



Fully distributed clock synchronization in wireless sensor networks under exponential delays[☆]



Bin Luo, Lei Cheng, Yik-Chung Wu^{*}

Department of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong

ARTICLE INFO

Article history:

Received 27 November 2015

Received in revised form

28 January 2016

Accepted 4 February 2016

Available online 12 February 2016

Keywords:

Distributed clock synchronization

Wireless sensor networks

Exponential delays

Alternating direction method of multipliers (ADMM)

ABSTRACT

In this paper, we study the global clock synchronization problem for wireless sensor networks under unknown exponential delays in the two-way message exchange mechanism. The joint maximum likelihood estimator of clock offsets, clock skews and fixed delays for the network is formulated as a linear programming (LP) problem. Then, based on the alternating direction method of multipliers (ADMM), we propose a fully distributed synchronization algorithm which has low communication overhead and computation cost. Simulation results show that the proposed algorithm achieves better accuracy than consensus algorithm and the distributed least squares algorithm, and can always converge to the centralized optimal solution.

© 2016 Elsevier B.V. All rights reserved.

1. Introduction

A variety of applications in wireless sensor networks (WSNs), such as object tracking, machine health monitoring, home appliance control, etc. [1,2], require cooperative execution of distributed tasks amongst a set of sensor nodes. To execute these tasks, nodes are usually required to run on a common time frame. However, this is far from trivial due to many factors, including imperfection of oscillators, environmental changes, and unknown delays in message delivery. Therefore, global clock synchronization constitutes an important prerequisite in many services of WSNs.

One critical component of clock synchronization is the modeling of the random network delays that perturb the message-exchange process. Existing probabilistic models for random delays in WSNs include Gaussian, exponential, Gamma, Weibull, and log-normal [3–5]. Not a single model

dominates, because different models are justified in different applications. For example, it was observed in [6] that when the point-to-point hypothesis reference connection topology is of interest, the cumulative link delay in WSN is approximately represented as a single server M/M/1 queue, and the random delays should be modeled as exponential random variables. Furthermore, the minimum link delay algorithm [6], which was later derived in [7] based on the assumption of exponentially distributed network delays, shows good performance in practice. In addition, among all distributions with support in $[0, +\infty)$ and a fixed mean, the exponential distribution has the maximum entropy, and thus least informative. Hence the assumption of exponential delay distribution is worth investigating.

The other crucial challenge in clock synchronization is to design algorithms enabling global clock calibration among all the nodes in the network. Earlier protocols were mainly based on pairwise synchronization. More specifically, before synchronization, a particular network structure (spanning tree or cluster) must be built, and then layer-by-layer pairwise synchronization was executed starting from a reference node. As an important component in this approach, pairwise synchronization under exponential

[☆] This work was presented in part at the 14th International Conference on Communication Systems 2014, Macau, November 2014 [38].

^{*} Corresponding author.

E-mail addresses: luobin@eee.hku.hk (B. Luo), leicheng@eee.hku.hk (L. Cheng), ycwu@eee.hku.hk (Y.-C. Wu).

delay distribution has been extensively studied in the past decade. For example, the maximum likelihood estimator (MLE) is studied in [7,8], while the linear unbiased estimator using order statistics (BLUE-OS) and the minimum variance unbiased estimator (MVUE) are derived in [9] under asymmetric exponential distributed delays in uplink and downlink. Recently, max-product algorithm is used in [10,11] to derive a message passing method for the clock offset tracking in the presence of exponential distributed random delays. Unfortunately, the clock skew is not considered in these works, resulting in potentially frequent re-synchronization. To avoid frequent offset resetting, skew compensation is adopted for a number of synchronization algorithms. The MLE for joint clock offset and skew estimation has been derived in [12,13], while a low-complexity MLE has been proposed in [14,15]. Furthermore, an approximate MLE for joint offset and skew estimation has been reported in [16].

However, in such pairwise-based synchronization approach, not only large communications overhead are required to maintain the specific network structure, synchronization errors also accumulate rapidly as distance from reference node increases. This hinders the scalability of the synchronization protocol and reduces its robustness to changes in the network topology. To relieve these problems, more recent synchronization algorithms were designed to work in a fully distributed way. Specifically, each node calibrates its own clock by exchanging messages with its direct neighbors and performing computations locally, with no special network structure maintained. Existing fully distributed synchronization schemes can be categorized into pulse-coupled based and packet-coupled based. For the pulse-coupled based approach, physical pulses are transmitted from different nodes to achieve a unified clock ticking rhythm [17,18]. On the other hand, for packet-coupled based synchronization, clock reading at each node is encoded in time stamps and transmitted as data packets to its neighbors. Within the packet-coupled based category, two major ideas have been frequently used: consensus principle [19–22] and factor graph based statistical inference [23–25]. Unfortunately, large mean-square-error (MSE) occurs in estimates from consensus based algorithms since the fixed message delays are not explicitly considered. Furthermore, for existing factor graph based algorithms, either only clock offset is considered [23], or the algorithms are derived explicitly for Gaussian delays [24,25].

In this paper, based on the two-way message exchange mechanism, a network-wide joint estimator of clock offsets, clock skews and fixed delays is derived under exponential delay model. The joint maximum likelihood estimation problem is first cast into a linear programming (LP) problem, and then a distributed solution is derived based on alternating direction method of multipliers (ADMM) [26,27]. Exploiting the special structure of the derived solution, further improvement in terms of computation efficiency is proposed. Both theoretical analysis and simulation results show that the proposed distributed algorithm approaches the MSE performance of the centralized LP solution.

The rest of the paper is organized as follows. In Section 2, the system model is first described, and then the MLE for global clock synchronization problem is cast as a LP-problem. In Section 3, a distributed synchronization algorithm based

on ADMM is derived. Simulation results are presented in Section 4. Finally, conclusions are drawn in Section 5.

Notation: The operator $\text{Tr}\{\mathbf{A}\}$ takes the trace of matrix \mathbf{A} , and $\mathbf{1}_N$ denotes the vector of N ones, while \mathbf{I}_N represents an $N \times N$ identity matrix. $\text{diag}(\cdot)$ denotes the block diagonal concatenation of input arguments. Superscript $(\cdot)^T$ denotes the transpose operator. \mathcal{N}_i denotes the set of neighbors of node S_i and $|\mathcal{N}_i|$ indicates the number of elements in set \mathcal{N}_i . Finally, $\|\mathbf{x}\|_2$ represents the Euclidean norm of vector \mathbf{x} .

2. System model and maximum likelihood estimator

Consider a network with N sensor nodes $\{S_1, S_2, \dots, S_N\}$. These sensors are randomly distributed in the field and can be self-organized into a network¹ by establishing connections between neighbor nodes lying within each other's communication range. An example of 25 sensor nodes is shown in Fig. 1, where each edge represents the ability to transmit and receive packets between the pair of nodes (i.e., the network is undirected). The communication network topology is described by the link set $\mathcal{L} \triangleq \{(i, j) : \text{there is a link between nodes } S_i \text{ and } S_j\}$. Each sensor S_i has a clock which gives clock reading $c_i(t)$ at real time t . The first-order model for the function $c_i(t)$ is [5,28]:

$$c_i(t) \triangleq \beta_i t + \theta_i, \quad (1)$$

where β_i and θ_i represent the clock skew and offset of S_i , respectively. In general, the value for θ_i would lie within $[-\infty, +\infty]$, while that of β_i would be around the value of 1 and its specific range depends on the quality of the oscillator being used.

In order to establish clock relationship between two neighboring nodes, two-way message exchange process [10,11,23,29] is performed. Assume that nodes S_i and S_j are in the communication range of each other, i.e., $(i, j) \in \mathcal{L}$, the k -th round message exchange between S_i and S_j is shown in Fig. 2. In the message exchange process, node S_i sends a synchronization message to node S_j with its sending time $T_{1,k}^{(i,j)}$, S_j records its time $T_{2,k}^{(i,j)}$ at the reception of that message and replies to S_i at time $T_{3,k}^{(i,j)}$. The replied message contains both $T_{2,k}^{(i,j)}$ and $T_{3,k}^{(i,j)}$. Then S_i records the reception time of S_j 's reply as $T_{4,k}^{(i,j)}$. Note that $T_{1,k}^{(i,j)}$ and $T_{4,k}^{(i,j)}$ are the time-stamps recorded by the clock of node S_i , while $T_{2,k}^{(i,j)}$ and $T_{3,k}^{(i,j)}$ are the time-stamps recorded by that of node S_j . Since both nodes require all the time-stamps for processing, we assume that the first message in the $(k+1)$ -th round contains the time-stamp $T_{4,k}^{(i,j)}$ as well. The message exchange process is repeated for K rounds, and node S_i finally sends an additional message² to node S_j containing the time-stamp $T_{4,K}^{(i,j)}$, resulting in both nodes having all the time-stamps $\{T_{1,k}^{(i,j)}, T_{2,k}^{(i,j)}, T_{3,k}^{(i,j)}, T_{4,k}^{(i,j)}\}_{k=1}^K$.

¹ We only consider networks where any two distinct nodes can communicate with each other through a finite number of hops.

² The additional message may not be required if piggybacking is adopted.

Download English Version:

<https://daneshyari.com/en/article/6958386>

Download Persian Version:

<https://daneshyari.com/article/6958386>

[Daneshyari.com](https://daneshyari.com)