Clock Synchronization over Wireless Sensor Network via a Filter-based Approach

Wenlun Yang
Department of Control Science and Engineering
Zhejiang University
Hangzhou, China
Email: yangwenlun@zju.edu.cn

Minyue Fu
School of Electrical Engineering and Computer Science
University of Newcastle
NSW 2308 Australia
Email: minyue.fu@newcastle.edu.au

Hao Xing
School of Automation
Hangzhou Dianzi University
Hangzhou, China
Email: xing.h@hdu.edu.cn

***Abstract*—This paper proposes a filter-based distributed protocol to realize time synchronization under time-varying clock parameters. The proposed protocol is derived from a first-order controller and is fully distributed, meaning that by relying merely on its local clock readings and reading announcements from its neighbouring sensor nodes, each node in WSNs can dynamically update its virtual clock and bound synchronization error to a steady state. We analyzing the input-to-state stable stability of the control system which could guarantee convergence properties in terms of time-vary clock parameters. Simulation results are given to illustrate the performance of this control protocol.**

***Index Terms*—Wireless sensor networks (WSNs), clock synchronization, first-order controller.**

I. Introduction

Synchronization of local clocks is an important requirement in mobile sensor networks for a series of applications, e.g. remote environmental monitoring, target tracking, etc. Clock synchronization protocol aims at synchronizing local clocks and achieve a common reference of time among sensors and thus is rather basic.

Basically two kinds of clock synchronization protocols are commonly used [1]: tree-based and distributed. A tree-based protocol assigns one node as a server and other nodes as clients and forms a hierarchical structure, e.g., reference broadcast synchronization (RBS) [2], timing-sync protocol for sensor networks (TPSN) [3] and flooding time synchronization protocol (FTSP) [4]. However, due to high dependence of tree-based structure, tree-based protocols are vulnerable to root-node failure and packet losses in some senarios.

Many researchers pay more attention to developing distributed protocols. Compared with tree-based ones, distributed protocols do not require a global reference by avoiding a specific construction among agents and no server node needs to be selected in advance. As all local nodes apply exactly the same algorithm with information from their neighbourhoods only, these protocols are robust to dynamic topology changes and are highly scalable. Some typical protocols are listed: [5] uses a recursive least squares estimation approach to estimate the best-fit offset values. [6] gives a further analysis of [5] and indicates that the least-squares solution provides a better performance for a range of typical network graphs. Other example, see [7], etc.

Among distributed protocols, consensus-based have been widely applied to study synchronization in a network as it can drive all agents to finally reach a state of agreement based on locally available information [8]. Based on its way of implementation, there are mainly two categories: asynchronous [9]–[11] and synchronous [12], [13]. In asynchronous form, node i randomly picks one of its neighbours and exchange information mutually. Then an updating rule is used to update their states. Compared with asynchronous ones, synchronous require concurrent update for every local node which is unrealistic before a common reference of time is reached. To tackle this problem, [12] presents a synchronous protocol to synchronize a network of controlled discrete-time double integrators. [13] makes a further step by proposing a realistic pseudo-synchronous implementation. However, most distributed protocols assume that the clock skews are constant, which is unrealistic in real applications. In fact, it is well known that a typical a sensor node can drift up to 30-100 ppm [14].

In this paper, we study the clock synchronization problem with the focus on handling slow drift of the clock skews. A filter-based protocol is then proposed, which is fully distributed in the sense that each node relies only on its local clock readings and reading announcements from its neighbours. The main contribution of the protocol is that it deals with a time-varying clock model and can reduce the synchronization error of clock skew to a steady state. Compared with the existing consensus-based distributed protocols, the proposed protocol shows better robustness against slowly time-varying clock parameters. Besides, due to its distributed nature it requires no global information. Simulation results show the performance of the proposed protocol.

II. Preliminaries

A. Notations

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

978-1-5386-3524-7/17/$31.00 ©2017 IEEE

B. Graph theory

An undirected and connected 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 an undirected graph. d_i denotes the cardinality of $\mathcal{N}_i$ and $d_{max} = \max d_i, \forall i \in \mathcal{V}$.

For an undirected and connected graph $\mathcal{G}$, the degree matrix $D(\mathcal{G})$, the adjacency matrix $A(\mathcal{G})$ and the associated Laplacian matrix $L(\mathcal{G})$ are defined as follows:

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

$$A(\mathcal{G})_{ij} = \begin{cases} 1 & i \neq j, j \in \mathcal{N}_i, \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

$$L(\mathcal{G})_{ij} = \begin{cases} -1 & i \neq j, j \in \mathcal{N}_i, \\ 0 & i \neq j, j \notin \mathcal{N}_i, \\ d_i & i = j. \end{cases} \tag{3}$$

C. Clock model

Considering an integral clock model [15] for each local clock i:

$$\tau_i(t) = \int_0^t \alpha_i(t')dt' + \beta_i, \tau_i(0) = \beta_i, \tag{4}$$

where $\tau_i(t)$ is local clock reading of node i; $\alpha_i(t)$ is i's clock skew and is slowly time-varying; β_i is local clock offset and t indicates absolute reference time. Assume that $\alpha_i(t)$ satisfies the following assumption.

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

$$1 - \rho_1 \leq \alpha_i(t) \leq 1 + \rho_1, \ \forall\, i \in \mathcal{V}, \tag{5}$$

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

Our distributed synchronization protocol will be in discrete-time form. Hence the sampling period is introduced and denoted by T. For the sake of simplicity, $\alpha_i(kT) = \alpha_i(k), \ \forall\, k \in \mathbb{Z}$, i.e., $T = 1$. Define $\Delta\alpha_i(k) = \alpha_i(k+1) - \alpha_i(k)$ as the variation of α_i during one sampling period. Another assumption is proposed to give a bound on $\Delta\alpha_i(k)$.

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

$$|\Delta\alpha_i(k)| \leq \rho_2, \tag{6}$$

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

Consider a WSN composed of n sensor nodes equipped with its local clocks. The communication topology of WSN is modeled as an undirected and connected graph $\mathcal{G}$ and $\mathcal{N}_i$ denotes the set of one-hop neighbours of node i in WSN. An edge between node i and j implies that they can communicate with each other by exchanging their information mutually. In our setup, communication and computational delays are both negligible.

The general linear updating protocol is proposed in (7) and each node i periodically updates its logical clock reading $\hat{\tau}_i(t)$ based only on its own information and its neighbours' information

$$\hat{\tau}_i(t) = G_i(\tau_i(t), \tau_j(t)), \tag{7}$$

where $G_i(.)$ is a linear function depending on the information available at node i and from node $j \in \mathcal{N}_i$, i.e, the local clock readings of clock i itself and its neighbour nodes $j \in \mathcal{N}_i$.

III. Clock synchronization protocol

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

A. Relative clock skew estimation

Some definitions are listed as follows:

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

$$\alpha_{ij}(t) = \frac{\alpha_j(t)}{\alpha_i(t)}. \tag{8}$$

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.

As the local clock skew $\alpha_i(t)$ is neither known nor measurable by node i, node i can not figure out $\alpha_{ij}(t)$. Instead, 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 $t_j(k)$. The estimation of relative clock skew $\alpha_{ij}(t_j(k))$ is proposed via low-pass filter introduced by [9].

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

$$\hat{\alpha}_{ij}(t^+) = \rho\hat{\alpha}_{ij}(t^-) + (1-\rho)\frac{T}{\tau_i(t_j(k)) - \tau_i(t_j(k-1))}, \tag{9}$$

where $\rho \in (0, 1)$ is a tuning parameter served as a tradeoff between the precision of new measurement and prior estimation based on the old measurement. t^+ and t^- represent, respectively, the right-hand limit and left-hand limit of t.

Lemma 3.1: [9] For constant α_i, applying (9) yields the following convergent result

$$\lim_{t\to\infty} \hat{\alpha}_{ij}(t) = \alpha_{ij} \tag{10}$$

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

For time-varying $\alpha_i(t)$, the relative skew $\alpha_{ij}(t)$ is also time-varying. However, considering small change of $\alpha_i(t)$, (10) holds approximately under time-varying clock skew. In the following we propose clock skew compensation protocol using relative skew algorithm proposed in (9).

B. Clock skew compensation

The filter-based clock synchronization protocol of skew compensation is presented 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:

$$\begin{cases}\hat{\alpha}_i(t^+)=\hat{\alpha}_i(t^-)-T\sum\limits_{j\in\mathcal{N}_i}(\omega_i(t^-)-\omega_j(t^-)\hat{\alpha}_{ji}(t^-)),\\ \omega_i(t^+)=(1-T\gamma)\omega_i(t^-)+T\sum\limits_{j\in\mathcal{N}_i}(\hat{\alpha}_i(t^-)-\hat{\alpha}_j(t^-)\\ \hat{\alpha}_{ij}(t^-)),\end{cases}\tag{11}$$

where $\hat{\alpha}_{ij}(t^-)$ is calculated in (9); γ is the information rate which depicts the proportion of how much new information is introduced; $T>0$ is discrete-time step size; both γ and T will be specified in the next section to guarantee stability and convergence of the control system.

By replacing t^+ or t^- with a common discrete time point k from a perspective of global clock, (11) can be further presented as:

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 $j\in\mathcal{N}_i$.

```
Initialize α̂_i(0) = 1, ω_i(0) = 0, ∀i ∈ V.
while 1 do
    α̂_i(k) ⇐ α̂_i(k) − T Σ_{j∈N_i} (ω_i(k) − ω_j(k) α̂_ji(k)) at t = t(k).
    ω_i(k) ⇐ (1 − Tγ) ω_i(k) + T Σ_{j∈N_i} (α̂_i(k) − α̂_j(k) α̂_ij(k)) at t = t(k).
    ω_i(t) = ω_i(k), α̂_i(t) = α̂_i(k), t ∈ [k, k+1).
    ω_i(k+1) = ω_i(k), α̂_i(k+1) = α̂_i(k) at t = t(k+1).
end while
```

Lemma 3.2 shows the bounded property of $\hat{\alpha}_i(k)$ and $\omega_i(k)$ when applying Algorithm 1.

Lemma 3.2: Protocol (11) leads to the boundedness of $\lim_{k\to\infty}\hat{\alpha}_i(k)$ and $\lim_{k\to\infty}\omega_i(k)$ under the following parameter constraints:

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

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

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

$\Lambda(k)$ 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:

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

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

Proof 1: The proof is similar to Proof 2 in main theorem and is ignored here.

The error of node i's clock skew is as follows:

$$\varepsilon_i(k)=\hat{\alpha}_i(k)\alpha_i(k)-\frac{1}{N}\sum_{i=1}^{N}\hat{\alpha}_i(k)\alpha_i(k).\tag{12}$$

Our main result is presented in Theorem 3.1.

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, Algorithm 1 leads to the boundedness of $\varepsilon_i(k)$ as

$$\lim_{k\to\infty}\varepsilon_i(k)\le\rho_3=|\frac{1+T\lambda_i}{T\lambda_i}|(\frac{4\rho_1T\omega_{sup}d_{max}}{1+\rho_1}+\rho_2\hat{\alpha}_{sup})$$

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

$$A_0=\begin{bmatrix}\mathbb{O}^n & -L(\mathcal{G})\\ L(\mathcal{G}) & -\gamma I_n\end{bmatrix},$$

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 3.2.

C. Stability and convergence analysis

We now give the proof for Theorem 3.1.

Proof 2: Let $\alpha_i(k)\omega_i(k)=\overline{\omega}_i(k)$, $\alpha_i(k)\hat{\alpha}_i(k)=\overline{\alpha}_i(k)$. Multiplying (11) with $\alpha_i(k)$ yields the following state space equation

$$\begin{cases}\overline{\alpha}_i(k+1)=\overline{\alpha}_i(k)-T\sum\limits_{j\in\mathcal{N}_i}(\overline{\omega}_i(k)-\overline{\omega}_j(k))+\Delta_i^{\alpha}(k)+\\ \Delta\alpha_i(k)\alpha_i(k+1),\\ \overline{\omega}_i(k+1)=(1-T\gamma)\overline{\omega}_i(k)+T\sum\limits_{j\in\mathcal{N}_i}(\overline{\alpha}_i(k)-\overline{\alpha}_j(k))+\\ \Delta_i^{\omega}(k)+\Delta\alpha_i(k)\omega_i(k+1),\end{cases}\tag{13}$$

where we have the following relationship with $\Delta_i^{\alpha}(k)$ and $\Delta_i^{\beta}(k)$:

$$\begin{aligned}\Delta_i^{\alpha}(k)&=-T\sum_{j\in\mathcal{N}_i}\overline{\omega}_j(k)(1-\frac{\alpha_i(k)}{\alpha_j(k)}\hat{\alpha}_{ji}(k)),\\ \Delta_i^{\omega}(k)&=T\sum_{j\in\mathcal{N}_i}\overline{\alpha}_j(k)(1-\frac{\alpha_i(k)}{\alpha_j(k)}\hat{\alpha}_{ij}(k)).\end{aligned}\tag{14}$$

Define

$$\begin{aligned}\boldsymbol{\overline{\omega}}(k)&=[\overline{\omega}_1(k),\overline{\omega}_2(k),...,\overline{\omega}_n(k)]^T,\\ \boldsymbol{\overline{\alpha}}(k)&=[\overline{\alpha}_1(k),\overline{\alpha}_2(k),...,\overline{\alpha}_n(k)]^T,\\ \boldsymbol{\alpha}(k)&=[\alpha_1(k),\alpha_2(k),...,\alpha_n(k)]^T.\end{aligned}\tag{15}$$

The synchronous form becomes

$$\begin{bmatrix}\boldsymbol{\overline{\alpha}}(k+1)\\ \boldsymbol{\overline{\omega}}(k+1)\end{bmatrix}=A_1\begin{bmatrix}\boldsymbol{\overline{\alpha}}(k)\\ \boldsymbol{\overline{\omega}}(k)\end{bmatrix}+\Theta(k)\begin{bmatrix}\boldsymbol{\overline{\alpha}}(k)\\ \boldsymbol{\overline{\omega}}(k)\end{bmatrix}+\begin{bmatrix}\boldsymbol{\Delta\alpha}(k)\\ \boldsymbol{\Delta\omega}(k)\end{bmatrix},\tag{16}$$

where

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

$$\Theta(k) = \begin{bmatrix} \mathbb{O}^n & -T(A(\mathcal{G}) - \Pi_1(k)) \\ T(A(\mathcal{G}) - \Pi_2(k)) & \mathbb{O}^n \end{bmatrix},$$

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

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

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

It can be simply verified that without external input $\Theta(k)\begin{bmatrix}\overline{\boldsymbol{\alpha}}(k) \\ \overline{\boldsymbol{\omega}}(k)\end{bmatrix}$ 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 (16).

According to Assumption 2.2 and Lemma 3.2, 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:

$$|\Delta\alpha_i\alpha_i(k+1)| \leq \rho_2\hat{\alpha}_{sup}, \ |\Delta\alpha_i\omega_i(k+1)| \leq \rho_2\omega_{sup}. \quad (17)$$

On the other hand, according to (9), $\alpha_{ij}(k) \in (\frac{1-\rho_1}{1+\rho_1}, \frac{1+\rho_1}{1-\rho_1})$. Hence the bound of $\Delta_i^{\alpha}(k)$, $\Delta_i^{\omega}(k)$ are given as

$$\begin{aligned} &|\Delta_i^{\alpha}(k)| \leq T\omega_{sup}(1+\rho_1)\sum_{j\in\mathcal{N}_i}\left(1 - \frac{(1-\rho_1)^2}{(1+\rho_1)^2}\right) \leq \Delta_1, \\ &|\Delta_i^{\omega}(k)| \leq T\hat{\alpha}_{sup}(1+\rho_1)\sum_{j\in\mathcal{N}_i}\left(1 - \frac{(1-\rho_1)^2}{(1+\rho_1)^2}\right) \leq \Delta_2, \\ &\Delta_1 = \frac{4\rho_1 T\omega_{sup} d_{max}}{1+\rho_1}, \ \Delta_2 = \frac{4\rho_1 T\hat{\alpha}_{sup} d_{max}}{1+\rho_1}. \end{aligned} \quad (18)$$

As 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, the linear system (16) is input-to-state stable as long as A_1 is Schur-stable.

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

$$\left(\lambda_i I_{2n} - \begin{bmatrix} \mathbb{O}^n & -L(\mathcal{G}) \\ L(\mathcal{G}) & -\gamma I_n \end{bmatrix}\right)\begin{bmatrix}\zeta \\ \eta\end{bmatrix} = 0 \quad (19)$$

By eliminating η, we have:

$$-L(\mathcal{G})^2\zeta = \lambda_i(\lambda_i + \gamma)\zeta, \quad (20)$$

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

$$\lambda_i{}^2 + \gamma\lambda_i + \sigma_i = 0, \ i = 1, ..., n \quad (21)$$

are the eigenvalues of A_0. (21) leads to the following explicit expression of root solution

$$\lambda_i = \frac{-\gamma \pm \sqrt{\gamma^2 - 4\sigma_i}}{2}, \ i = 1, ..., n. \quad (22)$$

The matrix $L(\mathcal{G})^2$ is symmetric and positive semi-definite with its rank the same as $L(\mathcal{G})$. The spectrum of matrix $L(\mathcal{G})^2$ the following inequality

$$0 = \sigma_1(\mathcal{G}) < \sigma_2(\mathcal{G}) \leq ... \leq \sigma_n(\mathcal{G}). \quad (23)$$

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

$$\gamma > 0. \quad (24)$$

Assume that the eigenvalues of A_0 satisfy the following relationship:

$$0 = \lambda_1(\mathcal{G}) > \lambda_2(\mathcal{G}) \geq ... \geq \lambda_{2n}(\mathcal{G}). \quad (25)$$

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

$$|1 + T\lambda_i| < 1, \ i = 2...n. \quad (26)$$

Hence T should be bounded by

$$0 < T < \min\{\frac{-2}{\lambda_{2n}}, \frac{-2}{\lambda_{2bn}}\}. \quad (27)$$

If (24) and (27) are satisfied, A_1 is Schur-stable with only one eigenvalue at the unit disk. Although calculation of $\lambda_2(\mathcal{G})$ and $\lambda_{2n}(\mathcal{G})$ require $\sigma_2(\mathcal{G})$ and $\sigma_n(\mathcal{G})$ which are global information, $\sigma_2(\mathcal{G})$ and $\sigma_2(\mathcal{G})$ can be solved in a distributed way by [16] or [17] due to symmetric nature of $L(\mathcal{G})^2$. By knowing $\sigma_2(\mathcal{G})$ and $\sigma_n(\mathcal{G})$ in a distributed way, then common parameter pair γ, T can be found to satisfy (24) and (27). This guarantees (11) being fully distributed. Finally, the eigenvalues of A_1 lie strictly inside the unit circle except for one eigenvalue, which means system (16) is input-to-state stable as A_1 is proven to be Schur-stable.

Consider the state coordinate transformation

$$V^{-1}\begin{bmatrix}\overline{\boldsymbol{\alpha}}(k) \\ \overline{\boldsymbol{\omega}}(k)\end{bmatrix} = \begin{bmatrix}\overline{\boldsymbol{\alpha}}'(k) \\ \overline{\boldsymbol{\omega}}'(k)\end{bmatrix}, \quad (28)$$

where V is the similar transformation matrix with its first column being $\begin{bmatrix}\mathbf{1} \\ \mathbf{0}\end{bmatrix}$ and $V^{-1}A_1V = \Lambda = diag\{1, \hat{\lambda}_2, ..., \hat{\lambda}_{2n}\}$. By similarity transformation V, system (16) is transformed into

$$\begin{aligned} \begin{bmatrix}\overline{\boldsymbol{\alpha}}'(k+1) \\ \overline{\boldsymbol{\omega}}'(k+1)\end{bmatrix} &= \Lambda\begin{bmatrix}\overline{\boldsymbol{\alpha}}'(k) \\ \overline{\boldsymbol{\omega}}'(k)\end{bmatrix} + V^{-1}\Theta(k)\begin{bmatrix}\overline{\boldsymbol{\alpha}}(k) \\ \overline{\boldsymbol{\omega}}(k)\end{bmatrix} \\ &+ V^{-1}\begin{bmatrix}\boldsymbol{\Delta\alpha}(k) \\ \boldsymbol{\Delta\omega}(k)\end{bmatrix}. \end{aligned} \quad (29)$$

If we let $\boldsymbol{\eta}^1(k) = [\overline{\alpha}_2'(k), ..., \overline{\alpha}_n'(k)]$ and $\boldsymbol{\eta}^2(k) = [\overline{\omega}_1'(k), ..., \overline{\omega}_n'(k)]$, an equivalent form of system (29) is

$$\begin{aligned} \begin{bmatrix}\overline{\alpha}_1'(k+1) \\ \boldsymbol{\eta}^1(k+1) \\ \boldsymbol{\eta}^2(k+1)\end{bmatrix} &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \Lambda_1 & 0 \\ 0 & 0 & \Lambda_2 \end{bmatrix}\begin{bmatrix}\overline{\alpha}_1'(k) \\ \boldsymbol{\eta}^1(k) \\ \boldsymbol{\eta}^2(k)\end{bmatrix} + V^{-1}\Theta(k) \\ &\begin{bmatrix}\overline{\boldsymbol{\alpha}}(k) \\ \overline{\boldsymbol{\omega}}(k)\end{bmatrix} + V^{-1}\begin{bmatrix}\boldsymbol{\Delta\alpha}(k) \\ \boldsymbol{\Delta\omega}(k)\end{bmatrix}, \end{aligned} \quad (30)$$

where

$$\Lambda_1 = diag\{\hat{\lambda}_2, ..., \hat{\lambda}_n\}, \ \Lambda_2 = diag\{\hat{\lambda}_{n+1}, ..., \hat{\lambda}_{2n}\}. \quad (31)$$

The zero input response of system (30) is

$$R'_1 = \lim_{k\to\infty} \Lambda^k \begin{bmatrix} \overline{\alpha}'_1(0) \\ \boldsymbol{\eta}^1(0) \\ \boldsymbol{\eta}^2(0) \end{bmatrix}. \tag{32}$$

As $\{(\overline{\alpha}, \overline{\omega}) : \overline{\alpha} \in \textbf{span}\{\mathbf{1}\}$ and $\overline{\omega} = \mathbf{0}\}$ is the equilibrium subspace of system (16),

$$\lim_{k\to\infty} \Lambda^k \begin{bmatrix} \overline{\alpha}'_1(0) \\ \boldsymbol{\eta}^1(0) \\ \boldsymbol{\eta}^2(0) \end{bmatrix} = \lim_{k\to\infty} \Lambda^k \begin{bmatrix} \overline{\boldsymbol{\alpha}}(k) \\ \overline{\boldsymbol{\omega}}(k) \end{bmatrix} = \overline{\alpha}'_1(0) \begin{bmatrix} 1 \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}. \tag{33}$$

From (33), 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 $k \to \infty$. Therefore, by the coordinate transformation, it is obtained that the zero-input response of (16) 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 (16) is

$$R_2 = \lim_{k\to\infty} \sum_{i=0}^{n-1} \Lambda^{k-i-1} (\Theta(k) \begin{bmatrix} \overline{\boldsymbol{\alpha}}(k) \\ \overline{\boldsymbol{\omega}}(k) \end{bmatrix} + \begin{bmatrix} \boldsymbol{\Delta\alpha}(k) \\ \boldsymbol{\Delta\omega}(k) \end{bmatrix}). \tag{34}$$

According to Lemma 3.1,

$$\lim_{k\to\infty} \Theta(k) \begin{bmatrix} \overline{\boldsymbol{\alpha}}(k) \\ \overline{\boldsymbol{\omega}}(k) \end{bmatrix} = \lim_{k\to\infty} \begin{bmatrix} -T(A(\mathcal{G}) - \Pi_1(k))\overline{\boldsymbol{\alpha}}(k) \\ \mathbf{0} \end{bmatrix}. \tag{35}$$

Taking (17) and (18) into account, we can deduce that the zero state response is bounded by

$$|R_2| \le \begin{bmatrix} |\frac{\hat{\lambda}_1}{1-\hat{\lambda}_1}|(\frac{4\rho_1 T\omega_{sup} d_{max}}{1+\rho_1} + \rho_2\hat{\alpha}_{sup}) \\ \vdots \\ |\frac{\hat{\lambda}_n}{1-\hat{\lambda}_n}|(\frac{4\rho_1 T\omega_{sup} d_{max}}{1+\rho_1} + \rho_2\hat{\alpha}_{sup}) \\ |\frac{\hat{\lambda}_{n+1}}{1-\hat{\lambda}_{n+1}}|\rho_2\omega_{sup} \\ \vdots \\ |\frac{\hat{\lambda}_{2n}}{1-\hat{\lambda}_{2n}}|\rho_2\omega_{sup} \end{bmatrix}. \tag{36}$$

Therefore,

$$\begin{aligned} \lim_{k\to\infty} |\varepsilon_i(k)| &\le |\frac{\hat{\lambda}_i}{1-\hat{\lambda}_i}|(\frac{4\rho_1 T\omega_{sup} d_{max}}{1+\rho_1} + \rho_2\hat{\alpha}_{sup}) \\ &= |\frac{1+T\lambda_i}{T\lambda_i}|(\frac{4\rho_1 T\omega_{sup} d_{max}}{1+\rho_1} + \rho_2\hat{\alpha}_{sup}). \end{aligned} \tag{37}$$

This completes the proof of Theorem 3.1.

D. Offset compensation

After the end of clock skew compensation procedure, the synchronization errors of logical clock skews are bounded by a relative small steady error bounded by ρ_3 under time-varying input $\alpha_i(t)$s. However, the final goal of clock synchronization is to compensate for possible offset errors and let all virtual clocks show identical clock readings. In previous studies [9], [10], [13], et al. synchronization errors of logical clock skews asymptotically converge to 0 under constant input α_i. Hence their focus is to drive offset errors to 0 as well. However, under time-varying input $\alpha_i(t)$s, the offset error does not converge to 0. As a consequence, clock reading compensation mainly focuses on reducing the errors of compensated clock reading $\hat{\tau}_i(t)$. We present an clock reading compensation protocol which is integrated into skew compensation in Algorithm 2.

Algorithm 2 Clock reading compensation Protocol

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

Specifically, for each broadcast, node i updates its virtual clock reading $\hat{\tau}_i(k)$ using the average consensus equation and then it is added by $\hat{\alpha}_i(k)(\tau_i(k+1) - \tau_i(k))$ as time moves until the next round of clock reading compensation.

Remark 3.1: This approach is inspired by [11] but uses synchronous form and pseudo-synchronous implementation [13]. 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.

IV. SIMULATION

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

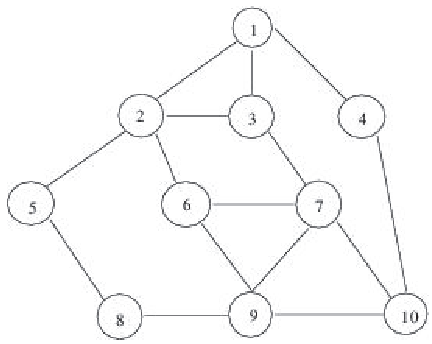

Fig. 1. Network topology composed of 10 labelled nodes

Initially we test the performance of clock skew compensation protocol under different ρ_2s. The parameter set is chosen as: $T = 0.1$, $\gamma = 4$. The initialization values of α_is are assumed to be randomly selected from $[0.9999, 0.99997] \cup [1.00003, 1.0001]$ since Crystal oscillators exhibit drift ρ_1 from $30\mu s$ to $100\mu s$ in one second. As each local clock skew experiences small drift, during one sampling period the local clock skew is added by a random noise ρ_2 as 0.01 ticks/s, 0.05 ticks/s, 0.1 ticks/s. It can be seen from Fig. 2 that it takes nearly 10 iterations (corresponding to 2 seconds) to reduce the maximum difference of skew below

1 ticks/s (1 ticks/s=1/32768Hz=30.5μs), i.e., the individual clock resolution.

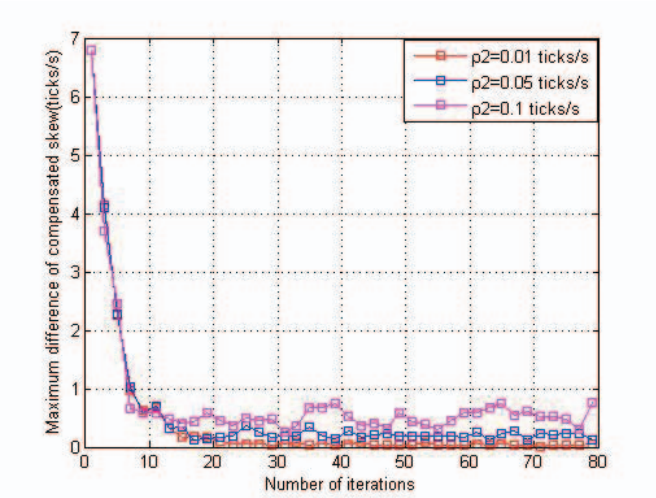

Fig. 2. Convergent performances of skew compensation

We also test the performance of clock reading compensation protocol. The random noise ρ_2s are chosen as 0.01 ticks/s, 0.05 ticks/s, 0.1 ticks/s. It can be seen from Fig. 3 that the maximum differences of clock readings finally converge to a small steady state error, which demonstrates a better performance in robustness against time-varying clock skew $\alpha_i(t)$s.

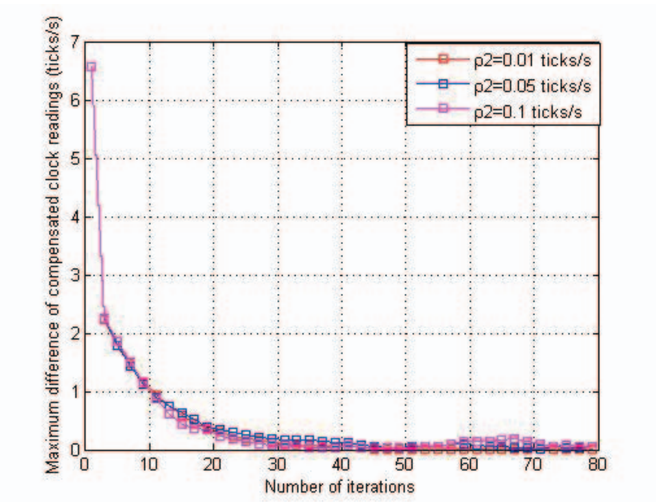

Fig. 3. Convergent performances of clock reading compensation

V. Conclusion

This paper studies clock synchronization via a filter-based approach which is fully distributed over wireless sensor network. A first-order controller is applied and the proposed protocol shows robustness under time-varying clock parameters. By analysing input-to-state stability, the convergence property of the control system is guaranteed by restricting synchronization error into a bounded range. Future research includes dealing with asynchronous implementation and checking whether the proposed protocol is still effective.

References

[1] B. Sundararaman, U. Buy, and A. D. Kshemkalyani, "Clock synchronization for wireless sensor networks: a survey," *Ad hoc Networks*, vol. 3, no. 3, pp. 281–323, 2005.

[2] J. Elson, L. Girod, and D. Estrin, "Fine-grained network time synchronization using reference broadcasts," *ACM SIGOPS Operating Systems Review*, vol. 36, no. SI, pp. 147–163, 2002.

[3] S. Ganeriwal, R. Kumar, and M. B. Srivastava, "Timing-sync protocol for sensor networks," in *Proceedings of the 1st international conference on Embedded networked sensor systems*. ACM, 2003, pp. 138–149.

[4] M. Maróti, B. Kusy, G. Simon, and Á. Lédeczi, "The flooding time synchronization protocol," in *Proceedings of the 2nd international conference on Embedded networked sensor systems*. ACM, 2004, pp. 39–49.

[5] R. Solis, V. S. Borkar, and P. Kumar, "A new distributed time synchronization protocol for multihop wireless networks," in *Proceedings of the 45th IEEE Conference on Decision and Control*. IEEE, 2006, pp. 2734–2739.

[6] A. Giridhar and P. Kumar, "Distributed clock synchronization over wireless networks: Algorithms and analysis," in *Proceedings of the 45th IEEE Conference on Decision and Control*. IEEE, 2006, pp. 4915–4920.

[7] N. M. Freris and A. Zouzias, "Fast distributed smoothing of relative measurements," in *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*. IEEE, 2012, pp. 1411–1416.

[8] Z.-W. Liu, Z.-H. Guan, T. Li, X.-H. Zhang, and J.-W. Xiao, "Quantized consensus of multi-agent systems via broadcast gossip algorithms," *Asian Journal of Control*, vol. 14, no. 6, pp. 1634–1642, 2012.

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

[10] J. He, P. Cheng, L. Shi, J. Chen, and Y. Sun, "Time synchronization in wsns: a maximum-value-based consensus approach," *IEEE Transactions on Automatic Control*, vol. 59, no. 3, pp. 660–675, 2014.

[11] M. K. Maggs, S. G. O'Keefe, and D. V. Thiel, "Consensus clock synchronization for wireless sensor networks," *IEEE sensors Journal*, vol. 12, no. 6, pp. 2269–2277, 2012.

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

[13] R. Carli and S. Zampieri, "Network clock synchronization based on the second-order linear consensus algorithm," *IEEE Transactions on Automatic Control*, vol. 59, no. 2, pp. 409–422, 2014.

[14] P. Sommer and R. Wattenhofer, "Gradient clock synchronization in wireless sensor networks," in *Proceedings of the 2009 International Conference on Information Processing in Sensor Networks*. IEEE Computer Society, 2009, pp. 37–48.

[15] N. M. Freris, V. S. Borkar, and P. Kumar, "A model-based approach to clock synchronization," in *Proceedings of the 48th IEEE Conference on Decision and Control*. IEEE, 2009, pp. 5744–5749.

[16] M. Franceschelli, A. Gasparri, A. Giua, and C. Seatzu, "Decentralized laplacian eigenvalues estimation for networked multi-agent systems," in *Proceedings of the 48th IEEE Conference on Decision and Control*. IEEE, 2009, pp. 2717–2722.

[17] T. Sahai, A. Speranzon, and A. Banaszuk, "Hearing the clusters of a graph: A distributed algorithm," *Automatica*, vol. 48, no. 1, pp. 15–24, 2012.