Decentralized Algorithms for Sequential Network Time Synchronization

Maxime Cohen
Department of Electrical Engineering
Technion
Haifa 32000, Israel
maxcohen@tx.technion.ac.il

Nahum Shimkin
Department of Electrical Engineering
Technion
Haifa 32000, Israel
shimkin@ee.technion.ac.il

***Abstract*[1]—Accurate clock synchronization is important in many distributed applications. Standard algorithms, such as the Network Time Protocol (NTP), essentially rely on pairwise offset estimation between adjacent nodes. Some recent work introduced more elaborate algorithms for clock offset estimation, which take into account the algebraic constraints imposed on the sum of offsets over network cycles, using a least-squares framework. These algorithms are iterative and decentralized in nature, requiring several cycles of local communication among neighbors for convergence. In this paper, we extend this approach towards a sequential estimation framework, which allows to incorporate initial time estimates along with their uncertainty, as well as multiple rounds of pairwise measurements. We propose a decentralized implementation of the estimation algorithm that employs only local broadcasts and establish its convergence to the optimal centralized solution. We also present some simulation results to illustrate the performance benefits of the suggested algorithms.**

Keywords: network clock synchronization; decentralized algorithms; Kalman filtering; recursive estimation.

I. INTRODUCTION

Accurate clock synchronization has been extensive studied and applied in the context of communication networks, from the Interent [10] to sensor networks [17]. The task of synchronizing clocks in distributed systems is usually accomplished via the exchange of time-stamped messages (probe packets) between the distributed entities in order to coordinate their time. There is a large literature on how to synchronize clocks in traditional networked systems; among these, the "Network Time Protocol" (NTP) is the most widely accepted standard for synchronizing clocks over the Internet [10], [11]. This protocol essentially uses a so-called hierarchical approach by sending probe messages along a layered spanning tree of the network.

More recently, a novel approach for time synchronization termed CTP – Classless Time Protocol [4] was proposed. This non-hierarchical approach exploits convex optimization theory in order to minimize a quadratic objective function of clock offsets. It was shown that CTP substantially outperforms hierarchical schemes such as NTP in terms of clock accuracy without increasing complexity. An related approach that relies on Least-Squares fit was proposed in [3], [15]. The accuracy of clock synchronization was improved by exploiting global network-wide constraints (i.e., the relative offsets must sum up to zero over network loops). The central characteristic of these methods is the use of a distributed algorithm that requires only local broadcasts among neighboring nodes. The algorithms are iterative, and typically converge within a small number or rounds. The work in [1] extends the same LS approach to a Weighted Least-Squares (WLS) framework, where each measurement may be assigned a different weight.

It is interesting to note, following [1], that the time synchronization problem is mathematically equivalent to a general class of distributed estimation problem of additive quantities over a network, a class that includes sensor localization over a sensor network (in Cartesian coordinates).

The basic Least-Squares framework of [3] considered the offset estimation problem using only a single set of measurements. Our goal here is to extend this framework into a sequential estimation one, which handles prior estimates of clock offsets as well as multiple measurement sets. These goals can be cast in the framework of Kalman Filtering, which indeed readily provides an optimal centralized solution. However, as the obtained equations are not readily amenable to a decentralized implementation, we resort to the equivalent least-squares formulation, and employ localized least squares iterations to obtain a decentralized algorithm. This algorithm employs only local broadcasts between neighbors when the initial covariance matrix is diagonal.

A rich literature exists on distributed implementation of the Kalman Filter (KF), dating back to [5], [14]. More recent work, such as [12], focused on network structures and developed consensus-based algorithms that employ only local communication. This line of work allows measurement to be processed locally, however each node keeps a copy of the entire state vector. In our framework, each node keeps only data related to its own offset. Interesting work in this line is presented in [9], where decentralized approximation schemes are presented. Our goal here is to develop exact algorithms, for the more limited problem that we consider.

Finally, we present simulation results over several network topologies for evaluating and comparing the accuracy of the proposed time synchronization schemes. We provide several interesting comparisons, where the Kalman Filter approach outperforms the existing algorithms.

[1] Submitted to NetCoop 2010. Corresponding author: N. Shimkin

The paper is organized as follows. In section II, we describe the model and formulate the estimation problem in state-space form. In section III, we present an iterative, decentralized algorithm that combines prior offset estimates with a single set of measurements, and establish its convergence. While the development is carried out for a general initial state covariance matrix, the local communication structure is maintained only if this matrix is diagonal. Hence, a recursive extension of this algorithm to the case of multiple measurement sets is not apparent. The case of multiple measurement sets is taken up in Section IV, where an exact algorithm is developed. We also consider for comparison purposes a simple sub-optimal algorithm that neglects the off-diagonal terms of the inverse covariance matrix. Section V presents some simulation results that evaluate and compare the performance of the proposed algorithms. Finally, the conclusions and some notes on future directions are reported in section VI.

II. MODEL AND PROBLEM DEFINITIONS

We model the network as a directed graph $G=(V,\mathrm{E})$ with $N=|V|$ nodes denoted $\{\Lambda_1,\Lambda_2,...,\Lambda_N\}$, and $m=|\mathrm{E}|$ edges. Each edge represents the ability to transmit and receive packets between the corresponding pair of nodes. The edge connecting nodes Λ_i and Λ_j is denoted by e_{ij}. We assume that all the edges are bidirectional, so that $e_{ij}\in\mathrm{E}$ implies $e_{ji}\in\mathrm{E}$, and that the network graph is connected, namely there exists a path between any pair of nodes in the network. Denote by N_i the set of nodes which are the neighbors of Λ_i, i.e., one edge away from node Λ_i, and let $|N_i|$ be the number of such neighbors. We assume that each node in the network keeps a local clock, and our goal is to estimate the time offset of each clock with respect to some global reference. Without loss of generality, we may assign node Λ_1 as the reference time node, so that all clock offsets are estimated with respect to this node's clock.

A. Clock Model

A standard model for the clock drift at a node follows the linear form: $T_i(t)=\alpha_i t+\tau_i$, where α_i and τ_i are the skew (rate deviation) and the offset parameters respectively, t is the real time (or reference time) and $T_i(t)$ is the local time at node Λ_i. This model is known as the two parameters linear model (see [15] and the references therein). As node 1 serves as the reference, we have by default $\tau_1=0$ and $\alpha_1=1$.

We focus here on the simplified model where all clocks run at the same speed, so that there is no skew ($\alpha_i=1$ for all i). This assumption is appropriate when the measurement time span is short so that the rate deviation is small relative to the skew.

B. Measurement Model

We consider a two-way offset measurement scheme. Each network node (Λ_i, $i=1,2,...,N$) sends probe packets to each of its neighbors. Upon sending a packet k_m the sender Λ_i stamps the packet with its local time $T_i(k_m)$, and the receiver Λ_j stamps the packet upon receiving it with its local time $R_j(k_m)$. Then, node Λ_j retransmits the packet back to the source with time stamp $T_j(k_m)$, and the source stamps its local time $R_i(k_m)$ when receiving the packet back. We thus obtain

$$\Delta T_{ij}\triangleq R_j(k_m)-T_i(k_m)=D_{ij}(k_m)-\tau_i+\tau_j+\varepsilon_{ij}.$$

Here $D_{ij}(k_m)$ is the propagation delay over link e_{ij}, ε_{ij} is an additive noise that represents the random queuing delay (and the other unknown influences) and $\tau_j-\tau_i$ is the difference between the two clock offsets. Assuming that $D_{ij}(k_m)=D_{ji}(k_m)$ (symmetric propagation delay) we obtain:

$$\hat{O}_{ij}\triangleq\frac{1}{2}\left(\Delta T_{ij}-\Delta T_{ji}\right)=\tau_j-\tau_i+v_{ij} \qquad (1)$$

where $v_{ij}=\left(\varepsilon_{ij}-\varepsilon_{ji}\right)/2$ is the effective measurement noise. We shall assume that v_{ij} is zero mean, with covariance $r_{ij}>0$.

Remark: In practice, the effective measurement $\hat{O}_{ij}$ may be obtained by processing several subsequent packet exchanges. One way to combine these successive measurements is to consider their average. More refined methods such as a minimum filter may favor measurements that reflect smaller delays, to account for changing congestion conditions (cf. [10]). In any case, the combined measurement is taken to be of the above form.

C. State-Space Equations

Our objective is to synchronize all the clocks in the network with the reference time. This is equivalent to estimating the offset τ_i at each network node. We start by formulating the statistical estimation problem in state space form. Define the state as the column vector $x\triangleq\left(\tau_2,...\tau_N\right)^T$ (recall that $\tau_1=0$ by definition). As x is assumed to be constant for the time frame of interest, we can write $x(n+1)=x(n)$, where $n\geq 0$ is the time (or step) index. The initial state $x(0)$ is assumed to have known first and second order statistics: $E\left[x(0)\right]=\bar{x}_0$, $\mathrm{cov}\left[x(0)\right]=P_0$. This allows to take into account prior information about the accuracy of the initial clock offsets in different nodes. For example, the network may contain several reference nodes which keep accurate time, which translates to small covariance entries in P_0.

The effective measurement for each pair of neighboring nodes is given by equation (1). Thus, $\hat{O}_{ij}$ measures the offset difference for these nodes, plus an additive noise v_{ij}. The noise v_{ij} is assumed to have zero mean, with finite covariance $r_{ij}>0$, and is independent across node pairs. We collect the individual measurements in a column vector $y=(\hat{O}_{ij},\ ij\in\mathrm{E})$,

which represents a single measurement set. We assume that a new measurement set $y(n)$ is obtained at each step $n \geq 1$. We note that n need not refer to actual time, but rather corresponds to the epoch when the n-th measurement set $y(n)$ become available.

To express the measurement equation in vector form, define the graph incidence matrix $\tilde{A}$ whose dimensions are N (nodes) $\times$ m (edges), and where in the row corresponding to node Λ_i, we have an entry +1 for all edges e_{ij} originating in Λ_i, an entry -1 for all edges e_{ji} terminating in Λ_i, and 0 otherwise. For a connected graph, the rank of the incidence matrix is $N-1$. Thus, deleting any row from the incidence matrix yields a full row rank matrix, which is called the reduced incidence matrix. Here, we will work with the $(N-1)\times m$ matrix A obtained by deleting the row corresponding to the reference node Λ_1.

The model may now be summarized in state space form:

$$\begin{cases} x(n) = x(n-1) \\ y(n) = A^T x(n) + v(n) \end{cases} \quad ; \quad n \geq 1$$

with initial conditions $E[x(0)] = \bar{x}_0$, $\mathrm{cov}[x(0)] = P_0$. The measurement noise sequence $\{v(n)\}$ is assumed to be a white noise sequence with zero mean and covariance $R(n) = R > 0$. We further assume that there is no correlation between the measurement noise over different links, so that $R = \mathrm{diag}\{r_{ij}, ij \in E\}$, as a diagonal matrix with elements $r_{ij} > 0$ along the diagonal. Finally, the measurement noise and initial state $x(0)$ are uncorrelated.

Our goal is to estimate the offset vector $x(n) = x$ based on the measurement sets $y(1),\ldots,y(n)$ and the prior information $\bar{x}_0, P_0$. This is of course a classical problem in sequential estimation theory. As is well known, the Kalman filter provides the optimal linear solution the MMSE (Minimal Mean Square Error) sense, which further coincides with the optimal (conditional expectation) solution under the Gaussian assumption. However, while the centralized KF equations are easily written, it is not readily seen how they may distributed. We thus proceed to develop distributed algorithms that converge to the optimal KF solution.

III. SINGLE MEASUREMENT SET

We start by considering a single measurement update, namely the problem of estimating $x(1) = x$ based on the measurement $y(1) = y$ and the prior information $\bar{x}_0, P_0$. As is well known (e.g., [7], Section 5.3), the KF equations are equivalently obtained as a solution to a Least-Squares deterministic problem, which in our case reduces to the minimum of the following objective function:

$$J(x) = (x - \bar{x}_0)^T P_0^{-1} (x - \bar{x}_0) + (y - A^T x)^T R^{-1} (y - A^T x) \qquad (2)$$

The first term is related to the initial information regarding the clock offsets, whereas the second term is associated with the single set of measurements and its corresponding covariance matrix R.

In the development of a distributed algorithm, we will find it more convenient to manipulate the above deterministic LS problem rather than starting with the KF equations.

A. Baseline Algorithm

We first review the existing algorithm introduced in [3], [4],[15]. In this case, the objective function is given by:

$$J(x) = (y - A^T x)^T (y - A^T x) = \sum_{\substack{i,j \\ j \in N_i}} \left(\hat{O}_{ji} - \tau_i + \tau_j \right)^2$$

The first order optimality conditions are

$$\frac{\partial J}{\partial \tau_i} = \left(AA^T \right)_i x - \left(A \right)_i y = -2 \sum_{j \in N_i} \left(\hat{O}_{ji} - \tau_i + \tau_j \right) = 0$$

From this, we obtain:

$$\tau_i = \frac{1}{|N_i|} \cdot \sum_{j \in N_i} \left(\hat{O}_{ji} + \tau_j \right) \qquad (3)$$

The above equation must be satisfied by the optimal solution of the offset estimation problem. While this is a set of linear equations, a direct solution cannot be carried out in a decentralized manner. Instead, a decentralized iterative algorithm was suggested and shown to convergence to the optimal centralized solution. This algorithm simply iterates the above equation, which can be simply interpreted as follows. Each node obtains from its neighbors their current estimates for their own clock offsets, and computes its offset estimate as the average of all its neighbors' estimates plus the corresponding relative measurements. This procedure is the same as in [3], [15] and one can easily show that this is equivalent to the algorithm in [4].

Our objective is to extend the previous result to a wider framework and we will obtain this procedure as a special case of a more general algorithm.

Next, we will consider the more general framework that includes the initial covariance matrix in the objective function in addition to a weighting matrix R^{-1} . The analysis is divided in two cases: non-diagonal and diagonal initial covariance matrix.

B. Adding Initial Conditions and Measurement Weights

We now consider the distributed solution of equation (2), that incorporates the initial estimate $\bar{x}_0$ with covariance P_0, as well as possibly different weights for the measurement as expressed by the covariance matrix R. We develop an iterative algorithm, and establish its convergence to the optimal solution of (2). However, it will be seen that the algorithm is not truly distributed, in the sense that each node must receive data from non-neighboring nodes, unless the initial covariance matrix is diagonal. The latter case will be considered in the next subsection.

Proceeding similarly to above, the first-order optimality conditions for (2) are

$$\frac{\partial J}{\partial \tau_i} = \tau_i \sum_{j\in N_i} \frac{1}{r_{ji}} - \sum_{j\in N_i} \frac{1}{r_{ji}}\left(\hat{O}_{ji} + \tau_j\right) + \sum_{k=1}^{N}\left(P_0^{-1}\right)_{ik}\left(\tau_k - \tau_k(0)\right) = 0$$

or, equivalently,

$$\tau_i = \frac{1}{I_i}\left[\sum_{j\in N_i}\frac{1}{r_{ji}}\left(\hat{O}_{ji}+\tau_j\right) + \left(P_0^{-1}\right)_{ii}\tau_i(0) - \sum_{\substack{m=1\\ m\neq i}}^{N}\left(P_0^{-1}\right)_{im}\left(\tau_m - \tau_m(0)\right)\right] \quad (4)$$

where

$$I_i = \sum_{j\in N_i}\frac{1}{r_{ji}} + \left(P_0^{-1}\right)_{ii}.$$

These equations motivate the following synchronous iterative algorithm for their solution:

$$\hat{\tau}_i^{(k+1)} = \frac{1}{I_i}\Big[\sum_{j\in N_i}\frac{1}{r_{ji}}\left(\hat{O}_{ji}+\hat{\tau}_j^{(k)}\right) + \left(P_0^{-1}\right)_{ii}\tau_i(0) - \dots - \sum_{\substack{m=1\\ m\neq i}}^{N}\left(P_0^{-1}\right)_{im}\left(\hat{\tau}_m^{(k)} - \tau_m(0)\right)\Big] \quad (5)$$

with initialization $\hat{\tau}_i^{(0)} = \tau_i(0),\ \ i = 2,3,\dots N$. Here, $k \geq 0$ is the iteration number. This algorithm can be interpreted as a the classical Jacobi iteration for the solution of the linear equations (5), or, equivalently, as a local least-squares algorithm where each node minimizes at each iteration the objective function (2) over its own offset, given the current offset estimates of the other nodes.

The following convergence result is next established.

Theorem 1. *Suppose that:*

(1) The matrix R^{-1} is diagonal and Positive Semi-Definite, that is: $0 \leq \left(r_{ji}\right)^{-1} < \infty \ \ \forall i, j$.

(2) The initial covariance matrix P_0 is an M-matrix, namely:

$$\begin{cases} \sum_j \left(P_0^{-1}\right)_{ij} \geq 0 \\ \left(P_0^{-1}\right)_{ii} \geq 0 \ \text{ and } \ \left(P_0^{-1}\right)_{ij} \leq 0 \ \ (i \neq j) \end{cases}$$

(3) The clock adjustment operation in (5) *is applied synchronously by all nodes ($i = 2,3,\dots N$) in all iterations.*

Then, the iterated estimators $\hat{\tau}_i^{(k)}(n) \ \ i = 2,3,\dots N$ converge (as $k \to \infty$) to the optimal offsets that minimize the objective function in (2).

The proof is provided in the Appendix.

The main problem with the last iterative algorithm is that, in general, each node needs to communicate with all the other nodes and not only with its neighbors. Thus, each node is required to be aware of the global topology of the network, and the algorithm does not satisfy the requirement of local information exchanges only. Fortunately, this problem does not exist when initial covariance matrix P_0 is diagonal.

C. Diagonal P_0

We henceforth specialize the discussion to the case where the initial covariance matrix P_0 (hence its inverse P_0^{-1}) is diagonal, with diagonal elements (p_i). This will be the case when the initial estimates of clock skews are obtained by the different nodes independently. For example, some nodes may have a GPS receivers which allows then to obtain an accurate estimate of the time. Or, starting from an initially accurate estimate, each node has after some time an added uncertainty due to the estimated drift of its clock.

The decentralized iterative procedure in (5) now reduces to:

$$\hat{\tau}_i^{(k+1)} = \frac{1}{\sum_{j\in N_i}\frac{1}{r_{ji}} + \frac{1}{p_i}} \cdot \left[\sum_{j\in N_i}\frac{1}{r_{ji}}\left(\hat{O}_{ji} + \hat{\tau}_j^{(k)}\right) + \frac{\tau_i(0)}{p_i}\right] \quad (6)$$

In words, τ_i is computed at each iteration as a weighted average between the modified estimates obtained from adjacent nodes, and its prior estimate. Observe that if the matrix P_0^{-1} is equal to zero and $R^{-1} = I$, we obtain the equation (3) as in the basic LS case described above. This algorithm requires only local broadcasts between adjacent nodes. Evidently, the convergence result in Theorem 1 applies here as a special case.

IV. MUTILPLE MEASUREMENT SETS

In the previous section we have developed a decentralized algorithm for the case of a single measurement set, under the assumption that P_0 is diagonal. We next consider the case when multiple measurement sets become available sequentially, with the goal of presenting a recursive version of the previous decentralized algorithms. As its turns out, a simple recursive extension of the single-measurement case, in the style of the Kalman Filter, will not work here. The reason is that even if the initial covariance matrix P_0 is diagonal, which we assume, after the first iteration the state covariance matrix P_1 will not be diagonal any more (as can be easily verified, the estimates will become correlated). Thus, although the iterative procedure in (5) can be formally repeated, it will not be distributed any more, as each node has to communicate and store information related to all the other nodes over the network and not only with the one-hop neighbors.

Our objective is therefore to derive an alternative recursion that is suitable for decentralized implementation. This is done in the next subsection. For comparison purposes we also mention, in the subsequent subsection, an approximate algorithm that simply neglects the off-diagonal elements of the covariance matrix after each measurement update.

A. Optimal Decentralized Algorithm

For the multiple measurement update case, the equivalent Least Squares objective function is given by:

$$J(n) = (x - \bar{x}_0)^T P_0^{-1}(x - \bar{x}_0) + \sum_{k=1}^{n}(y(k) - A^T x)^T R^{-1}(y(k) - A^T x) \quad (7)$$

For notational simplicity assume that the matrix R^{-1} is identical for each set of measurements. We propose the following iterative algorithm:

$$\hat{\tau}_i^{(k+1)}(n) = \hat{\tau}_i(n-1) + I_i(n)^{-1} \cdot \Big\{ \sum_{j \in N_i} \frac{1}{r_{ji}} \cdot [\hat{O}_{ji}^{(n)} - \big(\hat{\tau}_i(n-1) - \hat{\tau}_j^{(k)}(n)\big)] + (n-1) \cdot [\sum_{j \in N_i} \frac{1}{r_{ji}} \cdot \big(\hat{\tau}_j^{(k)}(n) - \hat{\tau}_j(n-1)\big)] \Big\}, \quad (8)$$

$$I_i(n) = I_i(n-1) + \sum_{j \in N_i} \frac{1}{r_{ji}} = \big(P_0^{-1}\big)_{ii} + n \cdot \sum_{j \in N_i} \frac{1}{r_{ji}}, \quad n \geq 1$$

for $i = 2,3,...,N$, initialized with $\hat{\tau}_i^{(0)} = \tau_i(0)$, $I_i(0) = (P_0^{-1})_{ii}$.

The above set of equations is a decentralized, synchronous and recursive algorithm that computes at each step the estimated offsets and their corresponding error variances. The main advantage of this algorithm is its local nature; each network node needs to communicate only with its neighbors.

We now describe in words the iterative procedure in (8). At time n, we assume that the estimate of $\hat{\tau}_i(n-1)$ is given. Then, $\hat{\tau}_i^{(k)}(n)$ $k = 1,2,...$ is computed based on $\hat{\tau}_i(n-1)$ and the last measurement set $y(n)$. We assume that a sufficient number of iterations is performed at each time n, so that the estimate $\hat{\tau}_i(n)$ is accurate.

Remarks:

1. We point out that the suggested recursion slightly deviates from a standard recursive estimation scheme due to the presence of n (time or measurement count) as a factor in the recursion.
2. It may be shown that the elements of $I_i(n)$ are the diagonal entries of the inverse covariance matrix in the Kalman filter equations, namely $I_i(n) = (P_n^{-1})_{ii}$. In fact, the above iteration for $I_i(n)$ is the same as the iteration over the diagonal elements in the information form of the Kalman covariance update:

$$\big(P_{n+1}\big)^{-1} = \big(P_n\big)^{-1} + AR^{-1}A^T \quad (9)$$

Observe however that we do not compute the non-diagonal elements of the inverse covariance matrix.

The proposed algorithm may be derived is by differentiating $J(n-1)$ and $J(n)$ with respect to the offsets vector x and set the partial derivatives to zero. The algebraic details (which can be found in [2]) are omitted since the procedure is similar to the previous case.

An alternative derivation, also presented in [2], may start with the KF equations in information form. However, the recursion in (8) is not equivalent to the KF recursion, and is not readily seen from these equations.

We next address the convergence the set of equations in (8) to the optimal centralized solution.

Theorem 2. *Suppose that:*

(1) Assumptions (a) and (b) from Theorem 1 hold.

(2) The clock adjustment operation in (8) *is applied synchronously by all nodes ($i = 2,3,...N$) in all iterations, recursively for n sets of measurements.*

(3) A sufficient number of iterations is performed after each measurement set n, so that $\hat{\tau}_i^{(k)}(n)$ converges to $\hat{\tau}_i(n)$.

Then, for each $n \geq 1$, the iterated estimators $\hat{\tau}_i^{(k)}(n)$ $i = 2,3,...N$ converge (as $k \to \infty$) to the optimal offsets that minimize the objective function in (7)*.*

The proof is provided in the Appendix.

Next, we propose a simple sub-optimal algorithm for the case where multiple sets of measurements are available.

B. A Sub-Optimal Decentralized Algorithm

For the case where P_0 is non-diagonal, we obtained in (5)) that the estimated offset of node Λ_i depends on all the other offsets and not only on those of its neighbors. One can consider the naïve sub-optimal algorithm that neglects the off-diagonal terms of the inverse covariance matrix:

$$\hat{\tau}_i^{(k+1)} = \frac{1}{\sum_{j \in N_i} \frac{1}{r_{ji}} + \big(P_0^{-1}\big)_{ii}} \left[\sum_{j \in N_i} \frac{1}{r_{ji}} \big(\hat{O}_{ji} + \hat{\tau}_j^{(k)}\big) + \big(P_0^{-1}\big)_{ii} \tau_i(0) \right]$$

The decentralized sub-optimal recursive algorithm for the multiple measurement scenario is given by:

$$\hat{\tau}_i^{(k+1)}(n) = \hat{\tau}_i(n-1) + \frac{1}{\big(P_0^{-1}\big)_{ii} + n \cdot \sum_{j \in N_i} \frac{1}{r_{ji}}} \cdot \left\{ \sum_{j \in N_i} \frac{1}{r_{ji}} \cdot \left[\hat{O}_{ji}^{(n)} - \big(\hat{\tau}_i(n-1) - \hat{\tau}_j^{(k)}(n)\big) \right] \right\}$$

Here, we may neglect the off-diagonal terms *before* inverting the information matrix $\big(P_{n-1}\big)^{-1}$, in order to reduce the algorithm complexity. In this case, we will invert a diagonal matrix and hence the time computation will significantly decrease.

V. NUMERICAL RESULTS

In this section, we implement some of the algorithms that we previously developed for typical problems and we compare the results with the existing algorithms. More extensive comparisons can be found in [2]. The convergence rates of the decentralized algorithm are not presented here as convergence is achieved after a relatively small number of iterations and the results are very similar than [4]. Consider two different network topologies:

- Network 1: a 400 node network with 997 edges.
- Network 2: a 170 node network with 1200 edges.

The first case we analyze is the one where 10% of the nodes are perfectly synchronized to the global time (through a GPS satellite receiver for example), and the remainder are not synchronized at all. Namely, for these arbitrary 40 nodes we take the initial variances to be very small (0.01) and the offsets equal to zero, and for the rest of the nodes, the variances tend

to infinity and the offsets are randomly chosen according to a uniform distribution. The graphical comparison between the decentralized CTP algorithm (equation (3)) and the Decentralized Kalman Filter (DKF) (equation (6)) is presented in Fig. 2. As expected, the DKF algorithm outperforms the decentralized CTP method in terms of clock accuracy. Fig. 2 shows the fraction of nodes with clock offset with respect to the reference time node that is not grater than t for the different algorithms. In other words, the y-axis represents the fraction of nodes with clock offset, relating to the global time, not greater than the value described by the x-axis.

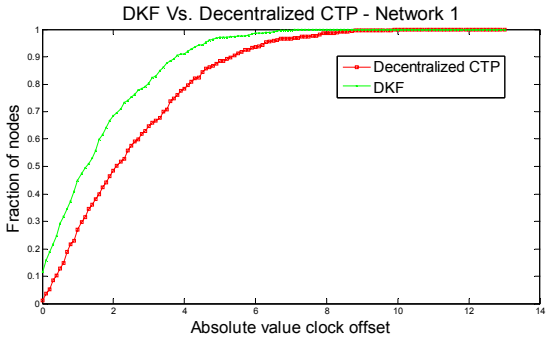

Figure 1. Comparison between the decentralized CTP and DKF algorithms (with 10% of nodes synchronized via GPS) in Network 1.

The second part of this section is devoted to the comparison of the recursive Centralized Kalman Filter (CKF) algorithm to the Sub-Optimal Algorithm (SOA) that neglects the off-diagonal terms of the inverse covariance matrix (see section V. B). We consider the topology of Network 2 and we check several values of n (the number of measurements). The queuing delay is randomized in accordance with the Kalman Filter assumptions, namely normally distributed with zero mean and covariance matrix R:

$$R \sim U[0.01,12] \qquad Q_{delay} \sim N(0,R)$$

In addition, we consider that 10% of the nodes are perfectly synchronized to the global time and the remainder are not synchronized at all (similar to the case in Fig. 2). In this analysis, we also compare the results to the Centralized Least-Squares (CLS) algorithm. Fig. 3 presents the results for the offsets obtained by applying the optimal CKF method, the SOA and the CLS algorithms for two different values of n. As expected, the optimal algorithm gives the best results. The sub-optimal algorithm gives relatively poor results but reduces the complexity and is not diverging. Moreover, we obtained that the sub-optimal algorithm is even worse (in terms of clock accuracy) than the basic centralized Least-Squares method (that does not take into account the initial covariance matrix).

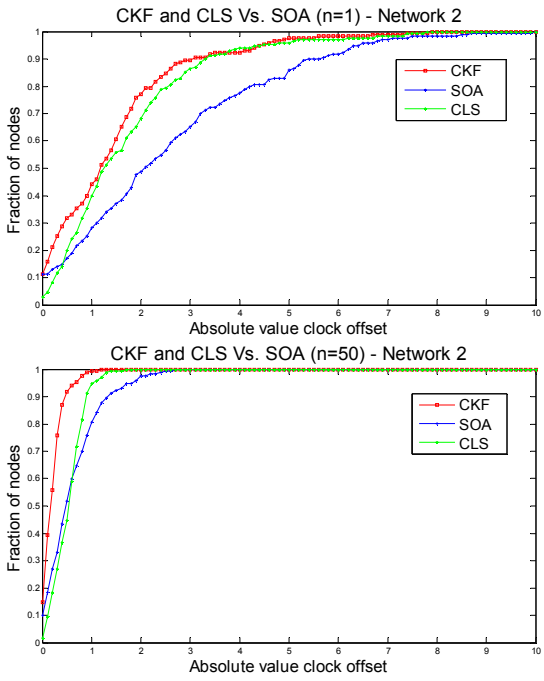

Figure 2. Comparison between CKF, SOA and CLS (with $R \sim U[0.01,12]$ and $P_0 \neq I$) in Network 2 for $n = 1, 50$.

VI. CONCLUSION

In this paper, we have developed several decentralized algorithms for estimating the offset at each network node with respect to the reference time, utilizing a sequential estimation framework. The essential characteristic of these algorithms is their decentralized nature; each node can estimate its clock offset by exchanging packets with its one-hop neighbors only. We extend the existing Least-Squares based algorithms so that we may assign different weights to the measurements according to their accuracy, include a-priori information, and provide a recursive estimation scheme. The main algorithm is both decentralized (requires only local broadcasts), recursive (works in on-line applications) and converges to the optimal centralized solution. Finally, some numerical results were presented to show that, as expected, the proposed algorithm outperforms the existing methods.

We close the paper by mentioning several extensions of interest. A discount factor is easily incorporated into the objective function (7) in order to give a higher weight to the more recent measurements, and leads to similar algorithms. This will be useful when the offsets are time-varying. The proposed algorithms may also be extended to handle dynamic changes in the communication topology by considering temporary link failures, following the treatment in [1]. Further details related to these two issues may be found in [2]. One may also consider more elaborate state dynamics to model possible time variations in the clock offsets. The simplest is adding a white system noise in the state space model. Interestingly, the results of this paper are not easily extendable to this model. Another major issue is the incorporation of the clock skew parameter into the clock model (see section II. A).

These scenarios were partially investigated in [2] and may be considered as directions for future research.

APPENDIX

Proof of Theorem 1: Let us recall that the general objective function is given by:

$$J=(x-\overline{x}_0)^T P_0^{-1}(x-\overline{x}_0)+(y-A^T x)^T R^{-1}(y-A^T x)$$

Let us analyze the convergence properties of the general case, where P_0 is not necessarily assumed to be a diagonal matrix. We recall that iteration (5) cannot be easily decentralized when P_0 is not diagonal as we previously explained. However, the iteration is still well defined mathematically.

The synchronous iteration can be written in vector form:

$$\hat{\tau}^{(k+1)}=\hat{\tau}^{(k)}-\left(\tilde{D}+\tilde{P}_0\right)^{-1}\left(AR^{-1}A^T\hat{\tau}^{(k)}-AR^{-1}y-P^{-1}\overline{x}_0+P_0^{-1}\hat{\tau}^{(k)}\right)$$

Here:

$$\left(\tilde{D}^{-1}\right)_{ij}=\begin{cases}\dfrac{1}{\sum_{j\in N_i}\left(r_{ji}\right)^{-1}} & i=j\\ 0 & otherwise\end{cases}\qquad \left(\tilde{P}_0\right)_{ij}=\begin{cases}\left(P_0^{-1}\right)_{ij} & i=j\\ 0 & otherwise\end{cases}$$

The optimal solution (equivalent to performing the centralized protocol) is given by:

$$\tau^*=\left(AR^{-1}A^T+P_0^{-1}\right)^{-1}\left(AR^{-1}y+P_0^{-1}\overline{x}_0\right)$$

Let us define: $\overline{\tau}^{(k)}\triangleq\hat{\tau}^{(k)}-\tau^*$. Then we obtain after some manipulations:

$$\overline{\tau}^{(k+1)}=M\overline{\tau}^{(k)}$$

where: $$M\triangleq I-\left(\tilde{D}+\tilde{P}_0\right)^{-1}\left(AR^{-1}A^T+P_0^{-1}\right)$$

Thus, the convergence of the sequence $\hat{\tau}^{(k)}$ to $\overline{\tau}^*$ is equivalent to the convergence of $\overline{\tau}^{(k)}$ to the zero vector, which is determined by the matrix M. The necessary and sufficient condition for this convergence is that the spectral radius of M is strictly smaller than 1. The following result is well known (see, e.g., [6], chapter 6).

Proposition 1.
Consider a non-negative square matrix A with the following properties:

a) All the row sums of A are smaller or equal than 1.
b) At least in one row this sum is strictly smaller than 1.
c) The matrix A is irreducible (i.e., there exists a path from any node to any other node in the network).

Then, $\rho(A)<1$.

According to Proposition 1, $\overline{\tau}^{(k)}\to 0$ if the sufficient conditions apply to the matrix M. In order to show that the spectral radius of M is strictly smaller than 1, we will require that the matrix M is both non-negative and sub-stochastic (the row sums are smaller than one).

The elements of the matrix M can be determined by inspection as the following: $M_{ii}=0$, and

$$M_{ij}=\begin{cases}\dfrac{-\left(P_0^{-1}\right)_{ij}+\left(r_{ji}\right)^{-1}}{\sum_{j\in N_i}\left(r_{ji}\right)^{-1}+\left(P_0^{-1}\right)_{ii}} & i\neq j \text{ and } i,j \text{ are neighbors}\\ \dfrac{-\left(P_0^{-1}\right)_{ij}}{\sum_{j\in N_i}\left(r_{ji}\right)^{-1}+\left(P_0^{-1}\right)_{ii}} & otherwise\end{cases}$$

Let us find the conditions for the row sums of the matrix M to be smaller than 1:

$$\sum\nolimits_j M_{ij}=\frac{\sum_{j\in N_i}\left[-\left(P_0^{-1}\right)_{ij}+\left(r_{ji}\right)^{-1}\right]-\sum_{\substack{j\notin N_i\\ j\neq i}}\left(P_0^{-1}\right)_{ij}}{\sum_{j\in N_i}\left(r_{ji}\right)^{-1}+\left(P_0^{-1}\right)_{ii}}$$

$$=\frac{\sum_{j\in N_i}\left(r_{ji}\right)^{-1}-\sum_{\forall j\neq i}\left(P_0^{-1}\right)_{ij}}{\sum_{j\in N_i}\left(r_{ji}\right)^{-1}+\left(P_0^{-1}\right)_{ii}}$$

It follows that $\sum_j M_{ij}\le 1$ if and only if: $\sum_j\left(P_0^{-1}\right)_{ij}\ge 0$.

In other words, we obtained that the necessary condition is that for each node Λ_i, the row sum of the matrix P_0^{-1} has to be non-negative.

Requiring that all the entries of the matrix M are non-negative leads to:

$$\left(P_0^{-1}\right)_{ii}\ge 0;\ \left(P_0^{-1}\right)_{ij}\le 0\ \ (i\neq j)$$

Hence, we can write: $\left(P_0^{-1}\right)_{ii}\ge-\sum_{\forall j\neq i}\left(P_0^{-1}\right)_{ij}\ge 0$

The above requirement can be seen as a diagonal dominance condition over the matrix P_0^{-1}.

In the case that the node Λ_i is adjacent to the reference node, the corresponding row sum of the M matrix is given by:

$$\frac{\sum_{j\in N_i}\left[\left(r_{ji}\right)^{-1}-\left(P_0^{-1}\right)_{ij}\right]-\left[\left(r_{1i}\right)^{-1}-\left(P_0^{-1}\right)_{i1}\right]+\sum_{\substack{j\notin N_i\\ j\neq i}}-\left(P_0^{-1}\right)_{ij}}{\sum_{j\in N_i}\left(r_{ji}\right)^{-1}+\left(P_0^{-1}\right)_{ii}}$$

$$=\frac{\left[\sum_{j\in N_i}\left(r_{ji}\right)^{-1}-\sum_{\forall j\neq i}\left(P_0^{-1}\right)_{ij}\right]-\left[\left(r_{1i}\right)^{-1}-\left(P_0^{-1}\right)_{i1}\right]}{\sum_{j\in N_i}\left(r_{ji}\right)^{-1}+\left(P_0^{-1}\right)_{ii}}<1$$

In the case that Λ_i is not adjacent to the reference node, the corresponding row sum of the M matrix is given by:

$$\frac{\left[\sum_{j\in N_i}\left(r_{ji}\right)^{-1}-\sum_{\forall j\neq i}\left(P_0^{-1}\right)_{ij}\right]-\left(P_0^{-1}\right)_{i1}}{\sum_{j\in N_i}\left(r_{ji}\right)^{-1}+\left(P_0^{-1}\right)_{ii}}<1$$

Hence, we have shown that at least in one row, the row sum of M is strictly smaller than 1. Actually, we proved that the iteration matrix verifies all the sufficient conditions for convergence. Namely, the row sums of the matrix M are less

or equal than 1 (and at least in one row this sum is strictly smaller than 1), the matrix M is irreducible and all its entries are non-negative.

As a result, we proved the convergence of the decentralized algorithm to the optimal solution performed by the centralized Kalman Filter for the most general case.

To sum up, the convergence conditions are given by:

$\sum_j \left(P_0^{-1}\right)_{ij} \geq 0; \left(P_0^{-1}\right)_{ii} \geq 0; \left(P_0^{-1}\right)_{ij} \leq 0 \ \ (i \neq j)$

■

Proof of Theorem 2: The case of $n=1$ was treated in Theorem 1. Our proof relies on the following lemma.

Lemma 1.

Suppose that P_0 satisfies the convergence conditions of Theorem 1, namely P_0 is an M-matrix. Let P_n be computed using (9)*, then P_n is an M-matrix for all $n \geq 1$.*

Proof

Equation (9) corresponds to the measurement update equation of the inverse covariance matrix of the KF. Recalling that the matrix R^{-1} is assumed to be diagonal, let us analyze the properties of the matrix $AR^{-1}A^T$. For the reduced incidence matrix, we have:

$$AR^{-1}A^T \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} = \underline{v}$$

Here, $\underline{v}$ is a vector with non-negative components. The structure of the matrix $AR^{-1}A^T$ is as follows:

$$\left(AR^{-1}A^T\right)_{ii} > 0; \ \left(AR^{-1}A^T\right)_{ij} \leq 0$$

The row sums are $\sum_j \left(AR^{-1}A^T\right)_{ij} = 0$ for each node Λ_i that is not adjacent to the reference node. Moreover, if Λ_i is adjacent to the reference, this sum is a strictly positive number. We conclude that if the a-priori inverse covariance matrix $\left(P_{n-1}\right)^{-1}$ verifies the convergence conditions, then the a-posteriori inverse covariance matrix $\left(P_n\right)^{-1}$ will verify them too.

■

This lemma immediately implies the convergence of the recursive extension (for several measurement sets) of equation (5) to the optimal solution, where at each step, the new covariance matrix is computed according to (9). Since the iterations in (8) are equivalent to the procedure in (5), we obtain the claimed convergence in Theorem 2.

■

REFERENCES

[1] P. Barooah, and J. P. Hespanha, "Distributed estimation from relative measurements in sensor networks," *Proc. 3rd Int. Conf. on Intelligent Sensing and Information Processing*, 2005.

[2] M. Cohen, "*Network Time Synchronization Using Decentralized Kalman Filtering*", M. Sc. Research Thesis, Technion, Aug. 2009, available at: http://webee.technion.ac.il/people/maxcohen/Thesis.pdf.

[3] A. Giridhar and P. R. Kumar, "Distributed clock synchronization over wireless networks: algorithms and analysis", *Proc. 45th IEEE Conference on Decision and Control*, San Diego, Dec. 2006.

[4] O. Gurewitz, I. Cidon and M. Sidi, "Network time synchronization using clock offset optimization", *IEEE International Conference on Network Protocols*, pp. 212-221, Atlanta, GA, November 2003.

[5] H. R. Hashemipour, S. Roy and A. J. Laub, "Decentralized structures for parallel Kalman filtering", *IEEE Trans. Automatic Control*, vol. 33, no. 1, pp. 88-94, Jan. 1988.

[6] R. A. Horn and C. R. Johnson, "*Matrix analysis*", Cambridge University Press, Campridge, UK, 1991 (pp. 344-347).

[7] A. H. Jazwinski, "*Stochastic processes and filtering theory",* Academic Press, NY, 1970. Reprinted by Dover, 2007.

[8] C. T. Kelley, "*Iterative methods for linear and nonlinear equations", SIAM*, Philadelphia, PA, 1995.

[9] U. A. Khan and J. M. F. Moura, "Distributing the Kalman Filter for large-scale systems", *IEEE Trans. on Signal Processing*, Vol. 56, No. 10, October 2008, pp. 4919-4935.

[10] D. L. Mills, "Internet time synchronization: the network time protocol", *IEEE Trans. Commun*., vol. COM-39, no. 10, pp. 1482-1493, Oct. 1991.

[11] D. L. Mills, Network Time Protocol (version 3) specification, implementation and analysis, *Network Working Group Report*. Univ. of Delaware, RFC-1305, pp. 113, 1992.

[12] R. Olfati-Saberi, "Distributed Kalman filtering for sensor networks", Proc. *46th IEEE Conference on Decision and Control*, New Orleans, Dec. 2007.

[13] K. Plarre and P. R. Kumar, "Object tracking by scattered directional sensors", *Proc. 44th IEEE Conference on Decision and Control*, pp. 3123-3128, Seville, Spain, Dec. 2005.

[14] B. S. Rao and H. F. Durant-Whyte, "Fully decentralized algorithm for multisensor Kalman filtering", *IEEE Proceedings-D,* Vol. 138, No. 5, September 1991.

[15] R. Solis, V. Borkar, and P. R. Kumar, "A new distributed time synchronization protocol for multihop wireless networks", *Proc. 45th IEEE Conference on Decision and Control*, pp. 2734-2739, San Diego, Dec. 2006.

[16] J. Stoer and R. Burlirsch, "*Introduction to numerical analysis*", *3rd Ed. Springer Verlag*, New York, 2002.

[17] F. Sivrikaya and B. Yener, "Time synchonization in sensor networks: a survey", *IEEE Network,* July 2004, pp. 45-50.