_0[k], v_1[k], ..., v_N[k]]^T, \mathcal{d} = [d_0, d_1, ..., d_N]^T, \mathcal{I} \text{ is the } N \times N \text{ identity matrix}, \text{ and } \mathcal{L}
\]

is the reduced network system (24) can be re-written in the following form:
\[
\hat{\mathcal{X}}[k+1] = ([\mathcal{I} \otimes A] - ([\mathcal{I} \otimes B] ([\mathcal{I} \otimes K] \mathcal{L} ([\mathcal{I} \otimes C])) \hat{\mathcal{X}}[k] - ([\mathcal{I} \otimes B] ([\mathcal{I} \otimes K] \mathcal{L} \mathcal{d} + ([\mathcal{I} \otimes B] \mathcal{F}[k] + ([\mathcal{I} \otimes B] ([\mathcal{I} \otimes K] \mathcal{L} \mathcal{V}[k] + \mathcal{O}[k]).
\]

Theorem 1. Given a blast wave monitoring network denoted by \( G \), consisting of an ideal master clock and \( N \) sensor node clocks with non-identical frequencies \( f_i : f_i \neq f_0 \) and \( i \in N^+ \), and a PI controller gain \( K \), if all the eigenvalues of the following matrix
\[
\begin{bmatrix}
A - BK(l_{11} - l_{01})C & ... & BK(l_{1N} - l_{0N})C \\
... & ... & ...
\end{bmatrix}
\]

are inside the unit circle, then the networked system is stable.

This means that, at steady synchronisation state, the \( i \)-th sensor node clock offset \( \theta_i[k] \) approaches to \(-d_i \).

Is this approach.
Using the \( z \)-transformation, it is easy to find that the characteristic polynomial of this system is

\[
\begin{vmatrix}
-zI - BK(l_{11} - l_{01}) + BK(l_{1N} - l_{0N})C & \cdots & -BK(l_{1N} - l_{0N})C \\
\vdots & \ddots & \vdots \\
-BK(l_{N1} - l_{01}) & \cdots & -zI + BK(l_{NN} - l_{0N})C
\end{vmatrix}
\]

(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

\[
\bar{X}[z] = \left( z\bar{I} - \left( (\bar{I} \otimes A) - (\bar{I} \otimes B) (\bar{I} \otimes K) \bar{L} (\bar{I} \otimes C) \right) \right)^{-1} \times \frac{z}{z - 1} \bar{X}[z] + \frac{z - 1}{z - 1} \bar{F}[z]
\]

(27)

Let \( E(\bar{X}[k]) \) denote the expected value of \( \bar{X}[k] \) at steady synchronised state. That is

\[
E(\bar{X}[k]) = \lim_{k \to \infty} \mathbb{E}(z - 1) \bar{X}[z]
\]

\[
= \left( z\bar{I} - \left( (\bar{I} \otimes A) - (\bar{I} \otimes B) (\bar{I} \otimes K) \bar{L} (\bar{I} \otimes C) \right) \right)^{-1} \times \frac{z}{z - 1} \bar{X}[z] + \frac{z - 1}{z - 1} \bar{F}[z]
\]

(28)

where \( \bar{\eta}_i + \delta_{ni} \) 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 of the clock offset (i.e. \( E(\theta[i][k]) = \lim_{k \to \infty} \frac{1}{2^k} \sum_{q=k}^{2^k} \theta[i][q] \)) approaches to \(-d_i\) at synchronised state. That is

\[
E(\theta[i][k]) = -d_i.
\]

(29)

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

\[
\lim_{k \to \infty} r[k] e^{j\psi[k]} = \frac{1}{N+1} \sum_{i=0}^{N} e^{j(E(\theta[i][k])+d_i)} = 1.
\]

(30)

This means that \( r[\infty] = 1 \) and \( \psi[\infty] = 0 \). At synchronised state, all the clocks (i.e. oscillators) fire at the specified time \( d_i \), and all the nodes transmit \texttt{Sync} packets at the allocated time slot \( d_i \) of each synchronisation cycle.
V. EXPERIMENTAL EVALUATION

This section presents the performance evaluation of the PI-based PkCOs protocol in two blast wave monitoring networks of Fig. 2. In the implementation, the clock state is represented by a 32-bit counter register of the RTC module, which uses a 32.768kHz 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 9.15ms and 3.66ms$^3$, 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 analyser [30] is adopted to collect the data for analysis.

In addition to the macroscopic metric order parameter $r[k]$, this work also uses the root synchronisation precision, local synchronisation precision and global synchronisation precision to microscopically measure the achieved performance. The root precision is defined as the time difference between the sensor node clock and master clock. The local synchronisation precision is the clock difference between two adjacent nodes, and the global synchronisation precision is the difference of the local clock between two arbitrary nodes in a network. The

$$\mu_i = 0 \quad \mu_i = \rho \gamma d_i$$

Fitted curve

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

RBS, PCO, and PI-Sync 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 synchronisation 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 $\rho \gamma_i d_i$ to further improve the PI-based PkCOs performance.

Fig. 8 shows the synchronisation 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 PI-Sync, 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 proportional-integral PkCOs accuracy. From Fig. 8a, 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 μs), in terms of the root precision (see Fig. 8b).

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

In this paper, we adopt a PI-based PkCOs protocol to synchronise drifting clocks in blast wave monitoring networks. To avoid the occurrence of Sync packet concurrent transmission at synchronised state, the packet-coupled synchronisation 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 is proposed to eliminate for the impacts subject to the drifting clock frequency and anti-phase synchronisation duration. We also study the theoretical performance of the PkCOs synchronisation scheme in the network. The experimental results show that the proportional-integral PkCOs protocol provides an alternative clock synchronisation solution for the blast wave monitoring network.

REFERENCES

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 in 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 (WSNs) and Industrial Internet of Things (IIoT).
2021-11-08

PkCOs: synchronisation of packet-coupled oscillators in blast wave monitoring networks

Zong, Yan

IEEE

https://doi.org/10.1109/JIOT.2021.3126059

*Downloaded from Cranfield Library Services E-Repository*