

Research Article

Achieving Relative Time Synchronization in Wireless Sensor Networks

Salman Ahmed,¹ Feng Xiao,² and Tongwen Chen¹

¹ Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, Canada T6G 2V4

² Department of Control Science and Engineering, Harbin Institute of Technology, Harbin 150080, China

Correspondence should be addressed to Salman Ahmed; salman2@ualberta.ca

Received 5 May 2013; Accepted 8 August 2013

Academic Editor: James Lam

Copyright © 2013 Salman Ahmed et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Time synchronization is required for a variety of distributed control applications in wireless sensor networks (WSNs). In this paper, a novel distributed time synchronization technique for WSNs is presented. The proposed technique does not require all the sensor nodes to have the same time periods or starting times. Furthermore, the communication topology of the WSN is not assumed to be static; it can change with time. The WSN is modeled by a time-varying discrete-time system. The objective is to design local control inputs to achieve relative time synchronization by using the neighbor sensor nodes information. By employing tools from nonnegative matrix and graph theories, the convergence analysis is presented. Numerical examples are presented to demonstrate the effectiveness of the proposed protocol.

1. Introduction

Distributed time synchronization in wireless sensor networks (WSNs) has been an important topic of research in the control community [1–3]. In most distributed control applications, coordination among sensor nodes requires precise time synchronization [4, 5]. Time synchronization in a WSN is also required for many other tasks such as data fusion, transmission scheduling, and nodes sleep and wake-up scheduling. Therefore, it is important to design distributed protocols to achieve time synchronization in a WSN.

A common approach to design time synchronization protocols is to create a hierarchical structure within the WSN. The most prominent examples are the reference broadcast Synchronization (RBS) [6], timing-sync protocol for sensor networks (TPSN) [7], and flooding time synchronization protocol (FTSP) [8]. The implementation of RBS requires that the WSN should be divided into distinct clusters and each cluster should have a leader or reference node, all the leader nodes synchronize themselves with respect to each other and sensor nodes within the same cluster synchronize themselves with respect to their leader node. The implementation of TPSN and FTSP requires that a spanning tree rooted at a reference node should be built in the WSN and the time difference of any sensor node can be obtained with respect to

the reference node. Although the hierarchy-based protocols have been experimentally tested in [8] and their performance is remarkable, they require substantial overhead to rebuild the spanning tree or clusters if a sensor node dies or a new sensor node is added.

Distributed time synchronization protocols which do not require a hierarchical structure have been proposed in [3, 9, 10], where the authors have considered a continuous time system to represent the clock of a sensor node. It is assumed that the clock of a sensor node has the same continuity property as the physical time and it can measure in any time interval. In actual implementation, the clock of a sensor node has some frequency or time period. A clock cannot measure a time interval which is shorter than its time period. Hence, it is more reasonable to use a discrete-time system to model a clock.

Discrete-time systems to model clocks in a WSN have been used in [1, 11, 12]. For the design of time synchronization protocol, the authors have assumed that all the sensor nodes should periodically exchange and update their clock states at the same time instants. This assumption is unrealistic since it requires synchronization of all clocks, which is the goal of the protocol. In this paper, we propose a time synchronization protocol which relaxes the three main assumptions used in

the literature. First, the proposed protocol does not require all the sensor nodes to have the same time periods or starting times. Second, all the sensor nodes are not restricted to exchange and update their clock states at the same time instants. Third, the communication topology of the WSN is not assumed to be static. We use the discrete-time clock model presented in [1]. We assume that each sensor node knows its time period but does not know its exact starting time. Each sensor node exchanges its clock readings with its neighbors. The objective of the time synchronization protocol is to synchronize the clocks in a WSN with respect to a virtual clock, which is not physically present or accessible to any sensor node. The virtual clock is only used to demonstrate the concept of time synchronization in a WSN as the time periods and starting times of sensor nodes can be different.

The remainder of this paper is structured as follows: Section 2 presents some basic definitions and results from graph and matrix theories. The problem is formulated in Section 3. The main results and analysis are presented in Section 4. Numerical examples to demonstrate the effectiveness of the proposed protocol are presented in Section 5. Finally, concluding remarks are given in Section 6.

2. Preliminaries

In this section, we present some definitions and results from graph and matrix theories which will be used in this paper.

Let $\mathbf{1} = [1, \dots, 1]^T \in \mathbb{R}^n$, $I_n \in \mathbb{R}^{n \times n}$ denote an identity matrix, and the symbol \mathbb{N} denote the set of nonnegative integers. A matrix S is nonnegative if all its elements are nonnegative [13]. A non-negative, finite, and square matrix S is called stochastic if the sum of each row is 1. A stochastic matrix S is called indecomposable and aperiodic (SIA) if there exists a column vector w such that $\lim_{k \rightarrow \infty} S^k = \mathbf{1}w^T$ [14]. Let $\prod_{i=1}^k S_i = S_k, S_{k-1}, \dots, S_1$ denote the left product of the matrices S_k, S_{k-1}, \dots, S_1 . For any matrix S , we write s_{pq} to represent the element which lies in the p th row and q th column of S .

Consider an undirected (directed) graph \mathcal{G} which consists of a vertex set, $\mathcal{V}(\mathcal{G}) = \{v_1, \dots, v_n\}$, and an edge set, $\mathcal{E}(\mathcal{G}) \subset \{(v_i, v_j) : v_i, v_j \in \mathcal{V}(\mathcal{G})\}$ where an edge is an unordered (ordered) pair of vertices [15]. The index set of neighbors of vertex v_i is denoted by $\mathcal{N}_i = \{j : (v_j, v_i) \in \mathcal{E}(\mathcal{G}), j \neq i\}$ and its cardinality by $|\mathcal{N}_i|$. A path from v_{i_1} to v_{i_k} is a sequence, v_{i_1}, \dots, v_{i_k} of distinct vertices such that $(v_{i_j}, v_{i_{j+1}}) \in \mathcal{E}(\mathcal{G})$ for any $j = 1, \dots, k-1$. An undirected (directed) graph \mathcal{G} is connected (strongly connected) if there is a path between any two distinct vertices. A directed tree is a directed graph where every vertex except the root vertex has exactly one parent vertex and the root vertex can be connected to any other vertices via paths. A spanning tree of directed graph \mathcal{G} is a directed tree whose edge set belong to $\mathcal{E}(\mathcal{G})$ and vertex set is the same as $\mathcal{V}(\mathcal{G})$. The Laplacian matrix of \mathcal{G} is denoted by $L = [l_{pq}] \in \mathbb{R}^{n \times n}$ and is defined as follows:

$$l_{pq} = \begin{cases} -1 & \text{if } p \neq q, q \in \mathcal{N}_p, \\ |\mathcal{N}_p| & \text{if } p = q, \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

Let $S = [s_{pq}] \in \mathbb{R}^{n \times n}$ be a nonnegative matrix. Then, $\mathcal{G}(S)$ is defined as a directed graph having vertex set $\mathcal{V}(\mathcal{G}(S)) = \{v_1, \dots, v_n\}$, and edges from v_q to v_p if and only if $s_{pq} > 0$. We present some lemmas which will be used later.

Lemma 1 (see [16]). *Let S be a stochastic matrix. If $\mathcal{G}(S)$ contains at least one spanning tree such that the root vertex of that spanning tree has a self-loop in $\mathcal{G}(S)$, then S is SIA.*

Lemma 2 (see [14]). *Let $\Gamma = \{S_1, S_2, \dots, S_k\}$ be a finite set of SIA matrices such that for each sequence $S_{i_1}, S_{i_2}, \dots, S_{i_j}$ of positive length, the matrix product $S_{i_j}, S_{i_{j-1}}, \dots, S_{i_1}$ is SIA. Then, for each infinite sequence S_{i_1}, S_{i_2}, \dots , there exists a column vector w such that the following holds:*

$$\lim_{j \rightarrow \infty} S_{i_j}, S_{i_{j-1}}, \dots, S_{i_1} = \mathbf{1}w^T. \quad (2)$$

3. Problem Formulation

We consider a WSN having n sensor nodes. Each sensor node has a local clock which consists of a software program and a hardware oscillator. The oscillator periodically generates time interrupts using its time period and the software computes the local time of the sensor node at those time instants when the interrupts are generated [1]. Let τ_i , x_i and t_0^i denote the time period, clock value, and starting time of sensor node i , respectively, where $i = 1, \dots, n$. Each sensor node knows its time period but does not exactly know its starting time. Rather, its estimate, denoted by \hat{t}_0^i for sensor node i , is known. Let $T_i = \{\tilde{t}_k^i : k = 0, 1, \dots\}$ represent the sequence of time instants when the clock of sensor node i updates its state, where $\tilde{t}_k^i := k\tau_i + t_0^i$. The time computed locally by sensor node i can be expressed as follows:

$$x_i(\tilde{t}_{k+1}^i) = x_i(\tilde{t}_k^i) + \tau_i, \quad x_i(\tilde{t}_0^i) = \hat{t}_0^i, \quad i = 1, \dots, n. \quad (3)$$

If each sensor node does not exactly know its starting time, then the clocks in the WSN would not be synchronized. To achieve time synchronization, we apply a control input to adjust the clock readings of sensor nodes. The control input is applied periodically at a certain known integer multiple of the local clock time period. Let $t_s^i := \tilde{t}_{0+sz_i}^i$ be the time instants when the control input is applied to the clock of sensor node i , where z_i is a known integer, $s \in \mathbb{N}$ and $i = 1, \dots, n$. We can write the following:

$$x_i(t_{s+1}^i) = x_i(t_s^i) + z_i\tau_i + u_i(t_s^i), \quad x_i(t_0^i) = x_i(\tilde{t}_0^i), \quad (4)$$

where $u_i(t_s^i)$ denotes the control input or protocol which is needed to be designed. A virtual continuous time clock can be expressed as follows:

$$v(t) = at + b, \quad (5)$$

where t represents the time of WSN, v represents the clock readings, and a and b represent the time period and initial offset of the virtual clock, respectively. The virtual clock is not available to any sensor node. It is only used to demonstrate

the concept of relative time synchronization. The clocks in a WSN are said to be synchronized if there exists a common virtual clock in the form of (5) such that the following holds:

$$\lim_{\bar{t}_k^i \rightarrow \infty} (x_i(\bar{t}_k^i) - (\bar{t}_k^i + b)) = 0, \quad \forall i = 1, \dots, n. \quad (6)$$

As each clock knows its own time period, we use $a = 1$ for the common virtual clock. The sensor nodes communicate with their neighbors and exchange clock readings. The communication topology of the WSN can be modeled by a time-varying undirected graph, $\mathcal{G}(t_s^i)$, where $i = 1, \dots, n$ and $s \in \mathbb{N}$. The neighbors of sensor node j at t_s^i consist of those sensor nodes which are able to communicate with node j at t_s^i , where $i, j = 1, \dots, n$ and $s \in \mathbb{N}$. Let $\mathcal{N}_i(t_s^i)$ denote the index set of sensor node i neighbors and let $L(t_s^i)$ represents the Laplacian matrix associated with $\mathcal{G}(t_s^i)$. The following assumptions are required to ensure that the clocks can be synchronized.

- (A1) There exists a known integer κ such that each sensor node i can obtain information at least once from any possible neighbor during $[t_s^i, t_{s+\kappa}^i]$, where $i = 1, \dots, n$ and $s \in \mathbb{N}$.
- (A2) There exists an integer $p \geq 0$ such that the union of graphs $\mathcal{G}(t_s^i), \mathcal{G}(t_{s+1}^i), \dots, \mathcal{G}(t_{s+p}^i)$, is connected, for all $i = 1, \dots, n$, and $s \in \mathbb{N}$.
- (A3) For any two sensor nodes i and j , the following holds:

$$t_s^i \in T_j, \quad \forall s. \quad (7)$$

It should be noted that (A1) is made to ensure that each sensor node can receive information from its possible neighbors in a finite interval of time. Assumption (A2) is made to ensure that union of communication topologies during a specific time interval remains connected, although at some particular time instants the communication topology may not be connected. Assumption (A3) is made so that by using the proposed protocol the relative time synchronization error in the WSN can be driven to zero. In actual implementation, if (A3) can not be ensured to hold for all sensor nodes, then it can be relaxed to consider neighboring nodes only. If (A3) is not ensured for neighboring nodes, then a small approximation error may exist due to the fact that the starting times and time periods of the sensor nodes may be different. In the next section, we present the main results to achieve relative time synchronization.

4. Main Results

We present the following protocol for achieving time synchronization:

$$u_i(t_s^i) = \alpha_i(t_s^i) \sum_{j \in \mathcal{N}_i(t_s^i)} (x_j(t_s^i) - x_i(t_s^i)), \quad i = 1, \dots, n, \quad (8)$$

where $\alpha_i(t_s^i)$ denotes a gain which needs to be designed. For synchronized distributed control systems, a protocol similar

to (8) has been presented in [17]. In this paper, we study protocol (8) for asynchronous distributed control systems. The following theorem presents the main results.

Theorem 3. Consider a WSN which satisfies assumptions (A2) and (A3). Then, the clocks of the sensor nodes achieve relative time synchronization using protocol (8) if $\alpha_i(t_s^i) \in (0, 1/|\mathcal{N}_i(t_s^i)|)$, $\forall i = 1, \dots, n$ and $s \in \mathbb{N}$.

Remark 4. If the communication topology of the WSN is static, then we can allow $\alpha_i(t_s^i) \in (0, 1/d)$, $\forall i = 1, \dots, n$ and $s \in \mathbb{N}$, where d denotes the maximum of the diagonal entries of L .

The proof of Theorem 3 consists of three main steps. In the first step, we combine the local clock models and define a new time sequence to represent the local clock models by a single time-varying system. In the second step, we use the new time sequence to compute an upper bound on the time interval in which each sensor node obtains information at least once from all its neighbors. Finally, we complete the proof by utilizing the properties of connected graphs and product of SIA matrices.

4.1. Representation by a Single System. Before grouping the local clock models to represent them by a single system, we define a new state as follows:

$$y_i(t_s^j) = x_i(t_s^j) - (t_s^j - t_0), \quad i, j = 1, \dots, n. \quad (9)$$

We rewrite the model expressed by (4) and control protocol (8) as follows:

$$\begin{aligned} y_i(t_{s+1}^i) &= y_i(t_s^i) + u_i(t_s^i), & y_i(t_0^i) &= x_i(t_0^i), \\ u_i(t_s^i) &= \alpha_i(t_s^i) \sum_{j \in \mathcal{N}_i(t_s^i)} (y_j(t_s^i) - y_i(t_s^i)), & i &= 1, \dots, n. \end{aligned} \quad (10)$$

Let $t_h = \max\{t_0^i, i = 1, \dots, n\}$, and we define a new time sequence $T := \{t_r\}$, where $r \in \mathbb{N}$, which is the union of the distinct values of the sequences $\{t_s^i \geq t_h, i = 1, \dots, n, s \in \mathbb{N}\}$. We order the values of T such that $t_r < t_{r+1}$. We mention here that the distances between consecutive elements of T are nonuniform. The time-synchronization objective can be rewritten as follows:

$$\lim_{t_r \rightarrow \infty} (y_i(t_r) - b) = 0, \quad i = 1, \dots, n, \quad (11)$$

for some b . Let $y(t_r) = [y_1(t_r), \dots, y_n(t_r)]^T$ denote the group state vector. The dynamics of the overall system can be written as follows:

$$y(t_{r+1}) = (I_n - A(t_r) L(t_r)) y(t_r), \quad (12)$$

where $A(t_r) = [a_{pq}]$, $L(t_r) = [l_{pq}]$,

$$a_{pq} = \begin{cases} \alpha_p(t_r) & \text{if } p = q, \exists s \text{ s.t. } t_r = t_s^p, \\ 0 & \text{otherwise,} \end{cases}$$

$$l_{pq} = \begin{cases} -1 & \text{if } q \in \mathcal{N}_p(t_r), \exists s \text{ s.t. } t_r = t_s^p, \\ |\mathcal{N}_p(t_r)| & \text{if } p = q, \exists s \text{ s.t. } t_r = t_s^p, \\ 0 & \text{otherwise.} \end{cases} \quad (13)$$

It should be noted that $L(t_r)$ is a time-varying Laplacian matrix which can be unsymmetric and the graph associated with $L(t_r)$ may not be strongly connected. Let $\Phi(t_r) = I_n - A(t_r)L(t_r)$, the solution of (12) can be written as follows:

$$y(t_r) = \Phi(t_r) \Phi(t_{r-1}) \cdots \Phi(t_0) y(t_0), \quad r = 0, 1, \dots \quad (14)$$

If $\alpha_i(t_s^i) \in (0, 1/|\mathcal{N}_i(t_s^i)|) \forall i, s$, then $\Phi(t_r)$ are always stochastic matrices with positive diagonal entries. The product $\Phi(t_r)\Phi(t_{r-1})\Phi(t_{r-2}) \cdots \Phi(t_0)$ is also a stochastic matrix with positive diagonal entries. With this preparation, we proceed to the next step.

4.2. Computation of an Upper Bound on the Time Interval. The following lemma is quite useful in the computation of an upper bound on the time interval in which each sensor node can obtain information at least once from its neighbors.

Lemma 5. *Let $g = \max\{z_i\tau_i\} / \min\{z_i\tau_i\}$ where $i = 1, \dots, n$. For any sensor node $i = 1, \dots, n$ and $s = 1, 2, \dots$, the number of elements in the set $\{t_p : t_p \in [t_s^i, t_{s+1}^i]\}$ is not greater than h , where $h = (\lceil g \rceil)(n-1) + 1$ and $\lceil g \rceil$ denotes the smallest integer not less than g .*

Proof. Let j be the sensor node such that $z_j\tau_j \geq z_i\tau_i$ where $i = 1, \dots, n$, $i \neq j$. When sensor node j updates its control input at t_s^j , the maximum number of times any sensor node i can update its control input before the time instant t_{s+1}^j cannot exceed $\lceil g \rceil$, where $i = 1, \dots, n$, $i \neq j$. Since there are $n-1$ possible values of sensor node i (excluding $i = j$), we conclude that the number of elements in $\{t_p : t_p \in [t_s^j, t_{s+1}^j]\}$ is not greater than h . \square

Using Lemma 5 and assumption (A1), we can state that each sensor node obtains information from all its neighbors at least once during the time interval $[t_{m-\kappa h}, t_m]$, where $m = \kappa h, \kappa h + 1, \dots$

4.3. Technical Analysis. Let $\bar{\mathcal{G}}(t_m)$ be the graph obtained by the union of $\mathcal{G}(\Phi(t_m)), \dots, \mathcal{G}(\Phi(t_{m-\kappa h}))$. Using (A2), we conclude that $\bar{\mathcal{G}}(t_m)$ is strongly connected. If $\bar{\mathcal{G}}(t_m)$ is strongly connected, then, by using Lemma 1 together with [18, Lemma 3.9], we establish that the product of $\Phi(t_m) \cdots \Phi(t_{m-q})$ is SIA.

Let $F_0 = \prod_{l=0}^{h-1} \Phi(t_l)$, $F_1 = \prod_{l=h}^{2h-1} \Phi(t_l)$, $F_2 = \prod_{l=2h}^{3h-1} \Phi(t_l)$ and so on. Then, it can be easily shown that all possible unique F_j 's are finite, where $j = 0, 1, \dots$. Any product which involves

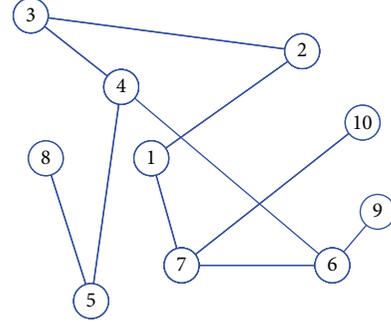


FIGURE 1: Communication topology of a WSN considered in Example 1 having 10 sensor nodes. The lines indicate the presence of communication channels for exchanging clock readings.

a finite combination of the unique F_j 's is SIA. So, we can apply Lemma 2 to get the following conclusion:

$$\lim_{k \rightarrow \infty} \prod_{j=0}^k F_j = \mathbf{1}w^T. \quad (15)$$

The vector w depends on the order in which the matrices $\Phi(t_l)$ are multiplied, where $l = 0, 1, \dots$, and it cannot be computed in advance. We can further write the following:

$$\lim_{t_r \rightarrow \infty} y(t_r) = \mathbf{1}w^T y(t_0) = \begin{pmatrix} w^T y(t_0) \\ \vdots \\ w^T y(t_0) \end{pmatrix}. \quad (16)$$

Hence, we complete the proof of time synchronization protocol by taking $b = w^T y(t_0)$.

5. Numerical Examples

In this section, we present two examples to demonstrate the effectiveness of the proposed time synchronization protocol.

Example 1 (static topology, small-size WSN). We consider a WSN having 10 sensor nodes and label them as $1, \dots, 10$. We assume that the communication topology of the WSN is not changing with respect to time ($\kappa = 0$) and is shown in Figure 1. Using Theorem 3, we choose $\alpha_i = 0.3, \forall i$. The values for different parameters satisfying (A2) and (A3) are listed in Table 1. By applying protocol (8), the WSN was simulated for 50 seconds. Figure 2 shows a plot of the clock readings for 50 seconds. Figure 3 illustrates a plot of the local control inputs applied to the sensor nodes. Based on the plots, it can be concluded that this WSN achieves distributed time synchronization using the proposed protocol.

Example 2 (switching topology, medium-size WSN). We consider a WSN having 45 sensor nodes and label them with $1, \dots, 45$. We consider that the communication links in the WSN are changing randomly with respect to time. Figure 4 shows the union of possible communication topologies of the WSN. We randomly select the parameter values for different sensor nodes from Table 1 and choose $\kappa = 3$. For simulation purposes, we compute the relative time synchronization error

TABLE 1: Parameter values used to simulate the WSN as shown in Figure 1.

Sensor nodes	1	2	3	4	5	6	7	8	9	10
Time period (sec)	0.01	0.02	0.01	0.05	0.04	0.02	0.1	0.01	0.2	0.4
Actual starting time (sec)	0.1	0	1.3	1.01	1	0.04	0.12	0.4	0.16	0
Estimated starting time (sec)	1.5	-12.5	3.19	-25.19	10.04	-7.34	2.19	3.25	0	14.58
Value of z	80	100	100	100	80	100	80	10	10	10

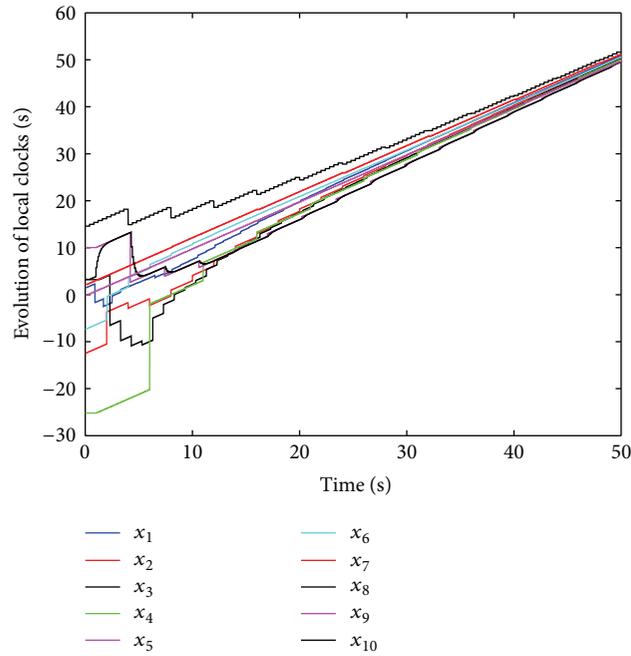


FIGURE 2: Plot of the local clock readings of a WSN considered in Example 1. The topology of the WSN is shown in Figure 1.

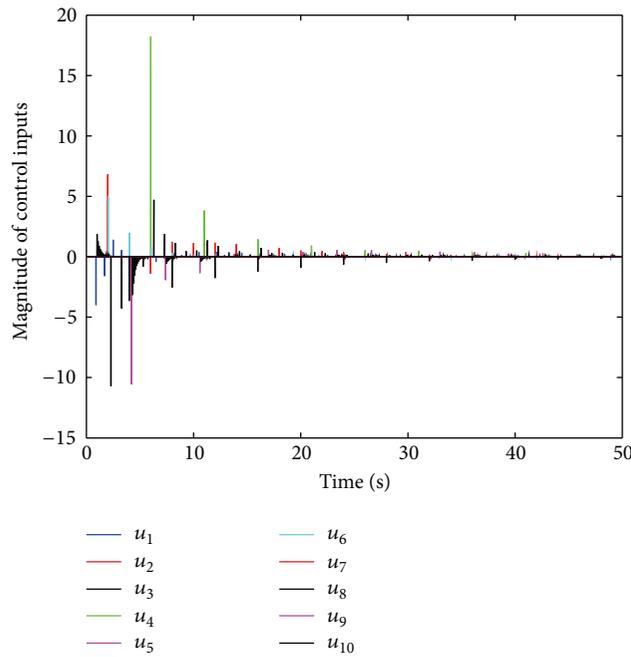


FIGURE 3: Plot of the local control inputs applied to the WSN considered in Example 1. The topology of the WSN is shown in Figure 1.

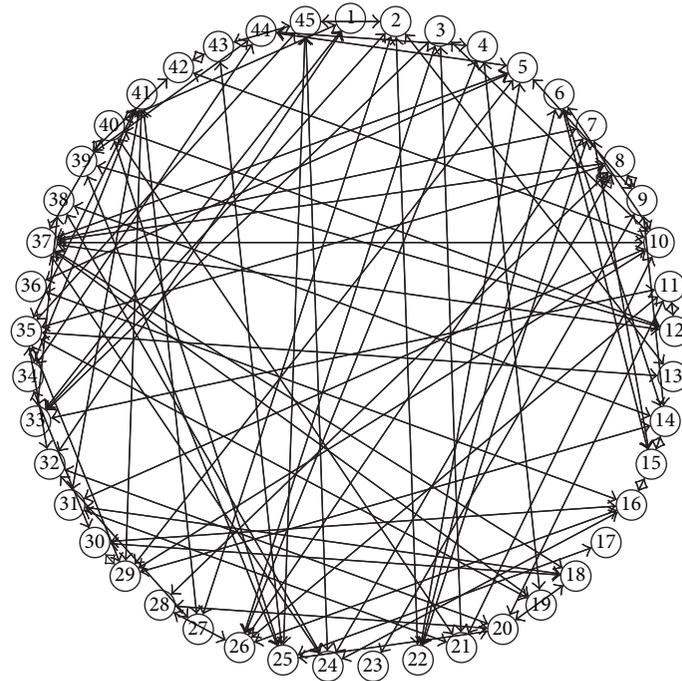


FIGURE 4: The graph of union of communication topologies of a WSN considered in Example 2 having 45 sensor nodes labeled as $1, \dots, 45$. The lines indicate the presence of communication channels only and do not indicate if at a particular time instant any information is exchanged on the communication link.

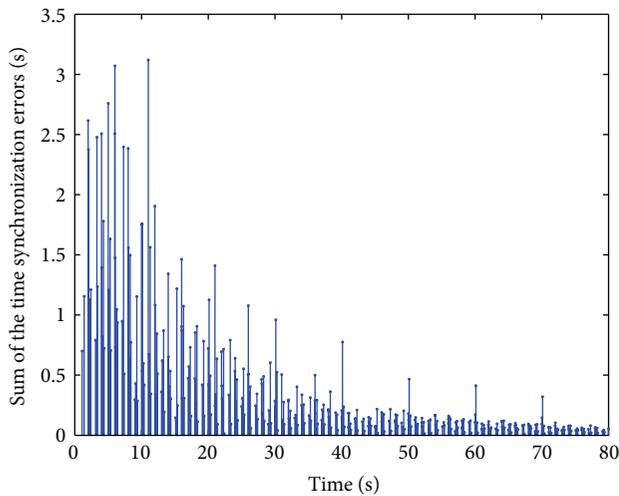


FIGURE 5: Plot of the sum of the relative time synchronization errors in the WSN considered in Example 2. The graph of the union of communication topologies of the WSN is shown in Figure 4.

of sensor node i with respect to its neighbor nodes, denoted by $e_i(t_r)$, as follows:

$$e_i(t_r) = \sum_{j \in \mathcal{N}(\mathcal{E}_i(t_r))} |(x_i(t_r) - x_j(t_r))|, \quad i = 1, \dots, 45. \quad (17)$$

Applying Theorem 3, we simulate the WSN for 80 seconds and select the values for α_i at run time based on the communication topology. Figure 5 shows a plot of the sum of

time synchronization errors. Based on the plots, it can be concluded that, by using protocol (8) and applying Theorem 3, this WSN also achieves distributed time synchronization.

6. Conclusions and Future Research Directions

In this paper, we presented a distributed time synchronization protocol for WSNs. The time periods and starting times of the sensor nodes were not restricted to be the same. By employing tools from stochastic matrix and algebraic graph theories, we presented the convergence analysis. In this paper, we considered that a sensor node could access the clock readings of its available neighbors without any delays. In actual implementation, time delay occurs between transmission and reception of information in a WSN. In the future work, we can consider this by modifying the protocol to handle time-delayed measurements from neighbors.

Acknowledgment

This research work was supported by the Natural Sciences and Engineering Research Council of Canada. Feng Xiao's work was supported in part by NSFC (61273030).

References

- [1] R. Carli, A. Chiuso, L. Schenato, and S. Zampieri, "A PI consensus controller for networked clocks synchronization," in *Proceedings of the 17th World Congress, International Federation of Automatic Control (IFAC '08)*, pp. 10289–10294, July 2008.

- [2] 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.
- [3] I. Shames and A. N. Bishop, "Distributed relative clock synchronization in wireless sensor networks," in *Proceedings of the World Congress, International Federation of Automatic Control (IFAC '11)*, pp. 11 256–11 270, 2011.
- [4] B. Ekstrand, "Some aspects on filter design for target tracking," *Journal of Control Science and Engineering*, vol. 2012, Article ID 870890, 15 pages, 2012.
- [5] L. Sheng, Y. J. Pan, and X. Gong, "Consensus formation control for a class of networked multiple mobile robot systems," *Journal of Control Science and Engineering*, vol. 2012, Article ID 150250, 12 pages, 2012.
- [6] J. Elson, L. Girod, and D. Estrin, "Fine-grained network time synchronization using reference broadcasts," in *Proceedings of the 5th Symposium on Operating Systems Design and Implementation (OSDI '02)*, vol. 36, pp. 147–163, 2002.
- [7] 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 (SenSys '03)*, pp. 138–149, November 2003.
- [8] 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 (SenSys '04)*, pp. 39–49, 2004.
- [9] R. Solist, V. S. Borkar, and P. R. Kumar, "A new distributed time synchronization protocol for multihop wireless networks," in *Proceedings of the 45th IEEE Conference on Decision and Control (CDC '06)*, pp. 2734–2739, December 2006.
- [10] L. Schenato and G. Gamba, "A distributed consensus protocol for clock synchronization in wireless sensor network," in *Proceedings of the 46th IEEE Conference on Decision and Control (CDC '07)*, pp. 2289–2294, December 2007.
- [11] O. Simeone, U. Spagnolini, Y. Bar-Ness, and S. H. Strogatz, "Distributed synchronization in wireless networks: global synchronization via local connections," *IEEE Signal Processing Magazine*, vol. 25, no. 5, pp. 81–97, 2008.
- [12] S. Bolognani, R. Carli, E. Lovisari, and S. Zampieri, "A randomized linear algorithm for clock synchronization in multi-agent systems," in *Proceedings of the IEEE Conference on Decision and Control (CDC '12)*, pp. 20–25, Maui, Hawaii, 2012.
- [13] R. A. Horn and C. R. Johnson, *Matrix Analysis*, Cambridge University Press, Cambridge, UK, 2012.
- [14] J. Wolfowitz, "Product of indecomposable, aperiodic and stochastic matrices," *Proceedings of the American Mathematical Society*, vol. 14, no. 5, pp. 733–737, 1963.
- [15] M. Mesbahi and M. Egerstedt, *Graph Theoretic Methods in Multiagent Networks*, Princeton University Press, Princeton, NJ, USA, 2010.
- [16] F. Xiao and L. Wang, "State consensus for multi-agent systems with switching topologies and time-varying delays," *International Journal of Control*, vol. 79, no. 10, pp. 1277–1284, 2006.
- [17] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, 2007.
- [18] W. Ren and R. W. Beard, "Consensus seeking in multiagent systems under dynamically changing interaction topologies," *IEEE Transactions on Automatic Control*, vol. 50, no. 5, pp. 655–661, 2005.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

