49th IEEE Conference on Decision and Control
December 15-17, 2010
Hilton Atlanta Hotel, Atlanta, GA, USA

TIME-SYNCHRONIZATION IN MOBILE SENSOR NETWORKS FROM DIFFERENCE MEASUREMENTS

Chenda Liao and Prabir Barooah

***Abstract*— We examine distributed time-synchronization in mobile ad-hoc and sensor networks. The problem is to estimate the skews and offsets of clocks of all the nodes with respect to an arbitrary reference clock. Pairs of nodes that can communicate with each other can obtain noisy measurements of the relative skews and offsets between them. We propose a distributed algorithm with which each node can estimate its offset/skew from these noisy relative measurements by communicating only with its neighbors. The algorithm is simple and easy to implement. We model the change in the communication network due to the moving nodes as a Markov chain whose state space is the set of graphs that can occur. Using tools from Markov Jump Linear Systems, we provide a sufficient condition for the mean square convergence of the estimation error. A conjecture on mean square convergence under weaker conditions is discussed. Monte Carlo simulations are provided that corroborate the predictions and justify the conjecture.**

I. INTRODUCTION

Time or clock synchronization is critical in the effective use of sensor and multi-agent networks; particularly in applications such as range finding for target tracking and localization, intrusion detection, time correlation of telemetry data, sensor fusion, slot assignment in TDMA, duty cycling protocols, and so on [1]. At a given global time t, the local clock time at node u can be approximately written as $t_u = \alpha_u t + \beta_u$, where α_u is the *skew* and β_u is the *offset*. The global time to which all nodes need to be synchronized can be the local clock time at an arbitrarily chosen "reference" node. The *time synchronization problem* is effectively a problem of estimating the skews and offsets of every node, since the nodes can infer the global time from their local clock times once they know their own skews and offsets.

A large number of protocols have been proposed to solve the problem of time-synchronization in static ad-hoc networks, such as the Timing-Sync Protocol for Sensor Networks [2], Flooding Time Synchronization Protocol [3] and Reference Broadcast Synchronization [4], etc. These are tree and cluster - based, and not fully distributed. The protocols in [5], [6], [7], [8] are distributed and iterative; nodes iteratively update their current estimates of skew and offset by exchanging information with their neighbors.

The methods described in all the papers mentioned above are geared to networks of static nodes. Due to the unreliability of the communication channels, edges in the graph that model communication between pairs of nodes may temporarily drop, but the graph does not change otherwise. Under these conditions, various properties of the estimation algorithm are established. For example, the averaging scheme proposed in [5], [6] are proven to converge to the centralized best linear unbiased estimate of the node variables (offsets and skews) even in the presence of temporary node and link failures. The schemes in [7], [8] have similar robustness properties.

In this paper we examine the question on how to estimate the offsets and skews accurately for a network consisting of mobile nodes. Due to motion, the set of neighbors of a node, i.e., the nodes it can communicate with, may change over time. To the best of our knowledge, distributed estimation of offsets and skews in this scenario has not been examined before.

When a pair of nodes can exchange time-stamped messages, they can obtain noisy measurements of the relative skews and offsets; see [9], [10] for details on how these measurements are obtained. The time-synchronization problem therefore can be posed as the problem of estimating scalar valued "node variables" from noisy measurements of their pairwise differences. The algorithm we propose to solve the estimation problem in a time-varying network of mobile nodes is similar in approach to the one in [5]. The algorithm is fully distributed and asynchronous. It involves only local communication among neighbors and simple computations. The algorithm consists of an iterative update law, and is related to the consensus-type algorithms [11], [12], [13]. The analysis of the algorithms, especially that of the variance of the estimation error, requires tools beyond that available from the literature on consensus.

We model the time-variation of the network topology due to nodes' motion as a Markov chain. In that case, the evolution of the estimation errors is a Markov Jump Linear System (MJLS) driven by a zero-mean wide sense stationary (WSS) white noise sequence. We establish a sufficient condition on the underlying Markov chain that guarantees that the estimation error dynamics are mean square convergent. In particular, the mean of the error converges to 0 and the correlation matrix to a positive-definite matrix, whose value is a function of the transition probability matrix, the measurement error variances, and structure of the graphs that constitute the state space of the Markov chain. Monte Carlo simulations are provided that corroborate the theoretical predictions. We conjecture that the conditions required for mean square convergence of the estimation errors are weaker than the sufficient condition provided here.

Since the variances converge to non-zero steady state values, the nodes can turn off the iterative updates after a

C. Liao and P. Barooah are with the Dept. of Mechanical and Aerospace Engineering, Univ. of Florida, Gainesville, FL; {cdliao,pbarooah}@ufl.edu. This work has been supported by the National Science Foundation by Grants CNS-0931885 and ECCS-0955023.

sufficiently long time. This will conserve on-board energy with little loss of estimation accuracy.

The rest of the paper is organized as follows. Section II states the estimation problem precisely. Section III describes the proposed algorithm and the main result, while Section IV provides simulation results.

II. Problem statement

By exchanging a number of time-stamped messages, a pair of nodes u and v can obtain measurements of their relative offsets $\beta_u - \beta_v$ and ratios of skews α_u/α_v. Methods for estimating offset differences when skews are unity are described in [9]. With 4 rounds of communication between u and v, ($u \to v, v \to u, u \to v, v \to u$ in that order), both offset differences and skew ratios can be obtained [10]. Due to various inaccuracies, the measurements will be noisy. That is, u and v can obtain $\eta_{u,v}$ and $\xi_{u,v}$ after a few rounds of communication, where $\eta_{u,v} = \beta_u - \beta_v + \epsilon_{u,v}$ and $\xi_{u,v} = \log(\alpha_u) - \log(\alpha_v) + \varepsilon_{u,v}$, where $\epsilon_{u,v}, \varepsilon_{u,v}$ are a zero-mean measurement noises. Thus, both relative offset and skew ratio measurements are special cases of

$$\zeta_{u,v} = x_u - x_v + \epsilon_{u,v}, \tag{1}$$

where $\zeta_{u,v}$ is a noisy measurement of the difference between the unknown scalar variables x_u and x_v, and $\epsilon_{u,v}$ is a zero-mean measurement error. For this reason, we restrict our attention to the problem of estimating scalar-valued variables from relative measurements of the type (1). For the sake of concreteness, we will frequently refer to offset estimation as a specific example of this problem.

Let the total number of sensor nodes in the network be n_{total}. These define a node set $\mathcal{V} = \{1, 2, \dots, n_{\text{total}}\}$. With each node u, a real-valued variable $x_u \in \mathbb{R}$ is associated that we call a *node variable*. The node variable could be either the log-of-skew or the offset of u's local clock with respect to the clock of an arbitrarily chosen reference. Without loss of generality, we fix the node variable of the reference node to 0: $x_r = 0$, and set the index of the reference node n_{total}. The number of nodes whose node variables have to be estimated is $n_{\text{total}} - 1 =: n$.

Time is measured by a discrete time-index $k = 0, 1, \dots$. The mobile nodes define a time-varying undirected graph $\mathcal{G}(k) = (\mathcal{V}, \mathcal{E}(k))$, where $(u, v) \in \mathcal{E}(k)$ if and only if u and v can obtain a relative measurement of the form (1) during the time interval between the time indices k and $k+1$. Specifically, for each $(u, v) \in \mathcal{E}(k)$, a measurement

$$\zeta_{u,v}(k) = x_u - x_v + \epsilon_{u,v}(k), \tag{2}$$

is obtained by both u and v at time k. We assume that $\epsilon_{u,v}(k)$ is zero mean wide sense stationary white noise process, and the measurement errors $\epsilon_{u,v}(k)$ on distinct edges are uncorrelated: $\mathrm{E}[\epsilon_{u,v}(k)\epsilon_{w,x}(\ell)] = 0$ unless $k = \ell$ and $u = w, v = x$, in which case it is equal to σ_0^2. Since multiple rounds of communication is required to obtain a measurement, if $(u, v) \in \mathcal{E}(k)$, then u and v can also communicate and exchange information at time k. The nodes that u can communicate with at k are the *neighbors* of u at time k, which are denoted by $\mathcal{N}_u(k)$.

The problem is to estimate the node variables x_u for $u = 1, \dots, n$ by using the relative measurements $\zeta_{u,v}(k), (u, v) \in \mathcal{E}(k)$ that become available over time $k = 0, 1, \dots$. In addition, the algorithm has to be distributed in the sense that each node has to estimate its own variables locally, and at every time k, a node u can only exchange information with its neighbors $\mathcal{N}_u(k)$.

III. Proposed algorithm

Each node u maintains in its local memory an estimate $\hat{x}_u(k)$ of its node variable x_u that is iteratively updated. The estimates are initialized to arbitrary values. In executing the algorithm at iteration k, node u communicates with its current neighbors to obtain from them their current estimates $\hat{x}_v(k)$, $v \in \mathcal{N}_u(k)$. It also collects the measurements $\zeta_{u,v}(k)$ for each $v \in \mathcal{N}_u(k)$ during this time. It then updates its estimate according to

$$\hat{x}_u(k+1) = \frac{\hat{x}_u(k) + \sum_{v \in \mathcal{N}_u(k)} (\hat{x}_v(k) + \zeta_{u,v}(k))}{d_u(k) + 1}, \tag{3}$$

where the *degree* $d_u(k) = |\mathcal{N}_u(k)|$ is the number of neighbors of node u. Nodes continue this iterative update as long as they keep moving; or until they see little change in their local estimates.

Note that $\hat{x}_v(k) + \zeta_{u,v}(k) = x_u + (\hat{x}_v(k) - x_v) + \epsilon_{u,v}(k)$; so each term inside the summation on the right hand side of (3) is an unbiased estimate of x_u, provided $\mathrm{E}[\hat{x}_v(k)] = x_v$. This condition is satisfied if the initial estimates are unbiased, i.e. $\mathrm{E}[\hat{x}_u(0)] = x_u$ for each u. It is possible to ensure that the initial estimates are unbiased by using the flagged initialization procedure with no additional communication or computation cost, which is described in [6]. The right hand side of (3) is therefore an average of multiple noisy estimates of x_u. The reason for including $\hat{x}_u(k)$ in the right hand side of (3) is to make it well-defined even at times when u has no neighbors. Note that if v in (3) is the reference node, i.e., $v = r$, then $\hat{x}_v(k) = x_r = 0$.

A. Main result

We consider the case when the sequence of graphs $\{\mathcal{G}(k)\}_{k=0}^{\infty}$ can be modeled as the realization of a Markov chain, whose state space $\mathbb{G} = \{\mathcal{G}_1, \dots, \mathcal{G}_N\}$ is the set of all the graphs that can be formed by the mobile nodes. The Markovian property means that if $\mathcal{G} \in \mathbb{G}$, then $\mathrm{P}(\mathcal{G}(k+1) = \mathcal{G}|\mathcal{G}(k)) = \mathrm{P}(\mathcal{G}(k+1) = \mathcal{G}|\mathcal{G}(k), \mathcal{G}(k-1), \dots, \mathcal{G}(0))$. A simple example in which the time variation of the graphs satisfies the Markovian property is that of a network of static nodes with unreliable communication links such that each link can fail temporarily, and the failure of each edge at every time instant k is independent of the failures of other links and the probability of its failure is time-invariant. Another example is a network of mobile agents whose motion obeys first order dynamics with range-determined communication. Specifically, suppose the position of node u at time k, denoted by $p_u(k)$, is restricted to lie on the unit sphere $\mathbf{S}^2 = \{x \in \mathbb{R}^3 | \|x\| = 1\}$, and suppose the position evolution obeys: $p_u(k+1) = f(p_u(k) + \Delta_u(k))$, where $\Delta_u(k)$ is a stationary zero-mean white noise sequence for every u, and $\mathrm{E}[\Delta_u(k)\Delta_v(k)^T] = 0$ unless $u = v$. The function $f(\cdot) : \mathbb{R}^3 \to \mathbf{S}^2$ is a projection function onto the unit-sphere. In addition, $(u, v) \in \mathcal{E}(k)$ if and only if the geodesic distance between them is less than or equal to some predetermined range. In this case, prediction of $\mathcal{G}(k+1)$ given $\mathcal{G}(k)$ cannot be improved by the knowledge of the graphs observed prior to k: $\mathcal{G}(k-1), \dots, \mathcal{G}(0)$, and hence the change in the graph sequence satisfies the Markovian property.

If no restriction is placed on the motion of the nodes or edge formation, the size of state space $N := |\mathbb{G}|$ is the total number of distinct graphs possible with n_{total} nodes, which is $2^{\frac{1}{2}n_{\text{total}}(n_{\text{total}}-1)}$. If certain nodes are restricted to move only within certain geographic areas, N is less than this maximum number. We assume that the Markov chain is homogeneous, and denote the transition probability matrix of the chain by $\mathcal{P}$. Define $\mathbf{x} := [x_1, \dots, x_n]^T$ as the vector of all the node variables and $\hat{\mathbf{x}}(k) := [\hat{x}_1(k), \dots, \hat{x}_n(k)]^T$ be the vector of the estimates at time k. The main result of the paper is the following, in which $\mathbf{e}(k) := \hat{\mathbf{x}}(k) - \mathbf{x}$ is the estimation error at time k, and $\mu(k) := \mathrm{E}[\mathbf{e}(k)]$, $\mathbf{Q}(k) := \mathrm{E}[\mathbf{e}(k)\mathbf{e}(k)^T]$ are its mean and correlation, respectively.

Theorem 1: Consider the iterative update (3) carried out by the nodes in a synchronous manner. Assume that the evolution of the graph $\mathcal{G}(k)$ is governed by a homogeneous Markov chain with transition probability matrix $\mathcal{P}$. If $\mathcal{P}$ is entry-wise positive, and at least one element of $\mathbb{G}$ is a connected graph, then the estimation error $\mathbf{e}(k)$ is mean square convergent, and in particular, $\mu(k) \to 0$ and $\|\mathbf{Q}(k) - \mathbf{Q}\| \to 0$ as $k \to \infty$, where $\mathbf{Q}$ is a positive definite matrix. □

A formula for $\mathbf{Q}$ is provided later in Lemma 1. The proofs of the theorem and the lemma are provided in [10]. The implication of the theorem is that the estimates converge to an unbiased estimate with a finite variance for every initial condition on the node variables. Thus, after a sufficiently long time, the nodes can turn off the synchronization updates without much loss of accuracy.

The hypothesis of the theorem asks that the Markov chain satisfy two conditions: (1) every entry of $\mathcal{P}$ is positive and (2) at least one element of $\mathbb{G}$ is a connected graph. The first condition will trivially lead to ergodicity of Markov chain, which guarantees the steady state distribution exists and is also required to use some of the results from the Markov jump linear system (MJLS) literature. In addition, both the conditions are used to show the spectral radius of a particular matrix is less than 1, which allows us to use results from MJLS theory to prove mean square stability. We believe the conditions are an artifact of our proof technique; they are not necessary.

Conjecture 1: If the Markov chain is ergodic and its state space $\mathbb{G}$ is such that the union of all the graphs in $\mathbb{G}$ is a connected graph, then $\mu(k)$ converges to 0 and $\mathbf{Q}(k)$ converges to a steady state matrix. □

Ergodicity implies that the unique steady state distribution of each state is non-zero, which means every graph in the state space of the chain occurs infinitely often. Since their union is connected, it means information from the reference will flow to each of the nodes, which is likely to lead to convergence. Simulations reported in Section IV provide evidence in support of this conjecture.

B. Convergence Analysis

To provide a formula for the steady state correlation matrix and prove the theorem, we need to express the iterative update (3) in a compact form, for which we introduce the following standard graph-theoretic terminology. Given an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, we first introduce a directed graph $\vec{\mathcal{G}} = (\mathcal{V}, \vec{\mathcal{E}})$ by assigning directions to the edges of $\mathcal{G}$ arbitrarily. For the sake of concreteness, however, we will follow the convention that $(u, v) \in \mathcal{E}$ then the edge direction in $\vec{\mathcal{G}}$ is from the node with lower index to that with a higher index. Let $|\mathcal{V}| = n_{\text{total}}$ and $|\mathcal{E}| = |\vec{\mathcal{E}}| = m$. The node-edge *incidence matrix* A of the directed graph $\vec{\mathcal{G}}$ is a $n_{\text{total}} \times m$ matrix whose entries are defined as follows: $A_{u,e} = 0$ if the edge e is not incident on node u; $A_{u,e} = +1$ if edge e is directed away from node u and $A_{u,e} = -1$ if edge e is directed towards node u. The *basis incidence matrix* A_b is the $(n_{\text{total}} - 1) \times m$ sub-matrix of A that is obtained by removing the row corresponding to the reference node. The matrix $L_b := A_b A_b^T$ is called the *Dirichlet or grounded Laplacian matrix* of the graph $\mathcal{G}$. In fact, L_b does not depend on the directions of the edges. The diagonal matrix M constructed from the diagonal entries of L_b is the *basis degree matrix* and $N := M - L_b$ is called the basis adjacency matrix. Explicitly, $M(u, u) = d_u$, and $N(u, u) = 0$, $N(u, v) = 1$ if $(u, v) \in \mathcal{E}$.

With the above notation, it follows from the update law (3) that the temporal evolution of $\hat{\mathbf{x}}(k)$ is given by

$$\hat{\mathbf{x}}(k+1) = J_{\mathcal{G}(k)}\hat{\mathbf{x}}(k) + B_{\mathcal{G}(k)}\boldsymbol{\zeta}(k) \tag{4}$$

where

$$\begin{aligned} J_{\mathcal{G}(k)} &:= (M_{\mathcal{G}(k)} + I)^{-1}(N_{\mathcal{G}(k)} + I), \\ B_{\mathcal{G}(k)} &:= (M_{\mathcal{G}(k)} + I)^{-1} A_{b\mathcal{G}(k)}, \\ \boldsymbol{\zeta}(k) &:= [\zeta_1(k), \dots, \zeta_{m(k)}(k)]^T = A_{b\mathcal{G}(k)}^T \mathbf{x} + \boldsymbol{\epsilon}(k), \end{aligned} \tag{5}$$

$m(k) = |\mathcal{E}(k)|$ is the total number of measurements at k and $\boldsymbol{\epsilon}(k)$ is vector of all measurement noise. $M_{\mathcal{G}(k)}$ and $N_{\mathcal{G}(k)}$ are the basis degree and basis adjacency matrices of the graph $\mathcal{G}(k)$, and $A_{b\mathcal{G}(k)}$ is the basis incidence matrix of the associated directed graph $\vec{\mathcal{G}}(k)$. To avoid making the notation more cumbersome than it already is, we refrain from using $\vec{\mathcal{G}}(k)$ in the subscripts. We rewrite the last term in (4):

$$\begin{aligned} B_{\mathcal{G}(k)}\boldsymbol{\zeta}(k) &= (M_{\mathcal{G}(k)} + I)^{-1} A_{b\mathcal{G}(k)} ({A_{b\mathcal{G}(k)}}^T \mathbf{x}) + B_{\mathcal{G}(k)}\boldsymbol{\epsilon}(k) \\ &= (M_{\mathcal{G}(k)} + I)^{-1}(M_{\mathcal{G}(k)} + I - (N_{\mathcal{G}(k)} + I))\mathbf{x} \\ &\quad + B_{\mathcal{G}(k)}\boldsymbol{\epsilon}(k) \\ &= -J_{\mathcal{G}(k)}\mathbf{x} + \mathbf{x} + B_{\mathcal{G}(k)}\boldsymbol{\epsilon}(k), \end{aligned} \tag{6}$$

where the second equality follows from $A_{b\mathcal{G}(k)} {A_{b\mathcal{G}(k)}}^T = L_{b\mathcal{G}(k)}$. Substituting (6) in (4) and subtracting $\mathbf{x}$ from both sides of (4), we obtain the dynamics of the estimation error $\mathbf{e}(k) := \hat{\mathbf{x}}(k) - \mathbf{x}$:

$$\mathbf{e}(k+1) = J_{\mathcal{G}(k)}\mathbf{e}(k) + B_{\mathcal{G}(k)}\boldsymbol{\epsilon}(k), \tag{7}$$

where the initial condition $\mathbf{e}(0)$ is a random vector that is independent of the Markov chain.

Now we introduce some non-standard notation to express the error dynamics in a more convenient form. Let $\tilde{m}$ be the number of edges in the union graph $\cup_i \mathcal{G}_i$, where the union is over the state space $\mathbb{G}$ of the Markov chain. For every graph $\mathcal{G}_i \in \mathbb{G}$, define an *extended incidence matrix* $\tilde{A}_i$ by introducing additional columns with all entries equal to 0 so that $\tilde{A}_i$ is an $n_{\text{total}} \times \tilde{m}$ matrix. Let $\tilde{A}_{bi}$ the corresponding extended basis incidence matrix, and define $\tilde{B}_i := (M_i + I)^{-1}\tilde{A}_{bi}$. Note that unlike B_i, the dimension of $\tilde{B}_i$ does not change with $\mathcal{G}_i$. At every time k, we also define an extended noise vector $\tilde{\boldsymbol{\epsilon}}(k) \in \mathbb{R}^{\tilde{m}}$, with entries that correspond to edges that do not exist in $\mathcal{G}(k)$ set to arbitrary values. As a result, $\tilde{A}_{b\mathcal{G}(k)}\tilde{\boldsymbol{\epsilon}}(k) = A_{b\mathcal{G}(k)}\boldsymbol{\epsilon}(k)$. Based on the assumptions made previously about the measurement noises $\epsilon_{u,v}$, we impose $\gamma := \mathrm{E}[\tilde{\boldsymbol{\epsilon}}(k)] = 0$ and $\Gamma := \mathrm{E}[\tilde{\boldsymbol{\epsilon}}(k)\tilde{\boldsymbol{\epsilon}}(k)^T] = \sigma_0^2 I > 0$ (positive definite), where I is an identity matrix. Note that the entries of $\tilde{\boldsymbol{\epsilon}}(k)$ (and their means and variances) that correspond to edges that do not exist in $\mathcal{E}(k)$ do not affect the analysis and can be chosen arbitrarily.

The state of the following system is identical to that of (7) for the same initial conditions:

$$\mathbf{e}(k+1) = J_{\mathcal{G}(k)}\mathbf{e}(k) + \tilde{B}_{\mathcal{G}(k)}\tilde{\boldsymbol{\epsilon}}(k). \tag{8}$$

The error dynamics (8) is a Markov jump linear system (MJLS) [14]. Our main result, Theorem 1, is established by using tools from the theory of MJLS. The reason for defining the "extended" matrices was to make the terms $\tilde{B}_{\mathcal{G}(k)}$ and $\tilde{\boldsymbol{\epsilon}}(k)$ have the same dimension irrespective of k.

We also need the following definitions and terminology from [14]. Let $\mathbb{R}^{m\times n}$ be the space of $m\times n$ real matrices. Let $V = (V_1, V_2, \ldots, V_N) \in \mathbb{H}^{m\times n}$ be the set of all N-sequences of real $m \times n$ matrices $V_i \in \mathbb{R}^{m\times n}$. The operators φ and $\hat{\varphi}$ is defined as follows: let $V_i = [(v_i)_1 \ldots (v_i)_n] \in \mathbb{R}^{m\times n}$, then

$$\varphi(V_i) := \begin{pmatrix} (v_i)_1 \\ \vdots \\ (v_i)_n \end{pmatrix} \text{ and } \hat{\varphi}(V) := \begin{pmatrix} \varphi(V_1) \\ \vdots \\ \varphi(V_N) \end{pmatrix} \tag{9}$$

Hence, $\varphi(V_i) \in \mathbb{R}^{mn}$ and $\hat{\varphi}(V) \in \mathbb{R}^{Nmn}$. Similarly, define an inverse function $\hat{\varphi}^{-1} : \mathbb{R}^{Nmn} \to \mathbb{H}^{m\times n}$ that produces an element of $\mathbb{H}^{m\times n}$ given a vector in Nmn. For $J_i \in \mathbb{R}^{n\times n}$, define

$$diag[J_i] := \begin{pmatrix} J_1 & \ldots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \ldots & J_N \end{pmatrix} \in \mathbb{R}^{Nn\times Nn} \tag{10}$$

and

$$\begin{aligned} \mathcal{C} &:= (\mathcal{P}^T \otimes I_n) diag[J_i] \in \mathbb{R}^{Nn} \\ \mathcal{D} &:= (\mathcal{P}^T \otimes I_{n^2}) diag[J_i \otimes J_i] \in \mathbb{R}^{Nn^2}, \end{aligned} \tag{11}$$

where I_n is the $n \times n$ identity matrix, $\mathcal{P}$ is transition probability matrix and $\otimes$ denotes Kronecker product. Define also

$$\psi_j := \sum_{i=1}^{N} p_{ij}\tilde{B}_i\gamma\pi_i \in \mathbb{R}^n, \quad \psi := [\psi_1^T, \ldots, \psi_N^T]^T \in \mathbb{R}^{Nn},$$

where p_{ij} is (i,j)-th entry of $\mathcal{P}$ and $\pi \in \mathbb{R}^{1\times N}$ is the stationary distribution of the Markov chain, which exists due to the ergodicity directly followed by assumption of positive $\mathcal{P}$. Now define

$$q = [q_1^T, \ldots, q_N^T]^T = (I - \mathcal{C})^{-1}\psi \in \mathbb{R}^{Nn} \tag{12}$$

$$R(q) := (R_1(q), \ldots, R_N(q)) \in \mathbb{H}^{n\times n}, \tag{13}$$

where

$$\begin{aligned} R_j(q) := \sum_{i=1}^{N} p_{ij}(&\tilde{B}_i\Gamma\tilde{B}_i^T\pi_i + J_i q_i \gamma^T \tilde{B}_i^T \\ &+ \tilde{B}_i\gamma q_i^T J_i^T)) \in \mathbb{R}^{n\times n} \end{aligned} \tag{14}$$

Finally, define

$$\begin{aligned} Q &= (Q_1, \ldots, Q_N) \in \mathbb{H}^{n\times n}, \\ &= \hat{\varphi}^{-1}\left((I - \mathcal{D})^{-1}\hat{\varphi}(R(q))\right) \\ \mu &:= \sum_{i=1}^{N} q_i \in \mathbb{R}^n, \quad \mathbf{Q} := \sum_{i=1}^{N} Q_i \in \mathbb{R}^{n\times n}. \end{aligned} \tag{15}$$

With all these definitions, we can finally state the steady state correlation of the estimation error when convergence occurs.

Lemma 1: Under the hypothesis of Theorem 1, the steady state correlation matrix is given by $\mathbf{Q}$ in (15), and $\mathbf{Q} > 0$. □

Remark 1 (Relationship to consensus): The proposed algorithm is closely related to average consensus protocols that have been widely studied. In fact, convergence of the mean of the estimation error can be independently established by using results from the consensus literature. If time-variation of the graph is examined in the deterministic setting, it can be shown that the mean of the error converges to 0 if there is an integer T so that the union graph $G^*(k) = \cup_{i=k}^{k+T}\mathcal{G}(i)$ at every k is connected. This follows upon taking expectation (over the noise alone) of both sides of (7) to obtain $\mathrm{E}[\mathbf{e}(k+1)] = J(k)\,\mathrm{E}[\mathbf{e}(k)]$, which turns out to be an average consensus algorithm with the state of the reference node fixed at 0, and then applying results from [11], [12]. The effect of stochastic disturbances on consensus algorithms in graphs that do not change with time has been examined by Xiao *et. al.* [13], Bamieh *et. al.* [15] and Kar et. al.[16]. The examination of the covariance of $\mathbf{e}(k)$ with time-varying graph topologies does not seem tractable with such techniques. □

IV. Numerical investigation

In this section we present results from simulations with two networks, the first one with 4 nodes and the second with 25 nodes. These two networks are referred to as network 4A and 25B, respectively. Because skew and offset estimation

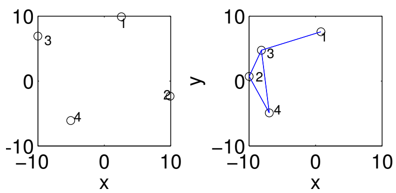
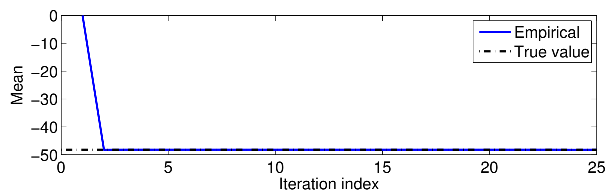
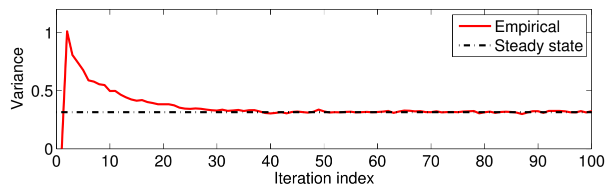


Fig. 1. The graph $G(k)$ at two distinct k; for network 4A.



(a) Mean



(b) Variance

Fig. 2. Mean and variance of the estimate of node 3's node variable (clock offset) as a function of time in network 4A; empirically estimated from 5000 Monte Carlo experiments. In (b), The dot-dash line is the theoretical steady state variance as predicted by Lemma 1.

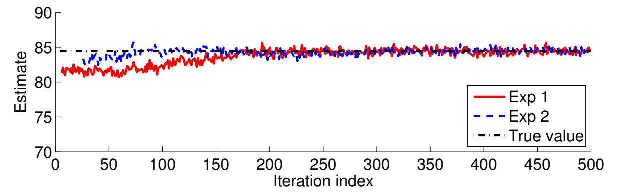
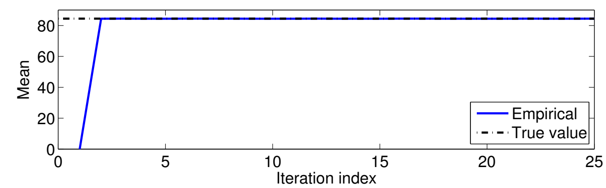
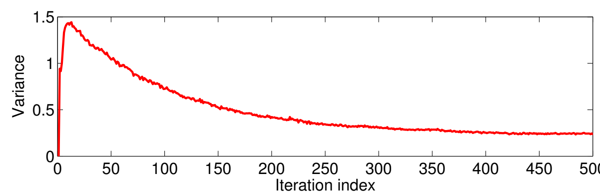


Fig. 3. The trajectory of node 24's estimate of its node variable (clock offset) as a function of time in network 25B, for two distinct numerical experiments.



(a) Mean



(b) Variance

Fig. 4. Mean and variance of the estimate of node 24's node variable (clock offset) as a function of time in network 25B; empirically estimated from 5000 Monte Carlo experiments.

are special cases of the node-variable estimation problem as discussed in Section II, we run simulations for the offset estimation problem only, with clock skews setting to 1 for all nodes. The clock offsets of the nodes with respect to the reference node (indexed as 4 and 25, respectively) are fixed arbitrarily. The noise on each measurement is a Normally distributed random variable with mean 0 and variance 1. The initial estimate of each node is chosen by using the flagged initialization method of [6]. It should be noted that the flagged initialization scheme does not alter the convergence properties of the algorithm since it only affects the initial conditions.

In both networks, the positions of the mobile nodes evolve according to

$$p_u(k+1) = f(p_u(k) + \Delta_u(k)), \tag{16}$$

where, for every u, $p_u(k) \in \mathbb{R}^2$ is the position of agent u at time k, and $\{\Delta_u(k)\}$ is a zero mean white noise sequence with covariance $4I$, where I is the 2×2 identity matrix. The function $f(\cdot)$ is a projection function acting on $p_u(k)$ such that $p_u(k)$ remains in the region $[-10, 10] \times [-10, 10]$ (for network 4A) and in $[-50, 50] \times [-50, 50]$ (for network 25B). In the network 4A and 25B, an edge between two nodes at time k exists if and only if the Euclidean distance between them is less than or equal to 10 and 30, respectively. Figure 1 shows two snapshots of the graph sequence $\mathcal{G}(k)$ resulting from the motion of the 4 mobile nodes in network 4A.

Monte Carlo experiments are conducted to empirically estimate the mean and variance of the estimation errors, by averaging over 5000 sample runs. Figure 2(a) shows the empirically estimated mean of node 3's estimate and the true value. As predicted by Theorem 1, the mean of the estimate converges to the true value. Figure 2(b) shows the empirically estimated variance of node 3's estimate and the theoretical prediction from Theorem 1 (Eq. (15)). Computation of the theoretical predictions requires knowledge of the Markov transition probability matrix $\mathcal{P}$ and the stationary distribution of the chain. For the network 4A, the number of possible graphs is 64, and the 64×64 transition probability matrix is estimated from the Monte Carlo experiments. The stationary distribution π is computed from the estimated $\mathcal{P}$.

Monte Carlo simulations are conducted with the network 25B. Figure 3 shows two distinct sample runs of the estimate of node 24 in the network 25B. Each sample run is obtained by running the algorithm with a distinct sequence of pseudo-random numbers generated in MATLAB. The plot shows that the estimates quickly approach the true value and then oscillate around it. Figure 4(a) and 4(b) show the mean and variance of the estimates of node 24. As predicted by Theorem 1, the mean of the estimate converges to the true value. Due to the motion model that does not preclude any of the $2^{25 \cdot 12}$ possible graphs, determining the steady state variance is computationally infeasible. Hence, in this case we are unable to provide comparisons of the variance with the predicted steady state value.

Remark 2 (Conjecture 1): To test the conjecture, we performed Monte Carlo simulations with another Markov chain whose state space consisted of the three graphs shown in

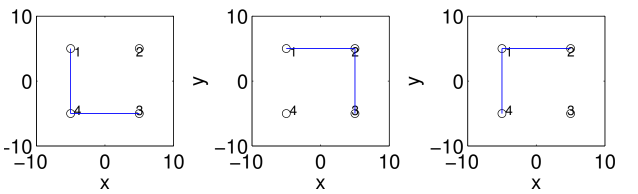
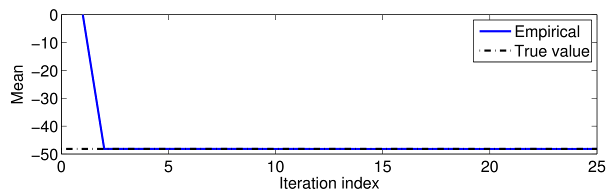
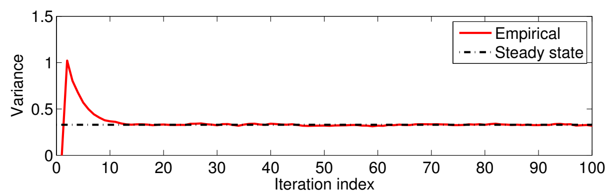


Fig. 5. All three graphs $\mathcal{G}_1, \mathcal{G}_2, \mathcal{G}_3$ that comprises $\mathbb{G}$ for the network of Remark 2. Node 4 is the reference.



(a) Mean



(b) Variance

Fig. 6. Mean and variance of the estimate of node 3's node variable (clock offset) as a function of time in the network discussed in Remark 2; empirically estimated from 5000 Monte Carlo experiments. In (b), the dot-dash line is theoretical steady state variance as predicted by Lemma 1.

Figure 5. The transition probability matrix of the chain was chosen as

$$\mathcal{P} = \begin{bmatrix} 0.3 & 0.2 & 0.5 \\ 0.1 & 0.5 & 0.4 \\ 0 & 0.5 & 0.5 \end{bmatrix}. \tag{17}$$

Notice that $\mathcal{P}$ is not entry-wise positive and none of the graphs in the state space $\mathbb{G}$ is connected. However, the conditions of the conjecture, i.e. the chain is ergodic and the union of the graphs in $\mathbb{G}$ is connected, are satisfied. Figure 6(a) and 6(b) show that the mean converges to 0 and variance converges to the value predicted by Lemma 1, even though the conditions are violated. □

V. SUMMARY

We proposed a distributed time-synchronization protocol for mobile ad-hoc and sensor networks. To the best of our knowledge, synchronization in networks with truly mobile nodes has not been analyzed earlier. The problem was converted to estimation of scalar node variables (offsets and log-skews) from noisy relative measurements between node pairs. The proposed algorithm is based on an iterative update law. When the time-evolution of the communication graph can be modeled as a Markov chain, we showed that under certain conditions the estimation algorithm is mean square convergent: the estimated values converge to their true values in mean, and their variances converge to positive constants. This means the mean and the variance of the estimation error does not change much after sufficient iterations. Therefore, nodes can stop updating their estimates to save bandwidth and on-board energy while incurring little loss of synchronization accuracy. Avenues for future work include establishing weaker sufficient conditions for mean square convergence and characterization of the convergence rate in terms of the Markov chain's properties.

REFERENCES

[1] B. M. Sadler and A. Swami, "Synchronization in sensor networks: an overview," in *IEEE MILCOM*, October 2006.

[2] S. Ganeriwal, R. Kumar, and M. B. Srivastava, "Timing-sync protocol for sensor networks," in *ACM Conference on Embedded Networked Sensor Systems (SenSys)*, 2003.

[3] M. Maróti, B. Kusy, G. Simon, and Ákos Lédeczi, "The flooding time synchronization protocol," in *Proc. of the 2nd international conference on Embedded networked sensor systems*. ACM Press, 2004.

[4] J. Elson, L. Girod, and D. Estrin, "Fine-grained network time synchronization using reference broadcasts," in *the Fifth Symposium on Operating Systems Design and Implementation (OSDI)*, 2002.

[5] P. Barooah and J. P. Hespanha, "Distributed estimation from relative measurements in sensor networks," University of California, Santa Barbara, Tech. Rep., September 2005.

[6] P. Barooah, N. M. da Silva, and J. P. Hespanha, "Distributed optimal estimation from relative measurements for localization and time synchronization," in *Distributed Computing in Sensor Systems DCOSS*, ser. LNCS, P. B. Gibbons, T. Abdelzaher, J. Aspnes, and R. Rao, Eds. Springer, 2006, vol. 4026, pp. 266 – 281.

[7] R. Solis, V. S. Borkar, and P. R. Kumar, "A new distributed time synchronization protocol for multihop wireless networks," in *Proc. of the 45th IEEE Conference on Decison and Control*, December 2006.

[8] L. Schenato and F. Fiorentin, "Average timesync: A consensus-based protocol for time synchronization in wireless sensor networks," in *Proceedings of 1st IFAC Workshop on Estimation and Control of Networked Systems (NecSys09)*, September 2009.

[9] P. Barooah and J. P. Hespanha, "Estimation from relative measurements : Algorithms and scaling laws," *IEEE Control Systems Magazine*, vol. 27, no. 4, pp. 57 – 74, August 2007.

[10] C. Liao and P. Barooah, "Time synchronization in mobile sensor networks from relative measurements," Mechanical and Aerospace Engineering, University of Florida, Tech. Rep., September 2010. [Online]. Available: http://plaza.ufl.edu/cdliao/

[11] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Transactions on Automatic Control*, vol. 48, no. 6, pp. 988–1001, June 2003.

[12] L. Moreau, "Stability of multiagent systems with time-dependent communication links," *IEEE Transactions on Automatic Control*, vol. 50, no. 2, pp. 169–182, February 2005.

[13] L. Xiao, S. Boyd, and S.-J. Kim, "Distributed average consensus with least-mean-square deviation," *Journal of Parallel and Distributed Computing*, vol. 67, pp. 33–46, 2007.

[14] O. Costa, M. Fragoso, and R. Marques, *Discrete-Time Markov Jump Linear Systems*, ser. Probability and its Applications. Springer, 2004.

[15] B. Bamieh, M. R. Jovanović, P. Mitra, and S. Patterson, "Effect of topological dimension on rigidity of vehicle formations: fundamental limitations of local feedback," in *Proceedings of the 47th IEEE Conference on Decision and Control*, Cancun, Mexico, 2008, pp. 369–374.

[16] S. Kar and J. Moura, "Distributed consensus algorithms in sensor networks with imperfect communication: Link failures and channel noise," *IEEE Transactions on Signal Processing*, vol. 57, no. 5, pp. 355–369, January 2009.