# Digital-Twin-Enabled Intelligent Distributed Clock Synchronization in Industrial IoT Systems

Pengyi Jia, *Member, IEEE*, Xianbin Wang, *Fellow, IEEE*, and Xuemin Shen, *Fellow, IEEE*

***Abstract*—Tight cooperation among distributively connected equipment and infrastructures of an Industrial-Internet-of-Things (IIoT) system hinges on low latency data exchange and accurate time synchronization within sophisticated networks. However, the temperature-induced clock drift in connected industry facilities constitutes a fundamental challenge for conventional synchronization techniques due to dynamic industrial environments. Furthermore, the variation of packet delivery latency in IIoT networks hinders the reliability of time information exchange, leading to deteriorated clock synchronization performance in terms of synchronization accuracy and network resource consumption. In this article, a digital-twin-enabled model-based scheme is proposed to achieve an intelligent clock synchronization for reducing resource consumption associated with distributed synchronization in fast-changing IIoT environments. By leveraging the digital-twin-enabled clock models at remote locations, required interactions among distributed IIoT facilities to achieve synchronization is dramatically reduced. The virtual clock modeling in advance of the clock calibrations helps to characterize each clock so that its behavior under dynamic operating environments is predictable, which is beneficial to avoiding excessive synchronization-related timestamp exchange. An edge-cloud collaborative architecture is also developed to enhance the overall system efficiency during the development of remote digital-twin models. Simulation results demonstrate that the proposed scheme can create an accurate virtual model remotely for each local clock according to the information gathered. Meanwhile, a significant enhancement on the clock accuracy is accomplished with dramatically reduced communication resource consumption in networks with different packet delay variations.**

***Index Terms*—Comprehensive clock model, digital twin, Industrial Internet of Things (IIoT), packet delay variation (PDV), resource saving, temperature compensated skew correction.**

## I. Introduction

THE ONGOING convergence of information and communication technologies (ICT) and vertical industry applications in the forms of advanced manufacturing, smart factories, and industry 4.0 will directly boost the efficiency, quality, and productivity of many industrial processes as well as reducing the corresponding production costs [1]. By integrating communications, networking, and computing technologies, future Industrial Internet of Things (IIoT) and cyber–physical systems (CPSs) will enable the ubiquitous connectivity and interactivity among large-scale industrial infrastructures [2], [3]. Assisted by the ongoing expansion and deployment of the fifth-generation (5G) wireless technologies, timely large-scale sensing and controlling information exchange among distributed physical infrastructures are becoming realities [4], leading to frequent interactions among the connected facilities. However, the involvement of a large number of devices and the associated massive data will cause unexpected latency, which hinders the intelligence of the distributed process in IIoT systems [5]. The long and nondeterministic network latency inevitably becomes a significant impediment for timely and efficient information exchange in distributed IIoT systems [6].

As one of the most critical requirements to guarantee efficient distributed interaction, accurate clock synchronization within an IIoT system empowers the collaborative synergy among distributed physical elements [7]. Without a consistent clock reference among the industrial infrastructures, data packets associated with misaligned time indices will be interchanged, leading to the inaccurate analysis of crucial data and decisions relying on stale data. However, the increasingly sophisticated industrial environment poses more challenges on clock synchronization among all involved entities. On the one hand, with the extensive development of ICT technologies, an increasing number of factories will rely on 5G wireless technologies as one of their fundamental ways to meet critical requirements on time-sensitive applications in the near future [8]. Consequently, a heterogeneous IIoT network based on wired standards, traditional wireless standards (e.g., wirelessHART), and arising 5G standards (e.g., URLLC) is emerging [9], leading to new synchronization challenges, including nondeterministic latencies and cross-standard synchronization for distributed nodes using various communication standards [10]. On the other hand, the dynamic industry operating environment of an IIoT system makes the synchronization process more challenging. As previously studied and validated [11], [12], clocks in complicated environments are susceptible to drift compared to those in moderate residential and office circumstances. The temperature-induced clock drifting inherent to inexpensive crystal oscillators will inevitably result in inconsistent clock outputs among IIoT

Manuscript received April 16, 2020; revised June 25, 2020, July 30, 2020, and September 21, 2020; accepted September 22, 2020. Date of publication October 6, 2020; date of current version March 5, 2021. This work was supported in part by the Natural Sciences and Engineering Research Council of Canada Discovery Program under Grant RGPIN-2018-06254, and in part by the Idea to Innovation Program under Grant I2IPJ 538563-19. *(Corresponding author: Xianbin Wang.)*

Pengyi Jia and Xianbin Wang are with the Department of Electrical and Computer Engineering, Western University, London, ON N6A 5B9, Canada (e-mail: pjia7@uwo.ca; xianbin.wang@uwo.ca).

Xuemin Shen is with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: sshen@uwaterloo.ca).

Digital Object Identifier 10.1109/JIOT.2020.3029131

devices, necessitating the situation-aware synchronization to fully address the susceptibility of distributed devices to the external influence.

Traditional synchronization techniques adopted in an IIoT environment, including precision time protocol (PTP) [13] and flooding time synchronization protocol (FTSP) [14], are typically achieved through frequent calibration of the target clock by minimizing the observed clock deviation from the time reference, instead of understanding the reason behind the ever-growing clock errors. The nondeterministic clock errors induced from the time-varying clock skews will necessitate a higher synchronization frequency to maintain clock consistency, leading to degraded synchronization efficiency with increased network resource consumption. Moreover, traditional point-to-point clock synchronization exclusively relies on the periodic exchange of timestamps, while its accuracy in large-scale IIoT systems will be deteriorated due to the uncontrollable issues incurred by intermediate communication processes, including suspectable Internet link quality, accumulated network latencies, and packet delay variations (PDVs). The expected performance of end-to-end network-wide synchronization will be significantly degraded in terms of the achievable clock accuracy and increment of network resource consumption with the extension of the network scale.

With the emergence of digital twin, a situation-aware and model-based distributed clock synchronization for IIoT systems can be enabled. By definition, digital twin is the digital counterpart of a physical infrastructure established through comprehensive observation, modeling, and simulation processes that can reflect the real-time relation between the physical entity and its virtual representation [15]. Consequently, digital twin is indispensable to bridge the gap between the physical world and the virtual world while enhancing the seamless integration between these two domains [16]. The cohesive physical-virtual collaboration exclusively hinges on the accurate and efficient clock synchronization among all physical and digital constituents in an IIoT system that couples with its digital twin equivalent. Conversely, the establishment of the digital twin of an IIoT system can enhance the clock synchronization performance by promoting the understanding of its physical entities. The behavior and real-time status of each clock in an IIoT synthesis can be thoroughly modeled and predicted by its digital counterpart so that more situation-aware synchronization protocols are attainable. By leveraging the strength of digital twin, the effect of the diversified operating environments and network conditions can be minimized.

Motivated by these considerations, a digital-twin-enabled distributed clock synchronization scheme is proposed in this article to tackle the previously mentioned synchronization challenges in IIoT systems. To be more specific, the primary contributions of this article comprise the following two aspects. First, an edge-cloud collaborative system architecture is proposed to establish digital-twin models. The data generated at each local device is efficiently processed at an edge device to minimize the effect of uncertain network latency on the timely analysis. Second, the characteristics of each clock are modeled by exploring the consecutive time information uploaded from the distributed devices, based on which a digital twin counterpart is established to predict the clock skew for each node under various operating environments. By leveraging the digital-twin-enabled clock skew estimation, the clock accuracy is improved with reduced consumption of network resources during the distributed clock synchronization.

The remainder of this article is organized as follows. Related work in terms of clock synchronization under network uncertainties, temperature-compensated skew correction, and digital twin-related designs are summarized in Section II. The overall physical system description, including the edge-cloud collaborative architecture and heterogeneous clock models is illuminated in Section III, while the realization of digital-twin-enabled clock synchronization (DTCS) is designed in Section IV, including information acquisition, model establishment, and clock skew correction. Simulations are carried out to demonstrate the effectiveness of the proposed scheme on the aspects of model validation, clock accuracy improvement, and network overhead reduction in Section V, followed by the conclusion in Section VI.

## II. Related Work

Clock synchronization in industrial environments has been widely investigated in recent years. In this section, related studies of clock synchronization considering network uncertainties and time-varying drifting issues are given first. Additionally, the recent advancements of digital twin in industrial applications are summarized in detail as well.

Many synchronization techniques have been developed to overcome the impact of uncertainty and asymmetrical network latency on clock synchronization accuracy. This effect is typically alleviated by designing more robust synchronization protocols, for example, using prioritized data transfer protocols [17] to guarantee the channel access for sensors with critical tasks so that the synchronization precision can be improved. However, those efforts generally depend on the feasibility of modifying communication protocols. In dealing with the random network latency, Leng and Wu [18] tried to jointly evaluate clock parameters and the network latency by proposing a low-complexity maximum-likelihood estimator. Meanwhile, a constant gain approach is designed [19] for consensus-based synchronization with random bounded communication delays. However, the effectiveness of these methods primarily relies on the accurate estimation and characterization of the random network latencies in the system.

Moreover, the time-varying clock drift for inexpensive oscillators due to defective manufacturing and dynamic environments is explored and analyzed in the literature. An intelligent-clustering mechanism is designed in [20] based on the continuous observation of the varying rate of skew from distributed nodes so that an appropriate synchronization frequency is assigned to each cluster for resource-saving. However, quantitative analysis of the clock skew is not sufficiently provided. According to the quadratic relation between temperature and oscillation frequency summarized in [11], a temperature-assisted and self-calibrated clock synchronization mechanism is proposed for sensor networks [21] by

dynamically compensating clock skews based on the working temperature. The improvement of clock accuracy depends on the assumption that the relation between its frequency and the corresponding temperature always follows the theoretical expression. A different scheme on environment-aware clock skew estimation and compensation is proposed for the synchronization in wireless sensor networks [22] by designing a Kalman filter to evaluate the impact of the operating temperature, which poses a more stringent requirement on the processing capability for each sensor node. Additionally, Cena *et al.* [23] proposed a neural network-based approach to explore the characteristics of the synchronization error, which is mainly affected by the external temperature. However, the feasibility of such a method is dubious due to the high cost of adopting the learning-based algorithm in distributed inexpensive devices. It also should be noted that none of these studies considered the effect of network latency on the clock relationship estimation and calibration, which motivated the design of the situation-aware clock synchronization protocol with dynamic network conditions.

Furthermore, as one of the enabling technologies in IIoT systems, digital twin has received increasing attention in the last few years. Digital twin can duplicate and imitate the physical characteristics of distributed entities to demonstrate the natural trend of each party based on historically collected data, which enables remote analysis, modeling and prediction of the future behavior for device of interest. Based on this observation, a digital twin-assisted fault diagnosis paradigm [24] is developed using the deep-learning methodology. A model with the high fidelity is designed in the virtual domain to train the learning model so that the dynamics of data is thoroughly traceable. Similarly, digital twin-based product design, manufacturing, and service provisioning with the assist of big data analysis are proposed in [25], although lacking a comprehensive demonstration of the data regulation and analysis. Inspired by the potential of digital twin that the data can be acquired and traced in a real-time manner, Zhou *et al.* [26] implemented an online analysis digital twin system to obtain a real-time solution for the power grid. These investigations indicate that digital twin is effective in dealing with the massive data on real-time acquisition, accurate analysis, and timely decision making.

## III. Cloud-Edge Collaborative Architecture and System Model

In this article, a three-tier cloud-edge collaborative architecture with heterogeneous physical clocks distributed in an IIoT system is adopted to form the foundation of the proposed DTCS.

### A. Overall System Architecture

The overall architecture of an IIoT system designed to maintain a desirable clock accuracy at every clock within the network is shown in Fig. 1, which is organized as a three-layer hierarchy, namely, cloud center, edge devices, and end nodes. In this section, the function of each layer and the interaction among them will be introduced in detail.

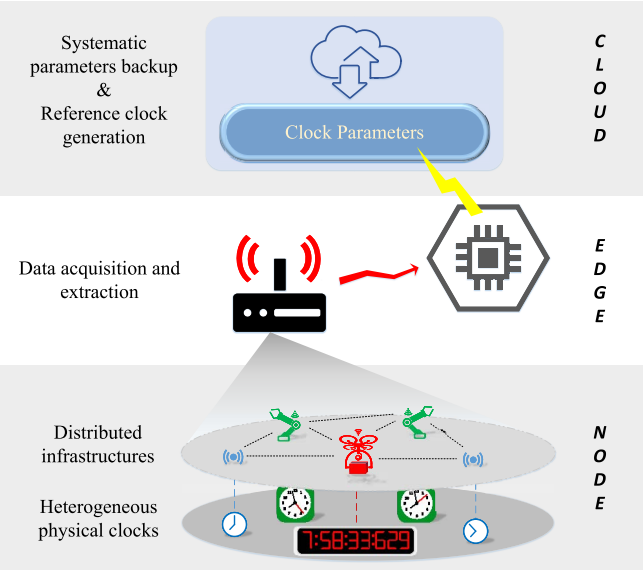



Fig. 1. Overall architecture of a physical system consists of a cloud center for data backup, some edge devices provide preliminary data processing capability, and end-devices with clocks required to be synchronized occasionally.

Node devices, at the most fundamental layer, are various classes of end-devices equipped with a clock of diverse conditions, which is required to be synchronized periodically regarding the time reference. All types of time-tracking devices, including sensors, actuators, controllers, schedulers, and unmanned vehicles, can be considered as node devices. All clocks in an IIoT system will function uniquely due to their diverse operating environments, inconsistent manufacturing specifications and packaging materials for the crystal oscillators, as well as their different capacities of receiving the time reference for local clock modification. Therefore, the clock of each facility will evolve with different drifting rates, and conduct an incoherent timestamp at any instant. This nondeterministic clock inaccuracy under complicated industrial operating and communication environments necessitates frequent and resource-consuming synchronization actions associated with massive time information conveyance for maintaining a global time consistency within the network.

Edge devices are intelligent devices with higher processing capacity assigned to accomplish preliminary and uncomplicated computation physically close to the node devices [27]. As the intermediate layer of the entire system, edge devices are responsible for collecting physical information, including timestamps and temperature, from the node devices, supervise relatively straightforward data processing, and convey the outputs to the cloud center for further computation or backup. As a consequence, edge devices can enhance the distributed intelligence during real-time raw data gathering and processing while alleviating the unsatisfactory impact of the significant network latency between node devices and the cloud center. Each edge device will be responsible for collecting and processing the clock information only from the node devices within its coverage so that excessive or unbalanced computation burden is avoided. Generally, edge devices

are gateways or IIoT devices assigned by the cloud center according to application-oriented requirements, including the sufficient processing capacity and appropriate physical distance, which can be investigated by adopting topology discovery algorithms [28].

The functions of a cloud center in the proposed scheme are threefold. Most fundamentally, the cloud center serves as the time reference, according to which every device in the network will synchronize its local clock. On recognizing the potential necessity of the clock calibration, the cloud center will disseminate this reference time to the entire system progressively. Meanwhile, the cloud center is responsible for the initial synchronization of the selected edge devices. The rest of this article will focus on the description of the digital-twin model establishment and clock synchronization between the edge and node devices under the assumption that a digital twin of each edge device is established at the cloud center, and its clock is accurately synchronized according to the same procedures. Furthermore, the cloud center acts as the data center of the overall system for data storage, management, and backup. The parameters obtained historically or updated periodically from the edge devices will be preserved in the cloud for potential long-term tracking and analysis.

The coherent collaboration among these three layers together with the efficient data exchange among all involved elements play critical roles in supporting the establishment of a situation-aware synchronization protocol. Consequently, the extensive understanding of the intrinsic parameters of each clock at distributed physical entities is indispensable for the accomplishment of the successive synchronization design.

### *B. Heterogeneous Clock Model*

In the node layer of the proposed hierarchical architecture, each end-device encloses a quartz-crystal oscillator-driven clock. For a large-scale IIoT system with the complicated operating environment, these clocks generally behave differently due to the existence of initial clock offsets and skews, which are mainly caused by the original manufacturing defect of their oscillators, as well as the successive clock drifting resulted from the effect of the external operating environment (e.g., temperature). Especially when an inexpensive oscillator embeds in each device for reducing the implementation cost, the inconsistent and varying drifting incurred will lead to long-term clock inaccuracy. To be more specific, the clock at node $i$ is a function of its time-varying clock skew $\alpha_i(t)$ caused by the dynamic frequency deviated from its dominated frequency and its constant clock offset $\beta_i$ incurred by the various initial phases. According to [12] and [29], the clock output at time instant $t$ can be written as

$$C_i(t) = \alpha_i(t)t + \beta_i. \tag{1}$$

Generally, the clock skew $\alpha_i(t)$ of an oscillator is a combination of its initial clock drift $\alpha_{i0}$ and a time-varying factor $\omega_i(t)$, so that the clock model can be further derived as

$$C_i(t) = (\alpha_{i0} + \omega_i(t))t + \beta_i. \tag{2}$$

The time-varying frequency-drifting of a node [i.e., $\omega_i(t)$] is originated from both the inherent defect and its congenial disposition, for instance, its susceptibility to the operating temperature. According to the correlation between temperature and oscillation frequency validated in [11], the real-time frequency of the crystal oscillator at node $i$ is defined as

$$f_i = f_{i0}\Big(1 + \eta_i(T(t) - T_{i0})^2\Big) \tag{3}$$

where $T(t)$ is the operating temperature at the instant $t$, while $f_{i0}$, $\eta_i$, and $T_{i0}$ are the dominated frequency, temperature sensitivity factor, and ideal operating temperature of the clock at node $i$, respectively. Therefore, it is reasonable to assume that the factor $\omega_i(t)$ will follow a quadratic relationship with the operating temperature $T(t)$ [21]. According to (3), the overall clock skew at the instant $t$ accumulated from its initial skew $\alpha_{i0}$ can be written as

$$\alpha_i(t) = \omega_i(t) + \alpha_{i0} = 1 + \eta_i(T(t) - T_{i0})^2 + \alpha_{i0}. \tag{4}$$

Consequently, the clock output at time $t$ can be derived by substituting (4) into (1), while the real-time clock error $e(t)$ regarding the absolute time $t$ is given by

$$e(t) = C_i(t) - t = \Big(\eta_i(T(t) - T_{i0})^2 + \alpha_{i0}\Big)t + \beta_i \tag{5}$$

which is affected by its initial inaccuracy as well as its intrinsic temperature-related parameters.

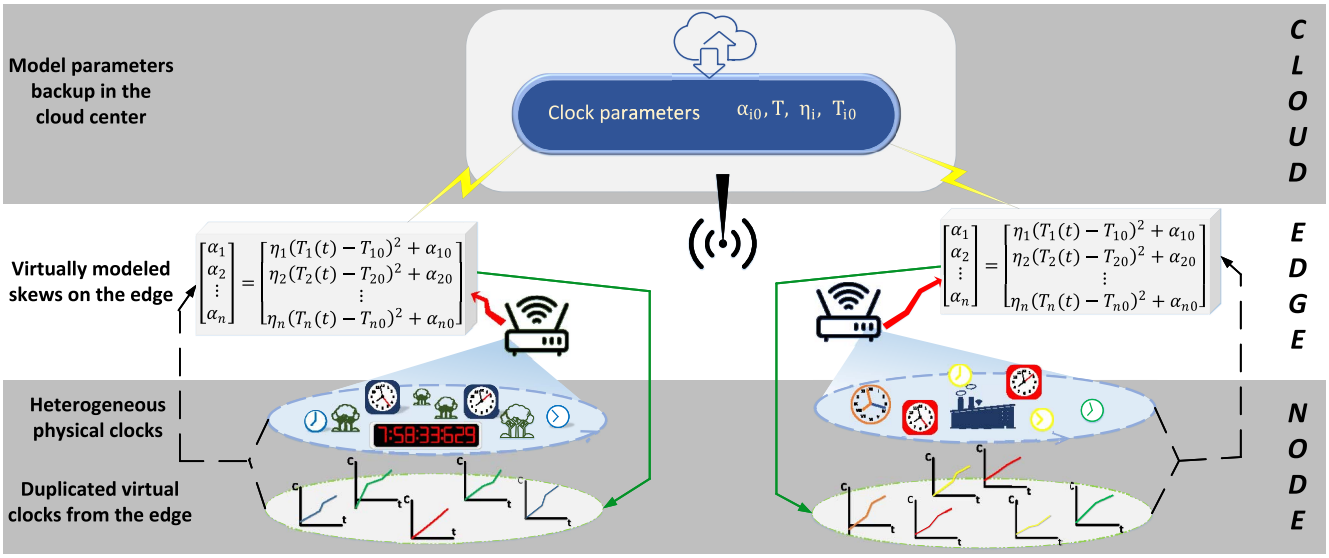
However, it is worth noting that none of these skew-related parameters for the distributed devices is identical, even if they are manufactured from the same batch. Meanwhile, this difference for heterogeneous entities in a IIoT synthesis will be even more critical, indicating that it is infeasible to derive a universal expression to represent all clocks. Especially when applications with extreme time-sensitivity are adopted in the IIoT system, the clock inconsistency incurred by the skew will be intolerable and fatal. Therefore, instead of adopting the general model given by (1), an appropriate approach to achieve parameter determination is of the utmost importance to further investigate an individualized model for each clock.

## IV. Digital-Twin-Enabled Clock Synchronization

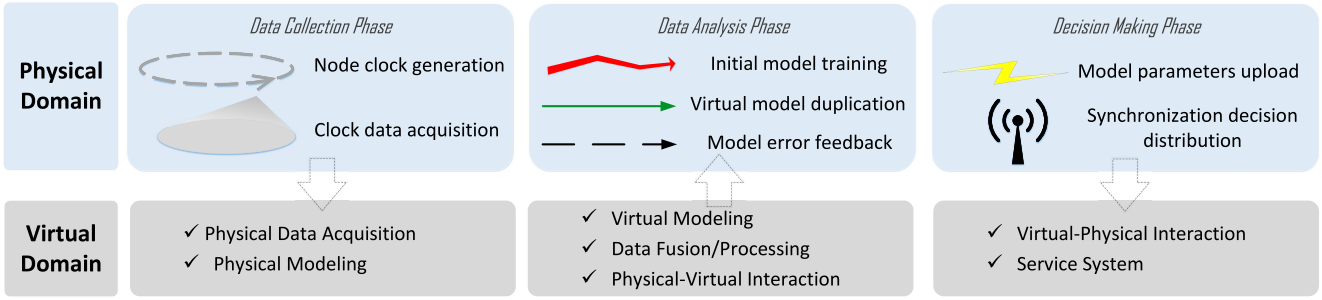
Based on the information from the heterogeneous clocks, a digital-twin system can be established to fully investigate their characteristics, which can be further used for clock synchronization. The expanded system architecture with the involvement of a digital-twin system is initially introduced.

### *A. Physical-Virtual-Collaborative System Design*

The seamless collaboration between the physical process and the virtual model is indispensable in realizing the timely data analysis and conscious decision making throughout its life cycle. In the proposed clock synchronization design, the physical elements extracted from the distributed infrastructures are a set of heterogeneous clock outputs, based on which models with various parameters are attainable. The physical-virtual-collaborated system architecture is shown in Fig. 2(a), which is an extension of the three-tier hierarchy introduced in Section III-A after initiating its digital twin counterpart.




(a)




(b)

Fig. 2. Expanded system architecture and the dataflow of the proposed DTCS scheme. (a) Cloud-edge-node hierarchy after inducing its virtual counterparts. (b) Icon description and the dataflow for the physical-virtual interaction during clock synchronization.

As previously stated in Section III-B, each device in the system is associated with an oscillator-driven clock, which generates a sequence of timestamps continuously. The timestamps given by each clock are always inconsistent, due to the heterogeneity of the enclosed oscillators. Each group of nodes will send the timestamps during the network initialization period continuously to their dedicated edge device, who is responsible for recording and investigating the behavior of these end-devices so that a virtual representative for each individual can be created in the digital twin domain. After conducting the initial digital model for all clocks within its coverage, the edge device will send a duplication of each model to every end-device for validating the correctness of the coefficients. After the training of several iterations based on the successive observation and feedback, a refined model can be established, while the model parameters will be transmitted to the cloud center as a backup. As shown in (5), the critical parameters required to be stored in the cloud center, including the initial clock skew $\alpha_i$, temperature sensitivity factor $\eta_i$, and its ideal working temperature $T_{i0}$. Meanwhile, the cloud center will make decisions on the reference time dissemination to guarantee the overall clock accuracy in the network.

The dataflow of the proposed digital twin-based clock synchronization scheme is shown in Fig. 2(b), where the entire dataflow comprises three successive phases, namely, data collection phase aiming at physical data acquisition, data analysis phase for the model establishment and its subsequent refinement, as well as the decision-making phase, which is responsible for finalizing the model parameters and calibrating clock offsets. Instead of organizing the devices into the cloud-edge-node hierarchy according to their classes, the overall dataflow is classified based on the intention of each interconnection among the entities involved. Moreover, those three phases are mapped into the digital twin domain with their corresponding components to illustrate their interactions.

A more detailed physical-virtual-collaborative procedure of the digital-twin model establishment between the edge device and its node devices is shown by the sequence diagram in Fig. 3, which consists of the three different phases. After the initialization period, all nodes in the network are orchestrated into a cloud-edge-node architecture with the assignment of the edge devices. The collaboration is initiated from the physical information collection at the local devices, where nodes will start to generate local timestamps under different temperatures. The local information consists of the generated timestamps

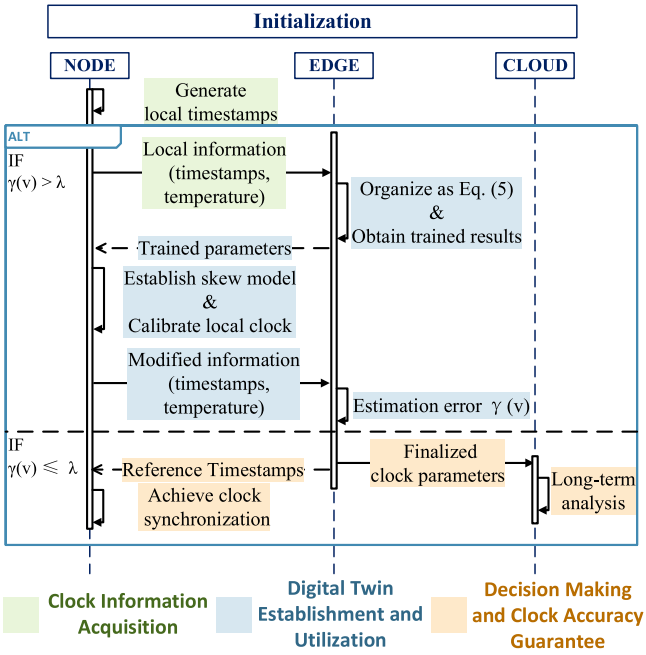



Fig. 3. Sequence diagram of the digital twin-based clock model establishment and skew synchronization between a node and its edge device in the proposed cloud-edge collaborative architecture.

and the corresponding temperatures will be successively transmitted to its superior edge device, which will be responsible for investigating the characteristics of each clock. Based on the obtained information, a digital-twin model will be formed in the node device, and corresponding feedback information will be delivered to the edge device for necessary update. The details of each phase will be introduced in the following sections.

### B. Clock Information Acquisition

As the external temperature has an apparent effect on the clock skew, it is meaningful to collect a sequence of timestamps under each operating temperature so that the relation between the clock variation and its temperature is discoverable. According to the observation that the temperature in an IIoT system typically varies gradually, a series of $l$ local clock timestamps at each node device is continuously generated at the same temperature. Meanwhile, a total number of $k$ distinct temperatures are selected for the relationship analysis. In other words, the node $i$ will deliver its local clock information as one timestamp to its edge device, while each timestamp is denoted in the form of $C_i(t_k^l)$, indicating the $l$th clock information obtained under the operating temperature $T_k$ at the node device $i$. After an apparent change of the operating temperature, each local device will update its operating temperature $T_k$ associated with the timestamps to its superior device for the successive relationship discovery purpose.

After the reception of each timestamp $C_i(t_k^l)$ from its subordinate end-device $i$, the edge device $E_j$ will record its local clock value $C_{ij}(t_k^l)$ for further calculation. As previously stated in Section III-A, we assume that all edge devices were already synchronized according to the cloud center with the same approach shown in Fig. 3. Therefore, once a packet is received at the edge device, its timestamp recorded is the addition of the absolute time that the packet is transmitted at the node device $i$ and the network latency inevitably induced between these two nodes, namely, $C_{ij}(t_k^l) = t_k^l + d_{ij}(t_k^l)$. The overall latency is assumed to be random and nondeterministic due to the existence of PDV inherent to the network uncertainties (e.g., access contention, channel asymmetry, and dynamic communication environment). The PDV is generally time-varying, manifested as the difference of delays when delivering two successive packets between the same pair of nodes. As the PDV dominates the uncertainty during timestamp calculation, larger PDV will lead to significantly degraded synchronization performance. The network delay can be denoted by $d_{ij}(t_k^l) = \bar{d}_{ij} \pm (1/2)\delta_{ij}$, where $\bar{d}_{ij}$ is the expectation of the delay between node $i$ and $E_j$, while $\delta_{ij}$ is the PDV affected by its communication protocol and network conditions, and its factor is adopted for further derivation simplicity. At the edge device $E_j$, the clock information delivered from one of its members $i$ can be recorded and regulated into a matrix, shown as

$$\hat{I}_{ij} = \begin{bmatrix} T_i(t_1) & T_i(t_2) & \cdots & T_i(t_k) \\ C_i(t_1^1) & C_i(t_2^1) & \cdots & C_i(t_k^1) \\ \vdots & \vdots & \ddots & \vdots \\ C_i(t_1^l) & C_i(t_2^l) & \cdots & C_i(t_k^l) \\ C_{ij}(t_1^1) & C_{ij}(t_2^1) & \cdots & C_{ij}(t_k^1) \\ \vdots & \vdots & \ddots & \vdots \\ C_{ij}(t_1^l) & C_{ij}(t_2^l) & \cdots & C_{ij}(t_k^l) \end{bmatrix} = \begin{bmatrix} \mathbf{T_i(t_k^l)} \\ \mathbf{C_i(t_k^l)} \\ \mathbf{C_{ij}(t_k^l)} \end{bmatrix} \tag{6}$$

which will be further used for the successive preprocessing and model training.

### C. Digital Twin Establishment and Utilization

The function of the data analysis layer in the proposed synchronization scheme can be classified into clock information preprocessing, initial clock model establishment, and feedback-based model update, which jointly develop a comprehensive understanding of each physical clock according to its time information (6) regulated at the superior device. Correspondingly, the data analysis layer can be mapped as data fusion, virtual modeling, and physical-virtual interaction in the digital twin domain.

*1) Clock Information Preprocessing:* The purposes of data fusion are twofold, namely, preliminarily alleviating the effect of random network latency and extracting critical information from the massive timestamps delivered. Based on (6), it is straightforward to obtain the clock error $e_i(t_k^l)$ at the time instant $t_k^l$ under the external temperature $T_k$ for the node $i$ as

$$e_i\left(t_k^l\right) = C_i\left(t_k^l\right) - C_{ij}\left(t_k^l\right). \tag{7}$$

By comparing (7) with (5), the clock error can be rewritten as

$$e_i\left(t_k^l\right) = \left(\eta_i(T_i(t_k) - T_{i0})^2 + \alpha_{i0}\right)t_k^l + \beta_i - d_{ij}\left(t_k^l\right). \tag{8}$$

Similarly, for the $(l+1)$th sampling under the same temperature $T_i(t_k)$, the clock error is given by

$$e_i\left(t_k^{l+1}\right) = \left(\eta_i(T_i(t_k) - T_{i0})^2 + \alpha_{i0}\right)t_k^{l+1} + \beta_i - d_{ij}\left(t_k^{l+1}\right). \tag{9}$$

By subtracting (9) from (8), the clock error evolution between the two consecutive samplings can be written as

$$\epsilon_i^l(T_k^i) = \left(\eta_i(T_i(t_k) - T_{i0})^2 + \alpha_{i0}\right)\tau_k + \left(d_{ij}\left(t_k^{l+1}\right) - d_{ij}\left(t_k^l\right)\right) \tag{10}$$

where $\tau_k$ is the fixed sampling interval between $t_k^l$ and $t_k^{l+1}$. By taking the average of $\epsilon_i^l(T_k^i)$ for all the $l$ samplings, the variation of clock errors can be obtained as

$$\hat{\epsilon}_i(T_k^i) = \left(\eta_i(T_i(t_k) - T_{i0})^2 + \alpha_{i0}\right)\tau_k \pm \delta_{ij} \tag{11}$$

which is the effect of the operating temperature on the clock outputs under the network uncertainty $\delta_{ij}$. According to (11), $\hat{\epsilon}_i(T_k^i)$ of node $i$ is quadratically related to its operating temperature, while the coefficients are discoverable from the consecutive observations in diverse environments. In summary, the critical information extraction based on the clock information recorded in (6) is given by

$$\begin{bmatrix} \mathbf{T_i}(\mathbf{t_k^l}) \\ \mathbf{C_i}(\mathbf{t_k^l}) \\ \mathbf{C_{ij}}(\mathbf{t_k^l}) \end{bmatrix} \rightarrow \begin{bmatrix} \mathbf{T_i}(\mathbf{t_k}) \\ \hat{\boldsymbol{\epsilon}}_\mathbf{i}(\mathbf{T_k^i}) \end{bmatrix}. \tag{12}$$

*2) Initial Model Training and Formation:* Based on the critical information extracted, the edge device will build a model for each of its subordinates to track the trend of its local clock, which is affected by the physical characteristics of its enclosed crystal oscillator, as shown in (11). Based on the error difference $\hat{\boldsymbol{\epsilon}}_\mathbf{i}(\mathbf{T_k^i})$ and its related temperature $\mathbf{T_i}(\mathbf{t_k})$ summarized in (12), a Vandermonde matrix can be established, which will result in a linear relationship, shown as

$$\begin{bmatrix} T_i^2(t_1) & T_i^1(t_1) & 1 \\ T_i^2(t_2) & T_i^1(t_2) & 1 \\ \vdots & \vdots & \vdots \\ T_i^2(t_k) & T_i^1(t_k) & 1 \end{bmatrix} \begin{bmatrix} \hat{p}_{i1} \\ \hat{p}_{i2} \\ \hat{p}_{i3} \end{bmatrix} = \begin{bmatrix} \hat{\epsilon}_i(T_1) \\ \hat{\epsilon}_i(T_2) \\ \vdots \\ \hat{\epsilon}_i(T_k) \end{bmatrix} \tag{13}$$

where the vector $\hat{\boldsymbol{\epsilon}}_\boldsymbol{i}$ is the system output correspondingly attained from the system input $\boldsymbol{T_i}$. Based on the observation, the polynomial coefficients $\boldsymbol{p_i}$ can be derived by solving

$$\hat{\boldsymbol{p}}_\boldsymbol{i} = \boldsymbol{T_i}^{-1}\hat{\boldsymbol{\epsilon}}_\boldsymbol{i} \tag{14}$$

from which the temperature-sensitivity factor $\hat{\eta}_i$, ideal operating temperature $\hat{T}_{i0}$, and the initial clock skew $\hat{\alpha}_{i0}$ can be estimated. These three coefficients will be distributed to each local device for further validation and updates.

Once the model coefficients are received at each local device, a virtual skew can be obtained as

$$\hat{\alpha}_i(t) = 1 + \hat{\eta}\left(T_i(t) - \hat{T}_{i0}\right)^2 + \hat{\alpha}_{i0} \tag{15}$$

which is the digital twin counterpart of the real skew at the local clock. This physical clock skew can be calibrated from dividing the local clock by the virtual skew, given by

$$\hat{C}_i(t) = \frac{C_i(t)}{\hat{\alpha}_i(t)} = \frac{\alpha_i(t)t + \beta_i}{\hat{\alpha}_i(t)} \tag{16}$$

where $\beta_i$ is the constant initial offset. A precise modeling of the clock skew $\hat{\alpha}_i$ (i.e., $\hat{\alpha}_i(t) = \alpha_i(t)$) will result in $\hat{C}_i(t) = t + \hat{\beta}_i$, indicating that the local clock at node $i$ will operate with the identical rate as the reference clock at its edge device with a tiny constant offset, which can be easily eliminated via any offset compensation process (OCP) once, for example, two-way packet exchange used in [20]. However, due to the network uncertainty $\delta_{ij}$ involved during clock modeling, the calibrated clock will operate with a slight deviation from its ideal scenario, leading to the necessity of successive model update according to the feedback information.

*3) Error Feedback and Model Update:* After the establishment of virtual clock models at distributed node devices, both the original and the calibrated clocks outputs will be recorded, while those timestamps will be periodically transmitted to their edge devices in improving the accuracy of the modeling. Each device will transmit its clock information in the same format of the initial training phase, while all the conveyed information will be recorded at the edge device as

$$\breve{I}_{ij}(t_k) = \begin{bmatrix} \mathbf{T_i}(\mathbf{t_k^l}) \\ \mathbf{C_i}(\mathbf{t_k^l}) \\ \hat{\mathbf{C}}_\mathbf{i}(\mathbf{t_k^l}) \\ \mathbf{C_{ij}}(\mathbf{t_k^l}) \end{bmatrix}. \tag{17}$$

Based on the feedback information received, the edge device will further update the model in the same procedures introduced above, while a group of feedback-based estimation of the coefficients $\breve{\boldsymbol{p}}_\boldsymbol{i}$ can be obtained as

$$\breve{\boldsymbol{p}}_\boldsymbol{i} = \boldsymbol{T_i}^{-1}\breve{\boldsymbol{\epsilon}}_\boldsymbol{i} \tag{18}$$

where $\breve{\boldsymbol{\epsilon}}_\boldsymbol{i}$ is the error evolution of the calibrated clock extracted from the feedback information $\hat{\mathbf{C}}_\mathbf{i}$. Therefore, an updated estimation of the coefficients in terms of temperature-sensitivity factor $\breve{\eta}_i$, ideal operating temperature $\breve{T}_{i0}$, and the initial clock skew $\breve{\alpha}_{i0}$ can be extracted with enhanced precision.

Since the edge device is aware of the two groups of coefficients obtained from both the training information and its corresponding feedback, a difference between these two groups of coefficients can be observed at the edge device for each training iteration. The estimation bias of the clock skew at the $v^{\text{th}}$ training iteration can be calculated as

$$\breve{\boldsymbol{\gamma}}_i(v) = \frac{\breve{\boldsymbol{p}}_i(v) - \hat{\boldsymbol{p}}_i(v)}{\hat{\boldsymbol{p}}_i(v)} \tag{19}$$

which can be evaluated as the evidence of the modeling credibility. The edge device will conduct continuous training processes to obtain more accurate coefficients until it reaches the upper bound of the training iterations, or the observed difference is smaller than an ideal threshold, indicating that the training results reach its convergence.

### D. Decision Making and Clock Accuracy Guarantee

After the accomplishment of the training process, each edge device will transmit the finalized parameters (i.e., $\breve{\eta}_i$, $\breve{T}_{i0}$, and $\breve{\alpha}_{i0}$) to both the cloud center and node devices for different purposes. On one hand, an updated digital twin counterpart of its physical skew will be established at each node device according to the delivered parameters, shown as

$$\breve{\alpha}_i(t) = 1 + \breve{\eta}_i\left(T_i(t) - \breve{T}_{i0}\right)^2 + \breve{\alpha}_{i0} \tag{20}$$

which will be used to achieve the situation-aware calibration of its local clock skew under different operating temperatures. A virtual clock model will be established at each node to generate time information that is calibrated by the estimated clock skew. For the ideal situation, the estimation of clock characteristics will be identical to its real behavior, indicating that the skew of each virtual clock is eliminated while the device will not require any synchronization action during its further operation. However, the modeling inaccuracy incurred by the network uncertainty will lead to defective correction at the distributed devices, so that further clock modification and model updates are necessary.

During the network operation stage, each node device will occasionally deliver its timestamps associated with other application-related messages to the edge device, which will be responsible for observing the difference between its local clock outputs and the conveyed timestamps. In the case that the clock difference is foreseeable to be larger than the accuracy requirement at one device, its edge device will be managed to eliminate the clock error by triggering the OCP at this subordinate. Since the OCP consumes critical communication and computation resources during network operation, a minimal triggering frequency of the OCP is desired.

On the other hand, a long-range analysis will be continuously executed at the cloud center based on the historical models trained at each edge device and the recently delivered timestamps from the distributed nodes. Another comprehensive training process illustrated in Fig. 3 will be conducted for the node only if its previous model stored in the cloud center presents a significantly different behavior, which violates its newly generated timestamps.

## V. Performance Evaluation

In this section, a series of simulations are carried out to evaluate the clock accuracy improvement and the network overhead incurred by adopting the proposed DTCS scheme. The modeling of an industrial environment and the distribution of local devices are introduced first.

### A. Temperature Modeling

In this simulation, a common industrial environment with different areas associated with diverse temperature variations is simulated in MATLAB. The outdoor temperature information is collected from real environment while the indoor data are generated by simulation. As shown in Fig. 4(a), 30 heterogeneous devices equipped with an oscillator-driven clock are randomly deployed in an industrial environment consists of four different temperature distributions, including the extreme and unpredictable outdoor environment where the temperature can be as low as $-40$ °C, well-controlled storage room with a constant temperature of $-5$ °C after its initialization, normal indoor areas with around 23 °C during daytime and around 16 °C during the night, as well as manufacturing space, which can reach 60 °C during its operation. The oscillation frequency of each clock will vary with a different extent in its environment, leading to nonidentical clock errors after a certain time.

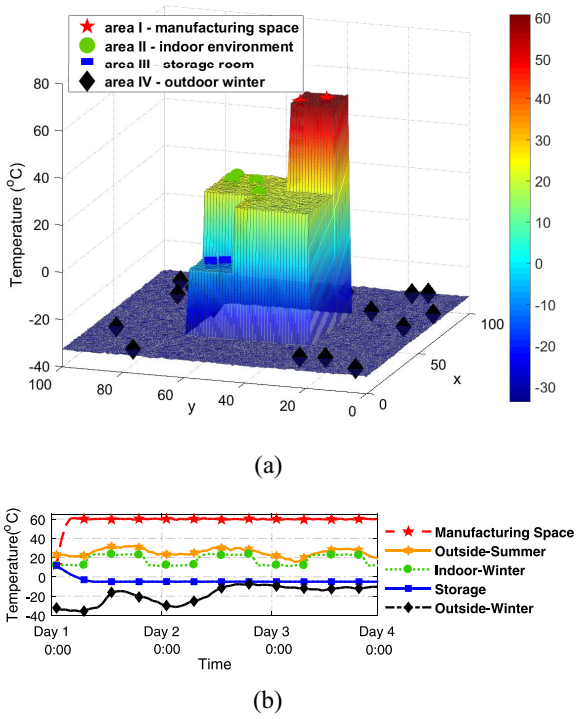



Fig. 4. Spatial and temporal distribution of the temperature in an IIoT environment with heterogenous devices. (a) Spatial distribution of the temperature in an industrial environment, where multiple facilities are deployed. (b) Temporal evolution of the temperature in different areas.

Meanwhile, the temperature in different areas will vary with time in its unique rate, as shown in Fig. 4(b). For instance, the fluctuation of outdoor temperature during both summer and winter periods will be much more nondeterministic, while temperatures in manufacturing and storage space will keep almost unchanged after their initialization. Due to the heterogeneous susceptibility of the distributed devices to the external environment, each clock will present a distinctive error, especially after long-term operation. Therefore, without an ideal approach to compensate for the effect of temperature, the overall coherence of the synchronization performance in the IIoT systems would be suspicious.

### B. Modeling Accuracy Evaluation

To deal with the effect of the temperature and enhance the understanding of the distributed clocks over the network, a unique skew model comprises three coefficients, namely, temperature sensitivity, ideal operating temperature, and initial clock skew, is created for each clock according to (15) and (20) for initial training and successive feedback, respectively. Due to the existence of network uncertainty, PDV incurred will inevitably cause observation error during parameter estimation. Furthermore, a larger PDV will naturally result in increasingly unpredictable variation between every two successive receptions of the timestamp, leading to more apparent randomness of the estimation error during both the training and calculation processes. According to the deployed network and different service contents provided by the IIoT system, dynamic PDVs fluctuating to a large extent will arise. In strictly controlled networks like guaranteed service transport [30], the

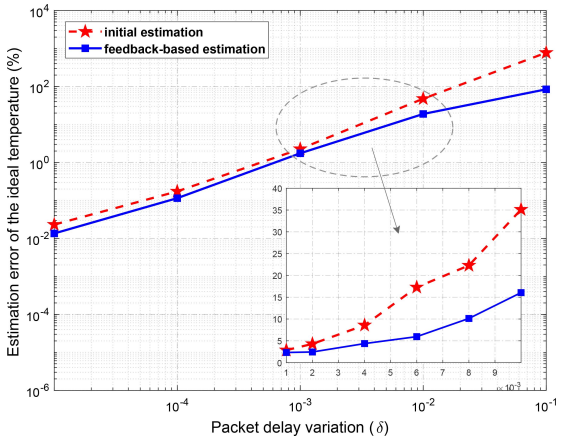



Fig. 5. Initial training and initial feedback of clock skew estimation are severely affected by the fluctuation of the network. The feedback-based estimation is more reliable with the increment of the network uncertainty.

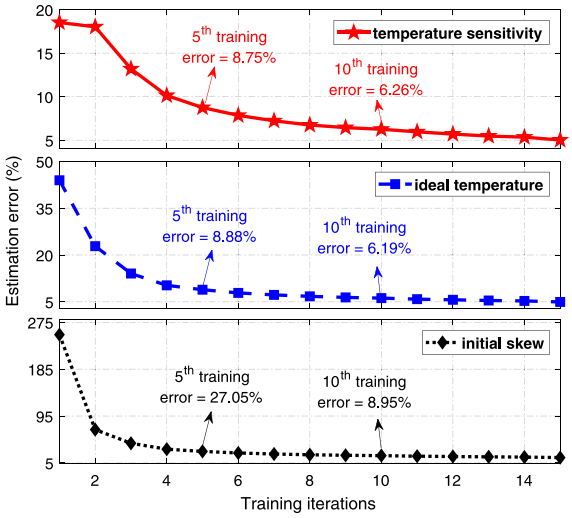



Fig. 6. Feedback-based coefficients estimation inaccuracy will decrease with the increment of training iterations, while an estimation error within 10% is achievable after the 10th iteration with a network that $\delta = 1 \times 10^{-2}$.

PDV can be eliminated by preserving dedicated communication resources for critical tasks. A small PDV is emerged in voice over LTE (VoLTE) supported by cellular networks with a PDV around $5 \times 10^{-5}$ s [31], while video-related services, including video over LTE and video conferencing services can lead to a larger PDV up to $2 \times 10^{-2}$ s [32]. Furthermore, as illustrated in [31], the amount of PDV can be as high as 0.1 s in some machine-type communication (MTC) scenarios.

Based on the above observation, simulations were conducted to validate the estimation accuracy under various PDV conditions ranging from $10^{-5}$ to $10^{-1}$, which can reveal most of the existing networks in practice. The estimation accuracy for the $v$th iteration is calculated by comparing the estimated clock coefficients with the ground truth value $\boldsymbol{p}_i$, given by

$$\boldsymbol{\gamma}_i(v) = \frac{\check{\boldsymbol{p}}_i(v) - \boldsymbol{p}_i}{\boldsymbol{p}_i}. \tag{21}$$

Taking the estimation of the ideal operating temperature $T_0$ as a representative, which is shown in Fig. 5, an accurate estimation is attainable when a well-controlled network (i.e., $\delta \leq 10^{-3}$) is employed. With the increment of PDV, the accuracy of the initial modeling and first-round feedback for $T_0$ are both severely degraded. A larger network uncertainty will even lead to over 100% initial estimation error, meaning that this estimation is not trustworthy. However, it can be observed from the enlarged linear inset that the feedback-based estimation can reduce the modeling inaccuracy from 35% to 15%, which will boost the precision of the model established. Similar observations are also found for the other two coefficients, while the feedback-based observations always increase the modeling accuracy significantly.

TABLE I
COMPARISON BETWEEN THE REAL CLOCK AND THE VIRTUAL CLOCK ESTABLISHED DURING THE OPERATION OF 10 MIN

| | *Indoor* | *Outdoor* | *Manuf.* | *Storage* |
|---|---|---|---|---|
| Averaged difference ($\mu s$) | 0.154 | 1.251 | 5.573 | 0.376 |
| Maximum difference ($\mu s$) | 1.210 | 3.760 | 7.806 | 0.617 |
| Minimum difference ($\mu s$) | 0.608 | 0.022 | 0.334 | 0.054 |

However, it can be observed that the initial training results still lack precision for the highly accurate clock synchronization purpose. Therefore, succeeding model training based on the continuously collected clock information is conducted to further decrease the error incurred during the model establishment. As shown in Fig. 6, an increasing training iteration will result in improved estimation accuracy of the three coefficients to different extents with the network uncertainty of $10^{-2}$. The averaged estimation error will drop from over 100% to under 15% within only five iterations. Moreover, the accuracy of the estimation on initial skew is improved the most dramatically compared to the other two parameters, due to its extreme estimation imprecision during initial training. It can be concluded that the feedback-based estimation can converge to an accurate result with an error of 7% after ten iterations in imperfect networks, while a much higher accuracy is obtainable more efficiently under better network conditions.

After obtaining accurate clock parameters, edge devices will transmit the coefficients to each corresponding local device, where a virtual skew model can be established accordingly. The modeling accuracy is exclusively related to the training results, while some differences can be observed due to modeling error. As shown in Table I, the difference between the real clock and the clock model established according to the virtual skew is calculated for different operating environments during the 10-min operation. Simulation results demonstrate that the clock errors are at the microsecond level, indicating that an extremely accurate virtual modeling of the clock skew is established for each local IIoT device. Consequently, the virtual skew can be further utilized for real-time skew correction with the assistance of the occasionally uploaded local temperatures.

### C. Clock Accuracy Improvement

As each clock under different environments varies with time at a distinctive rate, clock synchronization algorithms for

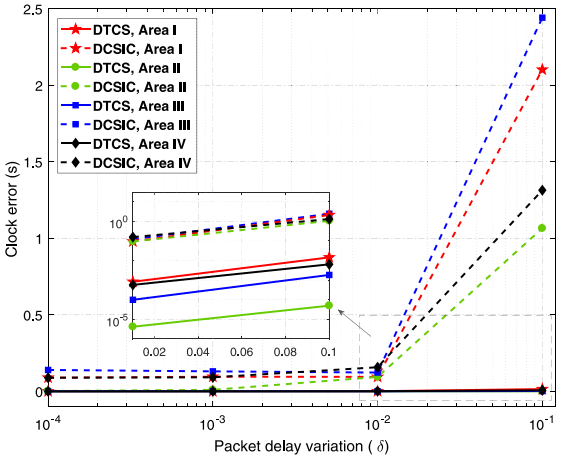



Fig. 7. Accumulated clock errors of 1 h in each environment will vary in different rates for both the proposed digital-twin-enabled method and the traditional calculation-based method.

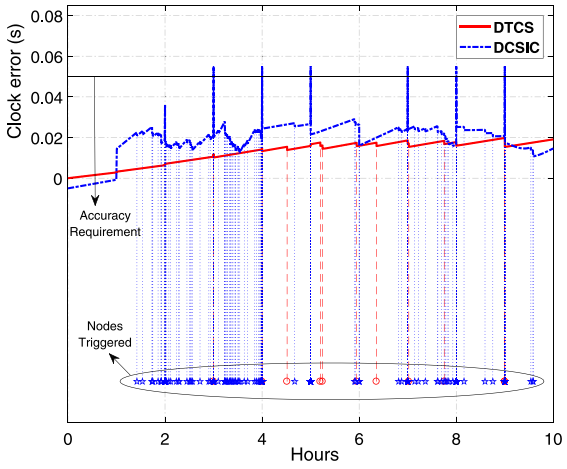



Fig. 8. Proposed synchronization scheme realizes a higher synchronization accuracy while requiring fewer actions compared to the DCSIC method to ensure the accuracy requirement for 10 h of long-term analysis.

the calibration of clock skews should be periodically adopted to ensure the overall clock accuracy within the network to be under the application-oriented requirements. Traditionally, synchronization methods adopting one-way or round-trip packet delivery are typically used in IoT systems, e.g., DCSIC [20] and MLE-based synchronization [33], where two successive messages containing timestamps will be exchanged between the reference node $i$ and each local node $j$ to calculate the clock skew for further correction. The corresponding estimation for the clock skew is based on four consecutive timestamps, which can be described as

$$\hat{\alpha}_e = \frac{C_j\big(t_2 + \delta_{ij}\big) - C_j(t_1)}{C_i\big(t_2 + \delta_{ij}\big) - C_i(t_1)} \tag{22}$$

where $\delta_{ij}$ is the random and variant PDV between the two nodes during different sequences of packet transmission. Moreover, another skew estimation method referred to as TACSC [21] was proposed in literature based on the experimental results, without considering the PDVs.

Both the traditional methods (i.e., DCSIC and TACSC) and the proposed DTCS approach are sensitive to the network uncertainty $\delta$, which has an everlasting effect on the performance of both the three schemes originally from the model initialization phase. Consequently, a simulation is conducted in four different operating areas to compare the performance of the DTCS scheme and the traditional methods under various network situations. As shown in Fig. 7, in a network of ideal conditions, the clock errors accumulated within 2 h are extremely tiny for all the four cases, while the clock errors in the proposed scheme are always slighter than the DCSIC scheme. Meanwhile, with the increment of PDV, the clock errors increase exponentially for the nodes adopting DCSIC. Specifically, when the network uncertainty exceeds $1\times10^{-2}$, the timestamps inaccuracy incurred during clock synchronization will affect the traditional skew calculation method significantly, however, the proposed DTCS approach is still able to obtain an accurate estimation with a much slighter increment of the clock error under worse network conditions, which is shown in the enlarged inset with the linear PDV and logarithmic clock error value.

Furthermore, since the proposed DTCS approach is aimed at achieving a better understanding of the environmental effect on the clock oscillators, another long-term observation of 10 h is conducted to evaluate its robustness under different operating temperatures. In this case, a synchronization requirement is set to be 50 ms so that any node violates such a threshold will trigger an OCP to eliminate its offset. It can be observed from Fig. 8 that, in the proposed scheme, the clock error increases much slower and smoother compared to the DCSIC scheme. In the meantime, there are 237 times that one of the devices was triggered in the traditional approach to guarantee the accuracy requirement, which is much higher than the proposed scheme of 11 times. Therefore, it can be conclude that both the achievable clock accuracy and the required actions in the DTCS scheme are dramatically improved.

### *D. Network Resource Consumption*

Besides the achievable clock accuracy empowered by different clock synchronization methods, a critical criteria to validate the synchronization algorithm is the network resource consumption during the network operation stage. In traditional synchronization methods (e.g., the DCSIC), both the skew correction and offset compensation are executed periodically to meet the synchronization accuracy. By contrast, in the proposed digital-twin-enabled skew estimation, due to the reason that the superior device has a better understanding of the clock skew at its subordinate devices, the network resource used for skew correction can be saved. Furthermore, since the clock skew estimated through the proposed algorithm is more accurate and environment-aware compared to the traditional approaches, the interval between two OCPs is significantly increased. As shown in Fig. 9, with a tighter synchronization requirement on the clock accuracy, an increasing number of packets is required for both the two scenarios. However, the total resource consumed by DTCS is always less than DCSIC, especially when the underlying network condition is poor, indicating that the proposed scheme outperforms traditional methods under diverse operating environments.

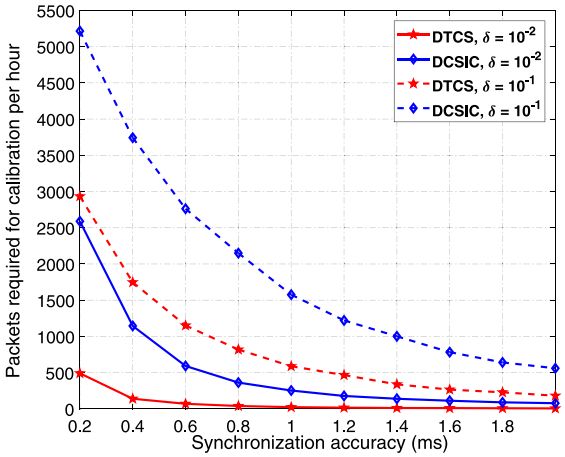



Fig. 9. Total number of packets required during clock synchronization for the proposed digital-twin-enabled scheme is always smaller than the DCSIC scheme, especially when the accuracy requirement is stringent.

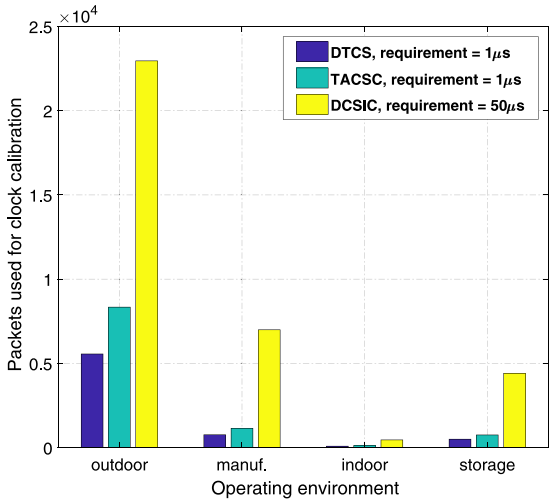



Fig. 10. Total number of packets used for clock calibration under different operating environments. Even with a tighter accuracy requirement, the proposed scheme still always requires fewer packets for calibration.

Moreover, the packets requirement for different operating environments is also evaluated by comparing the proposed scheme with the other two methods, one of which (i.e., TACSC) has a better understanding on the effect of the operating environment. As demonstrated in Fig. 10, the packets used for clock calibration in the digital-twin-enabled synchronization is always fewer than the other two approaches under different operating environments. Furthermore, the requirement for the proposed scheme is set to be 50 times of that in DCSIC scheme in a network with PDV equals $1 \times 10^{-5}$. Therefore, the overall packets used for clock calibration can be significantly reserved for other more critical applications.

Finally, it is worth noting that, the computation and communication resources of the proposed scheme is mainly consumed during the model training phase, which is achieved before the network operation. After the critical parameters of the digital-twin model for each clock are obtained, the proposed method can save resources significantly, and the tolerance against network uncertainties will be much stronger than traditional packet-based synchronization methods.

## VI. Conclusion

A digital-twin-enabled intelligent clock skew estimation and distributed synchronization approach had been proposed in this article to compensate for the effect of complicated environments on the heterogeneous oscillators in industrial IoT systems. A comprehensive digital twin model of each clock had been established by modeling the effect of external operating environments on the clock drift. By adopting the established virtual model at each device, the distributed local clock skews were compensated adaptively. Compared to the traditional clock synchronization approaches where clock skews are calculated frequently during network operation, the proposed method can avoid unnecessary and excessive packet exchange benefiting from a better understanding of clock behavior under different operating environments. The proposed digital-twin-enabled synchronization approach accomplished much higher clock accuracy with fewer packets required during network operation. Meanwhile, the performance improvements were even more significant under challenging network conditions, indicating that the proposed method is less sensitive to the PDV and the dynamic operating environments for maintaining higher clock accuracy.

## References

[1] Y. Liao, E. de Freitas Rocha Loures, and F. Deschamps, "Industrial Internet of Things: A systematic literature review and insights," *IEEE Internet Things J.*, vol. 5, no. 6, pp. 4515–4525, Dec. 2018.

[2] M. Wollschlaeger, T. Sauter, and J. Jasperneite, "The future of industrial communication: Automation networks in the era of the Internet of Things and industry 4.0," *IEEE Ind. Electron. Mag.*, vol. 11, no. 1, pp. 17–27, Mar. 2017.

[3] A. Humayed, J. Lin, F. Li, and B. Luo, "Cyber-physical systems security—A survey," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 1802–1831, Dec. 2017.

[4] S. Vitturi, C. Zunino, and T. Sauter, "Industrial communication systems and their future challenges: Next-generation Ethernet, IIoT, and 5G," *Proc. IEEE*, vol. 107, no. 6, pp. 944–961, Jun. 2019.

[5] M. Compare, P. Baraldi, and E. Zio, "Challenges to IoT-enabled predictive maintenance for industry 4.0," *IEEE Internet Things J.*, vol. 7, no. 5, pp. 4585–4597, May 2020.

[6] E. Sisinni, A. Saifullah, S. Han, U. Jennehag, and M. Gidlund, "Industrial Internet of Things: Challenges, opportunities, and directions," *IEEE Trans. Ind. Informat.*, vol. 14, no. 11, pp. 4724–4734, Nov. 2018.

[7] Y. Zou, H. Liu, and Q. Wan, "Joint synchronization and localization in wireless sensor networks using semidefinite programming," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 199–205, Feb. 2018.

[8] O. N. C. Yilmaz, Y.-P. E. Wang, N. A. Johansson, N. Brahmi, S. A. Ashraf, and J. Sachs, "Analysis of ultra-reliable and low-latency 5G communication for a factory automation use case," in *Proc. IEEE Int. Conf. Commun.*, London, U.K., Jun. 2015, pp. 1190–1195.

[9] L. Chettri and R. Bera, "A comprehensive survey on Internet of Things (IoT) toward 5G wireless systems," *IEEE Internet Things J.*, vol. 7, no. 1, pp. 16–32, Jan. 2020.

[10] A. Mahmood, M. I. Ashraf, M. Gidlund, and J. Torsner, "Over-the-air time synchronization for URLLC: Requirements, challenges and possible enablers," in *Proc. 15th Int. Symp. Wireless Commun. Syst. (ISWCS)*, Lisbon, Portugal, Aug. 2018, pp. 1–6.

[11] R. Sugihara and R. K. Gupta, "Clock synchronization with deterministic accuracy guarantee," in *Wireless Sensor Networks*. Heidelberg, Germany: Springer, 2011, pp. 130–146.

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

[13] *IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems*, IEEE Stdandard 1588-2008, pp. 1–300, 2008.

[14] K. S. Yildirim and A. Kantarci, "Time synchronization based on slow-flooding in wireless sensor networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 1, pp. 244–253, Jan. 2014.
[15] F. Tao, H. Zhang, A. Liu, and A. Y. C. Nee, "Digital twin in industry: State-of-the-art," *IEEE Trans. Ind. Informat.*, vol. 15, no. 4, pp. 2405–2415, Apr. 2019.
[16] Y. He, J. Guo, and X. Zheng, "From surveillance to digital twin: Challenges and recent advances of signal processing for industrial Internet of Things," *IEEE Signal Process. Mag.*, vol. 35, no. 5, pp. 120–129, Sep. 2018.
[17] S. Bhandari and X. Wang, "Prioritized clock synchronization for event critical applications in wireless IoT networks," *IEEE Sensors J.*, vol. 19, no. 16, pp. 7120–7128, Aug. 2019.
[18] M. Leng and Y. Wu, "Low-complexity maximum-likelihood estimator for clock synchronization of wireless sensor nodes under exponential delays," *IEEE Trans. Signal Process.*, vol. 59, no. 10, pp. 4860–4870, Oct. 2011.
[19] B. Wang and Y. Tian, "Time synchronization in WSNs with random communication delays: A constant gain design," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 657–662, 2017.
[20] P. Jia, X. Wang, and K. Zheng, "Distributed clock synchronization based on intelligent clustering in local area industrial IoT systems," *IEEE Trans. Ind. Informat.*, vol. 16, no. 6, pp. 3697–3707, Jun. 2020.
[21] Z. Yang, L. He, L. Cai, and J. Pan, "Temperature-assisted clock synchronization and self-calibration for sensor networks," *IEEE Trans. Wireless Commun.*, vol. 13, no. 6, pp. 3419–3429, Jun. 2014.
[22] F. Gong and M. L. Sichitiu, "Temperature compensated kalman distributed clock synchronization," *Ad Hoc Netw.*, vol. 62, pp. 88–100, Jul. 2017.
[23] G. Cena, S. Scanzio, and A. Valenzano, "A neural network clock discipline algorithm for the RBIS clock synchronization protocol," in *Proc. 14th IEEE Int. Workshop Factory Commun. Syst. (WFCS)*, Imperia, Italy, Jun. 2018, pp. 1–10.
[24] Y. Xu, Y. Sun, X. Liu, and Y. Zheng, "A digital-twin-assisted fault diagnosis using deep transfer learning," *IEEE Access*, vol. 7, pp. 19990–19999, 2019.
[25] F. Tao, J. Cheng, Q. Qi, M. Zhang, H. Zhang, and F. Sui, "Digital twin-driven product design, manufacturing and service with big data," *Int. J. Adv. Manuf. Technol.*, vol. 94, no. 9, pp. 3563–3576, Feb. 2018.
[26] M. Zhou, J. Yan, and D. Feng, "Digital twin framework and its application to power grid online analysis," *CSEE J. Power Energy Syst.*, vol. 5, no. 3, pp. 391–398, Sep. 2019.
[27] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, "Mobile edge computing: A survey," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 450–465, Feb. 2018.
[28] T. Yu, X. Wang, J. Jin, and K. McIsaac, "Cloud-orchestrated physical topology discovery of large-scale IoT systems using UAVs," *IEEE Trans. Ind. Informat.*, vol. 14, no. 5, pp. 2261–2270, May 2018.
[29] A. Mahmood, R. Exel, H. Trsek, and T. Sauter, "Clock synchronization over IEEE802.11—A survey of methodologies and protocols," *IEEE Trans. Ind. Informat.*, vol. 13, no. 2, pp. 907–922, Apr. 2017.
[30] R. Veisllari, S. Bjornstad, J. P. Braute, K. Bozorgebrahimi, and C. Raffaelli, "Field-trial demonstration of cost efficient sub-wavelength service through integrated packet/circuit hybrid network [invited]," *IEEE J. Opt. Commun. Netw.*, vol. 7, no. 3, pp. A379–A387, Mar. 2015.
[31] B. K. Al-Shammari, N. Al-Aboody, and H. S. Al-Raweshidy, "IoT traffic management and integration in the QoS supported network," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 352–370, Feb. 2018.
[32] C. Callegari *et al.*, "Experimental analysis of ViLTE service," *IEEE Access*, vol. 6, pp. 21129–21139, 2018.
[33] H. Wang, H. Zeng, M. Li, B. Wang, and P. Wang, "Maximum likelihood estimation of clock skew in wireless sensor networks with periodical clock correction under exponential delays," *IEEE Trans. Signal Process.*, vol. 65, no. 10, pp. 2714–2724, May 2017.

**Pengyi Jia** (Member, IEEE) received the B.S. degree in electronic information science and technology from Hebei University, Baoding, China, in 2014, and the M.Eng. degree in electrical and computer engineering from Western University, London, ON, Canada, in 2016, where he is currently pursuing the Ph.D. degree in electrical and computer engineering.

His current research interests include distributed clock synchronization, intelligent coordination of Internet-of-Things systems, digital twin, and networked control systems.

**Xianbin Wang** (Fellow, IEEE) received the Ph.D. degree in electrical and computer engineering from the National University of Singapore, Singapore, in 2001.

He was with Communications Research Centre Canada (CRC), Ottawa, ON, Canada, as a Research Scientist/Senior Research Scientist from July 2002 to December 2007. He is a Professor and the Tier 1 Canada Research Chair with Western University, London, ON, Canada. From January 2001 to July 2002, he was a System Designer with STMicroelectronics, Geneva, Switzerland. His current research interests include 5G and beyond, Internet of Things, communications security, machine learning, and intelligent communications. He has over 400 peer-reviewed journal and conference papers, in addition to 30 granted and pending patents and several standard contributions.

Prof. Wang has received many awards and recognitions, including the Canada Research Chair, CRC President's Excellence Award, the Canadian Federal Government Public Service Award, the Ontario Early Researcher Award, and six IEEE Best Paper Awards. He currently serves as an Editor/Associate Editor for IEEE TRANSACTIONS ON COMMUNICATIONS, IEEE TRANSACTIONS ON BROADCASTING, and IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY. He was also an Associate Editor of IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS from 2007 to 2011, and IEEE WIRELESS COMMUNICATIONS LETTERS from 2011 to 2016. He was involved in many IEEE conferences, including GLOBECOM, ICC, VTC, PIMRC, WCNC, and CWIT, in different roles, such as the Symposium Chair, the Tutorial Instructor, the Track Chair, the Session Chair, and the TPC Co-Chair. He is currently serving as the Vice Chair for IEEE London Section and the Chair for ComSoc Signal Processing and Computing for Communications Technical Committee. He is a Fellow of the Canadian Academy of Engineering, the Engineering Institute of Canada, and an IEEE Distinguished Lecturer.

**Xuemin (Sherman) Shen** (Fellow, IEEE) received the Ph.D. degree in electrical engineering from Rutgers University, New Brunswick, NJ, USA, in 1990.

He is currently a University Professor with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada. His research focuses on network resource management, wireless network security, Internet of Things, 5G and beyond, and vehicular *ad hoc* and sensor networks.

Dr. Shen received the R.A. Fessenden Award in 2019 from IEEE, Canada, the Award of Merit from the Federation of Chinese Canadian Professionals (Ontario) in 2019, the James Evans Avant Garde Award in 2018 from the IEEE Vehicular Technology Society, the Joseph LoCicero Award in 2015 and Education Award in 2017 from the IEEE Communications Society, the Technical Recognition Award from Wireless Communications Technical Committee in 2019 and AHSN Technical Committee in 2013, the Excellent Graduate Supervision Award in 2006 from the University of Waterloo, and the Premier's Research Excellence Award in 2003 from the Province of Ontario, Canada. He served as the Technical Program Committee Chair/Co-Chair for IEEE Globecom'16, IEEE Infocom'14, IEEE VTC'10 Fall, IEEE Globecom'07, and the Chair for the IEEE Communications Society Technical Committee on Wireless Communications. He was the elected IEEE Communications Society Vice President for Technical and Educational Activities, the Vice President for Publications, a Member-at-Large on the Board of Governors, the Chair of the Distinguished Lecturer Selection Committee, a Member of IEEE ComSoc Fellow Selection Committee. He was an Editor-in-Chief of IEEE INTERNET OF THINGS JOURNAL, IEEE NETWORK, and *IET Communications*. He is a Registered Professional Engineer of Ontario, Canada, a Fellow of the Engineering Institute of Canada, the Canadian Academy of Engineering, and the Royal Society of Canada, a Member of the Chinese Academy of Engineering Foreign, and a Distinguished Lecturer of the IEEE Vehicular Technology Society and Communications Society.