Low Overhead Minimum Variance Time Synchronization for Time-Sensitive Wireless Sensor Networks

Zhian Jia, Dongliang Cui, Xuewu Dai, *Member, IEEE,* Zhi-Wei Liu, *Senior Member, IEEE,* Shuxin Liu, and Tianyou Chai, *Life Fellow, IEEE*

***Abstract*—Targeting to improve the time synchronization accuracy of multi-hop Time-Sensitive Wireless Sensor Networks (TS-WSNs) for mission-critical industrial automation applications, a Minimum Variance Time Synchronization (MVTS) algorithm utilizing the concept of Packet-Coupled Oscillators (PkCOs) is proposed. This MVTS algorithm utlizes an output feedback approach to mitigate the impact of communication noise on the accumulation of synchronization errors. In addition, a Time-Division Multiple Access (TDMA) packet-exchange superframe is introduced to achieve efficient and low-overhead time synchronization. The optimal gain matrix of the MVTS algorithm is obtained by the Linear Matrix Inequality (LMI) optimization with theoretic analysis. The proposed MVTS algorithm is evaluated by both simulation and experiments on an IEEE 802.15.4 hardware testbed. The experimental results show that the proposed algorithm can effectively reduce the growth rate of clock offset along multi-hop nodes and improve the time synchronization accuracy of the TS-WSNs.**

***Note to Practitioners*—This paper explores a method to achieve precise time synchronization in TS-WSNs. The primary challenge being addressed is the accumulation of synchronization errors that occur in multi-hop TS-WSNs, which can compromise the accuracy of time synchronization. To mitigate this challenge while taking communication overhead into account, we propose a solution that combines a TDMA-based packet-exchange superframe structure with the MVTS algorithm. This approach introduces an output feedback consensus control scheme to minimize synchronization error variance. The optimal gain matrix for this consensus control scheme is derived through LMI optimization. The algorithm is implemented on an IEEE 802.15.4-compatable wireless node SAM R21 by Microchip and the experimental results of a 10-hop netowrk shows that the maximum synchronization error is 8.32μs, reduced by 56% and 32%, repsectivly, compared to the baseline method FTSP and the recent PISync.**

***Index Terms*—Time synchronization, time-sensitive wireless sensor networks (TS-WSNs), packet-coupled oscillators (PkCOs).**

Zhian Jia is with the State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, Shenyang, 110819, China, and also with the School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan, 430074, China (e-mail: jiazhian@hust.edu.cn).

Dongliang Cui, Shuxin Liu and Tianyou Chai are with the State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, Shenyang, 110819, China (e-mail: cuidongliang@mail.neu.edu.cn, shuxin.liu.neu@gmail.com, tychai@mail.neu.edu.cn).

Xuewu Dai is with the Department of Mathematics, Physics and Electrical Engineering, Northumbria University, NE1 8ST Newcastle-uponTyne, U.K. (e-mail: xuewu.dai@northumbria.ac.uk).

Zhi-Wei Liu is with the School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan, 430074, China (e-mail: zwliu@hust.edu.cn).

I. Introduction

TIME synchronization plays a key role in time-sensitive applications of Wireless Sensor Networks (WSNs), such as industrial automaton and cooperation of multiple mobile robots [1], condition monitoring and event detection [2], localization [3], smart homes [4], etc. It is worth pointing out that these require precise time synchronization among distributed nodes [5]. Wireless sensor devices usually deploy crystal oscillator to provide local time, and due to the manufacturing tolerance, aging and varying operation conditions (e.g. ambient temperature), the crystal oscillators' frequency is non-identical and time-varying [6], the local time of each device deviates from the reference time. The deviation with respect to the reference time is called *clock offset*, which needs to be regulated by time synchronization and retained as close to zero as possible. Since WSNs is a packet-exchange network, packet exchange time synchronization is the process of providing a common sense of timing for a set of distributed sensor nodes through clock information exchange and clock's frequency and phase adjustment. Time synchronization is indeed a typical joint communication and an optimal control design problem.

A. Related Work

Time synchronization research has been growing with the blooming of WSNs, such as in [7]–[28]. Two main challenges in time synchronization in battery-powered WSNs are the low overhead requirements (i.e. communication overhead of synchronization) and the network scale challenges (i.e. increased clock synchronisation errors due to the increased number of communication hops). Next, we will introduce the relevant research work of low overhead requirements and network scale challenges.

1) Low Overhead Requirements: Current research into the problem of low overhead requirements has focused on the study of event-triggered control strategies and communication strategies. As a practical industrial application, event-triggered control strategies are not always suitable for clock synchronisation nodes with hard real time constraint [11]. In terms of communication strategy design, the packet exchange communication protocols at the Media Access Control (MAC) layer encompass two typical types [13], namely, the contention based (i.e., the Carrier Sense Multiple Access

(CSMA) protocols [14]) and the contortions-free scheduling based (i.e., Time Division Multiple Access (TDMA) protocols [15], [16]). CSMA protocols check channel availability before transmitting, but collisions can still happen if multiple nodes send packets simultaneously. A random back-off delay is used to handle collisions. In contrast, TDMA protocols allocate unique time slots to nodes for packet transmission, preventing collisions. TDMA is a contention-free protocol, reducing communication bandwidth and energy usage, particularly during high network traffic. Existing studies such as [17] and [18] design low overhead communication protocols based on the TDMA mechanism. Recent research, such as [19], proposes an asymmetric timestamp-free time synchronization scheme with two estimation methods tailored for resource-constrained WSNs. Additionally, in [20], a one-way time synchronization scheme tailored for the energy-efficient LoRa network is proposed based on the reverse asymmetric framework.

2) Network Scale Challenges: In multi-hop WSNs, the number of hops is a major concern that affects the accuracy of all synchronization protocols, as the synchronization error accumulates and increases rapidly along the hops. Traditional time synchronization algorithms such as FTSP [21] and RBS [22] are a kind of proportional control with a control gain of unit, the local clock is adjusted by the same amount of the measured or estimated clock offset. A consensus-based protocol called Average TimeSynch (ATS) was proposed in [23], which has the advantages of being totally distributed, robust to sensor node failure, and it is adaptive to time-varying clock drifts. And then the proportional-integral clock phase and frequency regulation strategy was proposed in [24], which is mainly through the integral role to suppress the impact of clock noise and delay fluctuations. And the PISync time synchronization algorithm proposed in [25], which was later proposed on the basis of an adaptive proportional-integral controller, combined with a distributed protocol for flooding to improve the steady-state error and scalability of the system. Newer studies include the maximum-likelihood estimation algorithm [26], which achieves better clock offset compensation through a strategy of separating clock offset compensation from clock skew compensation. Additionally, Bayesian estimation [27] and Kalman estimation [28] time synchronization algorithms have been proposed, aiming to mitigate the impact of communication noise on clock offset estimates through tailored estimation strategies.

B. Contribution

This paper delves into the challenges posed by communication overhead and synchronization error accumulation in multi-hop Time-Sensitive Wireless Sensor Networks (TS-WSNs). To tackle these challenges, the Minimum Variance Time Synchronization (MVTS) algorithm with an output feedback regulation mechanism is investigated. Rigorous theoretical analysis of synchronization performance and convergence is provided, taking into account uncertainties and noises stemming from low-power embedded processors and communication delay jitters. The primary focus is on reducing both communication overhead and synchronization error accumulation as the hop count increases.

TABLE I
NOTATIONS USED IN THIS PAPER

Symbol	Definition
$\mathcal{G}$	The directed graph for the spanning tree path in TS-WSNs
$\mathcal{L}$	The laplacian matrix for the graph $\mathcal{G}$
f_0	Nominal frequency of the crystal oscillator for reference nodes
f_i	Non-nominal frequency of the crystal oscillator for the i-th sensor node
T	Synchronization cycle
M_0	Clock moment value for the reference node
M_i	Clock moment value for the i-th sensor node
θ_i	Clock offset for the i-th sensor node from the reference node
γ_i	Clock skew for the i-th sensor node from the reference node
ξ_i	The clock offset noise for the i-th sensor node
w_i	The clock skew noise for the i-th sensor node
ζ_i	The measurement noise for the i-th sensor node
$\hat{\theta}_i$	The observed clock offset for the i-th sensor node
$\hat{\gamma}_i$	The observed clock skew for the i-th sensor node
x_i	The clock state for the i-th sensor node
y_i	The observed clock state for the i-th sensor node
u_i	The control strategy for the i-th sensor node
K_i	The gain matrix for the i-th sensor node
N	The number of nodes in TS-WSNs
X	The clock state for the network system
Y	The observed clock state for the network system
U	The control strategy for the network system
P	The error variance matrix for the observed clock state.

The main work and contribution of this paper can be summaries as follows:

1) A collision-free TDMA communication protocol is proposed, which reduces the time-stamped packets for synchronization, thus effectively reduces communication overhead and energy consumption.
2) A MVTS algorithm is developed, featuring an output feedback control scheme to minimize the variance of synchronization errors. The convergence conditions and parameter optimization for the proposed MVTS algorithm is also proved.
3) An IEEE 802.15.4 hardware testbed is built to evaluate the proposed MVTS algorithm. The hardware experiments results show that, compared with the conventional time synchronization protocols (e.g. FTSP [21]), the increase rate of synchronization errors concerning hops nearly halved.

II. PROBLEM FORMULATION

In this section, we propose a hardware clock model and a network model for embedded systems, which we use for the presentation and analysis of the synchronization algorithms in this paper. And the variables and acronyms used in this paper can be found in Table I and Table II, respectively.

In TS-WSNs, as illustrated in Fig. 1, the reference node typically serves as a GPS clock node, while other nodes are often designated as non-GPS clock nodes (Non-ideal clock nodes) due to cost considerations. Consequently, the network can be viewed as a time synchronization control problem centered around the reference node (leader). Sensor nodes are typically not uniformly distributed, and due to the communication constraints of wireless networks, not all nodes have direct communication links with each other. For instance, the black lines between nodes in Fig. 1 indicate the existence of communication links. Therefore, nodes far away from the reference node generally need multi-hop communication to

TABLE II
LIST OF ACRONYMS OF TIME SYNCHRONIZATION PROTOCOLS

Acronym	Definition
ATS	Average TimeSynch
CSMA	Carrier Sense Multiple Access
FTSP	Flooding Time Synchronization Protocol
LMI	Linear Matrix Inequality
MAC	Media Access Control
MAS	Multi-Agent Systems
MVTS	Minimum Variance Time Synchronization
PkCOs	Packet-Coupled Oscillators
RBS	Reference Broadcast Synchronization
TDMA	Time Division Multiple Access
TS-WSNs	Time-Sensitive Wireless Sensor Networks
WSNs	Wireless Sensor Networks

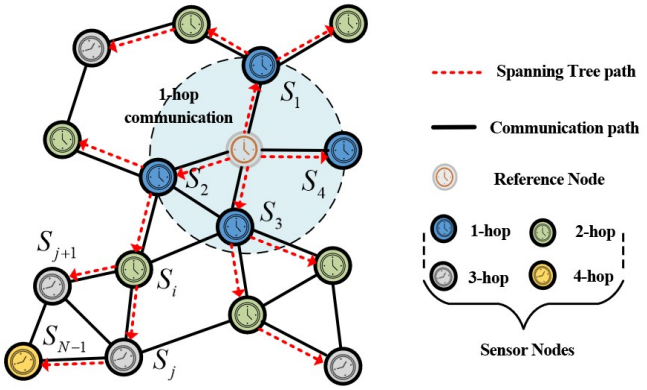


Fig. 1. Time-sensitive wireless sensor networks (e.g. UAV target tracking and localization)

communicate with the reference node, which will cause the synchronization accuracy of the network system to decline, this paper is mainly devoted to solve the problem of time synchronization accuracy decreases with the increase in the number of hops.

A. Graph Theory and Matrix Theory Knowledge

In the TS-WSNs depicted in Fig. 1, there are multiple spanning tree paths within their communication paths. In the event of node failures in sensor nodes, new spanning tree paths can be generated during network design, as long as there exists a spanning tree within the communication paths that ensures connectivity among all nodes. The spanning tree path can be described by a directed graph, let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ denoted a directed graph, where $\mathcal{V} = \{v_1, \cdots, v_N\}$ represents N clock nodes and the set of edges $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ represents the directional connection between clock nodes. For a given clock node i, the set of neighbours is defined as $\mathcal{N}_i = \{j \mid (i,j) \in \mathcal{E}\}$. Define the matrix $\mathcal{A} = [a_{ij}] \in \mathcal{R}^{N \times N}$ as the adjacency matrix of a graph, when $(i,j) \in \mathcal{E}, a_{ij} > 0$; otherwise, $a_{ij} = 0$. Therefore, the Laplacian matrix of the graph is defined as: $\mathcal{L} = \mathcal{D} - \mathcal{A}$, where $\mathcal{D} = diag\{d_1, \cdots, d_N\}, d_i = \sum_{j=1}^{N} a_{ij}$.

***Lemma 1** [29]*: For a real symmetric matrix S, the condition

$$S = \begin{bmatrix} S_{11} & S_{12} \\ * & S_{22} \end{bmatrix} < 0$$

is equivalent to the following two cases:

1) $S_{11} < 0,\ S_{22} - S_{12}^T S_{11}^{-1} S_{12} < 0.$
2) $S_{22} < 0,\ S_{11} - S_{12} S_{22}^{-1} S_{12}^T < 0.$

B. Crystal Oscillator Clock Model

For the actual embedded systems, the local clock module consists of two parts: a) crystal oscillator, ticking at a nominal frequency f_0, and b) a counting register to count the number of ticks from the crystal oscillator [7], where the count register accumulates the number of ticks of the crystal oscillator and when the threshold is reached, the value of the register is cleared and a pulse is generated to trigger an interrupt processing event, in this paper, the synchronous packet exchange will be triggered.

For an ideal crystal oscillator, define $t(q) = q/f_0$ as the q-th clock event of the crystal oscillator, $t(q)$ is accurate and referred to as the reference time. Each clock period $\tau_0 = 1/f_0$ can be treated as an event to record by a counter register. Such a perfect embedded clock is also called the reference clock. And we define reference clock threshold φ_0. We adopt PkCOs model [7], and its clock moment value $M_0(q)$ can be described as:

$$M_0(q) = t(q) - \sum_{h=0}^{k} \varphi_0(h), \tag{1}$$

where $k = \left\lfloor \frac{q\tau_0}{T} \right\rfloor$ represents how many clock resettings have occurred from $q = 0$ to q-th clock event, and T is the synchronization cycle, and q is the integer value of the register of the actual embedded system, and the event of the counter reaching q is referred to as the q-th clock event [10].

For sensor nodes, the crystal frequency usually deviates from f_0 because of the manufacturing tolerance, ambient temperature, power supply and currents (see [30], [31]). Additionally, the interrupt processing delay of the embedded system (microprocessor R21) affects the clock moment value, which represents the time stamping uncertainties and is treated as clock phase noise. Thus, the clock moment value of the i-th sensor node can be described as:

$$M_i(q) = t(q) + \frac{\sum_{h=0}^{q} \chi_i(h)\tau_0}{f_0} + \frac{\phi_i(q)}{2\pi f_0} - \sum_{h=0}^{k} \varphi_i(h), \tag{2}$$

where χ_i is the deviation of each tick count caused by the frequency skew of the crystal oscillator (i.e. $\chi_i = f_i - f_0$), and the effect on the clock moment value of the q-th event accumulation is $\frac{\sum_{h=0}^{q} \chi_i(h)\tau_0}{f_0}$. $\phi_i(q)$ is the effect of environmental factors on clock random noise, φ_i is the i-th sensor node clock threshold.

Let clock offset $\theta_i(q)$ denote the clock offset of the i-th sensor node from the reference node. And we can obtained:

$$\begin{aligned} \theta_i(q) &= M_i(q) - M_0(q) \\ &= \frac{\sum_{h=0}^{q} \chi_i(h)\tau_0}{f_0} + \frac{\phi_i(q)}{2\pi f_0} - \sum_{h=0}^{k} \Delta\varphi_i(h), \end{aligned} \tag{3}$$

where $\Delta\varphi_i = \varphi_i - \varphi_0$, and $\Delta\varphi_i$ can be seen as a reset behaviour of the clock due to a non-nominal frequency.

In order to have a better description of the deviation of the clock frequency from the reference clock frequency, we define

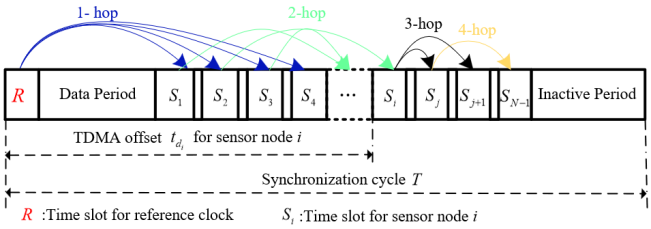


Fig. 2. Communication frame structure associated with Fig. 1

the clock skew $\gamma_i = \frac{f_i - f_0}{f_0}$, combined with the descriptive analysis provided in [7], and converting the clock offset from the q-dimension (i.e. the clock update period) to the k-dimension (i.e. the time synchronization cycle), we can obtain a simplified incremental form of the clock offset.

$$\theta_i(k+1) = \theta_i(k) + \gamma_i(k)T + \xi_i(k), \tag{4}$$

where T is the synchronization cycle, $\xi_i(k)$ means the clock offset noise, which can be described by a Gaussian stochastic process.

And the clock skew $\gamma_i(k)$ can be described as a stochastic process and the clock skew noise is $w_i(k)$, thus the clock skew at time $k+1$ can be expressed as:

$$\gamma_i(k+1) = \gamma_i(k) + w_i(k). \tag{5}$$

In combination with (4) and (5), the free clock model for the i-th clock can be expressed as:

$$\begin{cases} \theta_i(k+1) = \theta_i(k) + \gamma_i(k)T + \xi_i(k) \\ \gamma_i(k+1) = \gamma_i(k) + w_i(k) \end{cases}, \tag{6}$$

where $\xi_i(k)$ and $w_i(k)$ are the clock offset noise and skew noise, respectively.

Remark 1: In a point-to-point network, sensor nodes can obtain the free clock model as shown in (6) through point-to-point communication. However, for the time synchronization based on reference nodes with master-slave constraints depicted in Fig. 1, some nodes situated far away from the reference node typically require multi-hop communication to establish communication with the reference node. Hence, the actual free clock model for multi-hop nodes cannot be directly obtained and necessitates multi-hop communication to acquire. A detailed analysis will be provided in Section II-C.

C. TDMA packet exchange for clock state observation

In order to reduce the communication consumption in synchronization packet exchange, this paper adopts the TDMA mechanism for packet exchange between nodes. Base on our previous work [7], the TDMA superframe structure used in this paper is shown in Fig.2. This communication frame requires only one packet exchange to obtain the clock offset and only needs about 1/3 of communication costs compared with common two-way packet-exchange. And there exists a TDMA offset t_{d_i} between the reference node and the i-th sensor node, which is related to the definition of the frame format and can be calculated by (11) in [7].

In TS-WSNs, clock offset needs to be obtained through packet exchange. Under the communication frame structure depicted in Fig. 1, and in conjunction with the definition of clock offset (3), the clock offset of sensor node i for the k-th communication packet exchange is defined as $\theta_i(k) = M_i(k) - M_0(k) - t_{d_i}$. It's worth noting that $\theta_i(k)$ represents the ideal clock offset of sensor node i under perfect communication conditions. However, in practical environments, communication delay jitters and noise may be present. Therefore, the estimated value of the clock offset obtained from actual communication packet exchanges is denoted as $\hat{\theta}_i(k) = \theta_i(k) + \zeta_i(k)$, where $\zeta_i(k)$ represents the random disturbance due to the random asymmetry of communication delay jitters, following a Gaussian distribution with variance σ_v^2.

The observed clock skew is related to the clock offset, typically obtained from the clock offset estimate, which can be expressed as:

$$\hat{\gamma}_i(k) = \frac{\hat{\theta}_i(k) - \hat{\theta}_i(k-1)}{T}. \tag{7}$$

Here, $\hat{\gamma}_i(k)$ represents the observed clock skew, with its noise following a variance of $2\sigma v^2/\Delta T^2$. Define the clock state as $x_i(k) = [\theta_i(k)\ \gamma_i(k)]^T$, and the model for the i-th controlled clock observation is as follows:

$$\begin{cases} x_i(k+1) = Ax_i(k) + Bu_i(k) + d_i(k) \\ y_i(k) = Cx_i(k) + D\zeta_i(k) \end{cases}, \tag{8}$$

where $y_i(k)$ represents the observed clock state for the i-th sensor node. The state transition matrix $A = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}$, and the control input matrix $B = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}$. The state noise is represented by $d_i(k) = [\xi_i(k)\ w_i(k)]^T$. The observation matrix is $C = I_2$, where $C = I_2$ is the 2×2 identity matrix. $\zeta_i(k)$ represents the measurement noise, and the noise input matrix is $D = \begin{bmatrix} 1 & \frac{\sqrt{2}}{\Delta T} \end{bmatrix}^T$. The covariance of the state noise $d_i(k)$ is $R_0 = E\left(d_i(k)d_i^T(k)\right) = \text{diag}\begin{bmatrix} \sigma_\xi^2 & \sigma_w^2 \end{bmatrix}$, where σ_ξ^2 and σ_w^2 represent the variance of the phase noise and clock skew noise, respectively. The covariance of the observed noise $Q_0 = E\left(\zeta_i(k)\zeta_i^T(k)\right) = \begin{bmatrix} \sigma_\zeta^2 & \sigma_\zeta^2/\Delta T \\ \sigma_\zeta^2/\Delta T & 2\sigma_\zeta^2/\Delta T^2 \end{bmatrix}$. And in this paper, the following control objectives are considered.

Control objectives: Our objective for the system (8) is to design a control strategy to guarantee the existence of $\lim_{k\to\infty} x_i(k) = 0$ for any sensor node i.

In this paper, considering practical communication constraints, we define the following time synchronization control strategy:

$$u_i(k) = -K_i \sum_{j \in \mathcal{N}_i} l_{ij} \left(y_j(k) - y_i(k)\right), \tag{9}$$

where $\mathcal{N}_i$ represents node i's communication neighbor nodes, which is associated with the directed graph. K_i denotes the gain matrix for the i-th sensor node, and the design methodology will be provided in Section III.

D. Network Model

Assuming a TS-WSNs with N clock nodes, the local node clock observation model depicted in (8) can be extended to a network system model with N nodes:

$$\begin{cases} X(k+1) = (I_N \otimes A)\, X(k) + (I_N \otimes B)\, U(k) + D(k) \\ Y(k) = (I_N \otimes C)\, X(k) + Z(k) \end{cases}, \tag{10}$$

where the state $X(k) = \left[x_1^T(k), \cdots, x_N^T(k)\right]^T$, output $Y(k)$=$[y_1^T(k), \cdots, y_N^T(k)]^T$, control input $U(k) = [u_1^T(k), \cdots, u_N^T(k)]^T$, noise matrix $D(k) = [d_1^T(k), \cdots, d_N^T(k)]^T$, $Z(k) = [\zeta_1^T(k), \cdots, \zeta_N^T(k)]^T$. I_N represents the $N \times N$ identity matrix.

Furthermore, a network graph can be constructed through the knowledge of graph theory with a connected Laplace matrix $\mathcal{L}$, as illustrated in (11). In this matrix, the corresponding element $l_{ij} = -1$ if $(i,j) \in \mathcal{E}$, indicating that node i can receive information from node j; otherwise, $l_{ij} = 0$. Additionally, the diagonal element $l_{ii} = -d_i$, where d_i represents the degree of node i, indicating how many neighbor nodes node i has, calculated as $d_i = \sum_{j=1}^{N} l_{ij}$.

$$l_{ij} = \begin{cases} -1, & (i,j) \in \mathcal{E}, i \neq j \\ 0 & (i,j) \notin \mathcal{E}, i \neq j \\ -\sum_{j=1}^{N} l_{ij} & i = j \end{cases}, \tag{11}$$

where $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ represents a directed connection between clock nodes, $\mathcal{V} = (v_1 \cdots v_N)$ represents N clock nodes.

Thus, by combining (9) and (11), the networked system model depicted in (10) can be expressed recursively as follows:

$$Y(k+1) = (I_N \otimes A - \mathcal{L} \otimes BK)\, Y(k) + W(k), \tag{12}$$

where $\mathcal{L}$ is the Laplace matrix and $W(k) = D(k) + Z(k+1) - (I_N \otimes A)\, Z(k)$ is the noise matrix.

It's worth noting that this paper focuses on TS-WSNs with a reference node. In this scenario, the Laplacian matrix can be represented as $\mathcal{L} = \begin{bmatrix} 0 & \mathbf{0}_{N-1} \\ \mathcal{L}_1 & \mathcal{L}_2 \end{bmatrix}$, given that the reference node doesn't require control input. Therefore, Eq. (12) can be transformed as follows:

$$Y_2(k+1) = (\tilde{A} - \mathcal{L}_2 \otimes BK) Y_2(k) + W_2(k), \tag{13}$$

where $\tilde{A} = I_{N-1} \otimes A$, $W_2(k) = D_2(k) + Z_2(k+1) - \tilde{A} Z_2(k)$, $D_2(k) = \left[d_2^T(k), \ldots, d_N^T(k)\right]^T$, $Z_2(k) = \left[\zeta_2^T(k), \cdots, \zeta_N^T(k)\right]^T$.

III. MVTS Algorithm

In this section, we devise a minimum variance time synchronization algorithm, named MVTS, an optimization strategy is proposed for the gain matrix K in (9) and a TDMA-based one-way communication mechanism is presented.

In order to reduce the multi-hop accumulated synchronization errors, the concept of minimizing mean square error at steady state developed in the control community for Multi-Agent Systems (MAS) is adopted in [32] and modified to meet the time synchronization needs. We combines the practice of communication engineering and the theory of consensus control in MAS to develop an optimal output feedback control algorithm to minimize the variance of time synchronization errors among the whole multi-hop WSNs.

A. Minimum Variance Strategy

Due to the constraints imposed by packet exchange for the observed clock state and network communication, we have formulated the clock network system model as depicted in Eq. (13). Considering the control objectives of this paper, we can substitute it with the following optimization problem:

$$\min J = \lim_{k \to \infty} \operatorname{tr}\left(E\left(Y_2(k) Y_2^T(k)\right)\right), \tag{14}$$

where $E()$ represents the expected value and the ultimate goal. $\operatorname{tr}()$ represents the trace of a matrix, which refers to the sum of all diagonal elements of a square matrix.

Lemma 2 [33]: The networked system (13) is asymptotically stable if and only if there exists a symmetric matrix $\mathcal{R}$ such that:

$$\begin{bmatrix} \mathcal{R} & (\tilde{A} - \mathcal{L}_2 \otimes BK)\mathcal{R} \\ \mathcal{R}^T(\tilde{A} - \mathcal{L}_2 \otimes BK)^T & \mathcal{R} \end{bmatrix} > 0. \tag{15}$$

Theorem 1: For the networked system (13), the optimization objective shown in (14) is adopted. There exists $G > 0, H > 0, P > 0$ such that the gain matrix $K = HG^{-1}$, and the optimization parameters can be expressed based on the LMI as follows:

$$\begin{aligned} &\min_{G,H,P} \operatorname{tr} P \\ &\text{s.t.} \begin{bmatrix} P - g & \tilde{A}\tilde{G} - \mathcal{L}_2 \otimes BH \\ (\tilde{A}\tilde{G} - \mathcal{L}_2 \otimes BH)^T & \tilde{G} + \tilde{G}^T + P \end{bmatrix} > 0 \end{aligned}.$$

Proof:

For the networked system (13), our optimization problem is formulated as (14). Therefore, we define $P(k) = E\left[Y_2(k) Y_2^T(k)\right]$, and consider the independence of the process noise $D_2(k)$ and measurement noise $Z_2(k)$ in the networked system (13). Leveraging the characteristics of independent noises, we have $E\left[Y_2(k) D_2(k)\right] = \mathbf{0}_{N-1}$, $E\left[Y_2(k) Z_2(k)\right] = \mathbf{0}_{N-1}$, and $E\left[D_2(k) Z_2(k)\right] = \mathbf{0}_{N-1}$. Thus, we can express $P(k)$ as follows:

$$\begin{aligned} P(k+1) = (\tilde{A} - \mathcal{L}_2 \otimes BK) P(k) (\tilde{A} - \mathcal{L}_2 \otimes BK)^T \\ + R + Q_1 + Q_2, \end{aligned} \tag{16}$$

where $R = I_{N-1} \otimes R_0$, $Q_1 = I_{N-1} \otimes Q_0$, $Q_2 = I_{N-1} \otimes \left(A Q_0 A^T\right)$, R_0 and Q_0 are the covariance matrices of the state noise and observation noise, respectively.

Assuming the networked system (13) is asymptotically stable, the final steady state $P(k)$ also converges. According to Eq. (16), as $k \to \infty$, $P(k)$ is replaced by P, and letting $g = R + Q_1 + Q_2$, we have:

$$P = (\tilde{A} - \mathcal{L}_2 \otimes BK) P (\tilde{A} - \mathcal{L}_2 \otimes BK)^T + g. \tag{17}$$

The objective function yields its optimal K such that the expectation $tr(P)$ is minimized, and defining the diagonal

blocks of P as $P_1, P_2, \ldots, P_{N-1}$, with P having the same dimension as $\mathcal{L}_2$, the optimization problem (14) converts to:

$$\begin{aligned} &\min_{P} \operatorname{tr} P \\ &\text{s.t. } P = (\tilde{A} - \mathcal{L}_2 \otimes BK)P(\tilde{A} - \mathcal{L}_2 \otimes BK)^T + g \end{aligned}. \quad (18)$$

For the optimization problem as shown in (18), solving it directly may pose some difficulties. However, based on Lemma 1, this paper addresses this challenge by finding the optimal solution using a linear matrix inequality, as follows:

$$\begin{aligned} &\min_{P} \operatorname{tr} P \\ &\text{s.t. } \begin{bmatrix} P - g & (\tilde{A} - \mathcal{L}_2 \otimes BK)P \\ P^T(\tilde{A} - \mathcal{L}_2 \otimes BK)^T & P \end{bmatrix} > 0 \end{aligned}. \quad (19)$$

Therefore, the assumption made in Eq. (17) holds true, as the conditions derived from (19) under the assumption that the networked system (13) is asymptotically stable satisfy the asymptotic stability criteria of the networked system (13), as shown in Lemma 2.

According [33], there exist a general matrix $G \in \mathbb{R}^{2\times 2}$ and $H = KG$, an equivalent optimization problem for solving LMI can be obtained:

$$\begin{aligned} &\min_{G,H,P} \operatorname{tr} P \\ &\text{s.t. } \begin{bmatrix} P - g & \tilde{A}\tilde{G} - \mathcal{L}_2 \otimes BH \\ (\tilde{A}\tilde{G} - \mathcal{L}_2 \otimes BH)^T & \tilde{G} + \tilde{G}^T + P \end{bmatrix} > 0 \end{aligned}. \quad (20)$$

where $\tilde{G} = I_{N-1} \otimes G$. By solving the LMI optimization problem (20) for G, H, and P, one can calculate $K = HG^{-1}$. This provides the solution to the optimization problem (14). ■

Remark 2: This paper proposes an optimization solution to minimize the variance for the network topology and provides a method for determining the gain matrix using LMI solving. The gain matrix K remains constant unless there is a change in the network topology, and it is determined through network design (offline). Based on similar problems described in [34], the computational complexity of Eq. (20) is $O(c*(N-1)^2)$, where N represents the number of clock nodes and c is a constant. It is worth noting that various optimization methods can be employed to optimize Eq. (20), and the detailed complexity of optimization depends on the chosen optimization algorithm.

B. Minimum Variance Time Synchronization Algorithm

Next, we present an MVTS time synchronization algorithm with reference nodes, outlined in Algorithm 1. The communication frame structure of this algorithm is shown in Fig. 2, which is based on the TDMA communication mechanism, aiming to minimize communication costs and achieve low overhead.

In our algorithm, the clock offset and clock skew estimates are initialized to zero, and the adjustment threshold of the clock offset, denoted as θ_m is set 100 μs. The threshold θ_m is to avoid over-adjusted clock skew because the offset could be very large at the beginning due to the clock has not been synchronized yet or large jitters in packet exchange. Upon receiving the broadcasted SYNC packet from the parent node, local node's clock offset $\hat{\theta}_i(k)$ is estimated. Since the time synchronization period $T = 1s$ in this paper, the estimated clock skew is computed as $\hat{\gamma}_i(k) = \hat{\theta}_i(k) - \hat{\theta}_i(k-1)$. In lines 5-9, depending on whether the obtained clock offset estimate exceeds the predefined threshold θ_m, the gain matrix K_i is updated in two different ways. If $\hat{\theta}_i(k)$ is greater than or equal to the threshold θ_m, a constant gain $K_i = [1\ 0; 0\ 0]$ is used, meaning that only the clock offset is adjusted while the clock skew remains unchanged. This enables rapid adjustment for biased clocks. Conversely, if the clock offset $\hat{\theta}_i(k)$ is less than the threshold θ_m, the minimum variance strategy proposed in this paper is used to determine the corresponding gain matrix K_i. In lines 10 and 11, the control input $u_i(k)$ is generated according to the computed gain matrix K_i and applied to the embedded system's clock count register and threshold register, respectively. Finally, an synchronization packet SYNC is transmitted to the child nodes.

Algorithm 1: MVTS Algorithm Pseudocode for Node i With Reference Node

1: **Initialization**
2: set $\hat{\theta}_i(0) = 0$, $\hat{\gamma}_i(0) = 0$, $\theta_m = 100\mu s$;
3: **Upon Receiving k-th SYNC Packet from Parent Node**
4: **Updata** $\hat{\theta}_i(k)$, $\hat{\gamma}_i(k) = \hat{\theta}_i(k) - \hat{\theta}_i(k-1)$
5: **if** $\hat{\theta}_i(k) \geq \theta_m$ **then**
6: $K_i = [1\ 0; 0\ 0]$;
7: **else**
8: K_i update from (20)
9: **end if**
10: **Updata** $u_i(k)$
11: $u_i(k) = K_i * [\hat{\theta}_i(k), \hat{\gamma}_i(k)]^T$
12: **Broadcast SYNC Packet to Child Nodes**

Remark 3: Our paper focuses on designing a precision synchronization algorithm for time-sensitive applications with a global reference clock (e.g. a GPS-regulated clock), a "directed" clock tree is adopted, making it not a fully distributed system. It is more like a leader-follower system. This is different from most "fully" distributed consensus systems where no reference clock is provided. We conducted experiments comparing our algorithm with fully distributed time synchronization algorithms such as ATS [23]. The results show that our synchronization algorithm with a reference clock node significantly improves precision compared to ATS. The proposed method achieves higher precision but at the expense of reduced flexibility due to the introduction of clock tree.

Remark 4: This paper is to improve the precision of synchronization accuracy for industrial time-sensitive applications, with a specific focus on addressing the challenges posed by varying packet exchange delays (jitters) and clock stamping uncertainties (observation noises). The packet loss and node failure will easily degrade the synchronization accuracy significantly, rendering it unacceptable for practical use Therefore, in practice, industrial networks usually require extremely high reliability in packet transmission, often requiring

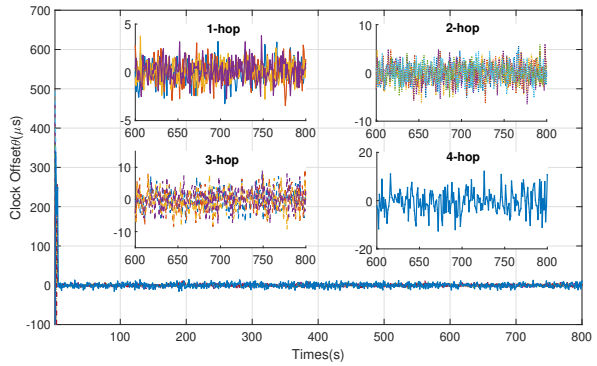


Fig. 3. Dynamic curve of the clock offset

close to 100% success rate, which is guaranteed at the network design stage through proper design of the network topologies, transmission power, and duty cycles, etc. Therefore, under the high-reliability assumption, this paper does not consider packet loss and node failure.

IV. Simulation Results

In order to verify the effectiveness of the MVTS control algorithm proposed in this paper for TS-WSNs, in this section we have carried out simulations using physical noise data from the Microchip SAM R21 in a complex physical environment [7]. Considering the requirements of a real industrial sensor network, we assume that the network connection topology is as shown in Fig. 1, where there exists a reference node, which is usually a GPS clock and can be considered as a perfect clock (reference clock). It is also set that there is communication noise between nodes and that each node is considered to have a crystal frequency skew. Corresponding to the system shown in (8), the phase noise standard deviation σ_ξ and clock skew noise standard deviation σ_w are both 10^{-6}, and the observed noise standard deviation $\sigma_v = 1 * 10^{-6}$.

It's worth noting that different minimum spanning trees for wireless sensor networks do not significantly affect the experimental results. The synchronization accuracy of the network is primarily influenced by the maximum number of hops in the network. Additionally, in the presence of node failures, new spanning tree paths can be generated during network design, as long as a spanning tree exists within the communication paths to ensure connectivity among all nodes. Without loss of generality, we conduct experiments using the spanning tree shown in dashed lines in Fig. 1.

For the synchronization spanning tree depicted in Fig. 1, the sensor nodes synchronize with the reference node in a network with a maximum of 4 hops. From Fig. 3, we observe that the clock offset for the 1-hop network is within $5\mu s$, while for the 4-hop network, it remains within $15\mu s$. To compare the performance of the MVTS algorithm, the simulation evaluates the FTSP [21], PI [7], and PISync [25] algorithms on the grid network shown in Fig. 1, obtaining their maximum clock offset and average clock offset for different hop nodes as presented in Table III and Table IV, respectively.

TABLE III
Maximum clock offset for simulation test

Hops	1	2	3	4
FTSP	$3.8\mu s$	$6.5\mu s$	$12.6\mu s$	$15.9\mu s$
PI	$3.7\mu s$	$6.4\mu s$	$9.8\mu s$	$15.4\mu s$
PISync	$5.2\mu s$	$8.5\mu s$	$13\mu s$	$14.6\mu s$
MVTS	$3.6\mu s$	$5.8\mu s$	$9.5\mu s$	$12.4\mu s$

TABLE IV
Average clock offset for simulation test

Hops	1	2	3	4
FTSP	$18ns$	$65ns$	$0.13\mu s$	$0.29\mu s$
PI	$17ns$	$60ns$	$0.11\mu s$	$0.17\mu s$
PISync	$30ns$	$80ns$	$0.11\mu s$	$0.12\mu s$
MVTS	$10ns$	$50ns$	$0.1\mu s$	$0.1\mu s$

From the comparison table in Table III, we observe that the FTSP algorithm exhibits the highest inefficiency, with its clock offset reaching maximum values that grow rapidly with the number of hops. In contrast, the PI algorithm demonstrates a slower growth rate in clock offset with increasing hop count. Although the PISync algorithm initially shows a larger maximum clock offset for single-hop nodes, its growth rate in maximum clock offset slows down as the number of hops increases. In contrast, the MVTS algorithm displays smaller growth rates in both maximum clock offset for single-hop nodes and with increasing hop count. This outcome aligns with our expectations. The comparison table in Table IV illustrates that the FTSP algorithm yields poorer results in terms of average clock offset, followed by the PI algorithm. Conversely, both the PISync and MVTS algorithms perform better, exhibiting less impact on the average error offsets.

In summary, the simulation results indicate that the algorithm proposed in this paper offers certain advantages over current algorithms in terms of improving time synchronization accuracy. Additionally, for time synchronization with reference nodes, synchronization errors increase significantly as the network size (measured in hops) increases. To allocate more space for emphasizing our hardware experimental results, we only present the simulation results of a 4-hop network and conduct experiments on multi-hop networks in the Experimental Results (Section V), thereby discussing the scalability of this algorithm.

Remark 5: It's worth noting that the simulation experiments in this paper utilize noisy data from complex industrial environments. Therefore, the synchronization accuracy of the simulation results is relatively poorer compared to the experimental results.

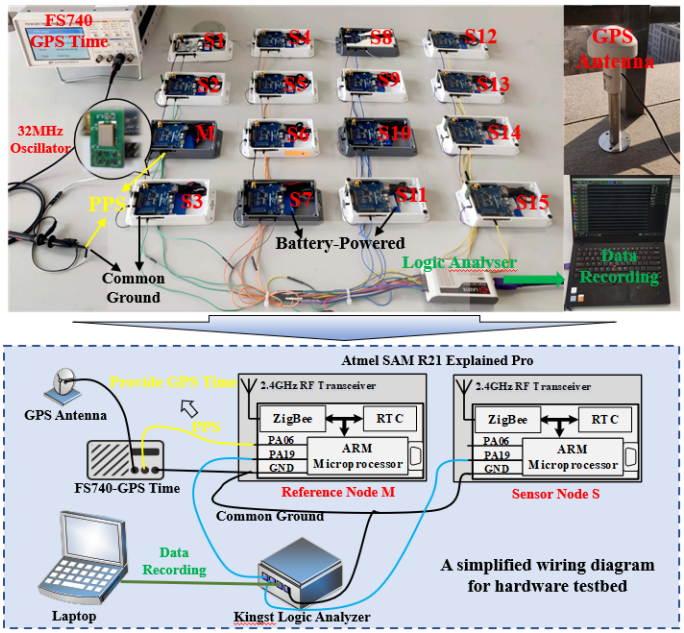


Fig. 4. Hardware testbed (M denotes the reference node, and S1-S15 represent fifteen sensor nodes).

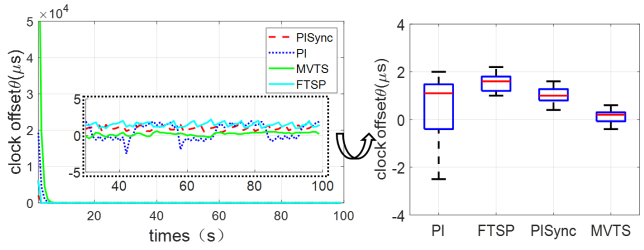


Fig. 5. Dynamic curve (left) and box diagram (right) of the clock offset

V. Experimental Results

The hardware experiments in this paper deviate from the common CSMA synchronization mechanism used in most papers, opting instead for the TDMA mechanism. This choice enables devices without allocated time slices to enter a sleep state, conserving energy through time division multiplexing. Wireless node is based on the Atmel SAM R21 Explained Pro, which is a low-cost wireless sensor platform integrated with the IEEE 802.15.4 physical layer and incorporates an ultra-low power 2.4GHz RF transceiver, AT86RF233 and ARM microprocessor. As shown in Fig.4, the platform utilizes the FS740-GPS Time System, which is connected to the PA06 pin (providing the Pulse Per Second (PPS) signal) of the reference node. And an Epson 32.768MHz oscillator of no more than $\pm 100ppm$ is used as the RTC clock source, and set the synchronization cycle $T = 1s$. The PA19 pin of each node is attached to a Kingst Logic Analyser, which captures time information and transmits it to a laptop for data recording and analysis.

The synchronization accuracy and communication overhead of different methods will be investigated. Let the number of communication packets of clock i be m_i. In two-way synchronization algorithms (e.g., Precision Time Protocol [35]), when

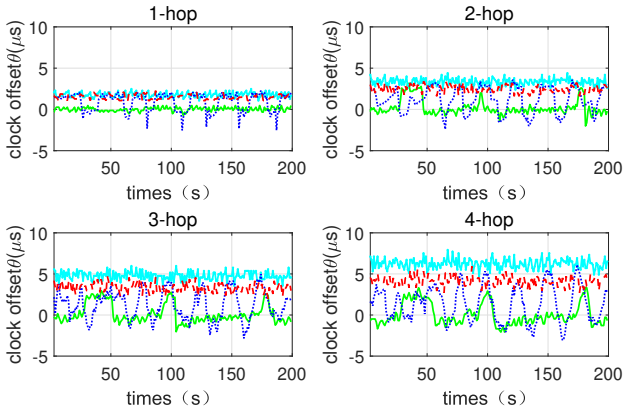


Fig. 6. Comparison of 1-4 hops node clock offset

a synchronization poll occurs, clock i has to request the state information from all its neighbors, receive the information, and then transmit the state to all its neighbors. Thus, the information transmission occurs twice, and each neighbor executes the transmission once. Therefore, the number of information transmissions for clock i is $3n_i m_i$. In one-way synchronization algorithms (e.g., the algorithm proposed in this paper), when a synchronization poll occurs, clock i transmits its state information to all its neighbors. Thus, the number of information transmissions for agent i is $n_i m_i$. This conclusion has been experimentally verified.

To verify the synchronization accuracy of the algorithm proposed in this paper, we have conducted validation tests on point-to-point network, line network, and grid network, respectively.

A. Point-to-Point Synchronization Errors

First, we conducted a point-to-point network test, comparing the FTSP [21], PI [7], PISync [25], and the MVTS algorithm proposed in this paper. We collected clock offset data over 100 seconds (100 synchronization cycles), as depicted in Fig. 5.

As depicted in Fig. 5, all four synchronization algorithms demonstrate fast synchronization for the point-to-point network. To illustrate the synchronization accuracy more effectively, we plotted a box plot of the clock offset data from 30 to 100 seconds (left side of Fig. 5), shown in Fig. 5 (right). It's evident that the FTSP strategy exhibits the largest average error, the PI control strategy shows significant error fluctuations, and the PISync algorithm performs better compared to the previous two. However, the MVTS algorithm proposed in this paper maintains both the mean and variance of the clock offset within $1\mu s$, presenting a significant advantage over current algorithms.

B. Synchronization in Grid Networks

In continuation, we evaluated the synchronization performance of the networked system. Using a network topology resembling that of Fig. 1, we conducted physical experiments and recorded clock offset data over a period of 200 seconds

TABLE V
MAXIMUM CLOCK OFFSET FOR EXPERIMENTAL RESULTS

Hops	1	2	3	4
FTSP	2.47μs	4.39μs	5.96μs	7.92μs
PI	2.27μs	3.59μs	4.96μs	6.12μs
PISync	2.27μs	3.39μs	4.76μs	5.92μs
MVTS	1.07μs	2.2μs	3.19μs	3.31μs

TABLE VI
AVERAGE CLOCK OFFSET FOR EXPERIMENTAL RESULTS

Hops	1	2	3	4
FTSP	1.77μs	3.32μs	4.74μs	6.21μs
PI	0.8μs	1.15μs	1.2μs	1.34μs
PISync	1.5μs	2.46μs	3.33μs	4.12μs
MVTS	0.07μs	0.16μs	0.23μs	0.28μs

for the four methods in a steady state. The dynamic curve of clock offset at different hops is illustrated in Fig. 6.

From Fig. 6, it can be observed that the control effect for 1-hop nodes is essentially similar to that of point-to-point networks. However, for multi-hop nodes, it is evident that the clock offset fluctuates significantly in the PI algorithm, indicating a large variance. Moreover, both the FTSP and PISync algorithms exhibit relatively large mean values in clock offset. In contrast, the MVTS algorithm for 1-4 hop nodes shows smaller mean values in clock offset. The average clock offset for MVTS algorithm remains close to zero for 1-4 hop nodes, although it may be influenced by communication noise, it quickly adjusts back. The maximum and average clock offsets for these four methods are presented in Table V and Table VI, respectively.

In Table V, the slope of the maximum clock offset for the MVTS algorithm increases by approximately $1\mu s$ as the number of hops increases. This growth rate is half of that observed in the FTSP algorithm and 4/5 of the rates seen in the PI and PISync algorithms. Conversely, in Table VI, the average clock offsets for the MVTS algorithm remain close to zero. In contrast, the average clock offsets for the FTSP and PISync algorithms exhibit a linear increase with the number of hops, characterized by larger slopes. This indicates that the MVTS algorithm effectively mitigates the impact of communication noise and delay.

C. Synchronization in Line Networks

As discussed earlier, the synchronization accuracy based on reference nodes is primarily influenced by the maximum communication hop count in the network. Therefore, to validate the synchronization effects of different control algorithms in a multi-hop network, we established a linear topology of

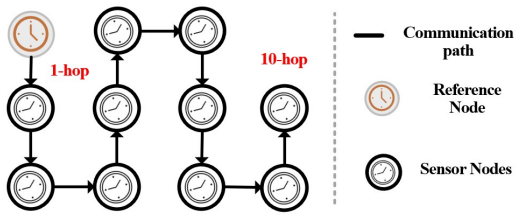


Fig. 7. Linear networks

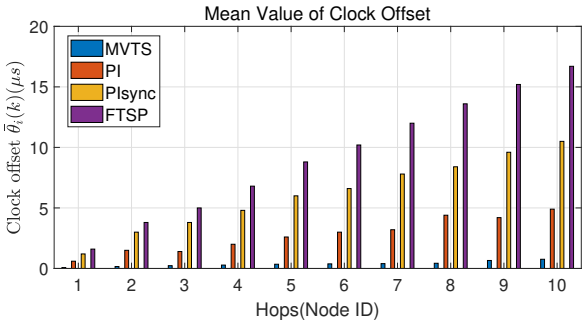


Fig. 8. Mean value of clock offset (From left to right MVTS, PISync, PI, FTSP)

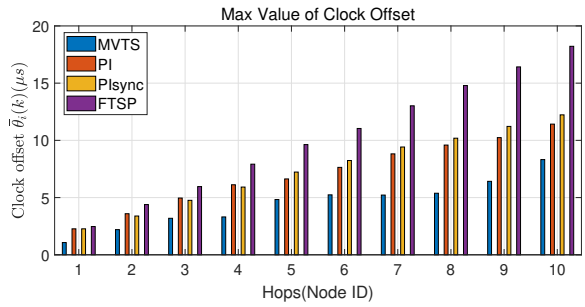


Fig. 9. Max value of clock offset (From left to right MVTS, PISync, PI, FTSP)

10 hops (which may represent a grid network of about 60 nodes), as illustrated in Fig. 7. Subsequently, we compared the performance of the four methods, collecting synchronization data over a duration of 200 seconds. Histograms depicting the average clock offset and maximum clock offset are presented in Fig. 8 and Fig. 9, respectively.

For the mean clock offset values depicted in Fig. 8, the MVTS algorithm exhibits a slow increase as the number of hops increases. In contrast, the FTSP algorithm demonstrates a rapid rise in the average clock offset. The order of average clock offset increase rate among the four methods is as follows: FTSP $>$ PISync $>$ PI $>$ MVTS. Regarding the histogram illustrating the maximum clock offset values in Fig. 9, the MVTS method appears to be the most effective. Consider a 10-hop node depicted in Fig. 9, where the maximum clock offsets for the four methods are observed as follows: FTSP$(18.82\mu s)$ $>$ PISync$(12.23\mu s)$ $\approx$ PI$(11.42\mu s)$ $>$ MVTS$(8.32\mu s)$. This indicates that the maximum synchronization error for the MVTS algorithm is 8.32 μs, which represents a reduction of 56% and 32%, respectively, compared to the baseline method FTSP and the recent PISync algorithm.

Additionally, the growth rate of maximum clock offset for the four methods follows the order: FTSP $>$ PISync $\approx$ PI $>$ MVTS. The results demonstrate that the MVTS algorithm proposed in this paper effectively reduces the rate at which synchronization errors increase with the number of hops.

In summary, we compared the synchronization performance of the FTSP [21], PI [7], PISync [25], and MVTS algorithms across point-to-point, grid, and line networks. Our analysis concludes that the MVTS algorithm proposed in this paper outperforms existing algorithms in both point-to-point and multi-hop networks. The effectiveness of time synchronization accuracy control follows the order: MVTS $>$ PISync $\approx$ PI $>$ FTSP.

VI. Conclusion

In this paper, we introduced a TDMA-based MVTS algorithm with output feedback consensus control for clock synchronization, aimed at minimizing time synchronization variance and improving synchronization accuracy. Theoretical analysis of the MVTS algorithm's performance in the network was also conducted. Experimental findings demonstrate that, in point-to-point networks, the MVTS algorithm achieves synchronization precision of $1\mu s$ on clocks based on 32MHz crystal oscillators. Furthermore, in multi-hop networks, the synchronization error variance of the MVTS algorithm increases at a rate half that of traditional algorithms such as FTSP.

References

[1] F. Liu, C. Jiang, and W. Xiao, "Multistep prediction-based adaptive dynamic programming sensor scheduling approach for collaborative target tracking in energy harvesting wireless sensor networks," *IEEE Transactions on Automation Science and Engineering*, vol. 18, no. 2, pp. 693–704, 2021.

[2] S. Gao, G. Y. Tian, X. Dai, M. Fan, X. Shi, J. Zhu, and K. Li, "A novel distributed linear-spatial-array sensing system based on multichannel lpwan for large-scale blast wave monitoring," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 9679–9688, 2019.

[3] S. Zhao, X.-P. Zhang, X. Cui, and M. Lu, "A new toa localization and synchronization system with virtually synchronized periodic asymmetric ranging network," *IEEE Internet of Things Journal*, vol. 8, no. 11, pp. 9030–9044, 2021.

[4] M. Khan, J. Seo, and D. Kim, "Modeling of intelligent sensor duty cycling for smart home automation," *IEEE Transactions on Automation Science and Engineering*, vol. 19, no. 3, pp. 2412–2421, 2022.

[5] K. Skiadopoulos, A. Tsipis, K. Giannakis, G. Koufoudakis, E. Christopoulou, K. Oikonomou, G. Kormentzas, and I. Stavrakakis, "Synchronization of data measurements in wireless sensor networks for iot applications," *Ad Hoc Networks*, vol. 89, pp. 47–57, 2019.

[6] L. Lo Bello and W. Steiner, "A perspective on ieee time-sensitive networking for industrial communication and automation systems," *Proceedings of the IEEE*, vol. 107, no. 6, pp. 1094–1120, 2019.

[7] Y. Zong, X. Dai, and Z. Gao, "Proportional-integral synchronization for nonidentical wireless packet-coupled oscillators with delays," *IEEE Transactions on Industrial Electronics*, vol. 68, no. 11, pp. 11 598–11 608, 2021.

[8] F. Shi, X. Tuo, S. X. Yang, J. Lu, and H. Li, "Rapid-flooding time synchronization for large-scale wireless sensor networks," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 3, pp. 1581–1590, 2020.

[9] S. Ding, Z. Wang, and X. Xie, "Periodic event-triggered synchronization for discrete-time complex dynamical networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 8, pp. 3622–3633, 2022.

[10] Y. Zong, X. Dai, Z. Wei, M. Zou, W. Guo, and Z. Gao, "Robust time synchronization for industrial internet of things by H_∞ output feedback control," *IEEE Internet of Things Journal*, vol. 10, no. 3, pp. 2021–2030, 2023.

[11] Z. Jia, X. Dai, D. Cui, and Y. Hu, "Asynchronous broadcast-based event-triggered control for discrete-time clock synchronization," *IET Control Theory & Applications*, vol. 17, no. 11, pp. 1543–1551, 2023.

[12] Z. Jia, X. Dai, D. Cui, F. Qin, D. Zhou, and Y. Hu, "General proportional integral observer (gpio) - based disturbance compensation for minimum variance time synchronization," *Journal of the Franklin Institute*, vol. 360, no. 8, pp. 5588–5608, 2023.

[13] N. R. Patel, S. Kumar, and S. K. Singh, "Energy and collision aware wsn routing protocol for sustainable and intelligent iot applications," *IEEE Sensors Journal*, vol. 21, no. 22, pp. 25 282–25 292, 2021.

[14] N. Cordeschi, F. De Rango, and M. Tropea, "Exploiting an optimal delay-collision tradeoff in csma-based high-dense wireless systems," *IEEE/ACM Transactions on Networking*, vol. 29, no. 5, pp. 2353–2366, 2021.

[15] M. Lévesque and D. Tipper, "A survey of clock synchronization over packet-switched networks," *IEEE Communications Surveys Tutorials*, vol. 18, no. 4, pp. 2926–2947, 2016.

[16] S. Zhuo, Z. Wang, Y.-Q. Song, Z. Wang, and L. Almeida, "A traffic adaptive multi-channel mac protocol with dynamic slot allocation for wsns," *IEEE Transactions on Mobile Computing*, vol. 15, no. 7, pp. 1600–1613, 2016.

[17] Y. Zong, X. Dai, S. Gao, P. Canyelles-Pericas, and S. Liu, "Pkcos: Synchronization of packet-coupled oscillators in blast wave monitoring networks," *IEEE Internet of Things Journal*, vol. 9, no. 13, pp. 10 862–10 871, 2022.

[18] S. Haider, G. Abbas, and Z. H. Abbas, "Vlcs: A novel clock synchronization technique for tdma-based mac protocols in vanets," in *2019 4th International Conference on Emerging Trends in Engineering, Sciences and Technology (ICEEST)*, 2019, pp. 1–6.

[19] X. Huan, H. He, T. Wang, Q. Wu, and H. Hu, "A timestamp-free time synchronization scheme based on reverse asymmetric framework for practical resource-constrained wireless sensor networks," *IEEE Transactions on Communications*, vol. 70, no. 9, pp. 6109–6121, 2022.

[20] X. Huan, W. Chen, T. Wang, H. Hu, and Y. Zheng, "A one-way time synchronization scheme for practical energy-efficient lora network based on reverse asymmetric framework," *IEEE Transactions on Communications*, vol. 71, no. 11, pp. 6468–6481, 2023.

[21] M. Maróti, B. Kusy, G. Simon, and Ákos Lédeczi, "The flooding time synchronization protocol," in *Proc International Conference on Embedded Network Sensor Systems*, 2004.

[22] J. Elson, L. Girod, and D. Estrin, "Fine-grained network time synchronization using reference broadcasts," *ACM SIGOPS Operating Systems Review*, vol. 36, 06 2002.

[23] L. Schenato and F. Fiorentin, "Average timesynch: A consensus-based protocol for clock synchronization in wireless sensor networks," *Automatica*, vol. 47, no. 9, pp. 1878–1886, 2011.

[24] R. Carli, A. Chiuso, L. Schenato, and S. Zampieri, "Optimal synchronization for networks of noisy double integrators," *IEEE Transactions on Automatic Control*, vol. 56, no. 5, pp. 1146–1152, 2011.

[25] K. S. Yıldırım, R. Carli, and L. Schenato, "Adaptive proportional-integral clock synchronization in wireless sensor networks," *IEEE Transactions on Control Systems Technology*, vol. 26, no. 2, pp. 610–623, 2018.

[26] F. Shi, S. X. Yang, X. Tuo, L. Ran, and Y. Huang, "A novel rapid-flooding approach with real-time delay compensation for wireless-sensor network time synchronization," *IEEE Transactions on Cybernetics*, vol. 52, no. 3, pp. 1415–1428, 2022.

[27] T. Yang, Y. Niu, and J. Yu, "Clock synchronization in wireless sensor networks based on bayesian estimation," *IEEE Access*, vol. 8, pp. 69 683–69 694, 2020.

[28] H. Wang, F. Yu, M. Li, and Y. Zhong, "Clock skew estimation for timestamp-free synchronization in industrial wireless sensor networks," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 1, pp. 90–99, 2021.

[29] Y.-E. Wang, J. Zhao, and B. Jiang, "Stabilization of a class of switched linear neutral systems under asynchronous switching," *IEEE Transactions on Automatic Control*, vol. 58, no. 8, pp. 2114–2119, 2013.

[30] Y. Zong, X. Dai, P. Canyelles-Pericas, Z. Gao, W. P. Ng, K. Busawon, and R. Binns, "Synchronization of packet coupled low-accuracy rc oscillator clocks for wireless networks," *IEEE Transactions on Wireless Communications*, vol. 22, no. 7, pp. 4843–4857, 2023.

[31] S. Schriegel and J. Jasperneite, "Investigation of industrial environmental influences on clock sources and their effect on the synchronization accuracy of ieee 1588," in *2007 IEEE International Symposium on Precision Clock Synchronization for Measurement, Control and Communication*, 2007, pp. 50–55.

[32] Q. Ling, W. Zheng, and H. Lin, "An iterative method for control gain design of multiagent systems with process noise," *IEEE Transactions on Control Systems Technology*, vol. 25, no. 5, pp. 1905–1911, 2017.
[33] M. C. D. Oliveira, J. C. Geromel, and J. Bernussou, "Extended H_2 and H_∞ norm characterizations and controller parametrizations for discrete-time systems," *International Journal of Control*, vol. 75, no. 9, pp. 666–679, 2002.
[34] K.-Y. Wang, A. M.-C. So, T.-H. Chang, W.-K. Ma, and C.-Y. Chi, "Outage constrained robust transmit optimization for multiuser miso downlinks: Tractable approximations by conic optimization," *IEEE Transactions on Signal Processing*, vol. 62, no. 21, pp. 5690–5705, 2014.
[35] X. Zhiqiang and F. Yue, "Precise time synchronization system for tdoa sound source localization," in *2021 IEEE 3rd International Conference on Civil Aviation Safety and Information Technology (ICCASIT)*, 2021, pp. 1107–1110.



Zhian Jia received the B.Eng. degree in Automation from Hefei University of Technology, HeFei, China, in 2020. and the M.Sc. degree in control science and engineering from Northeastern University, Shen Yang, China, in 2023.

He is currently working toward the Ph.D. degree in control science and engineering from the Huazhong University of Science and Technology, Wuhan. His current research interests include time synchronization, cooperative control and optimization of distributed network systems.



Dongliang Cui received the B.S. degree in electronic and information engineering from the HuaZhong University of Science and technology, Wuhan, China, in 2001, and the Ph.D. degree in mechanical design and manufacturing from Northeastern University, Shenyang, China, in 2013.

He is currently a Faculty Member with Northeastern University, His research interests include computer integrated manufacturing system, multi-objective optimization in complex product design, dispatch optimization, and timetable reschedule.



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

He is a Senior Lecturer with the Department of Mathematics, Physics and Electrical Engineering, Northumbria University, Newcastle upon Tyne, U.K. Prior to that he was a Postdoctoral Researcher with the Department of Engineering Science, University of Oxford (2011-2013)., and with the Department of Electronic and Electrical Engineering, University of College, London, U.K. (2009-2011). His research interests include robust state estimation, fault detection, networked control systems and synchronization, and their applications to the time-sensitive Industrial Internet of Things. Dr. Dai was awarded the Early Career Research Prize by SWIG U.K.



Zhi-Wei Liu (Senior Member, IEEE) received the B.S. degree in information management and information system from Southwest Jiaotong University, Chengdu, China, in 2004, and the Ph.D. degree in control science and engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2011.

He is currently a Professor with the School of Artificial Intelligence and Automation, Huazhong University of Science and Technology. His current research interests include cooperative control and optimization of distributed network systems.



Shuxin Liu received the B.Eng. degree in communication engineering from Dalian Jiaotong University, Dalian, China, in 2021. He is currently pursing 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 (IoT).



Tianyou Chai (Life Fellow, IEEE) received the Ph.D. degree in control theory and engineering from Northeastern University, Shenyang, China, in 1985.

He became a Professor with Northeastern University in 1988. He is the Founder and Director with the Center of Automation, which became a National Engineering and Technology Research Center and a State Key Laboratory. He has authored/coauthored 230 peer-reviewed international journal papers. His research interests include modeling, control, optimization, and integrated automation of complex industrial processes.

Prof. Chai is a Member of Chinese Academy of Engineering and IFAC Fellow. His paper titled Hybrid intelligent control for optimal operation of shaft furnace roasting process was selected as one of three best papers for the Control Engineering Practice Paper Prize for 2011 to 2013. He has developed control technologies with applications to various industrial processes. For his contributions, he was the recipient of four prestigious awards of National Science and Technology Progress and National Technological Innovation, the 2007 Industry Award for Excellence in Transitional Control Research from IEEE Multiple-conference on Systems and Control, and the 2017 Wook Hyun Kwon Education Award from Asian Control Association.