Asian Journal of Control, Vol. 21, No. 4, pp. 1–15, July 2019
Published online in Wiley Online Library (wileyonlinelibrary.com) DOI: 10.1002/asjc.1795

–Brief Paper–

A FILTER-BASED CLOCK SYNCHRONIZATION PROTOCOL FOR WIRELESS SENSOR NETWORKS

Wenlun Yang and Minyue Fu

ABSTRACT

Clock parameters in wireless sensors experience slow changes due to low-cost construction and environmental conditions. In this paper, a filter-based distributed protocol, called FBP, is proposed to dynamically achieve clock synchronization for wireless sensor networks. The idea of FBP is derived from a first-order filter which is robust to environmental noises. The proposed protocol is fully distributed, meaning that each node relies only on its local clock readings and reading announcements from its neighbouring sensor nodes. This will allow the proposed protocol applicable to large sensor networks. By applying FBP, the compensated clock skews can be bounded into a small steady-state error. Numerical simulations show that the proposed protocol yields better performances in both convergence property and synchronization accuracy.

Key Words: Wireless sensor networks (WSNs), clock synchronization, consensus control.

I. INTRODUCTION

Sensor networks are widely used for applications such as remote environmental monitoring, target tracking and genetic networks [1] where a common reference of time is often required [2,3]. Clock synchronization is a critical and rather basic requirement in mobile sensor networks for providing accurate time information of data collections and for energy conservation purposes. Synchronization protocols aim to synchronize local clocks and achieve a common reference of timescale among sensors in the network. Finding effective protocols with low overhead in communication and computation aspects still remains challenging.

Two kinds of clock synchronization protocols are commonly applied: tree-based and distributed [4]. A tree-based protocol establishes a hierarchical structure in the network, for example, assigns one node as a server (or master) and other nodes as clients. Typical examples include Reference Broadcast Synchronization (RBS) [5], Timing-sync Protocol for Sensor Networks (TPSN) [6] and Flooding Time Synchronization Protocol (FTSP) [7]. Due to high dependence on a specific predefined structure, tree-based protocols are often fragile to node failure and packet losses.

In contrast, distributed protocols do not require a master node and hence avoid a hierarchy structure. By adopting a fully distributed way, distributed protocols are robust to dynamic topology changes and become more scalable, especially for large networks. Typical examples include [8–11]. Consensus, which typically belongs to distributed protocols, has been widely used to design synchronization protocols as it can drive all agents finally reach a state of agreement based on locally available information [12]. We classify consensus-based synchronization algorithms into two categories: one is synchronous protocol with pseudo-synchronous implementation, *e.g.*[13–17]; the other is asynchronous protocol, also known as gossip-based protocol, *e.g.*[18–22]. Asynchronous protocol has faster convergence rate than synchronous ones. On the other hand, synchronous protocol can realize higher synchronization accuracy especially when physical clock parameters are time-varying [23]. Compared with asynchronous ones, synchronous protocols require concurrent updating process for every local node which is unrealistic before a common reference of time is reached. To tackle this problem, [15] proposes a realistic pseudo-synchronous implementation. Based upon proportional-integral controller, [16] comes up with an on-line adaptive strategy for the control parameters of the integral gain which could improve both steady state error and scalability. Reference [17] gives further analysis of the convergence result of [15] under three different communication scenarios: broadcast communication corresponding with pseudo-synchronous

Manuscript received December 28, 2016; revised September 16, 2017; accepted December 2, 2017.

W. Yang is with the State Key Laboratory of Industrial Control Technology and the College of Control Science and Engineering, Zhejiang University, Hangzhou 310027, China (e-mail: yangwenlun@zju.edu.cn).

M. Fu is with the School of Electrical Engineering and Computer Science, University of Newcastle, NSW 2308 Australia, and also with the State Key Laboratory of Industrial Control Technology and the College of Control Science and Engineering, Zhejiang University, Hangzhou 310027, China (corresponding author, e-mail: minyue.fu@newcastle.edu.au).

implementation, gossip communication corresponding with asynchronous implementation and hierarchical communication in wireless sensor networks.

However, most distributed protocols assume that the clock skews are either constant or they drift as a zero mean-noise [16], which is unrealistic in real applications. Typical sensor nodes are equipped with external crystal oscillators that are used as clock source for a counter register. These oscillators exhibit drift which is gradually changing depending on the environmental conditions such as ambient temperature, battery voltage and on oscillator aging. It is well known that typically an oscillator can drift up to 30–100 parts per million (ppm) [13], equating to $8.64s$ in 24 hours, assuming 1 MHz crystal frequancy [13]. As the clocks are subject to drift, adjusting the clock reading values only once is not enough as the clocks will be drifting away again later. Hence, one needs to apply the synchronization repeatedly [24].

In this paper, we study the clock synchronization problem with the focus on handling slow drift of time-varying clock parameters. We consider a fully distributed approach and develop a filter-based protocol (FBP) in both delay-free and random-delay cases. The main contributions of the proposed protocol are summarized as follows:

1. Improved synchronization precision under slowly time-varying clock parameters. The proposed FBP is in a synchronous form with pseudo-synchronous implementation. Comparison results between other two consensus-based algorithms indicate that FBP has improved the synchronization precision under slowly time-varying clock parameters.
2. Robustness to both multiplicative and additive noises. Synchronization protocols generally suffer from the noises (due to delays, measurements noises, quantization, etc) that enter the updating rules in both additive and multiplicative ways. By applying a filter-based approach, the proposed FBP has the advantage of being robust to both forms of noises. The convergence curve of skew compensation is smoother, resulting in a better performance in convergence rate.
3. Fully distributed property. FBP is fully distributed, meaning that each node relies only on its local clock readings, reading announcements from its neighbours and information that exchanges between neighbours. Due to its fully distributed property, it requires no global information. This property enlarges the scalability of the proposed protocol, thus suitable for large networks.

The remainder of this paper is organized as follows. In Section II, graph theory and a time-varying clock model for WSNs are introduced, together with the final goal of synchronization protocol. Section III mainly composes of three parts. Firstly, a low-pass filtering-based algorithm to estimate the relative clock skew is presented following the works in [18] and [19]. Secondly, a filter-based protocol for clock skew compensation is proposed. Thirdly the clock reading compensation protocol is put forward. Finally, we combine these three parts together and present the implementation of the proposed FBP. Simulation results are shown in Section IV. The conclusion of our work is given in Section V. The detailed proofs of Lemma 3.3 and Theorem 3.1 are given in the Appendix.

II. PRELIMINARIES

2.1 Notation

$\mathbb{R}$ denotes the set of real numbers and $\mathbb{Z}$ denotes the set of nonnegative integer numbers; $\boldsymbol{r}$ denotes n-dimensional vector of real numbers; $\mathbf{1}$ represents n-dimensional vector of ones and $\mathbf{0}$ represents n-dimensional vector of zeros; I_n indicates identity matrix with order n and O_n indicates zero matrix with order n.

2.2 Algebraic graph theory

An undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ consists of a non-empty node set $\mathcal{V} = \{1, 2, \cdots, n\}$ and an edge set $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. The neighbourhood $\mathcal{N}_i \in \mathcal{V}$ of the vertex v_i is the set $\{v_j \in \mathcal{V} | v_i v_j \in \mathcal{E}\}$, i.e, the set of all vertices that are adjacent to v_i. If $v_j \in \mathcal{N}_i$, it follows that $v_i \in \mathcal{N}_j$, since they are mutually adjacent to each other in a (undirected) graph. d_i denotes the cardinality of $\mathcal{N}_i$. $d_{max} = \max d_i, \forall i \in \mathcal{V}$. A path of length m in $\mathcal{G}$ is given by a sequence of distinct vertices:

$$v_{i_0}, v_{i_1}, \ldots, v_{i_m},$$

such that for $k = 0, 1, \ldots, m-1$, the vertices v_{i_k} and $v_{i_{k+1}}$ are adjacent. In this case, v_{i_0} and v_{i_m} are called the end vertices of the path; the vertices $v_{i_1}, \ldots, v_{i_{m-1}}$ are the inner vertices.

An undirected graph is called connected if for every pair of vertices in $\mathcal{G}$, there is a path that has them as its end vertices. For a connected undirected graph $\mathcal{G}$, the degree matrix $D(\mathcal{G})$ is defined as follows:

$$D(\mathcal{G})_{ij} = \begin{cases} d_i & i = j, \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

The adjacency matrix $A(\mathcal{G})$ is defined as follows:

$$A(\mathcal{G})_{ij} = \begin{cases} 1 & i \neq j, j \in \mathcal{N}_i, \\ 0 & \text{otherwise.} \end{cases}$$

The associated Laplacian matrix $L(\mathcal{G})$ is defined as follows:

$$L(\mathcal{G})_{ij} = D(\mathcal{G})_{ij} - A(\mathcal{G})_{ij} = \begin{cases} -1 & i \neq j, j \in \mathcal{N}_i, \\ 0 & i \neq j, j \notin \mathcal{N}_i, \\ d_i & i = j. \end{cases}$$

We now introduce a lemma in regard to the eigenvalue set of Laplacian matrices corresponding to a connected undirected graph $\mathcal{G}$ [25].

Lemma 2.1. For a connected undirected graph $\mathcal{G}$, the associated Laplacian matrix $L(\mathcal{G})$ is symmetric and positive semi-definite, and its eigenvalues can be ordered as

$$0 = \lambda_1(\mathcal{G}) < \lambda_2(\mathcal{G}) \leq ... \leq \lambda_n(\mathcal{G}).$$

2.3 Clock model

We start by modelling a physically time-varying clock model and stating the objective of a distributed synchronization protocol. Considering an integral clock model [26] for each local clock i:

$$\tau_i(t) = \int_0^t \alpha_i(t')dt' + \beta_i, \tau_i(0) = \beta_i, \tag{2}$$

where $\tau_i(t)$ is local clock reading of node i; $\alpha_i(t)$ is local clock skew (i.e, rate); β_i is local clock offset and t indicates absolute reference time. In practice, $\alpha_i(t)$ is a slowly time-varying variable. Assume that $\alpha_i(t)$ satisfies Assumption 2.1.

Assumption 2.1. Each local clock skew $\alpha_i(t)$ has an upper bound and lower bound:

$$1 - \rho_1 \leq \alpha_i(t) \leq 1 + \rho_1, \ \forall \ i \in \mathcal{V}, \tag{3}$$

where $0 < \rho_1 \ll 1$ indicates the maximum drift of the skew.

Our distributed synchronization protocol will be in discrete-time form. Hence the synchronization period is introduced and denoted by T. For the sake of simplicity, $\alpha_i(kT) = \alpha_i(k)$, $\forall\ k \in \mathbb{Z}$. Define $\Delta\alpha_i(k) = \alpha_i(kT + 1) - \alpha_i(kT)$ as the variation of α_i over one synchronization period. Another assumption concerning with $\Delta\alpha_i(k)$ is proposed:

Assumption 2.2. For any $k \in \mathbb{Z}$,

$$|\Delta\alpha_i(k)| \leq \rho_2, \tag{4}$$

where $0 < \rho_2 \ll 1$ is the bound of α_i's variation in one sampling period.

Assume a WSN composed of n sensor nodes, each of them equipped with its local clocks. The communication topology is modeled as a connected and undirected graph $\mathcal{G}$. $\mathcal{N}_i$ denotes the set of one-hop neighbours of node i in WSNs. An edge between node i and j in $\mathcal{G}$ indicates that they can communicate with each other by exchanging their information mutually.

The common virtual clock reading can be interpreted as:

$$\overline{\tau}(t) = \int_0^t \overline{\alpha}(t')dt' + \overline{\beta}. \tag{5}$$

Since $\alpha_i(t)$ experiences slow drift, it is unrealistic to let both virtual skew and offset converge exactly to certain values. Indeed, we try to bound the synchronization error of clock skew and the virtual clock reading as close as possible. As a result, the objective of a fully distributed synchronization protocol is to synchronize n local clocks with respect to a common virtual reference time as close as possible, namely, both synchronization errors of virtual clock skew and clock reading are within acceptable error range.

We consider a distributed linear updating protocol and each node i periodically keeps an update of its virtual clock reading $\hat{\tau}_i(t)$ based only on its own information and its neighbours' information

$$\hat{\tau}_i(t) = G_i(\hat{\alpha}_i(t)\tau_i(t), \tau_j(t)),$$

where $G_i(.)$ is a linear function depended on the information available at node i and from node $j \in \mathcal{N}_i$, that is, the local clock readings of clock i itself and its neighbour nodes $j \in \mathcal{N}_i$. $\hat{\alpha}_i(t)$ is the virtual clock skew compensation quantity. Node i's clock skew can be compensated by multiplying $\hat{\alpha}_i(t)$ with physical clock reading $\tau_i(t)$. Two cases are demonstrated to illustrate the final goals of synchronization protocol. By mentioning the goals of synchronization, we refers to synchronization rather than ordering of events, that is, to s et all clock displays to agreement and acquire a common notion of time [27].

Case 1. If α_i is constant, running synchronization protocol can drive the virtual clock skew and virtual clock reading for node i asymptotically converge to $\overline{\alpha}$ and $\overline{\tau}(t)$:

(i) $\lim_{t\to\infty} \hat{\alpha}_i(t)\alpha_i = \overline{\alpha}, \ \forall i \in \mathcal{V}.$

(ii) $\lim_{t\to\infty}(\hat{\tau}_i(t) - \overline{\tau}(t)) = 0,\ \forall i \in \mathcal{V}$.

Case 2. Define $\varepsilon_i(t) = \hat{\alpha}_i(t)\alpha_i(t) - \overline{\alpha}(t)$ where $\varepsilon_i(t)$ is called skew synchronization error for node i. If $\alpha_i(t)$ is slowly time-varying, the objective of synchronization protocol is to bound the synchronization error of compensated clock skews as

$$\lim_{t\to\infty} |\varepsilon_i(t)| \leq \rho_3,$$

where $0 < \rho_3 < \rho_1$ is the synchronization precision parameter of clock skew compensation, and it needs to be made as small as possible.

III. CLOCK SYNCHRONIZATION PROTOCOL

The proposed distributed protocol mainly includes three parts: relative clock skew estimation, clock skew compensation and clock reading compensation.

3.1 Relative clock skew estimation

Relative clock skew estimation aims to derive an algorithm to estimate the relative clock skew of each node i with respect to its neighbour node $j \in \mathcal{N}_i$. Some definitions are listed as follows.

Definition 3.1. The definition of relative clock skew for node i at time t is as follows:

$$\alpha_{ij}(t) = \frac{\alpha_j(t)}{\alpha_i(t)}. \tag{6}$$

The estimation of relative clock skew will be discussed in both delay-free and random-delay cases.

3.1.1 Relative clock skew estimation in delay-free case

We first consider the case without communication delays.

Assumption 3.1. Communication delays at all time instants are negligible.

Definition 3.2. $t_j(k)$ indicates the global time at which node j's clock reading $\tau_j(t_j(k))$ just reaches kT, where T is a common sampling period set as a default value.

Definition 3.3. $\tau_i(t_j(k))$ $(k \in \mathbb{Z},\ \forall j \in \mathcal{N}_i)$ indicates node i's local clock reading when node j announces that its local clock reading just reaches kT.

One thing to note is that local clock skew $\alpha_i(t)$ is neither known nor measurable by node i [18]. However, node i can acquire its relative clock skew estimation. Specifically, every node i tries to estimate relative clock skew $\alpha_{ij}(t_j(k))$ with respect to its neighbours $j \in \mathcal{N}_i$ at time instant $t_j(k)$. In delay-free case, if we take unavoidable measurement, quantization errors and small drift of clock skews into consideration, a low-pass filter introduced by [18] is as follows.

Initialization. $\hat{\alpha}_{ij}(0) = 1$.

Main Loop. At $t = t_j(k),\ k \in \mathbb{Z}$, the updating step of $\hat{\alpha}_{ij}(t^+)$ is

$$\hat{\alpha}_{ij}(t^+) = \rho\hat{\alpha}_{ij}(t^-) + (1-\rho)\frac{T}{\tau_i(t_j(k)) - \tau_i(t_j(k-1))}, \tag{7}$$

where $\rho \in (0,1)$ is a tuning parameter. t^+ and t^- represent, respectively, the right-hand limit and left-hand limit of t.

Lemma 3.1. [18]. For constant α_i, applying (7) yields the following convergence result in delay-free case

$$\lim_{t\to\infty} \hat{\alpha}_{ij}(t) = \alpha_{ij} \tag{8}$$

for any initial condition $\hat{\alpha}_{ij}(t)$.

3.1.2 Relative clock skew estimation in random-delay case

Now, we consider the case with communication delays.

Assumption 3.2. The communication delays at different time instants are now assumed to be positive random variables with constant mean μ and variance σ^2 and they are identically and independently distributed (i.i.d).

Some notation is listed as follows:

1. $t'_j(k)$ is the real broadcasting time at which node j's clock reading $\tau_j(t'_j(k))$ just reaches kT. At $t'_j(k)$, node j broadcasts its hardware clock reading $\tau_j(t'_j(k))$ to node i.
2. $t_j(k)$ indicates the real receiving time for node i. At $t_j(k)$, node i receives packets from node j and immediately records its hardware clock reading $\tau_i(t_j(k))$.
3. $d_k^{j\mapsto i} = t_j(k) - t'_j(k),\ k \in \mathbb{Z}$ is the communication delay from node j to node i. For different $d_k^{j\mapsto i}$s, they are mutually independent to each other.

In random-delay case, if we take unavoidable measurement, quantization errors and small drift of clock skews into consideration, a low-pass filter-based algorithm introduced by [19] is presented as follows.

Initialization. $\hat{\alpha}_{ij}(0) = 1$. Set a common broadcast T to each node.

Main Loop. At $t = t_j(k),\ k \in \mathbb{Z}$, the updating step of $\hat{\alpha}_{ij}(t^+)$ is

$$\begin{cases} \beta_{ij}(t) = \frac{T}{\tau_i(t_j(k)) - \tau_i(t_j(k-1))}, \\ \hat{\alpha}_{ij}(t^+) = \frac{\beta_{ij}(t) + (k-1)\hat{\alpha}_{ij}((t-1)^+)}{k},\ k \in \mathbb{Z}. \end{cases} \tag{9}$$

Lemma 3.2. [19]. For the case of constant clock skew α_i, applying (9) yields the following mean square convergence result in random-delay case

$$E\{\hat{\alpha}_{ij}(k)\} = \alpha_{ij} \tag{10}$$

and

$$\lim_{k\to\infty} Var\{\hat{\alpha}_{ij}(k)\} = 0. \tag{11}$$

The relative skew $\alpha_{ij}(t)$ is also slowly time-varying. As a result, (7) and (9) hold approximately under slowly time-varying clock skew. In the following we put forward a clock skew compensation protocol using relative skew algorithm proposed in (7) and (9).

3.2 Clock skew compensation

Clock skew compensation is the preliminary step of synchronization protocol. It constrains the synchronization error of compensated clock skew into a bounded range, which can guarantee long-term reliability of synchronization and reduce the number of re-synchronization frequencies. Consider a WSN of n sensors carrying their own local clocks. At time instant $k \in \mathbb{Z}$ each sensor node $i \in \mathcal{V}$ updates its own clock skew compensation quantity $\hat{\alpha}_i(k)$ using only the local information of their neighbouring sensor nodes. We propose the filter-based clock synchronization protocol of skew compensation as follows.

Initialization. $\hat{\alpha}_i(0) = 1,\ \omega_i(0) = 0,\ \forall i \in \mathcal{V}$.

Main Loop. At $t = t_j(k),\ k \in \mathbb{Z}$, $\hat{\alpha}_i(t^+)$ is updated as follows:

$$\begin{cases} \hat{\alpha}_i(t^+) = \hat{\alpha}_i(t^-) - T \sum\limits_{j\in\mathcal{N}_i} (\omega_i(t^-) - \omega_j(t^-)\hat{\alpha}_{ji}(t^-)), \\ \omega_i(t^+) = (1 - T\gamma)\omega_i(t^-) + T \sum\limits_{j\in\mathcal{N}_i} (\hat{\alpha}_i(t^-) - \hat{\alpha}_j(t^-) \\ \hat{\alpha}_{ij}(t^-)), \end{cases} \tag{12}$$

where $\omega_i(t^-) \in \mathbb{R}$ is called intermediate estimator state; $\hat{\alpha}_{ij}(t^-)$ is calculated in (7) or (9); γ is the information rate that measures the proportion of how much new information is introduced; T is discrete-time sampling period as discussed before; certain conditions of γ and T should be met to guarantee stability and convergence of the whole system, as depicted in Theorem 3.1.

By replacing t^+ or t^- with a common discrete time point k from a perspective of global clock, we get the algorithm for the protocol of (12) as follows:

Algorithm 1 (Clock skew compensation Protocol)

Input: $\hat{\alpha}_{ij}(k)$ for $i \in \mathcal{V}$.
Output: $\hat{\alpha}_i(k)$ for $i \in \mathcal{V}$ and $j \in \mathcal{N}_i$.

1: Initialize $\hat{\alpha}_i(0) = 1,\ \omega_i(0) = 0,\ \forall i \in \mathcal{V}$.
2: **while** 1 **do**
3: $\hat{\alpha}_i(k) \Leftarrow \hat{\alpha}_i(k) - T\sum_{j\in\mathcal{N}_i}(\omega_i(k) - \omega_j(k)\hat{\alpha}_{ji}(k))$ at $t = t(k)$.
4: $\omega_i(k) \Leftarrow (1 - T\gamma)\omega_i(k) + T\sum_{j\in\mathcal{N}_i}(\hat{\alpha}_i(k) - \hat{\alpha}_j(k)\hat{\alpha}_{ij}(k))$ at $t = t(k)$.
5: $\omega_i(t) = \omega_i(k),\ \hat{\alpha}_i(t) = \hat{\alpha}_i(k), t \in [k, k+1)$.
6: $\omega_i(k+1) = \omega_i(k),\ \hat{\alpha}_i(k+1) = \hat{\alpha}_i(k)$ at $t = t(k+1)$.
7: **end while**

Lemma 3.3 gives the upper bounds for both $\hat{\alpha}_i(k)$ and $\omega_i(k)$ when applying Algorithm 1.

Lemma 3.3. Under Assumptions 3.1, protocol (12) leads to the boundedness of $\lim_{k\to\infty} \hat{\alpha}_i(k)$ and $\lim_{k\to\infty} \omega_i(k)$ by choosing:

(i) $0 < T < \frac{-2}{\lambda_{2bn}}$,
(ii) $\gamma > 0$,

where λ_{2bn} is defined as the minimum eigenvalue of B_0 and

$$B_0 = \begin{bmatrix} O_n & -(D(\mathcal{G}) - \Lambda(k)^T) \\ D(\mathcal{G}) - \Lambda(k) & -\gamma I_n \end{bmatrix}.$$

$\Lambda(k)$ is defined as

$$\Lambda(k)_{ij} = \begin{cases} \hat{\alpha}_{ij}(k) & i \neq j, j \in \mathcal{N}_i, \\ 0 & \text{otherwise.} \end{cases}$$

Moreover, $\hat{\alpha}_i(k)$ and $\omega_i(k)$ are uniformly bounded for all $k \in \mathbb{Z}$, that is:

$$|\hat{\alpha}_i(k)| \leq \hat{\alpha}_{sup}, |\omega_i(k)| \leq \omega_{sup}, \ \forall k \in \mathbb{Z},$$

where $\hat{\alpha}_{sup} > 0$ and $\omega_{sup} > 0$ respectively denote the upper bounds for $\hat{\alpha}_i(k)$ and $\omega_i(k)$.

Proof. The proof is given in the Appendix.

Our main result for delay free case is presented in Theorem 3.1. It shows that (12) will lead to the boundedness of $\lim_{k\to\infty} |\varepsilon_i(k)|$.

Theorem 3.1. Consider the communication topology of WSNs to be a connected and undirected graph $\mathcal{G}$. If Assumptions 2.1, 2.2, 3.1 hold, (12) leads to the boundedness of $\varepsilon_i(k)$ as

$$\lim_{k\to\infty} |\varepsilon_i(k)| \leq \rho_3 = \left|\frac{1+T\lambda_i}{T\lambda_i}\right| \left(\frac{4\rho_1 T\omega_{sup} d_{max}}{1+\rho_1} + \rho_2 \hat{\alpha}_{sup}\right)$$

by choosing:

(i) $0 < T < \min\left\{\frac{-2}{\lambda_{2n}}, \frac{-2}{\lambda_{2bn}}\right\}$,
(ii) $\gamma > 0$,

where λ_i is defined as the ith eigenvalue of A_0 where

$$A_0 = \begin{bmatrix} O_n & -L(\mathcal{G}) \\ L(\mathcal{G}) & -\gamma I_n \end{bmatrix},$$

and λ_{2n} is defined as the minimum eigenvalue of A_0. λ_{2bn} is defined as the minimum eigenvalue of B_0 defined in Lemma 3.3.

Proof. The proof is given in Appendix.

Remark 3.1. As shown in the proof, even in the presence of both multiplicative and additive but exponentially decaying noises, synchronization error of virtual clock skew will eventually converge to a certain bound ρ_3 exponentially fast.

This protocol design is inspired by a continuous filter-based approach which is initially presented in the formation maneuvering control [28]. The introduction of auxiliary variable ω_i is to construct the first-order filter in order to enhance the robustness to noises. Drawn from the error form of $\varepsilon_i(k)$, under constant clock skew α_i, $\varepsilon_i(k)$ converges exponentially to zero as $k \to \infty$.

In the random-delay case, a similar result can be obtained.

Theorem 3.2. Consider the communication topology of WSNs to be a connected and undirected graph $\mathcal{G}$. If Assumption 3.2, 2.1, 2.2 hold, by applying (12) $|\varepsilon_i(k)|$ converges to exponentially to a ball at the origin of radius ρ_3 in mean square sense:

1. $\rho_3 = \left|\frac{1+T\lambda_i}{T\lambda_i}\right| \left(\frac{4\rho_1 T\omega_{sup} d_{max}}{1+\rho_1} + \rho_2 \hat{\alpha}_{sup}\right)$,
2. $\lim_{k\to\infty} E\{|\varepsilon_i(k)|\} \leq \rho_3$,
3. $\lim_{k\to\infty} Var\{|\varepsilon_i(k)|\} = 0$,

by choosing:

(i) $0 < T \min\{\frac{-2}{\lambda_{2n}}, \frac{-2}{\lambda_{2bn}}\}$,
(ii) $\gamma > 0$,

where λ_i is defined as the ith eigenvalue of A_0 where

$$A_0 = \begin{bmatrix} O_n & -L(\mathcal{G}) \\ L(\mathcal{G}) & -\gamma I_n \end{bmatrix},$$

and λ_{2n} is defined as the minimum eigenvalue of A_0. λ_{2bn} is defined as the minimum eigenvalue of B_0 defined in Lemma 3.3.

The proof of Theorem 3.2 follows similar to that of Theorem 3.1 and it is thus omitted.

Remark 3.2. As the proposed protocol (12) is in a synchronous form which is unrealistic, we use a pseudo-synchronous implementation [15] to implement (12).

3.3 Clock reading compensation

At the end of clock skew compensation procedure, the synchronization errors of virtual clock skews are bounded by ρ_3 under slowly time-varying input $\alpha_i(t)$s. The next step is to compensate for possible clock reading errors. Clock reading compensation mainly focuses on reducing possible errors of virtual clock reading $\hat{\tau}_i(t)$. The clock reading compensation protocol presented in Algorithm 2 can be integrated into skew compensation.

Algorithm 2 Clock reading compensation protocol

Input: $\hat{\alpha}_i(k)$ for $i \in \mathcal{V}$.
Output: $\hat{\tau}_i(k)$ for $i \in \mathcal{V}$.

1: Initialize $\hat{\tau}_i(0) = \hat{\alpha}_i(0)\tau_i(0), \ \forall i \in \mathcal{V}$.
2: **while** 1 **do**
3: $\hat{\tau}_i(k) \Leftarrow \frac{\hat{\tau}_i(k)+\sum_{j\in\mathcal{N}_i} \hat{\tau}_j(k)}{d_i+1}$ at $t = t(k)$.
4: $\hat{\tau}_i(t) \Leftarrow \hat{\tau}_i(k) + \hat{\alpha}_i(k)(\tau_i(t) - \tau_i(k)), t \in [t(k), t(k+1))$.
5: $\hat{\tau}_i(k+1) \Leftarrow \hat{\tau}_i(k) + \hat{\alpha}_i(k)(\tau_i(k+1) - \tau_i(k)) \ \forall i \in \mathcal{V}$ at $t = t(k+1)$.
6: **end while**

At $t = t(k)$, node i updates its virtual clock reading $\hat{\tau}_i(k)$ by taking the average of the sum of $\hat{\tau}_i(k)$ and $\hat{\tau}_j(k)$, $j \in \mathcal{N}_i$ and then it is added by $\hat{\alpha}_i(k)(\tau_i(t)-\tau_i(k))$, $t \in (t(k), t(k+1))$ as time moves until the next round of clock reading compensation.

Remark 3.3. This approach is inspired by standard consensus algorithm [29] but uses pseudo-synchronous implementation. The stability analysis is similar to [29] and it is not repeated. As the clock reading compensation protocol is integrated into skew compensation, every node i in WSNs can achieve clock skew compensation and clock reading compensation simultaneously.

Remark 3.4. Under slowly time-varying input $\alpha_i(t)$s, synchronization error of clock reading always exists. As time moves, clock reading errors can be increased and exceed the accuracy requirement. In this case, Algorithm 2 should be performed again. This operation is called re-synchronization.

After introducing the details of clock reading compensation, notations of the transmission and updating time instants should be specified in order to illustrate the pseudo-synchronous implementation of the proposed protocol.

The transmission time instants of $j \in \mathcal{N}_i$ are defined by

$$t^j_{tr}(k) = t_j(k), \text{ where } \tau_j(t_j(k)) = kT. \tag{13}$$

Hence the receiving time instants of $i \in \mathcal{N}_j$ are defined by

$$t^i_{re}(k) = t^j_{tr}(k), \ \forall\, i \in \mathcal{N}_j. \tag{14}$$

The updating time instants of node i are defined by

$$t^i_{up}(k) = \max\{t^j_{tr}(k) | j \in \mathcal{N}_i \cup \{i\}\}, \tag{15}$$

namely the ith clock updates its state right after all its neighbour nodes finish their transmission actions, included its own transmission. $t^i_{tr}(k)$ and $t^i_{up}(k)$ can be determined by node i relying only on its local information.

We then combine relative clock skew estimation, clock skew compensation and clock reading compensation of the proposed protocol together and propose the combined filter-based protocol as Algorithm 3.

Algorithm 3 Combined filter-based protocol

Input: $\tau_i(t)$, $\tau_j(t)$ for $i \in \mathcal{V}$ and $j \in \mathcal{N}_i$.
Output: $\hat{\tau}_i(t)$ for $i \in \mathcal{V}$ and $j \in \mathcal{N}_i$.

1: Set a common T to each node $i \in \mathcal{V}$.
2: $\hat{\alpha}_i(t^i_{tr}(0)) = 1, \omega_i(t^i_{tr}(0)) = 0, \hat{\tau}_i(t^i_{tr}(0)) = \tau_i(t^i_{tr}(0)), \hat{\alpha}_{ij}(t^i_{tr}(0)) = 1, \ \forall i \in \mathcal{V}$.
3: Node i broadcasts $\omega_i(t^i_{tr}(k))$, $\hat{\alpha}_i(t^i_{tr}(k))$ and $\tau_i(t^i_{tr}(k))$ to its neighbours $j \in \mathcal{N}_i$ at $t = t^i_{tr}(k)$.
4: Node i broadcasts $\hat{\tau}_i(t^i_{up}(k))$ to its neighbours $j \in \mathcal{N}_i$ at $t = t^i_{up}(k)$.
5: **while** 1 **do**
6: Node i receives $\omega_j(t^j_{tr}(k))$, $\hat{\alpha}_j(t^j_{tr}(k))$ from its neighbours $j \in \mathcal{N}_i$ at $t = t^j_{tr}(k)$ and immediately records $\tau_i(t^j_{tr}(k))$.
7: $\hat{\alpha}_{ij}(t^j_{tr}(k)) = \rho\hat{\alpha}_{ij}(t^j_{tr}(k)) + \frac{(1-\rho)T}{\Delta\tau_i(t^j_{tr}(k))}, \rho \in (0,1), \Delta\tau_i(t^j_{tr}(k)) = \tau_i(t^j_{tr}(k)) - \tau_i(t^j_{tr}(k-1))$.
8: $\hat{\alpha}_i(t^j_{up}(k)) = \hat{\alpha}_i(t^j_{tr}(k)) - T\sum_{j\in\mathcal{N}_i}\left(\omega_i(t^j_{tr}(k)) - \frac{\omega_j(t^j_{tr}(k))}{\hat{\alpha}_{ij}(t^j_{tr}(k))}\right)$ at $t = t^j_{up}(k)$.
9: $\omega_i(t^j_{up}(k)) = \omega_i(t^j_{tr}(k)) - \epsilon\gamma\omega_i(t^j_{tr}(k)) + T\sum_{j\in\mathcal{N}_i}(\hat{\alpha}_i(t^j_{tr}(k)) - \hat{\alpha}_j(t^j_{tr}(k))\hat{\alpha}_{ij}(t^j_{tr}(k))$ at $t = t^j_{up}(k)$.
10: Node i receives $\hat{\tau}_j(t^j_{up}(k))$ from its neighbours $j \in \mathcal{N}_i$ at $t = t^j_{up}(k)$.
11: $\hat{\tau}_i(t^j_{up}(k)) = \frac{\hat{\tau}_i(t^j_{up}(k)) + \hat{\tau}_j(t^j_{up}(k))}{2}$ at $t = t^j_{up}(k)$.
12: $\hat{\tau}_i(t) = \tau_i(t^j_{up}(k)) + \hat{\alpha}_i(t^j_{tr}(k))(\tau_i(t) - \tau_i(t^j_{up}(k))), \ t \in (t^j_{up}(k), t^j_{up}(k+1)]$.
13: $\hat{\alpha}_i(t) = \hat{\alpha}_i(t^j_{up}(k)), \omega_i(t) = \omega_i(t^j_{up}(k)), \hat{\alpha}_{ij}(t) = \hat{\alpha}_{ij}(t^j_{tr}(k)), \ t \in (t^j_{up}(k), t^j_{tr}(k+1)]$.
14: **end while**

IV. SIMULATION

This section provides some examples to illustrate the performances of the proposed filter-based protocol. We test the performances of FBP in delay-free and random-delay cases and give a simulated case study to prove the usefulness of the proposed protocol. Finally, a comparative analysis with the proposed FBP and other two representative algorithms: A Maximum-Value-Based Consensus Synchronization (MTS)[19] and Second-order Linear Consensus Algorithm (SLCA) [15] is proposed in two aspects: synchronization accuracy and convergence rate.

4.1 Convergence performance of skew compensation in delay-free case

Consider a network topology in Fig. 1 composed of 10 labelled nodes.

The performance of clock skew compensation protocol is performed under different ρ_2s in delay-free case.

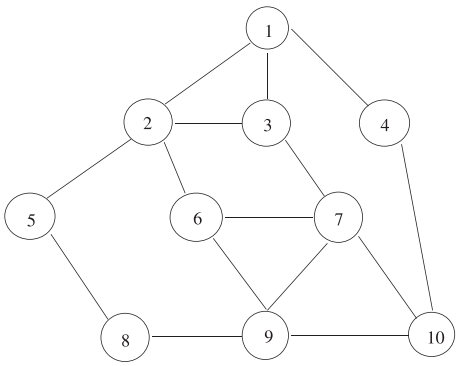

Fig. 1. Network topology composed of 10 labelled nodes.

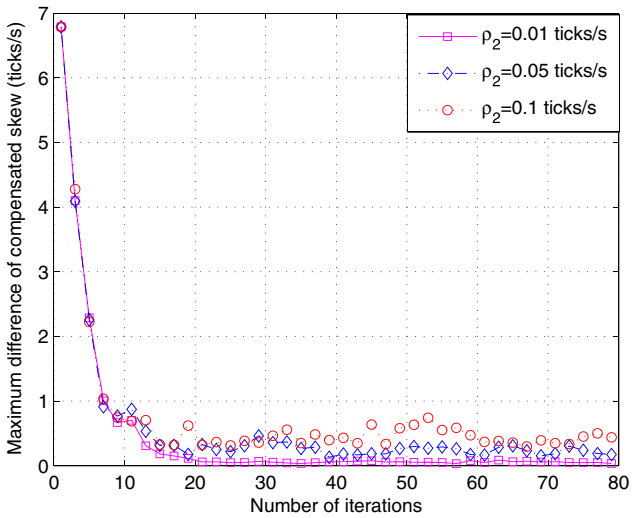

Fig. 2. Convergence performance of skew compensation in delay-free case. [Color figure can be viewed at wileyonlinelibrary.com]

Based on the topology shown in Fig. 1, $\lambda_{2n} = -4$ and $\lambda_{2bn} \simeq -4$. As $0 < T\min\{\frac{-2}{\lambda_{2n}}, \frac{-2}{\lambda_{2bn}}\} = 0.5$, the parameter set is chosen as: $T = 0.1s$, $\gamma = 3.5$, $K_i = 1.5$, $K_p = 1.5$.

The initialization values of $\alpha_i(0)$ are assumed to be randomly selected from $[0.9999, 0.99997] \cup [1.00003, 1.0001]$ since Crystal oscillators exhibit drift ρ_1 as $100\mu s$ in one second [13]. As each local clock skew experiences small drift, during one sampling period the local clock skew is added by a Gaussian random noise ρ_2 with its values respective as 0.01 ticks/s, 0.05 ticks/s, 0.1 ticks/s. It can be seen from Fig. 2 that it takes nearly 50 iterations to reduce the maximum difference of skew below 1 tick/s (1 tick/s=1/32768 Hz=30.5μs), *i.e.*, the individual clock resolution.

The performances of compensated clock readings are also tested as the compensated clock readings are

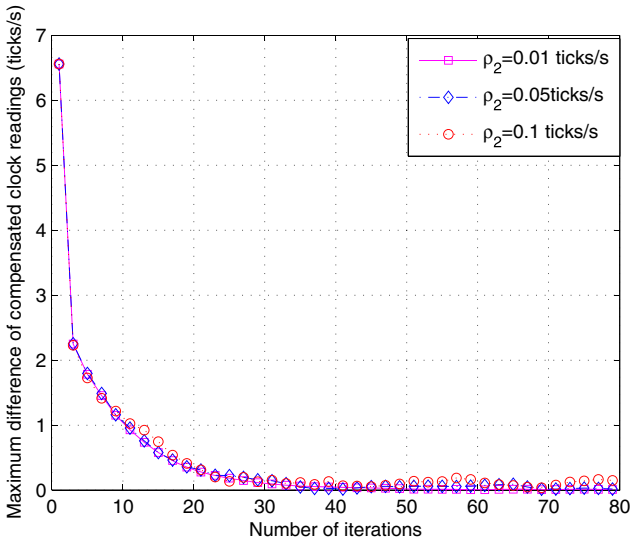

Fig. 3. Convergence performance of compensated clock readings in delay-free case. [Color figure can be viewed at wileyonlinelibrary.com]

the main focus of synchronization protocol. The random noises ρ_2s are respectively chosen as 0.01 ticks/s, 0.05 ticks/s, 0.1 ticks/s. It can be seen from Fig. 3 that the proposed FBP performs well in both convergence rate and synchronization accuracy under different ρ_2s, which demonstrates stronger robustness against slowly time-varying clock skew $\alpha_i(t)$.

4.2 Convergence performance of clock reading compensation in random-delay case

When calculating the estimate of relative clock skew α_{ij}, the communication delay can not be ignored. In this section, we evaluate how the skew compensation performances degrade when adding the presence of random delays. Based on the empirical results in [19], we assume that in the simulation $d_k^{j\mapsto i} \sim N(2.5\times10^{-4}, 10^{-8})$, which means that the delay is in the range of [0,550]μs with 99.97% confidence. The introduction of random delay $d_k^{j\mapsto i}$ degrades the accuracy of relative clock skew estimation α_{ij} and hence affects the performance of skew compensation.

Fig. 4 illustrates the convergence performance of clock reading compensation in delay-free case and random-delay case with $\rho_2 = 0.03$ ticks/s.

Drawn from the convergence result of clock reading compensation in Fig. 4, the steady-state error bound of random-delay case is larger compared with the error of delay-free case, but within the range of 1 ticks/s, just below the clock resolution.

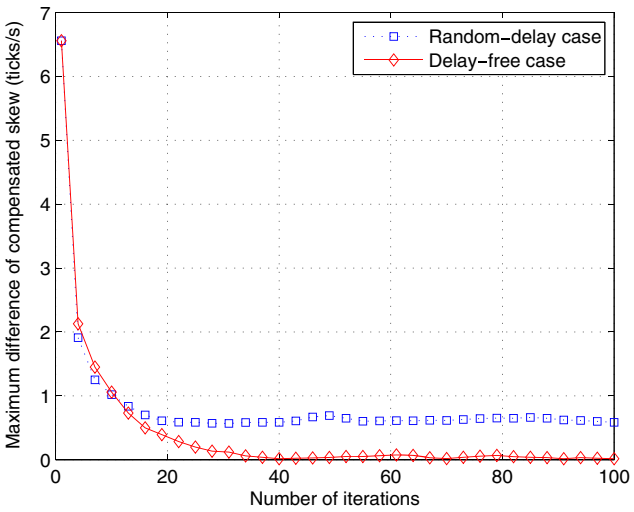

Fig. 4. Convergence performances of clock reading compensation in delay-free case and random-delay case with $\rho_2 = 0.03$ *ticks*/*s*. [Color figure can be viewed at wileyonlinelibrary.com]

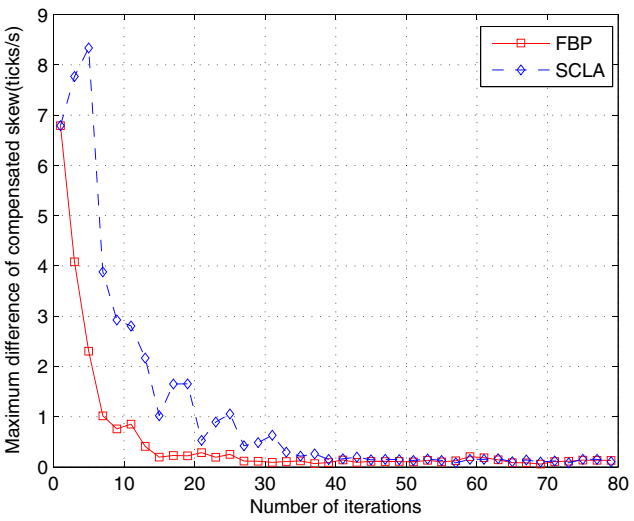

Fig. 5. Comparison in terms of skew compensation between FBP and SCLA under slowly time-varying $\alpha_i(t)$ with $\rho_2 = 0.03$ *ticks*/*s*. [Color figure can be viewed at wileyonlinelibrary.com]

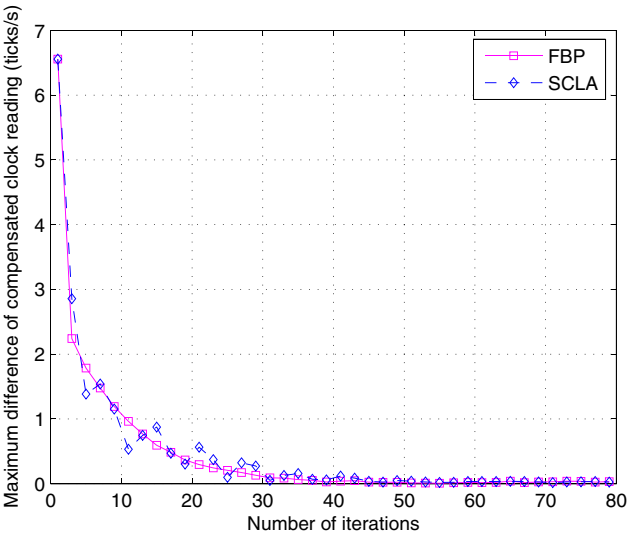

Fig. 6. Comparison in terms of compensated clock readings between FBP and SCLA under slowly time-varying $\alpha_i(t)$ with $\rho_2 = 0.03$ *ticks*/*s*. [Color figure can be viewed at wileyonlinelibrary.com]

4.3 Comparison between other techniques under time-varying clock parameters

Next we compare the proposed FBP with two existing state-of-the-art synchronization protocols: the second-order linear consensus algorithm (SCLA) proposed in [15] and a maximum-value-based consensus synchronization (MTS) proposed in [19]. Note that both SCLA and our FBP are synchronous protocols with pseudo-synchronous implementation, whereas MTS is asynchronous protocol with asynchronous implementation.

4.3.1 Comparison between FBP and SCLA under time-varying clock parameters

The second-order linear consensus algorithm applies standard consensus protocol to simultaneously compensate both clock skews and clock readings and it is an accurate-oriented protocol, meaning its synchronization precision is better compared with other existing protocols.

The sampling period of SCLA is chosen as $T = 0.2$, the same as FBP. The performances of skew compensation and clock reading compensation are shown respectively in Fig. 5 and Fig. 6. Fig. 5 shows that the proposed protocol FBP has faster convergence rate in skew compensation compared with SCLA. This results in a slightly better performance of clock reading compensation of FBP under slowly time-varying skew $\alpha_i(t)$, as shown in Fig. 6. On the whole, the proposed FBP shows better robustness in both convergence speed and synchronization precision compared with SCLA under time-varying clock parameters.

The second-order linear consensus algorithm mainly uses an iterative dynamics as follows:

$$\begin{aligned} x_i'(k^+) &= x_i'(k) + u_i'(k), \\ x_i''(k^+) &= x_i''(k) + u_i''(k), \end{aligned} \tag{16}$$

where $x_i'(k)$ represents the compensated clock reading for node i at time kT, and $x_i''(k)$ represents the estimate of physical clock skew. T is the sampling period. The second-order linear feedback control law $u_i'(k)$ and $u_i''(k)$ is given by

$$u_i'(k) = -0.5 * \sum_{j\in\mathcal{N}_i} K_{ij}[x_j'(k) - x_i'(k)],$$
$$u_i''(k) = -\frac{1}{T\max_i \alpha_i} \sum_{j\in\mathcal{N}_i} K_{ij}[x_j'(k) - x_i'(k)]. \tag{17}$$

where $K_{ij}s$ are elements of a Laplacian matrix based on the communication topology.

Drawn from the equation above, the second-order linear consensus algorithm mainly uses a relative clock reading difference $x_j'(k) - x_i'(k)$ as its control input. As the second-order linear consensus algorithm is in synchronous form, relative clock reading difference can also be treated as relative clock skew difference. This implies that the second-order linear consensus algorithm applies relative clock skew difference $\frac{x_j'(k)-x_i'(k)}{T}$ as its kind of feedback information.

The clock skew compensation part of FBP applies the following dynamic:

$$\hat{\alpha}_i(t^+) = \hat{\alpha}_i(t^-) - T\sum_{j\in\mathcal{N}_i}(\omega_i(t^-) - \omega_j(t^-)\hat{\alpha}_{ji}(t^-)),$$
$$\omega_i(t^+) = (1 - T\gamma)\omega_i(t^-) + T\sum_{j\in\mathcal{N}_i}(\hat{\alpha}_i(t^-) - \hat{\alpha}_j(t^-)\cdot$$
$$\hat{\alpha}_{ij}(t^-)), \tag{18}$$

By introducing the estimation of relative skew $\hat{\alpha}_{ij}(t^-)$, the proposed FBP also uses relative clock skew difference $\alpha_i(t^-) - \alpha_j(t^-)$ as its feedback information. This fact can be verified by multiplying $\alpha_i(t^-)$ to both sides of equations (18). Unlike the second-order linear consensus algorithm, as we bring in an auxiliary variable $\omega_i(k)$ into FBP, the proposed form of FBP has the functionality of a first-order filter. This contributes to better performances in both skew compensation and clock reading compensation.

4.3.2 Comparison between FBP and MTS under time-varying clock skew

The maximum-value-based consensus synchronization(MTS), which is initially proposed in [19], has its advantage of finite time convergence. The proposed FBP is then compared with MTS according to their performances in both convergence rate and synchronization accuracy. The comparison result of compensated clock readings is shown in Fig. 7.

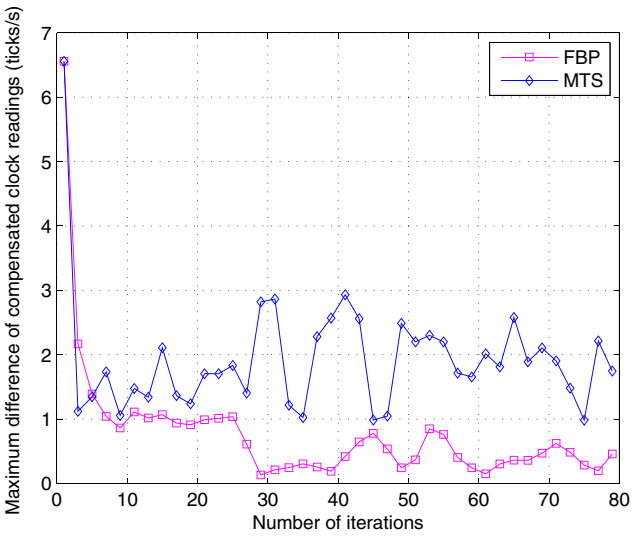

Fig. 7. Comparison in terms of compensated clock readings between FBP and MTS under time-varying $\alpha_i(t)$ with $\rho_2 = 0.5$ *ticks/s*. [Color figure can be viewed at wileyonlinelibrary.com]

Drawn from Fig. 7, FBP is inferior to MTS as MTS is a fast-convergent-oriented algorithm. However, after convergence, the the synchronization accuracy of FBP is more precise than MTS, indicating that the proposed FBP has smaller steady-state synchronization error.

In conclusion, the proposed FBP is shown to be more robust to noises, resulting in better convergence rate when compared with SCLA and better synchronization precision when compared with both SCLA and MTS.

V. CONCLUSION

This paper studies clock synchronization in wireless sensor network under time-varying clock parameters and proposes a fully distributed synchronization protocol called FBP. By applying FBP, a network of sensors can bound the synchronization error of compensated clock skews into a small steady-state range. The proposed protocol uses a first-order filter design method which shows better robustness against slowly time-varying clock parameters. We provide theoretical analysis as well as simulation results of the protocol and show that the proposed FBP achieves clock synchronization with better performances in both synchronization precision and convergence property. Future work includes extending the proposed protocol to the case of totally asynchronous form and checking whether the properties of FBP are still remained.

REFERENCES

1. Li, P. and J. Lam, "Synchronization in networks of genetic oscillators with delayed coupling," *Asian J. Control*, Vol. 13, No. 5, pp. 713–725 (2011).
2. Faizulkhakov, Y. R., "Time synchronization methods for wireless sensor networks: A survey," *Program Comp. Soft.*, Vol. 33, No. 4, pp. 214–226 (2007).
3. Freris, N. M., S. R. Graham, and P. Kumar, "Fundamental limits on synchronizing clocks over networks," *IEEE Trans. Autom. Control*, Vol. 56, No. 6, pp. 1352–1364 (2011).
4. Sundararaman, B., U. Buy, and A. D. Kshemkalyani, "Clock synchronization for wireless sensor networks: A survey," *Ad hoc Netw.*, Vol. 3, No. 3, pp. 281–323 (2005).
5. Elson, J., L. Girod, and D. Estrin, "Fine-grained network time synchronization using reference broadcasts," *ACM SIGOPS Oper. Syst. Rev.*, Vol. 36, No. SI, pp. 147–163 (2002).
6. Ganeriwal, S., R. Kumar, and M. B. Srivastava, "Timing-sync protocol for sensor networks," *Proc. 1st Int. Conf. Embedded Networked Sensor Systems*, Los Angeles, CA, USA, pp. 138–149 (2003).
7. Maróti, M., B. Kusy, G. Simon, and Á. Lédeczi, "The flooding time synchronization protocol," *Proc. 2nd Int. Conf. Embedded Networked Sensor Systems*, Baltimore, MD, USA, pp. 39–49 (2004).
8. Solis, R., V. S. Borkar, and P. Kumar, "A new distributed time synchronization protocol for multihop wireless networks," *Proc. 45th IEEE Conf. Decis. Control*, San Diego, CA, USA, pp. 2734–2739 (2006).
9. Giridhar, A. and P. Kumar, "Distributed clock synchronization over wireless networks: Algorithms and analysis," *Proc. 45th IEEE Conf. Decis. Control*, San Diego, CA, USA, pp. 4915–4920 (2006).
10. Freris, N. M. and A. Zouzias, "Fast distributed smoothing of relative measurements," *2012 IEEE 51st IEEE Conf. Decis. Control (CDC)*, Maui, HI, USA, pp. 1411–1416 (2012).
11. Barooah, P., J. P. Hespanha, and A. Swami, "On the effect of asymmetric communication on distributed time synchronization," *Proc. 46th IEEE Conf. Decis. Control*, New Orleans, LA, USA, pp. 5465–5471 (2007).
12. Liu, Z. W., Z. H. Guan, T. Li, X. H. Zhang, and J. W. Xiao, "Quantized consensus of multi-agent systems via broadcast gossip algorithms," *Asian J. Control*, Vol. 14, No. 6, pp. 1634–1642 (2012).
13. Sommer, P. and R. Wattenhofer, "Gradient clock synchronization in wireless sensor networks," *Proc. 2009 Int. Conf. Inform. Process. Sensor Networks*, San Francisco, CA, USA, pp. 37–48 (2009).
14. Carli, R., A. Chiuso, L. Schenato, and S. Zampieri, "Optimal synchronization for networks of noisy double integrators," *IEEE Trans. Autom. Control*, Vol. 56, No. 5, pp. 1146–1152 (2011).
15. Carli, R. and S. Zampieri, "Network clock synchronization based on the second-order linear consensus algorithm," *IEEE Trans. Autom. Control*, Vol. 59, No. 2, pp. 409–422 (2014).
16. Yıldırım, K. S., R. Carli, and L. Schenato, "Adaptive control-based clock synchronization in wireless sensor networks," *Eur. Control Conf. (ECC)*, Linz, Austria, pp. 2806–2811 (2015).
17. Bolognani, S., R. Carli, E. Lovisari, and S. Zampieri, "A randomized linear algorithm for clock synchronization in multi-agent systems," *IEEE Trans. Autom. Control*, Vol. 61, No. 7, pp. 1711–1726 (2016).
18. Schenato, L. 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).
19. He, J., P. Cheng, L. Shi, J. Chen, and Y. Sun, "Time synchronization in wsns: A maximum-value-based consensus approach," *IEEE Trans. Autom. Control*, Vol. 59, No. 3, pp. 660–675 (2014).
20. Maggs, M. K., S. G. O'Keefe, and D. V. Thiel, "Consensus clock synchronization for wireless sensor networks," *IEEE Sens. J.*, Vol. 12, No. 6, pp. 2269–2277 (2012).
21. Choi, B. J., L. Hao, X. Shen, and Z. Weihua, "Dcs: Distributed asynchronous clock synchronization in delay tolerant networks," *IEEE Trans. Parallel Distrib. Syst.*, Vol. 23, No. 3, pp. 491–504 (2012).
22. Heidarian, F., J. Schmaltz, and F. Vaandrager, "Analysis of a clock synchronization protocol for wireless sensor networks," *Theor. Comput. Sci.*, Vol. 413, No. 1, pp. 87–105 (2012).
23. Yang, W. and M. Fu, "A proportional integral estimator-based clock synchronization protocol for wireless sensor networks," *ISA Trans.*, Vol. 71, pp. 148–160 (2017).
24. Youn, S., "A comparison of clock synchronization in wireless sensor networks," *Int. J. Distrib. Sens. Netw.*, Vol. 9, No. 12, pp. 532–986 (2013).
25. Godsil, C. D. and G. Royle, *Algebraic Graph Theory*, Vol. 207, Springer, New York (2001). 32.
26. Freris, N. M., V. S. Borkar, and P. Kumar, "A model-based approach to clock synchronization," *Proc. 48th IEEE Conf. Decis. Control*, Shanghai, China, pp. 5744–5749 (2009).
27. Freris, N. M., H. Kowshik, and P. Kumar, "Fundamentals of large sensor networks: Connectivity, capacity, clocks, and computation," *Proc. IEEE*, Vol. 98, No. 11, pp. 1828–1846 (2010).

28. Han, Z., Z. Lin, and M. Fu, "A fully distributed approach to formation maneuvering control of multi-agent systems," *Proc. 53rd IEEE Conf. Decis. Control*, Los Angeles, CA, USA, pp. 6185–6190 (2014).
29. Olfati-Saber, R., J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proc. IEEE*, Vol. 95, No. 1, pp. 215–233 (2007).
30. Franceschelli, M., A. Gasparri, A. Giua, and C. Seatzu, "Decentralized laplacian eigenvalues estimation for networked multi-agent systems," *Proc. 48th IEEE Conf. Decis. Control*, Shanghai, China, pp. 2717–2722 (2009).
31. Sahai, T., A. Speranzon, and A. Banaszuk, "Hearing the clusters of a graph: A distributed algorithm," *Automatica*, Vol. 48, No. 1, pp. 15–24 (2012).

VI. APPENDIX

6.1 Proof of Lemma 3.3

The aggregated synchronous form of (12) is

$$\begin{bmatrix} \hat{\boldsymbol{\alpha}}(k+1) \\ \boldsymbol{\omega}(k+1) \end{bmatrix} = B_1 \begin{bmatrix} \hat{\boldsymbol{\alpha}}(k) \\ \boldsymbol{\omega}(k) \end{bmatrix} = \begin{bmatrix} \hat{\boldsymbol{\alpha}}(k) \\ \boldsymbol{\omega}(k) \end{bmatrix} + TB_0 \begin{bmatrix} \hat{\boldsymbol{\alpha}}(k) \\ \boldsymbol{\omega}(k) \end{bmatrix}, \tag{19}$$

where

$$B_1 = I + TB_0,\ B_0 = \begin{bmatrix} O_n & -(D(\mathcal{G}) - \Lambda(k)^T) \\ D(\mathcal{G}) - \Lambda(k) & -\gamma I_n \end{bmatrix},$$
$$\Lambda(k)_{ij} = \begin{cases} \hat{\alpha}_{ij}(k)\ i \neq j, j \in \mathcal{N}_i, \\ 0 \qquad \text{otherwise.} \end{cases} \tag{20}$$

It can be seen that $\{(\hat{\boldsymbol{\alpha}}, \boldsymbol{\omega}) : \lim_{k\to\infty}(\hat{\alpha}_i(k) - \hat{\alpha}_j(k)\alpha_{ij}(k)) = 0$ and $\lim_{k\to\infty} \omega_i(k) = 0,\ \forall k \in \mathbb{Z},\ \{i,j\} \in \mathcal{V}\}$ is the equilibrium point of (19).

Assume that the set of eigenvalues of B_0 is $\pi = \{\lambda_{b1},\ \lambda_{b2},\ \ldots\ \lambda_{b2n}\}$ while B_1 has eigenvalue set as $\hat{\pi} = \{\hat{\lambda}_{b1},\ \hat{\lambda}_{b2},\ \ldots\ \hat{\lambda}_{b2n}\}$ where $\hat{\lambda}_{bi} = 1 + T\lambda_{bi}$. Let $[\mu, \nu]^T$ be the associated eigenvector of B_0 corresponding to λ_{bi}, where $\mu, \nu \in \boldsymbol{R}$. Then we have

$$\left(\lambda_{bi} I_{2n} - \begin{bmatrix} O_n - (D(\mathcal{G}) - \Lambda(k)^T) \\ D(\mathcal{G}) - \Lambda(k) - \gamma I_n \end{bmatrix}\right) \begin{bmatrix} \mu \\ \nu \end{bmatrix} = 0. \tag{21}$$

or equivalently,

$$\left([\lambda_{bi} I_n \ \ O_n] - \left[\, O_n - (D(\mathcal{G}) - \Lambda(k)^T) \,\right]\right) \begin{bmatrix} \mu \\ \nu \end{bmatrix} = 0 \tag{22}$$

and

$$\left([O_n \ \ \lambda_{bi} I_n] - \left[\, D(\mathcal{G}) - \Lambda(k) - \gamma I_n \,\right]\right) \begin{bmatrix} \mu \\ \nu \end{bmatrix} = 0. \tag{23}$$

Combining (22)–(23) together yields the following result:

$$-(D(\mathcal{G}) - \Lambda(k))^T (D(\mathcal{G}) - \Lambda(k))\mu = \lambda_{bi}(\lambda_{bi} + \gamma)\mu, \tag{24}$$

indicating that $\lambda_{bi}(\lambda_{bi} + \gamma)$ is an eigenvalue of $-(D(\mathcal{G}) - \Lambda(k))^T (D(\mathcal{G}) - \Lambda(k))$ with μ being its associated eigenvector. Let $\Pi = \{\rho_1, \ldots, \rho_n\}$ be an eigenvalue of the matrix $(D(\mathcal{G}) - \Lambda(k))^T (D(\mathcal{G}) - \Lambda(k))$. The roots of the polynomial equation

$$\lambda_{bi}^2 + \gamma \lambda_{bi} + \rho_i = 0,\ i = 1, \ldots, n \tag{25}$$

are the eigenvalues of B_0. (25) leads to the following explicit expression of root solution

$$\lambda_{bi} = \frac{-\gamma \pm \sqrt{\gamma^2 - 4\rho_i}}{2},\ i = 1, \ldots, n. \tag{26}$$

The matrix $(D(\mathcal{G}) - \Lambda(k))^T (D(\mathcal{G}) - \Lambda(k))$ is symmetric and positive definite and its spectrum satisfies the following inequality

$$0 < \rho_1 \leq \rho_2 \leq \ldots \leq \rho_n, \tag{27}$$

which means its eigenvalue are all positive. Hence the eigenvalues of B_0 can be negative and lie in the negative half plane if and only if

$$\gamma > 0. \tag{28}$$

Assume that the eigenvalues of B_0 satisfy the following relationship:

$$0 > \lambda_{b1} \geq \lambda_{b2} \geq \ldots \geq \lambda_{b2n}. \tag{29}$$

To guarantee that $\hat{\pi} = \{\hat{\lambda}_{b1}, \hat{\lambda}_{b2},\ \ldots\ \hat{\lambda}_{b2n}\}$ lie strictly in the unit circle, the following condition should be satisfied:

$$|1 + T\lambda_{bi}| < 1,\ i = 1, 2, \ldots, 2n. \tag{30}$$

Hence T should be bounded by

$$0 < T < \frac{-2}{\lambda_{b2n}}. \tag{31}$$

Consequently, if (28) and (31) are satisfied, B_1 is Schur-stable, which means system (19) is BIBO stable. This guarantees the bounded output of $\hat{\alpha}_i(k)$ and $\omega_i(k)$ $\forall\ k \in \mathbb{Z}$ as the initial inputs of $\hat{\boldsymbol{\alpha}}(k) = \text{span}\{\mathbf{1}\}$ and $\boldsymbol{\omega}(k) = \mathbf{0}$ are bounded.

6.2 Proof of Theorem 3.1

Let $\alpha_i(k)\omega_i(k) = \overline{\omega}_i(k)$, $\alpha_i(k)\hat{\alpha}_i(k) = \overline{\alpha}_i(k)$. Multiplying (12) with $\alpha_i(k)$ yields the following state space equation

$$\begin{cases} \overline{\alpha}_i(k+1) = \overline{\alpha}_i(k) - T\sum_{j\in\mathcal{N}_i}(\overline{\omega}_i(k) - \overline{\omega}_j(k)) + \Delta_i^{\alpha}(k) + \\ \Delta\alpha_i(k)\alpha_i(k+1), \\ \overline{\omega}_i(k+1) = (1-\epsilon\gamma)\overline{\omega}_i(k) + T\sum_{j\in\mathcal{N}_i}(\overline{\alpha}_i(k) - \overline{\alpha}_j(k)) + \\ \Delta_i^{\omega}(k) + \Delta\alpha_i(k)\omega_i(k+1), \end{cases} \tag{32}$$

where $\Delta_i^{\alpha}(k)$ and $\Delta_i^{\beta}(k)$ are:

$$\begin{aligned} \Delta_i^{\alpha}(k) &= -T\sum_{j\in\mathcal{N}_i}\overline{\omega}_j(k)\left(1 - \frac{\alpha_i(k)}{\alpha_j(k)}\hat{\alpha}_{ji}(k)\right), \\ \Delta_i^{\omega}(k) &= T\sum_{j\in\mathcal{N}_i}\overline{\alpha}_j(k)\left(1 - \frac{\alpha_i(k)}{\alpha_j(k)}\hat{\alpha}_{ij}(k)\right). \end{aligned} \tag{33}$$

The synchronous form of (32) becomes

$$\begin{bmatrix} \overline{\boldsymbol{\alpha}}(k+1) \\ \overline{\boldsymbol{\omega}}(k+1) \end{bmatrix} = (A_1 + \Delta(k)) \begin{bmatrix} \overline{\boldsymbol{\alpha}}(k) \\ \overline{\boldsymbol{\omega}}(k) \end{bmatrix} + v(k), \tag{34}$$

where

$$A_1 = I_{2n} + TA_0, A_0 = \begin{bmatrix} O_n & -L(\mathcal{G}) \\ L(\mathcal{G}) & -\gamma I_n \end{bmatrix},$$

$$\Delta(k) = \begin{bmatrix} O_n & -T(A(\mathcal{G}) - \Pi_1(k)) \\ T(A(\mathcal{G}) - \Pi_2(k)) & O_n \end{bmatrix},$$

$$\Pi_1(k)_{ij} = \begin{cases} \frac{\alpha_j(k)}{\alpha_i(k)}\hat{\alpha}_{ij}(k) & i \neq j, j \in \mathcal{N}_i, \\ 0 & \text{otherwise.} \end{cases}$$

$$\Pi_2(k)_{ij} = \begin{cases} \frac{\alpha_i(k)}{\alpha_j(k)}\hat{\alpha}_{ij}(k) & i \neq j, j \in \mathcal{N}_i, \\ 0 & \text{otherwise.} \end{cases}$$

$$\boldsymbol{\Delta\alpha}(k) = [\Delta\alpha_1(k)\hat{\alpha}_1(k+1), \ldots, \Delta\alpha_n(k)\hat{\alpha}_n(k+1)]^T,$$

$$\boldsymbol{\Delta\omega}(k) = [\Delta\alpha_1(k)\hat{\omega}_1(k+1), \ldots, \Delta\alpha_n(k)\hat{\omega}_n(k+1)]^T,$$

$$v(k) = \begin{bmatrix} \boldsymbol{\Delta\alpha}(k) \\ \boldsymbol{\Delta\omega}(k) \end{bmatrix}.$$

$L(\mathcal{G})$ is a Laplacian matrix with zero as its simple eigenvalue; $\overline{\boldsymbol{\omega}}(k) = [\overline{\omega}_1(k), \ldots, \overline{\omega}_n(k)]^T$ is vector form of aggregated auxiliary variables; $\overline{\boldsymbol{\alpha}}(k) = [\overline{\alpha}_1(k), \ldots, \overline{\alpha}_n(k)]^T$ is vector form of aggregated virtual clock skew while $\boldsymbol{\alpha}(k) = [\alpha_1(k), \ldots, \alpha_n(k)]^T$ is vector form of aggregated physical clock skew. Without external input $\Delta\begin{bmatrix} \overline{\boldsymbol{\alpha}}(k) \\ \overline{\boldsymbol{\omega}}(k) \end{bmatrix}$ and $\begin{bmatrix} \boldsymbol{\Delta\alpha}(k) \\ \boldsymbol{\Delta\omega}(k) \end{bmatrix}$, $\{(\overline{\alpha}, \overline{\omega}) : \overline{\alpha} \in \mathbf{span}\{\mathbf{1}\}$ and $\overline{\omega} = \mathbf{0}\}$ is the equilibrium subspace of system (34).

According to Assumption 2.2 and Lemma 3.3, as long as $0 < T < \frac{-2}{\lambda_{2bn}}$ and $\gamma > 0$ are satisfied, $\Delta\alpha_i\alpha_i(k+1)$, $\Delta\alpha_i\omega_i(k+1)$ are uniformly bounded:

$$|\Delta\alpha_i\alpha_i(k+1)| \leq \rho_2\hat{\alpha}_{sup}, \ |\Delta\alpha_i\omega_i(k+1)| \leq \rho_2\omega_{sup}. \tag{35}$$

On the other hand, according to (7), $\alpha_{ij}(k) \in (\frac{1-\rho_1}{1+\rho_1}, \frac{1+\rho_1}{1-\rho_1})$. Hence the bound of $\Delta_i^{\alpha}(k)$, $\Delta_i^{\omega}(k)$ are given as

$$\begin{aligned} |\Delta_i^{\alpha}(k)| &\leq T\omega_{sup}(1+\rho_1)\sum_{j\in\mathcal{N}_i}\left(1 - \frac{(1-\rho_1)^2}{(1+\rho_1)^2}\right) \leq \Delta_1, \\ |\Delta_i^{\omega}(k)| &\leq T\hat{\alpha}_{sup}(1+\rho_1)\sum_{j\in\mathcal{N}_i}\left(1 - \frac{(1-\rho_1)^2}{(1+\rho_1)^2}\right) \leq \Delta_2, \\ \Delta_1 &= \frac{4\rho_1 T\omega_{sup}d_{max}}{1+\rho_1}, \ \Delta_2 = \frac{4\rho_1 T\hat{\alpha}_{sup}d_{max}}{1+\rho_1}. \end{aligned} \tag{36}$$

The external inputs $\Delta_i^{\alpha}(k)$, $\Delta_i^{\omega}(k)$, $\Delta\alpha_i(k)\hat{\alpha}_i(k+1)$, $\Delta\alpha_i(k)\hat{\omega}_i(k+1)$ are bounded. As long as A_1 is Schur-stable, system (34) is input-to-state stable.

A_1 can be divided into the sum of A_0 and I_{2n}. Assumes that the set of eigenvalues of A_0 is $\sigma = \{\lambda_1, \lambda_2, \ldots \lambda_{2n}\}$ while A_1 has eigenvalue set as $\hat{\sigma} = \{\hat{\lambda}_1, \hat{\lambda}_2, \ldots \hat{\lambda}_{2n}\}$ where $\hat{\lambda}_i = 1 + T\lambda_i$. Let $[\zeta, \eta]^T$ be an associated eigenvector corresponding to λ_i, where $\zeta, \eta \in \boldsymbol{r}$. Then we have

$$\left(\lambda_i I_{2n} - \begin{bmatrix} O_n - L(\mathcal{G}) \\ L(\mathcal{G}) - \gamma I_n \end{bmatrix}\right)\begin{bmatrix} \zeta \\ \eta \end{bmatrix} = 0 \tag{37}$$

or equivalently,

$$\left([\lambda_i I_n \ \ O_n] - \begin{bmatrix} O_n - L(\mathcal{G}) \end{bmatrix}\right)\begin{bmatrix} \zeta \\ \eta \end{bmatrix} = 0 \tag{38}$$

and

$$\left([O_n \ \ \lambda_i I_n] - \begin{bmatrix} L(\mathcal{G}) - \gamma I_n \end{bmatrix}\right)\begin{bmatrix} \zeta \\ \eta \end{bmatrix} = 0. \tag{39}$$

Drawn from (38)–(39), η can be eliminated:

$$-L(\mathcal{G})^2\zeta = \lambda_i(\lambda_i + \gamma)\zeta, \tag{40}$$

which means $\lambda_i(\lambda_i + \gamma)$ is an eigenvalue of $-L(\mathcal{G})^2$ with ζ being its associated eigenvector. Let σ_i be an eigenvalue of the matrix $L(\mathcal{G})^2$. The roots of the polynomial equation

$$\lambda_i{}^2 + \gamma\lambda_i + \sigma_i = 0, \ i = 1, \ldots, n \tag{41}$$

are the eigenvalues of A_0. (41) leads to the following explicit expression of root solution

$$\lambda_i = \frac{-\gamma \pm \sqrt{\gamma^2 - 4\sigma_i}}{2},\ i = 1, \ldots, n. \tag{42}$$

As $\mathcal{G}$ is a connected and undirected graph, $L(\mathcal{G})^2$ is symmetric and positive semi-definite with its rank the same as $L(\mathcal{G})$. By Lemma 2.1, the spectrum of matrix $L(\mathcal{G})$ satisfies the following inequality

$$0 = \lambda_1(\mathcal{G}) < \lambda_2(\mathcal{G}) \leq ... \leq \lambda_n(\mathcal{G}). \tag{43}$$

Hence the spectrum of matrix $L(\mathcal{G})^2$ satisfies the following inequality

$$0 = \sigma_1(\mathcal{G}) < \sigma_2(\mathcal{G}) \leq ... \leq \sigma_n(\mathcal{G}). \tag{44}$$

Therefore it has one zero eigenvalue and all other eigenvalues are positive and real. Hence the eigenvalues of A_0 can be negative and lie in the negative half plane except for one zero eigenvalue if and only if

$$\gamma > 0. \tag{45}$$

Assume that the eigenvalues of A_0 satisfy the following relationship:

$$0 = \lambda_1(\mathcal{G}) > \lambda_2(\mathcal{G}) \geq ... \geq \lambda_{2n}(\mathcal{G}). \tag{46}$$

To guarantee that $\hat{\sigma} = \{\hat{\lambda}_2,\ ...\ \hat{\lambda}_{2n}\}$ lie strictly in the unit circle, the following condition should be satisfied:

$$|1 + T\lambda_i| < 1,\ i = 2...n. \tag{47}$$

Hence T should be bounded by

$$0 < T < \min\left\{\frac{-2}{\lambda_{2n}}, \frac{-2}{\lambda_{2bn}}\right\}. \tag{48}$$

When (45) and (48) are satisfied, A_1 is Schur-stable with only one eigenvalue on the unit circle. Although calculation of $\lambda_2(\mathcal{G})$ and $\lambda_{2n}(\mathcal{G})$ require $\sigma_2(\mathcal{G})$ and $\sigma_n(\mathcal{G})$ that are global information, $\sigma_2(\mathcal{G})$ and $\sigma_2(\mathcal{G})$ can be solved in a distributed way proposed in [30] or [31] due to symmetric nature of $L(\mathcal{G})^2$. This guarantees (12) being fully distributed.

As the eigenvalues of A_1 lie strictly inside the unit circle except for one eigenvalue, system (34) is input-to-state stable as A_1 is proven to be Schur-stable.

Consider the state coordinate transformation

$$V^{-1}\begin{bmatrix}\overline{\boldsymbol{\alpha}}(k)\\ \overline{\boldsymbol{\omega}}(k)\end{bmatrix} = \begin{bmatrix}\overline{\boldsymbol{\alpha}}'(k)\\ \overline{\boldsymbol{\omega}}'(k)\end{bmatrix}, \tag{49}$$

where V is the similar transformation matrix with its first column being $\begin{bmatrix}\mathbf{1}\\ \mathbf{0}\end{bmatrix}$ and $V^{-1}A_1V = \Lambda = diag\{1, \hat{\lambda}_2, \ldots, \hat{\lambda}_{2n}\}$. By similar transformation V, system (34) is transformed into

$$\begin{bmatrix}\overline{\boldsymbol{\alpha}}'(k+1)\\ \overline{\boldsymbol{\omega}}'(k+1)\end{bmatrix} = \Lambda\begin{bmatrix}\overline{\boldsymbol{\alpha}}'(k)\\ \overline{\boldsymbol{\omega}}'(k)\end{bmatrix} + V^{-1}\Delta(k)\begin{bmatrix}\overline{\boldsymbol{\alpha}}(k)\\ \overline{\boldsymbol{\omega}}(k)\end{bmatrix} + V^{-1}\begin{bmatrix}\boldsymbol{\Delta\alpha}(k)\\ \boldsymbol{\Delta\omega}(k)\end{bmatrix}. \tag{50}$$

Let $\boldsymbol{\eta}^1(k) = [\overline{\alpha}'_2(k), \ldots, \overline{\alpha}'_n(k)]$ and $\boldsymbol{\eta}^2(k) = [\overline{\omega}'_1(k), \ldots, \overline{\omega}'_n(k)]$. An equivalent form of system (50) is

$$\begin{bmatrix}\overline{\alpha}'_1(k+1)\\ \boldsymbol{\eta}^1(k+1)\\ \boldsymbol{\eta}^2(k+1)\end{bmatrix} = \begin{bmatrix}1 & 0 & 0\\ 0 & \Lambda_1 & 0\\ 0 & 0 & \Lambda_2\end{bmatrix}\begin{bmatrix}\overline{\alpha}'_1(k)\\ \boldsymbol{\eta}^1(k)\\ \boldsymbol{\eta}^2(k)\end{bmatrix} + V^{-1}\Delta(k)\begin{bmatrix}\overline{\boldsymbol{\alpha}}(k)\\ \overline{\boldsymbol{\omega}}(k)\end{bmatrix} + V^{-1}\begin{bmatrix}\boldsymbol{\Delta\alpha}(k)\\ \boldsymbol{\Delta\omega}(k)\end{bmatrix}, \tag{51}$$

where

$$\Lambda_1 = diag\{\hat{\lambda}_2, \ldots, \hat{\lambda}_n\},\ \Lambda_2 = diag\{\hat{\lambda}_{n+1}, \ldots, \hat{\lambda}_{2n}\}. \tag{52}$$

The zero input response of system (51) is

$$R'_1 = \lim_{k\to\infty}\Lambda^k\begin{bmatrix}\overline{\alpha}'_1(0)\\ \boldsymbol{\eta}^1(0)\\ \boldsymbol{\eta}^2(0)\end{bmatrix}. \tag{53}$$

As $\{(\overline{\alpha}, \overline{\omega}) : \overline{\alpha} \in \mathbf{span}\{\mathbf{1}\}$ and $\overline{\omega} = \mathbf{0}\}$ is the equilibrium subspace of system (34),

$$\lim_{k\to\infty}\Lambda^k\begin{bmatrix}\overline{\alpha}'_1(0)\\ \boldsymbol{\eta}^1(0)\\ \boldsymbol{\eta}^2(0)\end{bmatrix} = \lim_{k\to\infty}\Lambda^k\begin{bmatrix}\overline{\boldsymbol{\alpha}}(k)\\ \overline{\boldsymbol{\omega}}(k)\end{bmatrix} = \overline{\alpha}'_1(0)\begin{bmatrix}1\\ \mathbf{0}\\ \mathbf{0}\end{bmatrix}. \tag{54}$$

From (54), it follows that $\overline{\alpha}_1'(0) = \overline{\alpha}_1'(k)$, and $\boldsymbol{\eta}^1(k), \boldsymbol{\eta}^2(k)$ globally asymptotically converges to $\mathbf{0}$ as $k \to \infty$. By the coordinate transformation, the zero-input response of (34) globally asymptotically converges to $R_1 = \overline{\alpha}_1'(0) \begin{bmatrix} \mathbf{1} \\ \mathbf{0} \end{bmatrix}$ as $k \to \infty$.

The zero state response of system (34) is

$$R_2 = \lim_{k\to\infty} \sum_{i=0}^{n-1} \Lambda^{k-i-1}(\Delta(k) \begin{bmatrix} \overline{\boldsymbol{\alpha}}(k) \\ \overline{\boldsymbol{\omega}}(k) \end{bmatrix} + \begin{bmatrix} \boldsymbol{\Delta\alpha}(k) \\ \boldsymbol{\Delta\omega}(k) \end{bmatrix}). \quad (55)$$

By Lemma 3.1, the multiplicative part of noise $\Delta(k)$ is time-decaying:

$$\lim_{k\to\infty} \Delta(k) \begin{bmatrix} \overline{\boldsymbol{\alpha}}(k) \\ \overline{\boldsymbol{\omega}}(k) \end{bmatrix} = \lim_{k\to\infty} \begin{bmatrix} -T(A(\mathcal{G}) - \Pi_1(k))\overline{\boldsymbol{\alpha}}(k) \\ \mathbf{0} \end{bmatrix}. \quad (56)$$

In view of (35) and (36), the zero state response is bounded by

$$|R_2| \le \begin{bmatrix} \left|\frac{\hat{\lambda}_1}{1-\hat{\lambda}_1}\right| \left(\frac{4\rho_1 T\omega_{sup} d_{max}}{1+\rho_1} + \rho_2\hat{\alpha}_{sup}\right) \\ \vdots \\ \left|\frac{\hat{\lambda}_n}{1-\hat{\lambda}_n}\right| \left(\frac{4\rho_1 T\omega_{sup} d_{max}}{1+\rho_1} + \rho_2\hat{\alpha}_{sup}\right) \\ \left|\frac{\hat{\lambda}_{n+1}}{1-\hat{\lambda}_{n+1}}\right| \rho_2\omega_{sup} \\ \vdots \\ \left|\frac{\hat{\lambda}_{2n}}{1-\hat{\lambda}_{2n}}\right| \rho_2\omega_{sup} \end{bmatrix}. \quad (57)$$

Finally,

$$\lim_{k\to\infty} |\varepsilon_i(k)| \le \rho_3 = \left|\frac{\hat{\lambda}_i}{1-\hat{\lambda}_i}\right| \left(\frac{4\rho_1 T\omega_{sup} d_{max}}{1+\rho_1} + \rho_2\hat{\alpha}_{sup}\right) = \left|\frac{1+\epsilon\lambda_i}{\epsilon\lambda_i}\right| \left(\frac{4\rho_1 T\omega_{sup} d_{max}}{1+\rho_1} + \rho_2\hat{\alpha}_{sup}\right). \quad (58)$$

This completes the proof of Theorem 3.1.