

TIMING ERROR NOISE REDUCTION IN DISTRIBUTED WIRELESS NETWORK USING
KALMAN FILTER

A Thesis by

Hanumantharaju T Veerabasavaiah

Bachelor of Engineering, M.S.Ramaiah Institute of Technology and Science, Bangalore

VTU, Belgaum, 2010

Submitted to the Department of Electrical Engineering and Computer Science
and the faculty of the Graduate School of
Wichita State University
in partial fulfillment of
the requirements for the degree of
Master of Science

July 2013

© Copyright 2013 by Hanumantharaju T Veerabasavaiah
All Rights Reserved

Note that thesis work is protected by copyright, with all rights reserved. Only the author and advisor Dr. Hyuck M. Kwon have the legal right to publish, produce, sell, or distribute this work. Author or advisor permission is needed for others to directly quote significant amounts of information in their own work or to summarize substantial amounts of information in their own work. Limited amounts of information cited, paraphrased, or summarized from the work may be used with proper citation of where to find the original work.

TIMING ERROR NOISE REDUCTION IN DISTRIBUTED WIRELESS NETWORK USING
KALMAN FILTER

The following faculty members have examined the final copy of this thesis for form and content, and recommend that it be accepted in partial fulfillment of the requirement for the degree of Master of Science with a major in Electrical Engineering.

Hyuck M. Kwon, Committee Chair

Animesh Chakravarthy, Committee Member

Xiaomi Hu, Committee Member

DEDICATION

To my brother Bruno, family members, and all my teachers

ACKNOWLEDGEMENTS

Foremost, I would like to extend my heartfelt thanks to my advisor, Dr. Hyuck M. Kwon, for his valuable assistance, support, motivation, and patience. His guidance helped me throughout my research and writing this thesis. Also, I would like to thank Dr. Animesh Chakravarthy and Dr. Xiaomi Hu for being part of the adjudicating committee. I thank my colleagues in the Wireless Research and Development Group (WiRed group): Prashanth, Youvaraj, Kanghee, Wenhao, Kenny, Jessie, Sangku, Mahesh, Kim, Shuang, and others for their support and encouragement throughout my research.

ABSTRACT

The distributed network has been considered the future's wireless networking. A key challenge to implementing a distributed network is time synchronization of nodes in a network in a noisy environment. A few parallel synchronization schemes have been proposed. But when the data rate is high, the performances of these algorithms are degraded because of timing error noise. Thus thesis deals with the problem of timing error noise using the Kalman filter, which is highly effective for additive white Gaussian noise (AWGN) but also restrains other kinds of noise, although not as effectively as for AWGN. By means of the Kalman filter, the effect of timing error noise can be reduced. Simulation results provide the effectiveness of the algorithm used.

TABLE OF CONTENTS

Chapter	Page
1. INTRODUCTION	1
1.1 Motivation	1
1.2 Background Literature Survey	3
1.3 Objective	4
1.4 Main Idea behind Thesis	4
1.5 Main Contribution of Thesis	4
1.5 Outline of Thesis	4
2. SIGNAL MODEL.....	6
3. EXISTING TIME SYNCHRONIZATION ALGORITHM	8
4. PRATICAL IMPLEMENTATION OF DISTRIBUTED COUPLED DISCRETE TIME OSCILLATOR	10
5. TIMING ERROR NOISE	13
6. KALMAN FILTER AND ITS EFFECT ON NOISE.....	15
6.1 Kalman Filter Algorithm	15
6.2 Implementation of Kalman Filter to Reduce Effects of Noise on Time Synchronization Algorithm.....	17
7. SIMULATION RESULT.....	20
8. CONCLUSION.....	23
LIST OF REFERENCES	24

LIST OF FIGURES

Figure	Page
1. Transmitted signal by nodes in the network	6
2. Block diagram of practical implementation of distributed synchronization scheme [31]	11
3. Difference between m^* and m , cause of measurement noise.....	13
4. Block diagram showing addition of noise	14
5. Block diagram of working Kalman filter used in proposed algorithm	17
6. Rectangular topology with four nodes.....	20
7. Standard deviation of timing vector $\xi(n)$ for synchronization algorithm without averaging with respect to channel noise and measurement noise. No Kalman filter was employed.....	21
8. Standard deviation of timing vector $\xi(n)$ for synchronization algorithm without averaging with respect to channel noise and measurement noise. Kalman filter was employed.....	21
9. Standard deviation of timing vector $\xi(n)$ for synchronization algorithm with averaging with respect to channel noise and measurement noise. No Kalman filter was employed.....	22
10. Standard deviation of timing vector $\xi(n)$ for synchronization algorithm with averaging with respect to channel noise and measurement noise. Kalman filter was employed.....	22

LIST OF ABBREVIATIONS

AWGN	Additive White Gaussian Noise
BS	Base Station
Gbps	Giga Bits per Second
GPS	Global Positioning System
MIMO	Multiple-Input Multiple-Output
MISO	Multiple-Input Single-Output
MS	Mobile Station
SNR	Signal-to-Noise Ratio

CHAPTER 1

INTRODUCTION

1.1 Motivation

Distributed wireless networks are some of the most promising wireless networks for the future, especially compared to conventional networks. The reason that distributed wireless networks are so promising is that they meet most of the necessities of a user because it has a high mobility for the users. Furthermore, there is no need for maintaining a base station, which will reduce the cost, reduce the power consumption, and allow for a higher data rate. A data rate of 1 gigabit per second (Gbps) can be achieved using distributed wireless networks, but to achieve 1 Gbps [1] in conventional networks, the mobile device should encompass more power. However, some of the research in the field of wireless communication has proven that distributed networks are capable of achieving a data rate of 1 Gbps with modest power [2]. One more advantage of distributed wireless networks is that they are easy to deploy in the case of natural disasters and war emergencies.

In current centralized networks, both the user's device and the base station are synchronized in order to allow networks to accurately function, but distributed wireless networks are decentralized, which means there are no base stations to monitor each user. Instead, every user is independent. Research concerning distributed networks assumes each and every user is absolutely synchronized with each other. The performances of the distributed wireless networks are degraded drastically if the users are not synchronized properly. Even though time synchronization in conventional networks is accomplished by exchanging timing stamps via packets, requirements such as power consumption, energy efficiency, high data rate, complex computation, and scalability are not achieved as anticipated.

Future wireless communications systems (e.g., fifth generation [5G] systems) will be able to achieve a data rate of more than 1 Gbps, which is considerably greater than that of emerging 4G systems. Potential, but theoretical, candidates for these future wireless communication systems include massive multiple-input multiple-output (MIMO) systems [3] - [11], interference alignment [12] - [18], network coding [19] - [23], lattice coding [24] - [26], two-way relay networks [27] - [28], polar codes [29] - [30], and so on. However, these candidates have assumed perfect synchronization among multiple mobile distributed nodes and have neglected the synchronization issue. This is probably because the analysts and researchers are familiar with a centralized global positioning system (GPS)-based synchronization and have assumed that distributed multiple-node synchronization is feasible in a centralized synchronization way.

This thesis does not consider a master-slave type plan or a centralized network synchronization scheme, but rather a mutual synchronization among distributed nodes in the network is considered. This is because centralized networks they typically require the assistance of a GPS and feedback from receiver nodes, which are complicated; however, a node can move dynamically and can be located at any place where there is no GPS. In other words, it is desirable to achieve and maintain synchronization among distributed nodes by employing a decentralized synchronization scheme under a no-GPS environment.

In this project, a node can represent a sensor, a relay, a mobile station (MS), or a base station (BS), all of which have the capabilities of transmitter and receiver processing. Distributed synchronization means there is a simple decentralized synchronization process at an individual node, which has the ability to acquire and track a common global clock time, among multiple nodes, where each node is not necessarily collocated. A mobile wireless network means that each node can be mobile, and because of that, the network geometry can keep changing but is still

connected through dynamic wireless communications. Some of the nodes can be closely placed, while others can be located farther away in geometry. A cluster is a sub-network consisting of multiple nodes located close to each other.

The algorithm proposed by Simeone and Spagnolini [31] for decentralized time synchronizations for wireless sensors using discrete time oscillators is admirable. However, when the effect of timing error noise increases, accuracy in synchronization decreases. The main goal here is to reduce the effect of timing error noise in the time synchronization process. To reduce the effect of this noise, the Kalman filter, also called linear quadratic estimation is used. The synchronization algorithm with the Kalman filter may be a good alternative for distributed coupled discrete-time oscillators in noisy environments. This algorithm is also a good substitute for the conventional time synchronization process using timing stamps in packets.

1.2 Background Literature on Synchronization

The majority of synchronization studies in the literature have focused on point-to-point synchronization for a single-transmitter and single-receiver system. Lindsey et al. have addressed multipoint-to-multipoint synchronization for a distributed multiple node network in [32]. The proposed algorithm by Simeone and Spagnolini [31] will fail even with the small power of timing error noise. Time synchronization in distributed networks with decode-and-forward (DF) relays have been discussed [33], [34], and time synchronization errors in cooperative networks have been reported [35]-[37]. Furthermore, their synchronization acquisition time increases as the number of nodes in the network increases, which is undesirable.

This project also employed a discrete-time consensus-based approach [31] but has proposed unique decentralized network synchronization methods that can be effective when the timing error noise level is high.

1.3 Objective

The objective of this project was to propose a physical-layer distributed mobile network time synchronization scheme that reduces the effect of timing error noise on the timing synchronization process using the Kalman filter with a discrete time oscillator, and to analyze the efficiency of the proposed scheme with varying measurement noise power levels.

1.4 Main Idea behind Thesis

To overcome one of the main issues in distributed wireless networks, which is the time synchronization between nodes present in the network, the proposed algorithm in the work of Simeone and Spagnolini [31] is examined through a simulation of the algorithm in a noisy environment. As a result, it was observed that the algorithm can completely fail. The purpose of this thesis is to propose a new scheme that uses the Kalman filter to reduce the noise effect on the time synchronization process.

1.5 Main Contribution of Thesis

- Examination of the algorithm proposed in the work of Simeone and Spagnolini [31] in a noisy environment with different noise power levels.
- Proposal of an algorithm that uses the Kalman filter to suppress the noise that affects the time synchronization process.
- Comparison of the proposed algorithm using the Kalman filter and the algorithm proposed in the work of Simeone and Spagnolini [31].

1.6 Outline of Thesis

The remainder of this thesis is organized as follows: Chapter 2 describes the system model; Chapter 3 describes the synchronization algorithm given by Simeone and Spagnolini [31]; Chapter 4 describes a practical implementation of a distributed coupled discrete time

oscillator; Chapter 5 describes the cause of measurement noise and its effects on timing synchronization process; Chapter 6 provides a brief introduction to the Kalman filter, its effect on noise, implementation, and functioning of the Kalman filter on the proposed system; Chapter 7 provides simulation results, and Chapter 8 concludes the thesis.

CHAPTER 2

SYSTEM MODEL

Let K be the number of nodes in the network. Consider two nodes in the network, i and k , which are separated from each other by a distance $d_{ki}(n)$, where $d_{ki}(n) = d_{ik}(n)$. The k th node has a discrete time clock with period T_k . The timing phase of the k th clock at the n th period is $\tau_k(n)$. Figure 1 shows the signal radiated by the nodes [31].

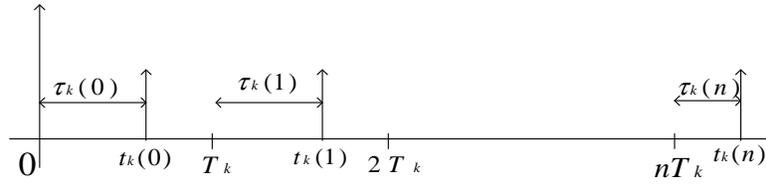


Figure 1. Transmitted signal by nodes in network.

Assume that all the nodes are isolated, and as a result the k th clock evolves as

$$t_k(n) = nT_k + \tau_k(n) \quad (1)$$

where $0 \leq \tau_k(n) < T_k$, an initial $\tau_k(0)$ is an arbitrary phase, and $n = 1, 2, \dots$ runs over the periods of the timing signal.

If all the K clocks are synchronized, then two conditions will be met

- Frequency Synchronization:

$$t_k(n+1) - t_k(n) = T \quad (2)$$

- Phase Synchronization:

$$t_1(n) = t_2(n) = \dots = t_K(n) \text{ for } n \rightarrow \infty \quad (3)$$

The clocks of all nodes are coupled in order to attain synchronization among them. All nodes in the network receive pulse $t_k(n)$ transmitted by the k th node through a flat fading channel, as shown in Figure 1. Suppose that all nodes in the network transmit with the same power. For instance, the power received by the i th node from the k th node is given by

$$P_{ki}(n) = \frac{C}{d_{ki}^\gamma(n)} \cdot F_{ki}(n) \quad (4)$$

where $d_{ki}(n)$ is the distance between node i and node k at the n th period, $F_{ki}(n)$ is the random variable accounting for the fading power process, and γ is the path loss exponent ($\gamma = 2-4$). The fading channel is reciprocal, and the power between nodes i and k will be equal, i.e., $F_{ki}(n) = F_{ik}(n)$ and $P_{ki}(n) = P_{ik}(n)$. For $i \neq k$, the same carrier frequency is used for all transmissions. C is an appropriate constant that depends on the transmitted power and is equal for all nodes.

CHAPTER 3

EXISTING TIME SYNCHRONIZATION ALGORITHM

In this section, the algorithm is the work of Simeone and Spagnolini [31]. Assume that for the k th node to be synchronized with other nodes in the network, it should update its clock solely based on the received signal by means of which the node should be able to calculate the timing difference and power of the received signal from other nodes. To update its clock, a node should calculate the weighted sum of timing difference Δt_k using equation (5). Say the k th node is receiving signals from $K - 1$ nodes, then

$$\Delta t_k(n + 1) = \sum_{i=1, i \neq k}^K \alpha_{ki}(n)(t_i(n) - t_k(n)) \quad (5)$$

The k th node should have the knowledge of the timing difference between itself and each node in the network, i.e., $t_i(n) - t_k(n), i \neq k$. Also, the k th node should have the knowledge of the powers $P_{ki}(n)$ received from the rest of the nodes in the network to calculate the weighting coefficient α_{ki} using equation (6).

$$\alpha_{ki}(n) = \frac{p_{ki}(n)}{\sum_{j=1, j \neq k}^K p_{kj}(n)} \quad (6)$$

The weighting coefficient, i.e., $\alpha_{ki}(n)$ is selected according to two conditions as shown in equations (7) and (8). The selection of the weighting coefficients in (6) is inspired by the algorithms proposed in the work of Tong and Akaiwa [38] and Sourour and Nakagawa [39].

$$\sum_{i=1, i \neq k}^K \alpha_{ki}(n) = 1 \quad (7)$$

$$\alpha_{ki}(n) \geq 0 \quad (8)$$

Since the k th node possesses the calculated value of the weighted sum of the timing difference $\Delta t_k(n + 1)$, the k th node will update its clock using (9).

$$t_k(n + 1) = t_k(n) + \varepsilon \cdot \Delta t_k(n + 1) + T_k \quad (9)$$

This design is more useful because the algorithm is robust against measurement errors over the fading channels since unreliable channels (low power) should be weighted less when updating the clock. In the work of Mohammed and Larsson (5), it is assumed that propagation delays are very small when compared to the time resolution and thus can rule out the propagation delay. A method to handle propagation delays is described by Tong and Akaiwa [38]. As a final remark made by Simeone and Spagnolini [31], it can be seen that the dynamic system shown in equation (9) updates the clock $t_k(n + 1)$ as a convex combination of the time $\{t_i(n)\}_{i=1}^K$ [40].

CHAPTER 4

PRATICAL IMPLEMENTATION OF DISTRIBUTED COUPLED DISCRETE TIME OSCILLATOR

For practical implementation of the scheme proposed, the ideal assumptions, such as the arrival times from the other nodes as discussed in Chapter 3, cannot be considered: each node is able to measure the timing difference and power of signal solely based on the received signal. Instead, a band-limited noisy channel is considered, say the signal $g(t)$ transmitted by the k th node, which is a band-limited wave form and a square root raised cosine pulse centered at times $t_k(n)$ with the symbol period $1/F_s$, and it defines the timing resolution of the system. But the algorithm is based on the instantaneous received power as done in the work of Simeone and Spagnolini [31]. For this reason, the algorithm can be functional for both time-invariant and time-variant systems.

The received signal is observed by the node on an interval of duration T_k around the current timing instant $t_k(n)$ because each node is assumed to function in a half-duplex mode, which means that the node can either transmit or receive at once but not both together. As a result of the half-duplex constraints and the finite switching times between the transmitting and receiving modes, each sensor is not able to measure the received signal in an interval of size θ around the firing instant $t_k(n)$. The observation window reads $t \in \left(t_k(n) - \frac{T_k}{2}, t_k(n) - \theta \right) \cup \left(t_k(n) + \theta, t_k(n) + \frac{T_k}{2} \right)$.

How the proposed scheme works is similar to the way that discrete time phase-locked loops function, which can be seen in Figure 2. As can be seen, the feedback loop uses the previous timing to regulate the present time of the node with respect to the weighted sum of the timing difference $\Delta t_k(n + 1)$ in the timing update equation, as shown in equation (9).

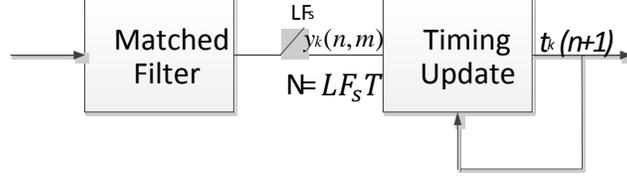


Figure 2. Block diagram of practical implementation of distributed synchronization scheme [31].

Later, at the receiver baseband, filtering is performed to match with the transmitted waveform. At some multiple L of the symbol frequency F_s , i.e., LF_s with $L \geq 1$, samples of the received signal are taken. In the n th observation window, samples are received at the receiver on the basis of $N = LF_s T$; the k th node computes the update using equation (9). In this case, a first-order loop is employed, and the second-order loop can also be considered. Since the ideal assumptions are not taken into consideration, the k th node does not encompass the knowledge of the timing difference and power, and as a result, the node is not capable of calculating the weighted sum of the timing difference $\Delta t_k(n+1)$ using equation (5). After the process of sampling and filtering at the receiver end, the k th node reads the signal received at the n th period as (sampling index m ranges within $-\frac{N}{2} < m \leq \frac{N}{2}$ with $m = 0$ corresponding to the firing time $t_k(n)$ of the k th node):

$$y_k(n, m) = \sum_{i=1, i \neq k}^k \sqrt{E_{ik}} \cdot \beta_{ki} \cdot g \left(\frac{m}{LF_s} - (t_i(n) - t_k(n)) \right) + w(n, m) \quad (10)$$

where the average energy per symbol reads

$$E_{ki} = \frac{C}{d_{ki}^{\gamma}(n)} \cdot G_{ki}(n) \quad (11)$$

$w(n, m)$ is the additive white Gaussian noise (AWGN) with zero mean and power N_0 , and β_{ki} denotes the fading coefficient. As explained previously, due to the half-duplex constraint of the node and the switching time between the receiving and transmitting modes of node k , the sample in the interval $-\Delta LF_s \leq m \leq \Delta LF_s$ is not calculated (i.e., near zero) [31]. A simple estimate of

$\Delta t_k(n+1)$ as the one in the work of Simeone and Spagnolini [31] can then be obtained as

$$\Delta t_k(n+1) = \sum_{m \in g} \bar{\alpha}_{km} \cdot \frac{m}{LF_s} \quad (12)$$

where

$$\bar{\alpha}_{km} = \frac{|y_k(n,m)|}{\sum_{i \in g} |y_i(n,i)|} \quad (13)$$

The timing update is given by the previous equation (9) as

$$t_k(n+1) = t_k(n) + \varepsilon \cdot \Delta t_k(n+1) + T_k$$

Here, we do not need the exact information of $t_i(n) - t_k(n)$ and power p_{ki} , but we can use the distributed consensus control scheme for the time synchronization. Every node shares its outputs with the other nodes through which the consensus is achieved [31].

CHAPTER 5

TIMING ERROR NOISE

Assume that the k th node receives the signal $y_k(n, m)$ and, while calculating $\Delta t_k(n + 1)$, reads m^* instead of m , as shown in Figure 3. This results in timing error noise, which is assumed to be additive white Gaussian noise $V_k \sim (0, R)$.

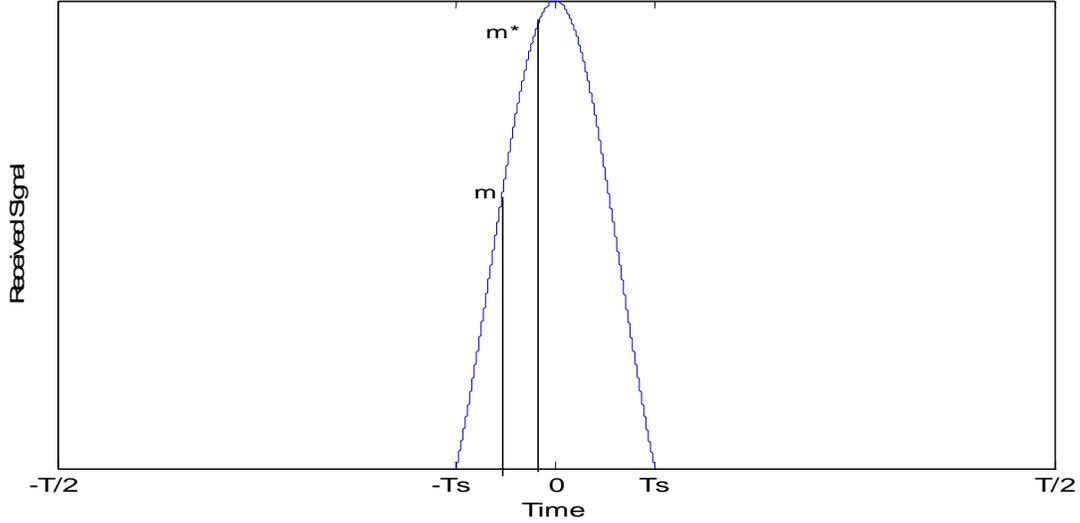


Figure 3. Difference between m^* and m , cause of measurement noise.

The interpretation of m^* instead of m may be attributable to the over sampling $N = LF_s T$ in order to achieve a high data rate. The sampling index m has normal ranges within $-N/2 < m \leq N/2$, with $m = 0$, which corresponds to the firing time $t_k(n)$ of the k th node and $N = LF_s T$. As a consequence, the timing update of equation (9), which depends on the $\Delta t_k(n + 1)$, is affected by this noise $V_k \sim (0, R)$ and results in a change of the timing update equation as

$$t_k(n + 1) = t_k(n) + (\varepsilon \cdot \Delta t_k(n + 1) + V_k) + T_k \quad (14)$$

A simple block diagram for this equation is shown in Figure 4.

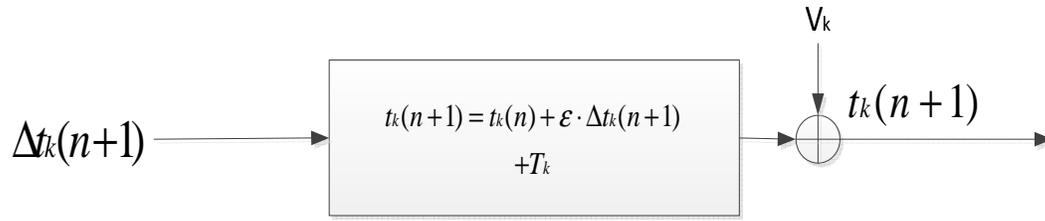


Figure 4. Block diagram showing addition of noise V_k .

CHAPTER 6

KALMAN FILTER AND ITS EFFECT ON NOISE

6.1 Kalman Filter Algorithm

The Kalman filter is an algorithm that estimates the system's state using the measurements with noise and other inaccuracies to estimate the state of the system more precisely. To make an estimate using the Kalman filter, a system model should be processed according to the Kalman filter framework and should have knowledge about the state-transition model "A," the observation model "H," the power of the process noise Q_k which is "Z," and the power of the measurement noise V_k which is "R." The present state of the system is evolved from the previous state of the system. Table 1 shows the dimensions of the matrix of different variables in the Kalman filter algorithm, with slight variation in notation of variables. N is the number of states, and M is the number of measurement variables [41].

TABLE 1

DIMENSIONS OF MATRIX OF DIFFERENT VARIABLES IN KALMAN FILTER ALGORITHM

Variable Notation	Matrix Dimension	Variables
$\hat{x}(n)$	$N \times 1$	State estimate
$\hat{x}(n)^-$	$N \times 1$	State estimate prior to the new measurement
$P(n)^-$	$N \times N$	Covariance matrix of the error in the state measurement. The (-) prefix signifies that this value is considered prior to the new measurement.
$G(n)$	$N \times M$	Kalman gain matrix
H	$1 \times N$	Observation model
$Z(n)$	$N \times N$	Process noise power
A	$N \times N$	State transitions matrix
$R(n)$	$M \times 1$	Measurement noise power

Consider a system with some process noise $Q(n) \sim (0, Z)$, where Z is power of the process noise with zero mean, along with measurement noise $V(n) \sim (0, R)$, where R is power of the measurement noise. The system model will be written as

$$x(n + 1) = A \cdot x(n) + Q(n) \quad (15)$$

$$y(n) = H \cdot x(n) + V(n) \quad (16)$$

The algorithm has two stages: prediction and measurement update. In the measurement update stage, by assuming initial uncertainty $P(0)^- = R$ and initial state $x(0)^-$, the first step of the Kalman filter algorithm can be initiated, which is calculating the Kalman gain $G(n)$ as

$$G(n) = P(n)^- \cdot H^T \cdot (H \cdot P(n)^- \cdot H^T + R)^{-1} \quad (17)$$

Next, the state estimate is updated as

$$\hat{x}(n)^+ = \hat{x}(n)^- + G(n) \cdot (y(n) - H \cdot \hat{x}(n)^-) \quad (18)$$

Next, the error covariance matrix is calculated as

$$P(n)^+ = (I - G(n) \cdot H) \cdot P(n)^- \quad (19)$$

Next, using the transition matrix, the state is projected as

$$\hat{x}(n + 1)^- = A \cdot \hat{x}(n)^+ \quad (20)$$

Error covariance is projected as

$$P(n + 1)^- = A \cdot P(n)^+ \cdot A^T + Z \quad (21)$$

By repeating these steps, the Kalman filter algorithm will be able to reduce the noise effect on the system, and hopefully the system will attain the steady state.

6.2 Implementation of Kalman Filter to Reduce Effects of Noise on Time Synchronization Algorithm

The actual main source that produces the noise V_k was discussed in the previous chapter. In this chapter, how to reduce or suppress the effects of this noise on timing synchronization process will be discussed, and for this purpose, the most effective filter for the

additive white Gaussian noise, i.e., a Kalman filter, was chosen, since it is assumed that the noise V_k is Gaussian. The Kalman filter restrains the effect of all kind of noises, but it is not as efficient on AWGN.

For the proposed system in the work of Simeone and Spagnolini [31] with the measurement noise shown in Figure 4, the Kalman filter can be implemented, as shown in Figure 5. One of the interesting things about the Kalman filter is that it is a replica of the system that can be observed in order to reduce the noise effect on the system, as shown in Figure 5. This is not an actual physical system but rather a virtual system that was designed in this thesis to perform a Kalman filter algorithm according to the scheme proposed in the work of Simeone and Spagnolini [31].

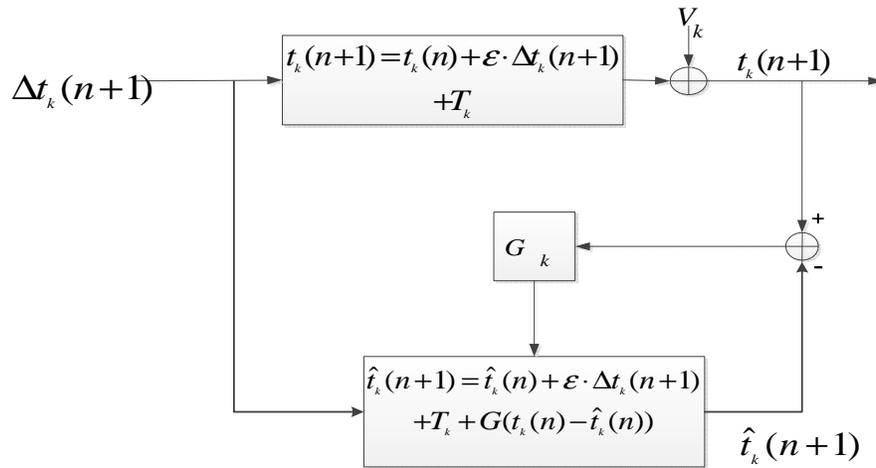


Figure 5. Block diagram of working Kalman filter used in proposed algorithm.

Because the dimension of the state variable is 1, the Kalman filter becomes scalar, but not vice versa. This implies that the state transition model is $A = 1$, the observation model is $H = 1$, and, because the process noise $Q(n)$ is not considered, the power of this noise $Z = 0$.

Therefore, the Kalman filter procedure for the system proposed in this paper is as explained in the measurement update stage: the Kalman filter algorithm calculates the Kalman

gain $G(n)$ using equation (18). Assuming initial covariance or initial uncertainty, i.e., $P_k(0)^- = R$, where R is the power of measurement noise and observation model $H_k = 1$, equation (17) will be modified as

$$G(n) = P_k(n)^- \cdot (P_k(n)^- + R)^{-1} \quad (22)$$

Updating the state estimate $\hat{t}_k(n + 1)^+$ of the proposed algorithm using equations (9) and (18), this can be written as

$$\hat{t}_k(n)^+ = \hat{t}_k(n)^- + \varepsilon \cdot \Delta t_k(n + 1) + T_k + G_k(n) \cdot (t_k(n) - H \cdot \hat{t}_k(n)^-) \quad (23)$$

And updating the covariance using equation (20), since $H_k = 1$, equation (19) will be modified as

$$P_k(n)^+ = (1 - G(n)) \cdot P_k(n)^- \quad (24)$$

In the next stage, or prediction stage, $A = 1$ is a transition value and $Z = 0$ is the power of process noise. This stage, which has two steps is not required in the case here, i.e., the predicted covariance estimate as shown in equation (25) and the prediction prior to the state estimate as shown in equation (26), respectively, are not necessary:

$$P_k(n + 1)^- = A \cdot P_k(n)^+ \cdot A^T + Z \quad (25)$$

$$\hat{t}_k(n + 1)^- = A \cdot \hat{t}_k(n)^+ \quad (26)$$

As a result of the transition value and power of the process noise, the predicted covariance estimate will be the same as the updated covariance, and the predicted prior state estimate will be the same as the updated state as shown in equations (27) and (28) respectively.

$$P_k(n + 1)^- = P_k(n)^+ \quad (27)$$

$$\hat{t}_k(n + 1)^- = \hat{t}_k(n)^+ \quad (28)$$

Notice that a discrete time Kalman filter is considered. If the predicted covariance estimate has no values equal to those of the updated covariance, and the predicted prior state

estimate is not equal to the update state, then it is necessary to take the average, as shown in equation (29), to obtain the state update more precisely.

$$\hat{t}_k(n + 1) = \frac{\hat{t}_k(n + 1)^- + \hat{t}_k(n + 1)^+}{2} \quad (29)$$

The steps shown in equations (23), (24), and (25) are repeated to eliminate the measurement noise effect from V_k on the timing vector. After some period of time n , the Kalman filter reduces the noise V_k effect on the timing vector $\hat{t}_k(n + 1)$.

CHAPTER 7

SIMULATION RESULT

Considering the same example given in the work of Simeone and Spagnolini [31] in which the channel is AWGN, a rectangular network topology having four nodes, as shown in Figure 6, was used for simulation. Consider an equal clock period $T_k = T = 1$, assuming that $K = 4$ nodes, that every node transmits with the same power, and that the $SNR = \frac{E_{12}}{N_0} = 15dB$. Other parameters are $\varepsilon = 0.9$, $\frac{D}{d} = 2$, $1/F_s = 0.01$, the waveform $g(t)$ is a raised cosine filter with roll-off $\delta = 0.2$, and the switching time is set to $\theta = 1/F_s$.



Figure 6. Rectangular topology with four nodes.

The standard deviation of the timing vector $t_k(n)$ was calculated, and plot iterations n versus the standard deviation of timing vector $\xi(n)$ with and without averaging with respect to channel noise and measurement noise are shown in Figures 7 to 10.

Figure 7 shows the standard deviation of the timing vector versus the iteration index n with no averaging and no Kalman filter under the measurement noise. Observe that the distributed network synchronization algorithm in the work of Simeone and Spagnolini [31] fails

completely, even when a small measurement noise of negative (-)10 dB is applied where signal power is normalized to unit, i.e., 0 dB. However, Figure 8 shows that when the Kalman filter is applied, the synchronization algorithm is successful, regardless of the noise power. The algorithm is working well, even if the noise power is 50 dB.

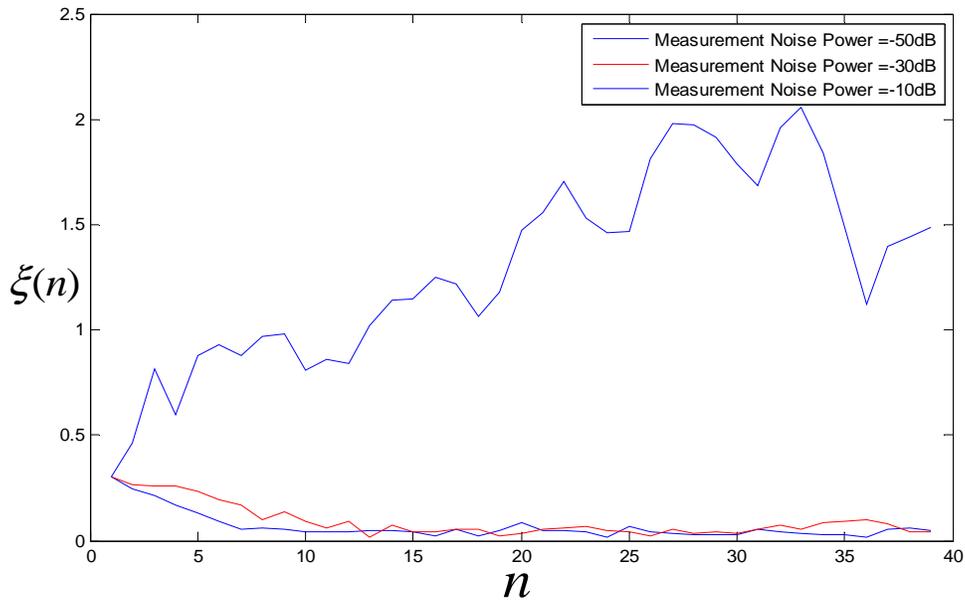


Figure 7. Standard deviation of timing vector $\xi(n)$ for synchronization algorithm without averaging with respect to channel noise and measurement noise. No Kalman filter was employed.

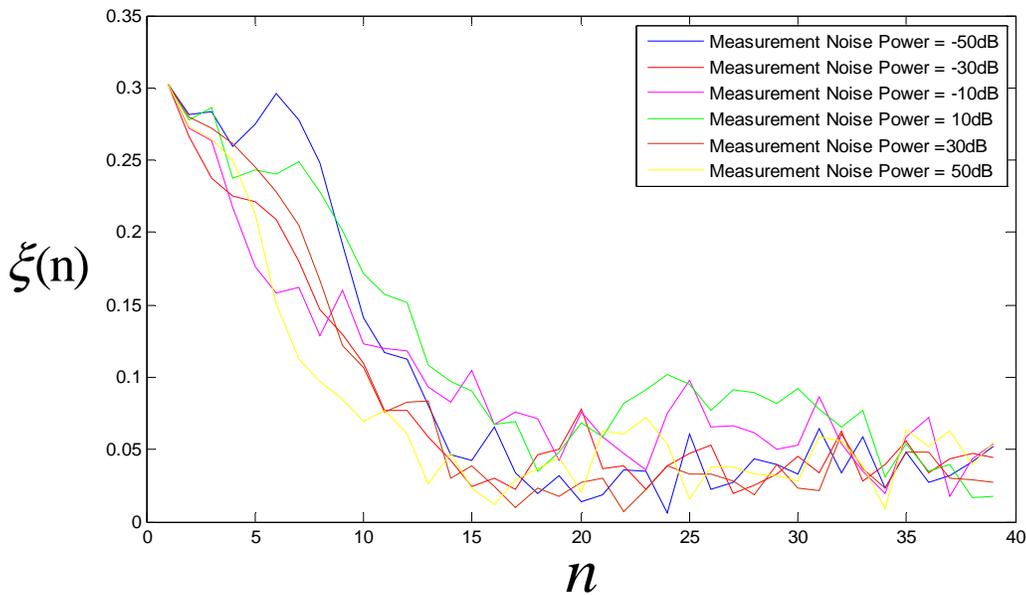


Figure 8. Standard deviation of timing vector $\xi(n)$ for synchronization algorithm without averaging with respect to channel noise and measurement noise. Kalman filter was employed.

Similarly, even if the averages are taken, Figures 9 and 10 show that the Kalman filter can make the distributed synchronization work successfully, regardless of the measurement noise power, whereas synchronization fails without the Kalman filter.

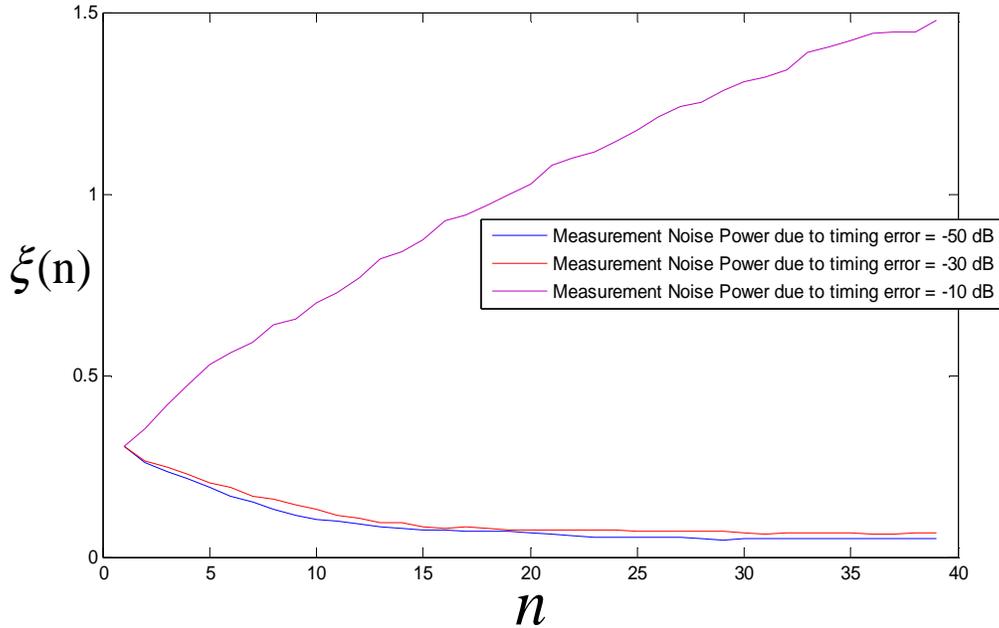


Figure 9. Standard deviation of timing vector $\xi(n)$ for synchronization algorithm with averaging with respect to channel noise and measurement noise. No Kalman filter was employed.

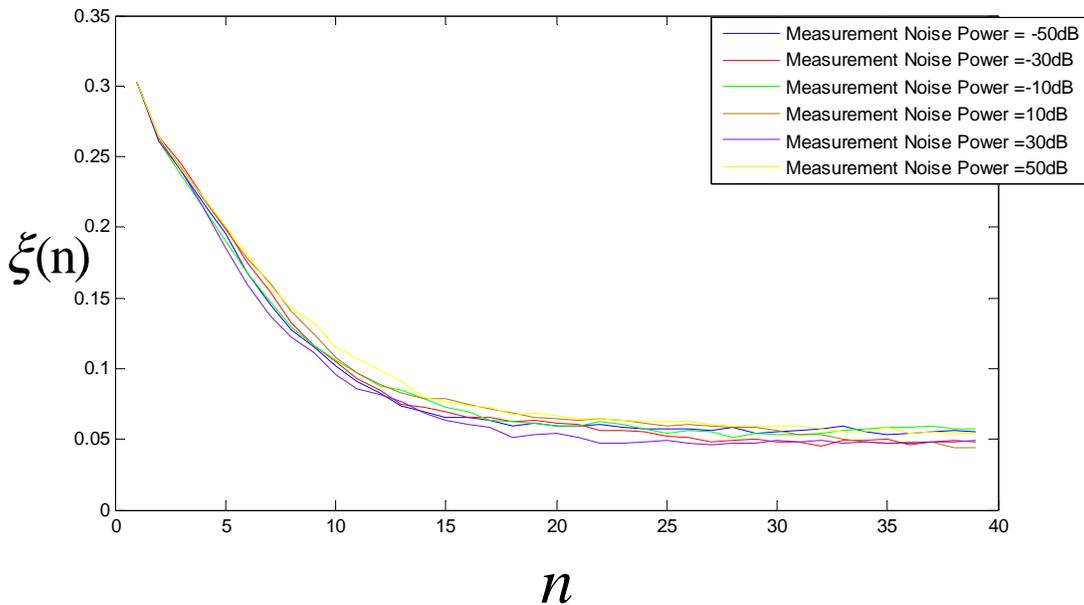


Figure 10. Standard deviation of timing vector $\xi(n)$ for synchronization algorithm without averaging with respect to channel noise and measurement noise. Kalman filter was employed.

CHAPTER 8

CONCLUSION

This paper considered the distributed consensus-based timing synchronization algorithm in the work of Simeone and Spagnolini [31] and showed that this algorithm can fail completely even under mild measurement noise. This paper also proposed to use the Kalman filter to suppress the negative effects of the measurement noise and demonstrated that the synchronization algorithm with the Kalman filter can be effective, regardless of measurement noise power, even 50 dB.

REFERENCES

LIST OF REFERENCES

- [1] Fumiyuki Adachi, Kazuki Takeda, Tetsuya Yamamoto, Ryusuke Matsukawa, and Shinya Kumagai, "Recent advances in single-carrier distributed antenna network," Special Issue Paper, published November 3, 2011, in Wiley Online Library (wileyonlinelibrary.com), DOI: 10.1002/wcm.1212.
- [2] Shidong Zhou, Ming Zhao, Xibin Xu, Jing Wang, and Yan Yao, "Distributed wireless communication system: A new architecture for future public wireless access," *IEEE Communications Magazine*, March 2003.
- [3] F. Rusek, D. Persson, B. K. Lau, E. G. Larsson, T. L. Marzetta, O. Edfors, and F. Tufvesson, "Scaling up MIMO: Opportunities and challenges with large arrays," arXiv:1201.3210, 2011.
- [4] H. Q. Ngo, E. G. Larsson, and T. Marzetta, "Energy and spectral efficiency of very large multiuser MIMO systems," arXiv:1112.3810, 2011.
- [5] S. K. Mohammed and E. G. Larsson, "Per-antenna constant envelope precoding for large multi-user MIMO systems," arXiv:1201.1634v1, 2011.
- [6] C. Studer and E. G. Larsson, "PAR-Aware large-scale multi-user MIMO-OFDM downlink," arXiv:1202.4034, 2012.
- [7] J. Hoydis, S. ten Brink, M. Debbah, "Massive MIMO: how many antennas do we need?," arXiv:1107.1709, 2011.
- [8] X. Gao, O. Edfors, F. Rusek, and F. Tufvesson, "Linear pre-coding performance in measured very-large MIMO channels," *IEEE VTC*, 2011.
- [9] A. Pitarokoilis, S. K. Mohammed, and E. G. Larsson, "On the optimality of single-carrier transmission in large scale antenna systems," *IEEE Wireless Communication Letters*, submitted 2012.
- [10] T. L. Marzetta, "Noncooperative MU-MIMO with unlimited numbers of base station antennas," *IEEE Trans. Wireless. Comm.*, 2010.
- [11] J. Jose, A. Ashikhmin, T. L. Marzetta, and S. Vishwanath, "Pilot contamination and precoding in multi-cell TDD systems," *IEEE Trans. Wireless Commun.*, 2011.
- [12] Viveck R. Cadambe and Syed Ali Jafar, "Interference alignment and degrees of freedom of the k-user interference channel," *IEEE Transactions on Information Theory*, vol. 54, no. 8, pp. 3425–3441, August 2008.

LIST OF REFERENCES (continued)

- [13] Abinesh Ramakrishnan, Abhik Das, Hamed Maleki, Athina Markopoulou, Syed Jafar, and Sriram Vishwanath, “Network coding for three unicast sessions: Interference alignment approaches,” Forty-Eighth Annual Allerton Conference, Allerton House, UIUC, Illinois, USA, September 29–October 1, 2010.
- [14] Krishna Gomadam, Viveck R. Cadambe, and Syed A. Jafar, “A distributed numerical approach to interference alignment and applications to wireless interference networks,” *IEEE Transactions on Information Theory*, vol. 57, no. 6, pp. 3309–3322, June 2011.
- [15] Meisam Razaviyayn, Maziar Sanjabi, and Zhi-Quan Luo, “Linear transceiver design for interference alignment: Complexity and computation,” *IEEE Transactions on Information Theory*, vol. 58, no. 5, pp. 2896–2910, May 2012.
- [16] Syed Ali Jafar, “Blind interference alignment,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 6, no. 3, pp. 210–227, June 2012.
- [17] Hamed Maleki, Syed Ali Jafar, and Shlomo Shamai, “Retrospective interference alignment over interference networks,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 6, no. 3, pp. 228–240, June 2012.
- [18] Bobak Nazer, Michael Gastpar, Syed Ali Jafar, and Sriram Vishwanath, “Ergodic interference alignment,” *IEEE Transactions on Information Theory*, vol. 58, no. 10, pp. 6355–6370, October 2012.
- [19] R. Ahlswede, N. Cai, S.Y.R. Li, and R.W. Yeung, “Network information flow,” *IEEE transactions on information theory*, vol. 46, no. 4, pp. 1204–1216, July 2000.
- [20] Ning Cai and Raymond W. Yeung, “Secure network coding on a wiretap network,” *IEEE Transactions on Information Theory*, vol. 57, no. 1, pp. 424–435, January 2011.
- [21] Zhu Han, Xin Zhang, and H. Vincent Poor, “High performance cooperative transmission protocols based on multiuser detection and network coding,” *IEEE Transactions on Wireless Communications*, vol. 8, no. 5, pp. 2352–2361, May 2009.
- [22] Raymond W. Yeung, “Network Coding: A historical perspective,” *Proceedings of the IEEE*, vol. 99, no. 3, pp. 366–371, March 2011.
- [23] R. W. Yeung, S.-Y. R. Li, N. Cai, and Z. Zhang, “Network coding theory,” *Found. Trends Commun. Inf. Theory*, vol. 2, no. 4 and 5, pp. 241–381, 2005.
- [24] Uri Erez and Ram Zamir, “Achieving $1/2\log(1 + \text{SNR})$ on the AWGN channel with lattice encoding and decoding,” *IEEE Transactions on Information Theory*, vol. 50, no. 10, pp. 2293–2313, October 2004.

LIST OF REFERENCES (continued)

- [25] Ram Zamir, Shlomo Shamai (Shitz), and Uri Erez, “Nested linear/lattice codes for structured multi terminal binning,” *IEEE Transactions on Information Theory*, vol. 48, no. 6, pp. 1250–1276, June 2002.
- [26] Uri Erez, Shlomo Shamai (Shitz), and Ram Zamir, “Capacity and lattice strategies for canceling known interference,” *IEEE Transactions on Information Theory*, vol. 51, no. 11, pp. 3820–3833, November 2005.
- [27] Wooseok Nam, Sae-Young Chung, and Yong H. Lee, “Capacity of the Gaussian two-way relay channel to within 1/2 bit,” *IEEE Transactions on Information Theory*, vol. 56, no. 11, pp. 5488–5494, November 2010.
- [28] Bobak Nazer and Michael Gastpar, “Compute-and-Forward: Harnessing interference through structured codes,” *IEEE Transactions on Information Theory*, vol. 57, no. 10, pp. 6463–6486, October 2011.
- [29] Erdal Arkan, “Channel Polarization: A method for constructing capacity-achieving codes for symmetric binary-input memory less channels,” *IEEE Transactions on Information Theory*, vol. 55, no. 7, pp. 3051–3073, July 2009.
- [30] Satish Babu Korada, Eren S Sasoglu, and Rüdiger Urbanke, “Polar codes: Characterization of exponent, bounds, and constructions,” *IEEE Transactions on Information Theory*, vol. 56, no. 12, , pp. 6253–6264, December 2010.
- [31] O. Simeone and U. Spagnolini, “Distributed time synchronization in wireless sensor networks with coupled discrete-time oscillators,” *EURASIP Journal on Wireless Communications and Networking*, vol. 2007, article ID 57054, 13 pages, DOI:10.1155/2007/57054.
- [32] William C. Lindsey, Farzad Ghazvinian, Walter C. Haggmann, and Khaled Dessouky, “Network synchronization,” *Proceedings of the IEEE*, vol. 73, no. 10, pp. 1445–1467, October 1985.
- [33] X. Li, Y.-C.Wu, and E. Serpedin, “Timing synchronization in decode-and-forward cooperative communication systems,” *IEEE Transactions on Signal Processing*, vol. 57, no. 4, pp. 1444–1455, April 2009.
- [34] X. Li, “Space-time coded multi-transmission among distributed transmitters without perfect synchronization,” *IEEE Signal Processing Letters*, vol. 11, no. 12, pp. 948–951, December 2004.

LIST OF REFERENCES (continued)

- [35] Y. Mei, Y. Hua, A. Swami, and B. Daneshrad, “Combating synchronization errors in cooperative relays,” *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '05)*, March 18–23, 2005, vol. 3, pp. iii/369–iii/372.
- [36] R. C. Palat, A. Annamalai, and J. H. Reed, “Upper bound on bit error rate for time synchronization errors in bandlimited distributed MIMO networks,” *Proc. IEEE Wireless Communications and Networking Conference (WCNC) 2006*, April 3–6, 2006, vol. 4, pp. 2058–2063.
- [37] K. Raghunath and A. Chockalingam, “Cooperative OFDM with amplify-and-forward relaying with timing offset,” *Proc. IEEE Global Telecommunications Conference (GLOBECOM 2008)*, November 30, 2008–December 4, 2008, pp. 1–5.
- [38] F. Tong and Y. Akaiwa, “Theoretical analysis of interbasestation synchronization systems,” *IEEE Transactions on Communications*, vol. 46, no. 5, pp. 590–594, 1998.
- [39] E. Sourour and M. Nakagawa, “Mutual decentralized synchronization for intervehicle communications,” *IEEE Transactions on Vehicular Technology*, vol. 48, no. 6, pp. 2015–2027, 1999.
- [40] L. Moreau, “Stability of multi agent systems with time-dependent communication links,” *IEEE Transactions on Automatic Control*, vol. 50, no. 2, pp. 169–182, 2005.
- [41] Andrew Huard and Chris Chiccone, “Implementation of four states Kalman filter,” Project, Electrical Engineering Department, California Polytechnic State University, California, December 2009.