

Network Time Synchronization Using a Decentralized Kalman Filtering Framework

M. Cohen and N. Shimkin, IEEE Senior Member

Abstract—Accurate clock synchronization is important in many distributed applications. This problem was both extensively treated in the literature and applied in practice. In the Internet, for example, the “Network Time Protocol” (NTP) is the most widely accepted standard for clock synchronization. In some recent work, improved algorithms that rely on Least-Squares (LS) estimation were introduced. A central characteristic of these methods is their decentralized structure that requires only local communication among neighbors. In this paper, we extend this LS framework by developing algorithms that estimate the offset of the local clock at each network node, using a Kalman Filter (KF) induced framework that takes account of initial data and multiple measurement sets. We propose a synchronous decentralized implementation of the estimation algorithm that employs only local broadcasts and prove that it converges to the optimal centralized solution. We also present some simulation results to illustrate the performance benefits of the suggested algorithms.

I. INTRODUCTION

ACCURATE clock synchronization is required in many distributed applications in computer networks (e.g., sleep scheduling in the case of low duty cycle [11], and tracking in wireless sensor networks [15]). 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. We will assume for simplicity that the links are bi-directional, the network topology is time-invariant and that each node is capable of sending and receiving messages from its neighbors. 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 [12], [13]. This protocol essentially uses a hierarchical approach by sending probe messages along a spanning tree of the network.

More recently, a novel approach for time synchronization termed CTP – Classless Time Protocol [5] 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 alternative proposed approach relies on the well known

Least-Squares Estimator in [4], [16]. The accuracy of clock synchronization was improved by exploiting global network-wide constraints (e.g., the relative offsets are summing up to zero over loops) and the use of a completely asynchronous, distributed algorithm employing only local broadcasts. The central characteristic of these methods relies in their decentralized structure that requires only local communication with neighbors. In addition, the work in [1] extends the same LS approach to a Weighted Least-Squares (WLS) framework, where each measurement is pre-multiplied by a weighting factor. It is interesting to note that the time synchronization problem is mathematically equivalent to any related distributed estimation problem stemming from relative additive measurements in sensor networks [1]. For example, one can apply the same algorithms to the sensor localization problem.

In estimation theory, for a linear dynamic system under the Gaussian assumption the Kalman Filter is the optimal MMSE (Minimum Mean Squared-Error) state estimator. If the Gaussian assumption is relaxed, we will obtain the linear optimal MMSE state estimator. The implementation of the KF in a decentralized manner was extensively treated in the literature [2], [6], [10], [15]. Our objective is to develop efficient decentralized estimation algorithms in order to synchronize the different clocks over the network with respect to the reference time. Without loss of generality, we may assume that Node 1 is synchronized with the universal clock, and we thus have to synchronize the other clocks with respect to it. Our objective is therefore to estimate the clock offsets at each network node relative to the clock reference.

The basic Least-Squares framework of [4] considered the estimation problem using only a single set of measurements. We extend this framework to a recursive one by first taking account of a-priori knowledge (namely, initial estimates of the local time at each node, together with their confidence level), and then considering recursive algorithms for multiple measurement sets. We also rely on the well known result that claims the equivalence between the KF solution and the minimizing vector of a deterministic constrained LS problem. In this way, we will be able to obtain the existing LS solution as a special case. The problem is conveniently formulated within a Kalman filtering framework. However, as will become clear, under our assumptions on the state vector, this is equivalent to a recursive Least-Squares problem. This equivalence will be used to develop the decentralized algorithms that are the main goal of this paper.

The first step is to formulate the model in the state space form where the state is the vector of biases of the clocks in the network. Then, we will show that a single measurement vector update can be done using a distributed iterative

Submitted to ACC, September 2009.

M. Cohen and N. Shimkin are with the Department of Electrical Engineering, Technion, Haifa 32000, Israel (e-mail addresses: maxcohen@tx.technion.ac.il; shimkin@ee.technion.ac.il).

scheme that converges to the optimal centralized estimator. In addition, the decentralized implementation converges in a relatively small number of iterations, like in [5]. We make the natural assumption that the initial state covariance matrix is diagonal, however we observe that after the first measurement update of the KF, the state covariance matrix does not remain diagonal. Hence, from this step the standard KF equations cannot be decentralized and each node has to communicate with every other node in the network. This is not a desirable situation since it is prohibitively expensive in terms of communication time. We solve this issue by proposing a decentralized recursive algorithm that relies on manipulating the standard equations. The next step is to consider the multiple measurement case and to present a recursive version. The recursive algorithm computes the optimal offsets and the corresponding variances in a decentralized manner after receiving each set of measurements. We also consider a simple sub-optimal algorithm that neglects the off-diagonal terms of the inverse covariance matrix. This method reduces significantly the complexity, but loses its optimal property.

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.

The remainder of this paper is organized as follows. In section II, we describe the model and formulate the problem. Then, in section III, we present the different algorithms for the single measurement update case. Section IV is devoted to show the convergence of the most general decentralized algorithm to the optimal centralized solution. In section V, we provide the recursive version of our algorithm for multiple measurements. Sections VI treat several extensions of the basic algorithm. Numerical results are presented in section VII. Finally, the conclusions and some notes on future directions are reported in section VIII.

II. MODEL AND PROBLEM DEFINITIONS

We model the network as a directed graph $G=(V,E)$ with $|V|=N$ nodes $\{\Lambda_1, \Lambda_2, \dots, \Lambda_N\}$ and $|E|=m$ 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, namely that if $e_{ij} \in E$, then $e_{ji} \in E$ and that the graph that corresponds to the network 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 consider a model in which only one out of the N nodes is a "reference time node" (the generalization for several reference time nodes is straightforward). Without loss of generality, we may assume that the reference time

node is Λ_1 . Our objective is to construct the optimal offset estimate for every node $u \in V \setminus \{1\}$.

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 [16] and the references therein).

The time synchronization problem relates to the task of setting the clocks in the network so that they all agree. We suppose that node Λ_1 is synchronized to the reference time:

$$\tau_1 = 0 \text{ and } \alpha_1 = 1$$

We will focus on the simplified model where the offsets are time-invariant and that all the clocks run at the same speed (there is no skew), namely: $\alpha_i = \alpha_j = 1, \forall i, j$.

These assumptions are appropriate when the effective measurement span (or the memory of the estimation procedure) is short relative to the rate deviation. In the conclusion, we will relate to the case in which the previous assumptions are relaxed.

B. The Measurements

Each network node ($\Lambda_i, i=1, 2, \dots, N$) sends probe packets to each one of its neighbors. Time is stamped on packet k_m by the sender Λ_i upon transmission ($T_i(k_m)$) and by the receiver Λ_j upon reception of the packet ($R_j(k_m)$). Then, the node Λ_j retransmits the packet back to the source ($T_j(k_m)$) and the source stamps its local time when receiving the packet back ($R_i(k_m)$). Then:

$$\Delta T_{ij}(k_m) \triangleq R_j(k_m) - T_i(k_m) = x_{ij}(k_m) - \tau_i + \tau_j + \tilde{\varepsilon}_{ij}$$

Here, $x_{ij}(k_m)$ is the propagation delay of the link e_{ij} , $\tilde{\varepsilon}_{ij}$ 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 $x_{ij}(k_m) = x_{ji}(k_m)$ (symmetric propagation delay) we obtain:

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

where: $\varepsilon_{ij} = \frac{1}{2}(\tilde{\varepsilon}_{ij} - \tilde{\varepsilon}_{ji})$

C. Problem Formulation and State Space Model

Our objective is to synchronize all the clocks in the network with the reference time. This is equivalent to estimating τ_i at each network node. The algorithm is required to be decentralized and to converge to the optimal centralized solution.

Let us define the state vector by the following column vector: $x(n) \triangleq (\tau_1 = 0, \tau_2, \dots, \tau_N)^T$, where, τ_i is the offset of

node Λ_i . As we previously explained, the measurements for each pair of neighboring nodes is given by:

$$y_{ij} = \hat{O}_{ij} \triangleq \frac{1}{2}(\Delta T_{ij} - \Delta T_{ji}) = \tau_j - \tau_i + \varepsilon_{ij}$$

Thus, the relative measurement y_{ij} for each pair of neighboring nodes is given by their offset difference plus an additive noise ε_{ij} . We note that \hat{O}_{ij} is the standard notation for this measurements, and we therefore retain this notation. The measurement equation of the state space model is related to the incidence matrix A whose dimensions are N (nodes number) \times m (edges number) and where in the row corresponding to node Λ_i , we have an entry +1 for all edges of the form (i, j) , an entry -1 for all edges of the form (j, i) , and 0 otherwise. For a connected graph, the rank of the incidence matrix is $N-1$, or one less than the number of nodes. 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 obtained by deleting the row corresponding to the reference node Λ_1 . For notational convenience, we use A to henceforth denote the reduced incidence matrix. Consequently, the state space model is given by:

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

Here, $n \geq 0$ is the discrete time index and $y(n)$ is the measurement set of every pair of neighboring nodes at time n . We note that n need not refer to the actual time, but rather corresponds to the epoch when the n -th measurement set $y(n)$ become available. The initial state of the system $x(0)$ has the following first and second order statistics: $E[x(0)] = \bar{x}_0$, $\text{cov}[x(0)] = P_0$. $\{v(n)\}$ is the measurement noise modeled as a white noise with zero mean and covariance $R(n) = R > 0$. We assume that $\{v(n)\}$ is uncorrelated and therefore the matrix R is diagonal and Positive Semi-Definite (PSD). Its i - j element corresponds to the pair of neighboring nodes Λ_i and Λ_j :

$$(R)_{ij} = r_{ji}$$

$\{v(n)\}, x(0)$ are uncorrelated, that is: $E[x(0)v^T(n)] = 0 \quad \forall n$.

III. SINGLE MEASUREMENT SET

First, we consider the single measurement update case. We start with the pair of parameters \bar{x}_0, P_0 and our goal is to find \hat{x}_{opt} by using the Kalman Filter equations. \bar{x}_0 and P_0 represent the a-priori knowledge and we want to include this information together with the measurements to find an optimal estimate. This is important, because this initial knowledge can improve the quality of the estimation.

The KF solution is the optimal linear estimate of $x = x(1)$, given $y = y(1)$ in the MMSE sense. As is well known (e.g.,

[8]), this solution is equivalent to a Least-Squares deterministic problem, which in our case reduces to the minimum of the following objective function:

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

The first term of the objective function is related to the initial knowledge of 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 more convenient to manipulate the above deterministic LS problem rather than starting with the KF equations.

A. The Basic Algorithm

We first present the existing algorithm introduced in [4], [5], [15]. In this case, the objective function is given by:

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

Differentiating J with respect to each one of the coordinates τ_i and set the derivatives to zero leads to:

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

From this, we get:

$$\tau_i = \frac{1}{|N_i|} \cdot \sum_{j \in N_i} (\hat{O}_{ji} + \tau_j) \quad (2)$$

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, we will implement a decentralized iterative algorithm and show its convergence to the optimal centralized solution. This algorithm follows the classical Jacobi iteration model. We will define the iterative procedure for the general later in this section and we will prove its convergence. The above equation has a very simple interpretation. Each node computes its offset estimate as the average of all its neighbors' estimates plus the corresponding relative measurements. This procedure is the same as in [4], [16] and one can easily show that this is equivalent to the algorithm in [5]. 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. General Framework

Let us solve the original problem where the objective function J is composed of two distinct terms, like in (1). In general, the initial covariance matrix P_0 need not be a diagonal matrix. Moreover, even if P_0 is chosen to be diagonal, after the first iteration of the Kalman Filter, the inverse covariance matrix will not preserve its diagonal structure. Hence, if we have a-priori knowledge of the system or if multiple sets of measurements are available, we

must consider the case in which the covariance matrix is not assumed to be diagonal. In this more general case, we get:

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

This implies:

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

The above optimality equations can be applied to a network in order to estimate the clock offsets at each node with respect to the reference time. There are many iterative methods that can be used [9], [17]. The suggested decentralized algorithm uses the synchronous iterative updates:

$$\hat{\tau}_i^{(k+1)} = \frac{1}{\left(\sum_{j \in N_i} \frac{1}{r_{ji}} + (P_0^{-1})_{ii} \right)} \left[\sum_{j \in N_i} \frac{1}{r_{ji}} (\hat{\tau}_{ji} + \hat{\tau}_j^{(k)}) + (P_0^{-1})_{ii} \tau_i(0) - \sum_{\substack{m=1 \\ m \neq i}}^N (P_0^{-1})_{im} (\hat{\tau}_m^{(k)} - \tau_m(0)) \right] \quad (3)$$

with initialization $\hat{\tau}_i^{(0)} = \tau_i(0)$ $i = 2, 3, \dots, N$. Here, $k \geq 0$ is the iteration number. The main problem in the last formula is that each node needs to communicate with all the other nodes and not only with its neighbors. Thus, unless the matrix P_0 is diagonal, each node has to know the global topology of the entire network. As we previously explained, the initial covariance matrix P_0 can be assumed to be diagonal. However, after applying the Kalman Filter equations, the covariance matrix P_n will not be diagonal anymore.

C. Diagonal P_0

One can make the logical assumption that the initial inverse covariance matrix P_0^{-1} is a diagonal matrix, with elements: $(P_0^{-1})_{ii} = \frac{1}{p_i}$. Indeed, P_0^{-1} represents the initial

correlation between the different clocks in the network, and there is no reason to have some a-priori knowledge of the cross correlation terms but only on the variances of each clock (diagonal terms).

Now, The decentralized iterative procedure is given by:

$$\hat{\tau}_i^{(k+1)} = \frac{1}{\left(\sum_{j \in N_i} \frac{1}{r_{ji}} + \frac{1}{p_i} \right)} \left[\sum_{j \in N_i} \frac{1}{r_{ji}} (\hat{\tau}_{ji} + \hat{\tau}_j^{(k)}) + \frac{\tau_i(0)}{p_i} \right] \quad (4)$$

In words, τ_i is given by the weighted average between the adjacent measurements and its a-priori estimate. Observe that if the matrix P_0^{-1} is equal to zero and $R^{-1} = I$, we obtain the equation (2) as in the basic LS case described above. This algorithm requires only local broadcasts and as we will show in section IV, converges to the optimal centralized solution (after an infinite number of iterations in all the nodes).

IV. CONVERGENCE ANALYSIS

We now establish the convergence of the previous decentralized clock synchronization algorithms to the optimal centralized solution. We consider the most general case where the initial covariance matrix P_0 need not be diagonal. In this case, the synchronous decentralized algorithm is given by (3).

Theorem 1.

Suppose that:

- The matrix R is diagonal and Positive Semi-Definite, that is: $0 \leq (r_{ji})^{-1} < \infty \quad \forall i, j$.
- The initial covariance matrix P_0 is an M-matrix, namely:

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

- The clock adjustment operation in (3) 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 $k \rightarrow \infty$) to the optimal offsets that minimize the objective function in (1).

The proof is provided in the Appendix.

V. RECURSIVE ALGORITHMS FOR MULTIPLE MEASUREMENT SETS

So far, the case of one measurement update was considered. The next step is to consider the multiple measurement case and to present a recursive version of the previous decentralized algorithms. Several sets of measurements become successively available and our goal is to develop an algorithm that recursively estimates the offsets after each set of measurements. In the subsequent analysis, we still focus on the case where the initial covariance matrix P_0 is diagonal. We can then try to apply the algorithm in (3) recursively after each set of measurements is received. The problem here is that after the first measurement update in the KF equations, the covariance matrix P_1 will not be diagonal anymore. Then, each node has to communicate with all the other nodes over the network and not only with the one-hop neighbors. We therefore cannot apply the equations that were previously developed in (3), and our objective is to derive an alternative form of the iteration (3), which makes it suitable for decentralized implementation.

The multiple measurement update case can be solved using several approaches. First, we present an optimal decentralized recursive algorithm based on the KF equations. Then, we propose a sub-optimal decentralized algorithm that neglects the off-diagonal terms of the inverse covariance matrix. An alternative method is to wait for all the measurements and then to perform the estimation procedure (batch non-recursive estimation). This last method is not presented here and can be found in [3].

A. Optimal Decentralized algorithm

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

$$J(n) = (x - \bar{x}_0)^T P_0^{-1} (x - \bar{x}_0) + \sum_{k=1}^n (y(k) - A^T x)^T R^{-1} (y(k) - A^T x) \quad (5)$$

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

$$\begin{aligned} \hat{\tau}_i^{(k+1)}(n) = & \hat{\tau}_i(n-1) + [I_i(n)] \cdot \left\{ \sum_{j \in N_i} \frac{1}{r_{ji}} \cdot [\hat{O}_{ji}^{(n)} - (\hat{\tau}_i(n-1) - \hat{\tau}_j^{(k)}(n))] \right\} + \\ & + (n-1) \cdot \left[\sum_{j \in N_i} \frac{1}{r_{ji}} \cdot (\hat{\tau}_j^{(k)}(n) - \hat{\tau}_j(n-1)) \right] \end{aligned} \quad (6)$$

$$[I_i(n)]^{-1} = [I_i(n-1)]^{-1} + \sum_{j \in N_i} \frac{1}{r_{ji}} = (P_0^{-1})_{ii} + n \cdot \sum_{j \in N_i} \frac{1}{r_{ji}} \quad i = 2, \dots, N$$

$$[I_i(0)]^{-1} = [P_i(0)]^{-1} = (P_0^{-1})_{ii}$$

The above set of equations is a decentralized, synchronous and recursive algorithm that computes at each step, the estimated offsets and the 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 (6). At time n , we assume that the estimate of $\hat{\tau}_i(n-1)$ is given.

Then, $\hat{\tau}_i^{(k)}(n)$ $k = 1, 2, \dots$ 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.

It may be shown that the elements of $[I_i(n)]^{-1}$ are the diagonal entries of the inverse covariance matrix in the Kalman Filter equations. Thus, the variances of the estimation errors at each step are available. This is a desirable property since it gives information on the estimation quality. Observe however that we do not compute the non-diagonal elements of the inverse covariance matrix. We point out that the suggested recursion slightly deviates from the standard structure of a recursive algorithm due to the term n (time explicit index) in the denominator and in the internal term.

The proposed algorithm may be derived in two ways, which lead to the same optimal equations:

1. Differentiate $J(n-1)$ 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 [3]) are omitted since the procedure is similar to the previous case.
2. Algebraic manipulations of the standard recursive extension of (3), with the following KF update inverse covariance equation:

$$(P_{k+1})^{-1} = (P_k)^{-1} + AR^{-1}A^T \quad (7)$$

In other words, the set of iterative equations in (6) is mathematically equivalent to perform (3) separately for each measurement set in addition to (7). This result will be useful in the convergence analysis of Theorem 2. This can be shown easily by some appropriate mathematical manipulations and is not presented here due to space

limitations. We note that an alternative derivation of these equations that starts with the KF equations in information form is given in [3]. However, the obtained recursion in (6) is not equivalent to the KF equations.

Now, let us show the convergence of the set of equations in (6) to the optimal centralized solution.

Theorem 2.

Suppose that:

- a) Assumptions a) and b) from Theorem 1 hold.
- b) The clock adjustment operation in (6) is applied synchronously by all nodes ($i = 2, 3, \dots, N$) in all iterations, recursively for n sets of measurements.
- c) 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 $\hat{\tau}_i^{(k)}(n)$ $i = 2, 3, \dots, N$ converge (as $k \rightarrow \infty$) to the optimal offsets that minimize the objective function in (5).

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 (3) that the estimated offset of node Λ_i depends on all the other offsets and not only on those of its neighbors. One can think about the naïve sub-optimal algorithm that neglects the off-diagonal terms of the inverse covariance matrix:

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

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

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

Here, we neglect the off-diagonal terms before inverting the information matrix $(P_{n-1})^{-1}$, in the goal to improve the complexity. In this case, we will invert a diagonal matrix and hence the time computation will significantly decrease. We can interpret the previous equation in a very logical manner. The new estimate is given by the sum of the previous estimate and a correction term. This correction term is composed of the latest measurement minus the estimated measurement multiplied by the measurement variance and the total is normalized by the accumulative variance. In the numerical results section, we will compare the decentralized recursive algorithm that converges to the optimal solution to the above sub-optimal scheme.

VI. 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 [3]. The convergence

analysis of the decentralized algorithm is not presented because the convergence is achieved after a relatively small number of iterations and the results are very similar than [5]. 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 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 (2)) and the Decentralized Kalman Filter (DKF) (equation (4)) 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 greater 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.

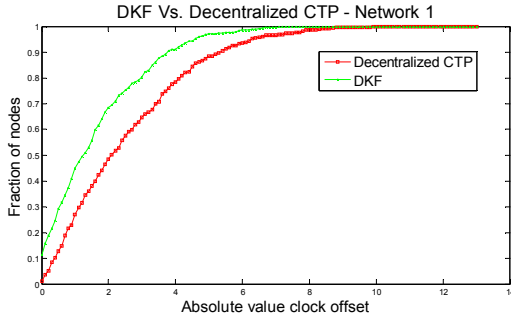


Fig. 2. Comparison between the decentralized CTP and DKF algorithms (with 10% of nodes synchronized via GPS) in Network 1.

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 :

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

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).

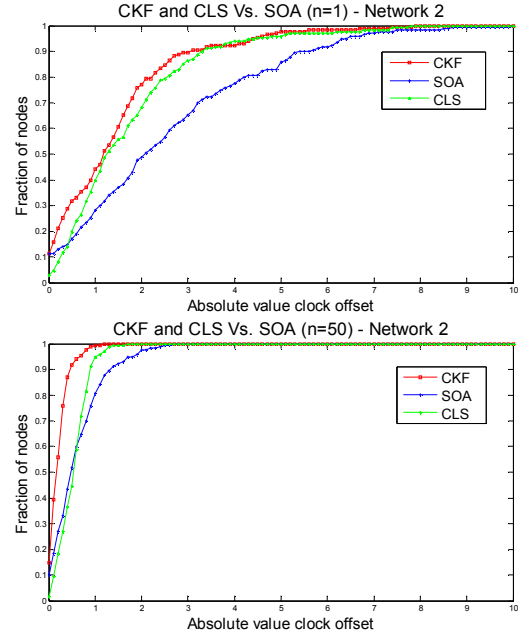


Fig. 3. Comparison between CKF, SOA and CLS (with $R \sim U[0.01, 12]$ and $P_0 \neq I$) in Network 2 for $n = 1, 50$.

VII. 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 Kalman Filter framework. The essential characteristic of these algorithms is their decentralized nature; each node can estimate its clock offset by only exchanging packets among its one-hop neighbors. We extend the existing Least-Squares based algorithms using a Kalman Filter framework so we can assign different weights to the measurements according to their accuracy, include a-priori knowledge 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 (5) 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 [3].

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 [3] and may be considered as directions for future research.

APPENDIX

First, we present the proof of Theorem 1. Let us recall that the general objective function is given by:

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

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 (3) 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:

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

Here:

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

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

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

Let us define: $\bar{\tau}^{(k)} \triangleq \hat{\tau}^{(k)} - \tau^*$

Then we obtain after some manipulations:

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

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

Thus, the convergence of the sequence $\hat{\tau}^{(k)}$ to τ^* is equivalent to the convergence of $\bar{\tau}^{(k)}$ 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., [7], 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, $\bar{\tau}^{(k)} \rightarrow 0$ 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

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

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

$$\begin{aligned} \sum_j M_{ij} &= \frac{\sum_{j \in N_i} [-(P_0^{-1})_{ij} + (r_{ji})^{-1}] - \sum_{\substack{j \in N_i \\ j \neq i}} (P_0^{-1})_{ij}}{\sum_{j \in N_i} (r_{ji})^{-1} + (P_0^{-1})_{ii}} \\ &= \frac{\sum_{j \in N_i} (r_{ji})^{-1} - \sum_{\forall j \neq i} (P_0^{-1})_{ij}}{\sum_{j \in N_i} (r_{ji})^{-1} + (P_0^{-1})_{ii}} \end{aligned}$$

It follows that $\sum_j M_{ij} \leq 1$ if and only if: $\sum_j (P_0^{-1})_{ij} \geq 0$.

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:

$$\begin{cases} (P_0^{-1})_{ii} \geq 0 \\ (P_0^{-1})_{ij} \leq 0 \quad (i \neq j) \end{cases}$$

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

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:

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

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

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

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

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:

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

We next present the proof of Theorem 2. The case in which $n=1$ 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 (7), then P_n is an M -matrix for all $n \geq 1$.

Proof

Equation (7) 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:

$$\begin{cases} (AR^{-1}A^T)_{ii} > 0 \\ (AR^{-1}A^T)_{ij} \leq 0 \end{cases}$$

The row sums are $\sum_{j \in N_i} (AR^{-1}A^T)_{ij} = 0$ for each node Λ_i that is not adjacent to the reference node. Moreover, if the node Λ_i is adjacent to the reference node, this sum is a strictly positive number. Hence, we conclude that if the a-priori

inverse covariance matrix $(P_{n-1})^{-1}$ verifies the convergence conditions, then the a-posteriori inverse covariance matrix $(P_n)^{-1}$ will verify them too. ■

This lemma immediately implies the convergence of the recursive extension (for several measurement sets) of equation (3) to the optimal solution, where at each step, the new covariance matrix is computed according to (7). Since the iterations in (6) are equivalent to the procedure in (3), we obtain the claimed convergence in Theorem 2. ■

REFERENCES

- [1] P. Barooah, and J. P. Hespanha, "Distributed estimation from relative measurements in sensor networks," *Proceedings of the 3rd Int. Conf. on Intelligent Sensing and Information Processing*, 2005.
- [2] W. H. Chung and J. L. Speyer, "A general framework for decentralized estimation", *Proceedings of the American Control Conference*, Seattle, Washington, June 1995.
- [3] 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>.
- [4] A. Giridhar and P. R. Kumar, "Distributed clock synchronization over wireless networks: algorithms and analysis", *Proceeding of the 45th IEEE Conference on Decision and Control*, San Diego, Dec. 2006.
- [5] 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.
- [6] H. R. Hashemipour, S. Roy and A. J. Laub, "Decentralized structures for parallel Kalman filtering", *IEEE Trans. on Automatic Control*, vol. 33, no. 1, pp. 88-94, Jan. 1988.
- [7] R. A. Horn and C. R. Johnson, "Matrix analysis", Cambridge, UK: Cambridge University Press, 1991 (pp. 344-347).
- [8] A. H. Jazwinski, "Stochastic processes and filtering theory", Dover ed., Mineola, N. Y., 2007.
- [9] C. T. Kelley, "Iterative methods for linear and nonlinear equations", SIAM, Philadelphia, PA, 1995.
- [10] 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.
- [11] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk and J. Anderson, "Wireless sensor networks for habitat monitoring", *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications*, ACM press, pp. 88-97, 2002.
- [12] D. L. Mills, "Internet time synchronization: the network time protocol", *IEEE Trans. Commun.*, vol. COM-39, no. 10, pp. 1482-1493, Oct. 1991.
- [13] D. L. Mills, Network Time Protocol (version 3) specification, implementation and analysis, *Network Working Group Report*. Univ. of Delaware, RFC-1305, pp. 113, 1992.
- [14] K. Plarre and P. R. Kumar, "Object tracking by scattered directional sensors", *Proceedings of the 44th IEEE Conference on Decision and Control*, pp. 3123-3128, Seville, Spain, Dec. 2005.
- [15] B. S. Rao and H. F. Durant-Whyte, "Fully decentralized algorithm for multisensor Kalman filtering", *IEEE Proceedings-D*, Vol. 138, No. 5, September 1991.
- [16] R. Solis, V. Borkar, and P. R. Kumar, "A new distributed time synchronization protocol for multihop wireless networks", *Proceeding of the 45th IEEE Conference on Decision and Control*, pp. 2734-2739, San Diego, Dec. 13-15 2006.
- [17] J. Stoer and R. Burlirsch, "Introduction to numerical analysis", 3rd Ed. Springer Verlag, New York, 2002.