FULLY DISTRIBUTED CLOCK SKEW AND OFFSET ESTIMATION IN WIRELESS SENSOR NETWORKS

Jian Du and Yik-Chung Wu

Department of Electrical and Electronic Engineering,
The University of Hong Kong
Email:{dujian,ycwu}@eee.hku.hk

ABSTRACT

In this paper, we propose a fully distributed algorithm for joint clock skew and offset estimation in wireless sensor networks. With the proposed algorithm, each node can estimate its clock skew and offset by communicating only with its neighbors. Such algorithm does not require any centralized information processing or coordination. Simulation results show that estimation mean-square-error at each node converge to the centralized Cramér-Rao bound with only a few number of message exchanges.

Index Terms— Clock synchronization, wireless sensor network, factor graphs.

1. INTRODUCTION

Wireless sensor networks (WSNs) have been widely used in environmental and emergency monitoring [1] [2], event detection [3] and object tracking [4]. To perform distributed information processing in WSNs, a common clock across the network is usually required. Unfortunately, clock in each sensor node has its own imperfection and both clock offset (phase difference) and clock skew (frequency difference) are present. Therefore, clock synchronization [5] is a crucial component in WSNs.

Traditionally, clock synchronization in WSNs relies on spanning tree or clustered-based structure. Under such structures, synchronization is achieved through layer-by-layer pairwise synchronization [8]- [12]. Such protocols, like time-synchronization protocol for sensor network (TPSN) [6] and pairwise broadcast synchronization (PBS) [7], suffer large overhead in building and maintaining the tree or cluster structure, and are vulnerable to sudden node failures.

Without global structure or special nodes, fully distributed synchronization based on averaged consensus algorithms have been proposed in [13], [14]. However, as shown in [15], consensus protocol is not optimal and the performance will deteriorate when message delay exists. More recently, [16] pioneered the fully distributed network-wide clock offset estimation algorithm based on belief propagation (BP), and found that its performance is superior to consensus algorithms.

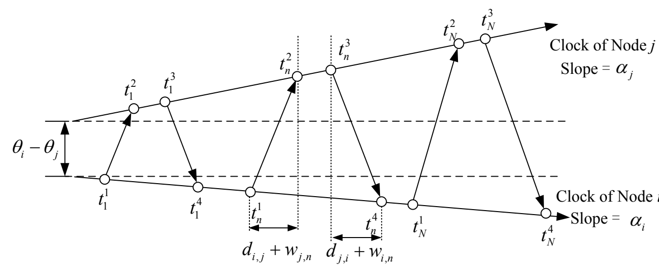

Fig. 1. Two way message exchange between node i and j in the WSNs.

In view of the fact that ignoring the effect of clock skew would significantly increase the re-synchronization frequency, in this paper, we take a step further and derive a fully distributed estimation algorithm for both clock skew and clock offset in WSNs. It is shown that the performance of the derived algorithm touches the centralized Cramér-Rao bound (CRB).

2. SYSTEM MODEL

Considering a general multi-hop sensor network with M sensor nodes distributed in 2-dimensional space. Each node can communicate with its neighboring nodes that lie within its communication range and any two nodes can communicate with each other through finite hops. With imperfection of oscillators and environmental changes, each node has a local clock with possibly different clock skew and offset. The relationship between real time t and the local clock reading is modeled as

$$c_i(t) = \alpha_i t + \theta_i, \qquad i = 1, \cdots, M, \tag{1}$$

where α_i and θ_i are the clock skew and offset of node i, respectively.

To estimate and compensate such clock skews and offsets, two-way time-stamp message exchange mechanism has been proposed for pairwise clock synchronization [7]. Specifically, as shown in Fig. 1, between one-hop neighboring nodes i and

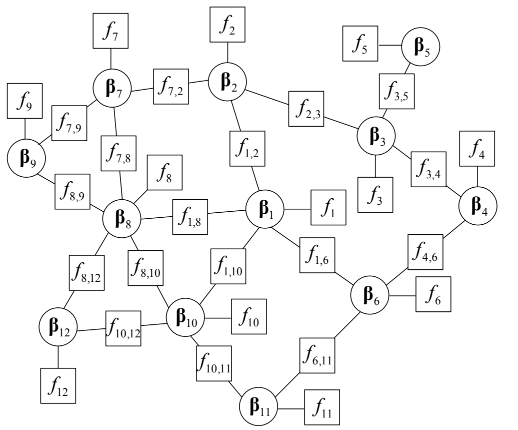

Fig. 2. The factor graph for the network of a WSNs.

j, at the n^{th} round of time-stamp exchange, node i sends a synchronization message to node j at t_n^1 with its local clock reading $c_i(t_n^1)$ embedded in the message. Node j records its time $c_j(t_n^2)$ at the reception of that message and replies to node i at $c_j(t_n^3)$. The replied message contains both time stamps $c_j(t_n^2)$ and $c_j(t_n^3)$. Then, node i records the reception time from node j's reply as $c_i(t_n^4)$. N rounds of such message exchange are performed between each pair of nodes to establish a relationship between the nodes i's and j's clocks. In particular, for the n^{th} round time-stamp exchange, we can write

$$\begin{aligned} \tfrac{1}{\alpha_j}[c_j(t_n^2)-\theta_j] &= \tfrac{1}{\alpha_i}[c_i(t_n^1)-\theta_i]+d_{i,j}+w_{j,n}, \\ \tfrac{1}{\alpha_j}[c_j(t_n^3)-\theta_j] &= \tfrac{1}{\alpha_i}[c_i(t_n^4)-\theta_i]-d_{j,i}-w_{i,n}, \end{aligned} \tag{2}$$

where $w_{j,n}$ and $w_{i,n}$ denote independent and identically distributed (i.i.d.) Gaussian random delay during the n^{th} round of time-stamp exchange, with zero mean and variances σ_j^2, σ_i^2, respectively; $d_{i,j}$ and $d_{j,i}$ represent the fixed message delay during node i/j sends message to node j/i, respectively. Under the assumption that the network topology does not change during clock synchronization process, we have $d_{i,j} = d_{j,i}$. Adding the two equations in (2) and stacking all the equations for all N rounds of time-stamp exchange, we obtain

$$\boldsymbol{A}_{j,i}\boldsymbol{\beta}_j + \boldsymbol{A}_{i,j}\boldsymbol{\beta}_i = \boldsymbol{z}_{j,i}, \tag{3}$$

where $\boldsymbol{A}_{j,i}$ and $\boldsymbol{A}_{i,j}$ are N-by-2 matrix with the n^{th} row being $[c_j(t_n^2)+c_j(t_n^3), -2]$ and $-[c_i(t_n^2)+c_i(t_n^3), -2]$, respectively; $\boldsymbol{\beta}_j \triangleq [\frac{1}{\alpha_j}, \frac{\theta_j}{\alpha_j}]^T$ and $\boldsymbol{\beta}_i \triangleq [\frac{1}{\alpha_i}, \frac{\theta_i}{\alpha_i}]^T$; and $\boldsymbol{z}_{j,i}$ is an N dimensional vector with the n^{th} element being $w_{j,n} - w_{i,n}$. Since $w_{j,n}$ and $w_{i,n}$ are both i.i.d. Gaussian, it is easy to obtain $\boldsymbol{z}_{j,i} \sim \mathcal{N}(\boldsymbol{z}_{j,i}; \boldsymbol{0}, \sigma_{i,j}^2\boldsymbol{I}_N)$, where $\sigma_{i,j}^2 = \sigma_i^2 + \sigma_j^2$. The goal is to establish global synchronization (i.e., estimate $\boldsymbol{\beta}_i$ in each node) based on local observations $\boldsymbol{A}_{j,i}$ and $\boldsymbol{A}_{i,j}$.

3. DISTRIBUTED ESTIMATION

In this section, distributed clock parameter estimation algorithm is derived based on BP. In the following, message exchange means BP message passing since two-way time-stamp exchange has been completed.

3.1. BP Framework

For the reason that the established clock relationships during two-way time-stamp exchanges involve interaction between neighboring nodes, the optimal clock estimate at each node requires the marginalization of joint posterior distribution of all $\boldsymbol{\beta}_i$, which is

$$\begin{aligned} g_i(\boldsymbol{\beta}_i) \;=\; & \int \ldots \int \prod_{i=1}^{M} p(\boldsymbol{\beta}_i) \prod_{i,\, j \text{ are neighbors}} p(\boldsymbol{A}_{i,j}, \boldsymbol{A}_{j,i}|\boldsymbol{\beta}_i, \boldsymbol{\beta}_j) \\ & d\boldsymbol{\beta}_1 \ldots d\boldsymbol{\beta}_{i-1} d\boldsymbol{\beta}_{i+1} d\boldsymbol{\beta}_M, \end{aligned} \tag{4}$$

where $p(\boldsymbol{\beta}_i)$ is the prior distribution of $\boldsymbol{\beta}_i$; $p(\boldsymbol{A}_{i,j}, \boldsymbol{A}_{j,i}|\boldsymbol{\beta}_i, \boldsymbol{\beta}_j) = \mathcal{N}(\boldsymbol{A}_{j,i}\boldsymbol{\beta}_j; \boldsymbol{A}_{i,j}\boldsymbol{\beta}_i, \sigma_{i,j}^2\boldsymbol{I}_2)$ is the likelihood function obtained from (3). It can be seen that the integral (4) is computationally demanding and needs to gather all the information in a central processing unit.

In order to compute the marginal distribution in a distributed way, conditional independence relationships among variables, which can be revealed from the factor graph [17] should be exploited. One example of factor graph is shown in Fig. 2. In this factor graph, local synchronization parameters $\boldsymbol{\beta}_i$, $i = 1, \cdots, M$, are represented by variables nodes (circles). If two sensor nodes i and j are within communication range of each other, the corresponding variable $\boldsymbol{\beta}_i$ and $\boldsymbol{\beta}_j$ are linked by factor node $f_{i,j} = p(\boldsymbol{A}_{i,j}, \boldsymbol{A}_{j,i}|\boldsymbol{\beta}_i, \boldsymbol{\beta}_j)$. On the other hand, the factor node $f_i = p(\boldsymbol{\beta}_i)$ denotes the prior information.

The message passing algorithm operated on the factor graph involves two kinds of messages: One is the message from factor node $f_{i,j}$ to a variable node $\boldsymbol{\beta}_i$, defined as [17]

$$m_{f_{i,j}\to i}^{(l)}(\boldsymbol{\beta}_i) = \int m_{j\to f_{i,j}}^{(l-1)}(\boldsymbol{\beta}_j) f_{i,j} d\boldsymbol{\beta}_j, \tag{5}$$

where $m_{j\to f_{i,j}}^{(l-1)}(\boldsymbol{\beta}_j)$ is the other kind of message from variable node to factor node which is simply the product of the incoming messages on other links, i.e.,

$$m_{j\to f_{i,j}}^{(l)}(\boldsymbol{\beta}_j) = \prod_{f\in\mathcal{B}(\boldsymbol{\beta}_j)\setminus f_{i,j}} m_{f\to j}^{(l)}(\boldsymbol{\beta}_j), \tag{6}$$

where $\mathcal{B}(\boldsymbol{\beta}_j)$ denotes the set of neighboring factors of $\boldsymbol{\beta}_j$ on the factor graph. In particular, under such message computation rule, the message from factor node f_i to $\boldsymbol{\beta}_i$ always equals to the prior distribution $p(\boldsymbol{\beta}_i)$.

In the message passing procedure, messages are iteratively updated at variable nodes and factor nodes, respectively. In any round of message exchange, a belief of $\boldsymbol{\beta}_i$ can be computed as the the product of all the incoming messages from neighboring factor nodes and an estimate of $\boldsymbol{\beta}_i$ can be obtained by maximizing the belief:

$$\boldsymbol{\mu}_i^{(l)} = \arg\max_{\boldsymbol{\beta}_i} \prod_{f \in \mathcal{B}(\boldsymbol{\beta}_i)} m_{f\to i}^{(l)}(\boldsymbol{\beta}_i). \qquad (7)$$

3.2. BP Message Computation

During the first round of message passing, it is reasonable to set initial messages from factor node to variable node $m_{f_i\to i}^{(1)}(\boldsymbol{\beta}_i)$ and $m_{f_{i,j}\to i}^{(1)}(\boldsymbol{\beta}_i)$ as $p(\boldsymbol{\beta}_i)$ and non-informative message $\mathcal{N}(\boldsymbol{\beta}_i; \mathbf{0}, +\infty \boldsymbol{I}_2)$, respectively. Assuming $p(\boldsymbol{\beta}_i) = m_{f_{i,j}\to i}^{(1)}(\boldsymbol{\beta}_i)$ is in Gaussian form (if there is no prior information, we can set the mean to be zero and set the variance to be a large value, i.e., non-informative prior). Then, based on the fact that the likelihood function $f_{i,j}$ is also Gaussian, according to (5), $m_{f_{i,j}\to i}^{(2)}(\boldsymbol{\beta}_i)$ is a Gaussian function. Furthermore, $m_{j\to f_{i,j}}^{(2)}(\boldsymbol{\beta}_j)$ being the product of Gaussian functions in (6) is also a Gaussian function [18]. Thus during each round of message exchange, all the messages are Gaussian functions and only the mean vectors and covariance matrices need to be exchanged between factor nodes and variable nodes.

In general, for the l^{th} ($l = 2, 3, \cdots$) round of message exchange, factor nodes $f_{i,j}$ receive message $m_{j\to f_{i,j}}^{(l-1)}(\boldsymbol{\beta}_j)$ from their neighboring variable nodes and then compute message using (5). It can be shown that

$$m_{f_{i,j}\to i}^{(l)}(\boldsymbol{\beta}_i) \propto \mathcal{N}(\boldsymbol{\beta}_i; \boldsymbol{v}_{f_{i,j}\to i}^{(l)}, \boldsymbol{C}_{f_{i,j}\to i}^{(l)}), \qquad (8)$$

where the covariance matrix and mean are given by

$$\begin{aligned}\big[\boldsymbol{C}_{f_{i,j}\to i}^{(l)}\big]^{-1} = &\frac{1}{\sigma_{i,j}^2}\boldsymbol{A}_{i,j}^T\boldsymbol{A}_{i,j} - \frac{1}{\sigma_{i,j}^2}\boldsymbol{A}_{i,j}^T\boldsymbol{A}_{j,i} \\ &\Big\{\boldsymbol{A}_{j,i}^T\boldsymbol{A}_{j,i} + \sigma_{i,j}^2\big[\boldsymbol{C}_{j\to f_{i,j}}^{(l-1)}\big]^{-1}\Big\}^{-1}\boldsymbol{A}_{j,i}^T\boldsymbol{A}_{i,j}, \qquad (9)\\ \boldsymbol{v}_{f_{i,j}\to i}^{(l)} = &\boldsymbol{C}_{f_{i,j}\to i}^{(l)}\boldsymbol{A}_{i,j}^T\boldsymbol{A}_{j,i}\Big\{\boldsymbol{A}_{j,i}^T\boldsymbol{A}_{j,i} + \sigma_{i,j}^2\big[\boldsymbol{C}_{j\to f_{i,j}}^{(l-1)}\big]^{-1}\Big\}^{-1} \\ &\big[\boldsymbol{C}_{j\to f_{i,j}}^{(l-1)}\big]^{-1}\boldsymbol{v}_{j\to f_{i,j}}^{(l-1)}. \qquad (10)\end{aligned}$$

On the other hand, using (6), the the message passed from variable node to factor node is given by

$$m_{i\to f_{i,j}}^{(l)}(\boldsymbol{\beta}_i) \propto \mathcal{N}(\boldsymbol{\beta}_i; \boldsymbol{v}_{i\to f_{i,j}}^{(l)}, \boldsymbol{C}_{i\to f_{i,j}}^{(l)}), \qquad (11)$$

where

$$\big[\boldsymbol{C}_{i\to f_{i,j}}^{(l)}\big]^{-1} = \sum_{f\in\mathcal{B}(\boldsymbol{\beta}_i)\setminus f_{i,j}} \big[\boldsymbol{C}_{f\to i}^{(l)}\big]^{-1}, \qquad (12)$$

$$\boldsymbol{v}_{i\to f_{i,j}}^{(l)} = \boldsymbol{C}_{i\to f_{i,j}}^{(l)} \sum_{f\in\mathcal{B}(\boldsymbol{\beta}_i)\setminus f_{i,j}} \big[\boldsymbol{C}_{f\to i}^{(l)}\big]^{-1}\boldsymbol{v}_{f\to i}^{(l)}. \qquad (13)$$

Furthermore, during each round of message passing, each sensor can compute the estimation for $\boldsymbol{\beta}_i$ using (7) which can be shown to be

$$\boldsymbol{\mu}_i^{(l)} = \Big\{\sum_{j\in\mathcal{B}(i)}\big[\boldsymbol{C}_{f_{i,j}\to i}^{(l)}\big]^{-1}\Big\}^{-1}\sum_{j\in\mathcal{B}(i)}\big[\boldsymbol{C}_{f_{i,j}\to i}^{(l)}\big]^{-1}\boldsymbol{v}_{f_{i,j}\to i}^{(l)}. \qquad (14)$$

When the algorithm converges or the maximum number of message exchange is reached, each sensor computes its clock skew and offset according to

$$\hat{\alpha}_i = 1/\boldsymbol{\mu}_i^{(l)}(1), \quad \hat{\theta}_i = \boldsymbol{\mu}_i^{(l)}(2)/\boldsymbol{\mu}_i^{(l)}(1), \qquad (15)$$

where (1) and (2) denote the indexes of the vector $\boldsymbol{\mu}_i^{(l)}$.

The iterative algorithm based on BP is summarized as follows. The algorithm is started by setting the message from factor node to variable node as $m_{f_i\to i}^{(1)}(\boldsymbol{\beta}_i) = p(\boldsymbol{\beta}_i)$ and $m_{f_{i,j}\to i}^{(1)}(\boldsymbol{\beta}_i) = \mathcal{N}(\boldsymbol{\beta}_i; \mathbf{0}, +\infty\boldsymbol{I}_2)$ respectively. At each round of message exchange, every variable node computes the output message to factor nodes according to (12) and (13). After receiving the message from neighboring variable nodes, each factor computes its output message according to (9) and (10). Such iteration is terminated when (14) converges or the maximum number of iteration is reached. Then the estimate of each clock skew and offset is obtained by (15).

In practical WSNs, there is neither factor nodes nor variable nodes. The two kinds of message $m_{i\to f_{i,j}}^{(l)}(\boldsymbol{\beta}_i)$ and $m_{f_{i,j}\to j}^{(l)}(\boldsymbol{\beta}_j)$ are computed locally at node i, and only $m_{f_{i,j}\to j}^{(l)}(\boldsymbol{\beta}_j)$ are passed from node i to node j during each round of message exchange of BP. It can be seen the algorithm is fully distributed and each sensor only needs to exchange message with neighboring nodes.

4. SIMULATION RESULTS

This section presents numerical results to assess the performance of the proposed algorithm. Simulation results of estimation mean square error (MSE) are presented for random networks. One example topology with 25 nodes is shown in Fig. 3. In each simulation, clock skews α_i and clock offsets θ_i are uniformly distributed in the range $[-5.5, 5.5]$ and $[-0.955, 1.055]$, respectively. Variance of random delay $\sigma_i^2 = 0.1$ and is assumed to be identical for all nodes. 5000 simulation trials were performed to obtain the average performance of each point in the figures. Without loss of generality, Node 1 is selected as the reference node with $\boldsymbol{\beta}_1 = [1, 0]^T$, and $p(\boldsymbol{\beta}_1) = \mathcal{N}(\boldsymbol{\beta}_1; [1, 0]^T, \mathbf{0})$. For other nodes, non-informative prior is assumed $p(\boldsymbol{\beta}_i) = \mathcal{N}(\boldsymbol{\beta}_i; \mathbf{0}, +\infty\boldsymbol{I}_2)$, since it is difficult to obtain a prior distribution of skew and offset in practice.

To serve as a reference of the performance, the centralized CRB is computed. By stacking (3) for all the pairs of i and j and moving all the terms containing $\boldsymbol{\beta}_1$ to one side,

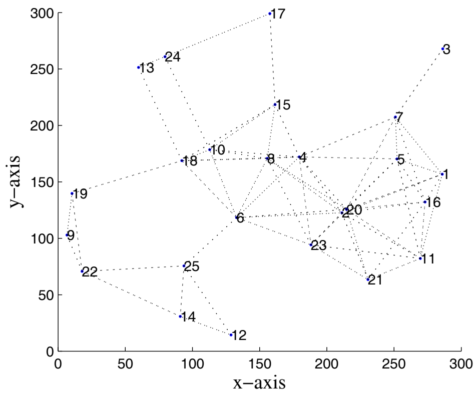

Fig. 3. WSN topology with 25 node randomly distributed.

a linear model $\boldsymbol{y} = \boldsymbol{H}\boldsymbol{\beta} + \boldsymbol{n}$, where $\boldsymbol{\beta} = [\boldsymbol{\beta}_2, \cdots, \boldsymbol{\beta}_M]^T$ and $\boldsymbol{n} \sim \mathcal{N}(\boldsymbol{n}; \boldsymbol{0}, \boldsymbol{P})$, can be obtained. Defining $\boldsymbol{\gamma} = [\theta_2, \alpha_2, \cdots, \theta_M, \alpha_M]^T$ for clock offsets and skews, the centralized CRB for $\boldsymbol{\gamma}$ is obtained as [19]

$$\mathrm{CRB}(\boldsymbol{\gamma}) = (\partial\boldsymbol{\gamma}/\partial\boldsymbol{\beta})(\boldsymbol{H}^T\boldsymbol{P}^{-1}\boldsymbol{H})^{-1}(\partial\boldsymbol{\gamma}/\partial\boldsymbol{\beta})^T, \quad (16)$$

where $\partial\boldsymbol{\gamma}/\partial\boldsymbol{\beta}$ can be computed to be a $2(M-1)$-by-$2(M-1)$ block diagonal matrix with $[-\alpha_{m+1}\theta_{m+1}, \alpha_{m+1}; -\alpha_{m+1}^2, 0]$ as the m^{th} sub-matrix on the diagonal. Due to space limitations, the detail of derivations is omitted.

Fig. 4 shows the MSE of both the clock skews and offsets of nodes 19, 18 and 5 as a function of BP iteration number for the WSN in Fig. 3. The number of time-stamp exchange round is $N = 20$. It can be seen from the figure that the MSEs decrease quickly and touch the corresponding CRBs in only a few iterations.

Fig. 5 shows the MSE for clock skews and offsets averaged over all nodes versus the number of time-stamp exchange rounds. The number of sensor node is 25 and the topology of WSNs is randomly generated within an area of 300×300 in each trial. As shown in the figure, the MSEs of proposed distributed estimator achieve the best performance as the MSEs touch the corresponding CRBs.

It is well known that BP may diverge in loopy graph. However in this synchronization algorithm, it always converges under different network topology. The proof of convergence of the proposed algorithm will be presented in another paper.

5. CONCLUSIONS

In this paper, a fully distributed clock skew and offset estimation algorithm for WSNs is proposed. The algorithm is based on BP and is easy to be implemented by exchanging limited information between neighboring sensor nodes. Although BP is an approximate method for loopy graph, simulation results show that the proposed method touches CRB after a few rounds of iterations even under random network topology.

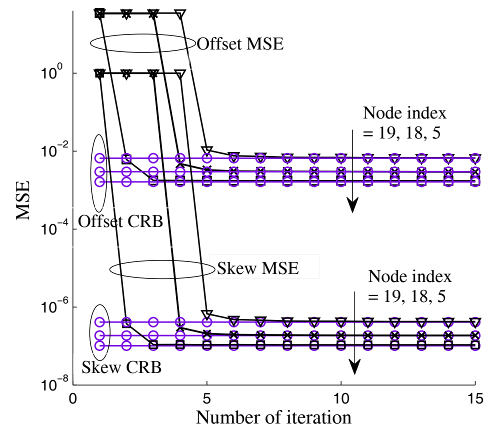

Fig. 4. Convergence performance of the proposed algorithm at different nodes.

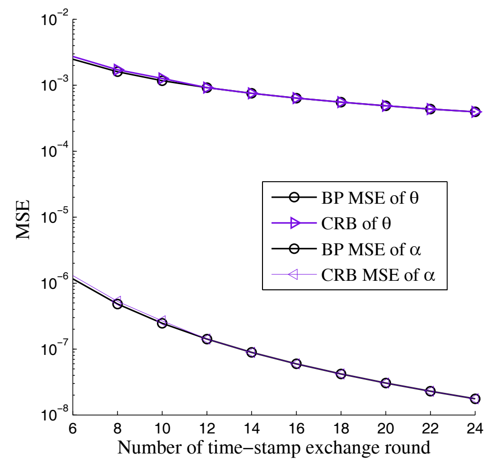

Fig. 5. MSE of clock skew and offset averaged over the whole network with respect to the number of time-stamp exchange round N.

6. REFERENCES

[1] T. Arampatzis, J. Lygeros, and S. Manesis, "A survey of applications of wireless sensors and wireless sensor networks, in *Proc. IEEE Int. Symp. Intell. Control, Limassol, Cyprus*, Jun. 2005, pp. 719–724.

[2] E. Msechu and G. Giannakis, "Sensor-centric data reduction for estimation with WSNs via censoring and quantization, *IEEE Trans. Signal Process.*, vol. 60, no. 1, pp. 400–414, Jan. 2012.

[3] M. Erol-Kantarci and H. T. Mouftah, "Wireless Sen-

sor Networks for Smart Grid Applications, in *Proc. Int. Conf. on Elec. Commun. and Photonic. (SIECPC)*, 24-26 April 2011, pp. 1–6.

[4] Y.-C. Wu, Q. Chaudhari, and E. Serpedin, "Clock synchronization of wireless sensor networks, *IEEE Signal Process. Mag.* , vol. 28, no. 1, pp.124–138, Jan. 2011.

[5] B. Sadler and A. Swami, "Synchronization in sensor networks: an overview, *IEEE Military Communication Conference (MILCOM)*, pp.1–6, Oct. 2006.

[6] S. Ganeriwal, R. Kumar, and M.B. Srivastava, Timing-Sync Protocol for Sensor Networks, *Proc. First Intl Conf. Embedded Networked Sensor Systems (SenSys 03)*, pp. 138-149, 2003.

[7] K.-L. Noh, E. Serpedin, and K. Qaraqe, "A new approach for time synchronization in wireless sensor networks: Pairwise broadcast synchronization," *IEEE Trans. Wireless Commun.*, vol. 7, no. 9, pp. 3318–3322, Sep. 2008.

[8] D. R. Jeske, "On maximum-likelihood estimation of clock offset," *IEEE Trans. on Commun.*, vol. 53, no. 1, pp. 53–54, Jan. 2005.

[9] K.-L. Noh, E. Serpedin, and B. Suter, "Novel clock phase offset and skew estimation using two-way timing message exchanges for wireless sensor networks," *IEEE Trans. on Commun.*, vol. 55, no. 4, pp. 766–777, Apr. 2007.

[10] M. Leng and Y.-C. Wu, "Low complexity maximum likelihood estimators for clock synchronization of wireless sensor nodes under exponential delays," *IEEE Trans. Signal Process.*, vol. 59, no. 10, pp. 4860–4870, Oct. 2011.

[11] A. Ahmad, D. Zennaro, E. Serpedin, and L. Vangelista, "A Factor Graph Approach to Clock Offset Estimation in Wireless Sensor Networks, *IEEE Trans. Inf. Theory,* vol. 58, no. 7, pp. 4244–4260, Jul. 2012.

[12] ——, "Time-varying clock offset estimation in two-way timing message exchange in wireless sensor networks using factor graphs, *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSAP)*, Mar 2012.

[13] Q. Li and D. Rus, "Global clock synchronization in sensor networks, *IEEE Trans. Comput.*, vol. 55, no. 2, pp. 214–225, Feb. 2006.

[14] A. Giridhar and P. R. Kumar, "Distributed clock synchronization over wireless networks: Algorithms and analysis," in *Proc. 45th IEEE Conf. Decision and Contr.(CDC)*, San Diego, CA, Dec. 2006, pp. 4915–4920.

[15] G. Xiong and S. Kishore, "Analysis of distributed consensus time synchronization with Gaussian delay over wireless sensor networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2009, 9 pages, 2009.

[16] M. Leng and Y.-C. Wu, "Distributed clock synchronization for wireless sensor networks using belief propagation," *IEEE Trans. Signal Process.*, vol. 59, no. 11, pp. 5404–5414, Nov. 2011.

[17] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inf. Theory,* vol. 47, no. 2, pp. 498–519, Feb. 2001.

[18] A. Papoulis and S. U. Pillai, *Random Variables and Stochastic Processes,* 4th ed. New York: McGraw-Hill, 2002.

[19] S. M. Kay, *Fundamentals of Statistical Signal Processing Estimation Theory,* Upper Saddle River, NJ: Prentice-Hall, 1993.