

DISTRIBUTED NETWORK TIME SYNCHRONIZATION USING PSEUDO NOISE CODE

A Thesis by

Prashanth Paladugula

Bachelor of Technology, Kamala Institute of Technology and Science, JNTU, Hyderabad, 2008

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

May 2013

© Copyright 2013 by Prashanth Paladugula

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.

DISTRIBUTED NETWORK TIME SYNCHRONIZATION USING PSEUDO NOISE CODE

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 requirements 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 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): Youvaraj, Kanghee, Wenhao, Kenny, Raju, Jessie, Sangku, Mahesh, Kim, Shuang, and others for their support and encouragement throughout my research. I thank my friends Karthik Vepuri, Harish, Shreyas, Pramod, Sai Mithil, Sujeeth, and Karthik Kallam for their moral support and encouragement. Lastly, I thank the Department of Electrical Engineering and Computer Science for providing funding through my graduate teaching assistantship position.

ABSTRACT

In recent years, distributed network systems have been studied in order to yield a higher data rate. However, performance of a distributed network degrades drastically when all nodes are not properly synchronized. This thesis discusses the practical issues involved in existing distributed network synchronization algorithms, and proposes an alternative solution for time synchronization in distributed networks using a pseudo noise (PN) code. The proposed algorithm has the following advantages over the existing algorithm: (1) increased time width of the pulse-shaping filter, i.e., narrowband waveform-shaping filter; (2) ability to work under a lower signal-to-noise ratio; (3) capability of reaching a steady state with fewer iterations; and (4) lower steady-state error and fewer samples required per one update interval. Chapter 6 will elaborate and explain these advantages. Simulation results will show the effectiveness of the algorithm proposed and the advantages of the proposed algorithm over existing algorithms.

TABLE OF CONTENTS

Chapter	Page
1. INTRODUCTION	1
1.1 Motivation	1
1.2 Literature Survey	1
1.3 Main Idea behind Thesis	2
1.4 Main Contribution of Thesis	3
1.5 Outline of Thesis	3
2. SIGNAL MODEL	4
3. EXISTING TIME SYNCHRONIZATION ALGORITHM	6
3.1 Synchronization Algorithm	6
3.2 Implementation using Discrete-Time Oscillators	6
4. PROPOSED PN CODE-BASED TIME SYNCHRONIZATION ALGORITHM	8
5. SIMULATION RESULTS	11
5.1 Triangular Network Topology	12
5.2 Linear Network Topology	13
5.3 Rectangular Network Topology	14
5.4 Star Network Topology	15
5.5 Effect of Fading	16
5.6 Comparison of Number of Samples per Symbol	17
5.7 Comparison of Existing and Proposed Algorithms	18
6. DISCUSSION	21
7. CONCLUSION	22
LIST OF REFERENCES	23

LIST OF FIGURES

Figure	Page
1. (a) Clock $t_k(n)$ of the k -th node; (b) sample PN code generated at $t_k(n)$ for $M = 7$	4
2. Sample sketch of received signal at k -th node when discrete-time oscillator implementation is used.....	7
3. Sample plot of correlation vector of PN code.....	8
4. Sketch of sample correlation vector for AWGN channel	9
5. Triangular network topology.....	12
6. Standard deviation of timing vectors for triangular network shown in Figure 5	12
7. Linear network topology.....	13
8. Standard deviation of timing vectors for linear network shown in Figure 7	13
9. Rectangular network topology.....	14
10. Standard deviation of timing vectors for rectangular network shown in Figure 9	14
11. Star network topology.....	15
12. Standard deviation of timing vectors for star network shown in Figure 11.....	15
13. Comparison of standard deviation of timing vectors of different fading channels for linear network shown in Figure 7.....	16
14. Comparison of number of samples taken per symbol for rectangular network shown in Figure 9 and AWGN channel.....	17
15. Comparison of standard deviation of timing vectors of discrete-time oscillator and PN code algorithms for rectangular network and AWGN channel	19
16. Comparison of standard deviation of timing vectors of discrete-time oscillator and PN code algorithms for rectangular network and Rayleigh fading channel	20

LIST OF ABBREVIATIONS

Gbps	Giga Bits per Second
MIMO	Multiple Input Multiple Output
MISO	Multiple Input Single Output
PN	Pseudo Noise
RRC	Root Raised Cosine
SNR	Signal-to-Noise Ratio

CHAPTER 1

INTRODUCTION

1.1 Motivation

In order to realize a gigabit wireless technology that can achieve a data rate of more than 1 Gbps, a distributed network may be needed [1]. Currently considerable research using distributed networks is in progress. Potential candidates for future generation wireless communication include a massive multiple-input multiple-output (MIMO) system [2]-[5], interference alignment [6]-[8], lattice coding [9]-[11], etc. Almost all of these candidates assume perfect synchronization in a network. However, performance degradation will be very high if all nodes are not synchronized perfectly. Jagannathan et al. [12] and Palat et al. [13] have discussed the effects of synchronization error on the performance of the multiple-input single-output (MISO) system and showed that the synchronization error will degrade performance of the MISO system.

The existing algorithms have some issues. The literature survey in this chapter will discuss these issues and explore the possible ways to overcome them.

1.2 Literature Survey

Conventional timing synchronization algorithms exchange timing information through packets with a timestamp [14]. Conventional methods do not satisfy the requirements of wireless networks, such as energy efficiency, computational complexity, and scalability. To overcome the issue of scalability, Kida et al. [15] proposed a sequential synchronization algorithm for a network synchronization scheme for a scalable two-way multi-hop network. This synchronization scheme is achieved by a pair (say node 1, node 2) synchronization, first using the conventional synchronization scheme and then the next pair (say node 2, node 3). Node 1 has

a master clock, and node 2 is used twice. Hence, all nodes are synchronized. If the number of nodes increases in this sequential scheme, then it will take time for them to be synchronized, thus becoming unstable and out of lock. Using Lindsey's discrete-time consensus approach [16], Simeone and Spagnolini [17] proposed an algorithm for decentralized parallel synchronization of wireless sensor networks using discrete-time oscillators. In this scheme, each node transmits a band-limited waveform, such as a square-root raised cosine (RRC) pulse, with a symbol period $1/F_s$. The symbol period $T_s = \frac{1}{F_s} \ll T$, where T is the clock-update period of each node. At the receiver, more than 100 oversamples per symbol were taken from the received signal for synchronization. This scheme can be used for high-power sensor networks but not for future wireless networks with a 1,000 Gbps data rate, because the clock-update period T will also become very small in the case of high data rate networks, and achieving an RRC filter with period $1/F_s$, which is much less than T , is impossible or very difficult. Even if this filter can be achieved, it will be highly affected by noise. Simeone and Spagnolini [17] took one sample per update to estimate the timing-error differences between nodes. Because only one sample at a time is considered, the effect of noise will be very high, and their scheme will not work under lower signal-to-noise ratios (SNRs). Based on the considerable amount of literature surveyed for this thesis, it was found that the algorithm in Simeone and Spagnolini's work [17] was a reasonable distributed time synchronization algorithm, and hence, this thesis will use this algorithm for comparison with the proposed algorithm.

1.3 Main Idea behind Thesis

To overcome the above issues, this thesis proposes a new scheme that uses a pseudo noise (PN) code as well as instantaneous correlation values instead of instantaneous power measurement. Here, the PN code correlation interval is an interval of multiple symbols, and a

symbol in the proposed scheme corresponds to a chip in the conventional PN code spread spectrum. Also, each symbol (or chip) can be transmitted with the same RRC band-limited waveform. Hence, there is no spread spectrum expansion in frequency. The bandwidth of the proposed system is unchanged and equal to that of the no-spreading case. The correlation is taken over many samples, whereby the effect of noise on one sample can be cancelled by the noise on another sample in the correlation processing and can be used for lower signal-to-noise ratio (SNR) values. This is the main idea behind the proposed distributed network timing synchronization.

1.4 Main Contribution of Thesis

- Revises the algorithm in the work of Simeone and Spagnolini [17] and proposes distributed network time synchronization using a pseudo noise code.
- Compares the proposed algorithm and Simeone and Spagnolini's algorithm [17] to show the effectiveness of the proposed algorithm.

1.5 Outline of Thesis

The remainder of this thesis is organized as follows: Chapter 2 describes the system model and data transmission strategy applied. Chapter 3 describes the synchronization algorithm given by Simeone and Spagnolini [17]. Chapter 4 describes the proposed algorithm. Chapter 5 provides simulation results to evaluate the effectiveness of the algorithm. Chapter 6 provides discussion to compare the proposed algorithm and Simeone and Spagnolini's algorithm [17]. Chapter 7 concludes the thesis.

CHAPTER 2

SIGNAL MODEL

Assume that the network has K nodes, each with a common clock period T_0 , i.e., a common frequency. Each node transmits a common pseudo noise code of length M symbols ($M = 2^m - 1$, where m is an integer). Also, consider a PN code of length M as one correlation interval with time period T_0 , where $T_0 = MT_s$.

Since the nodes are isolated, the k th sensor evolves as $t_k(n) = nT_0 + \tau_k(n)$, where $-\frac{T_0}{2} \leq \tau_k(n) < \frac{T_0}{2}$. The aim of synchronization is to make all phases equal, as shown below:

$$\tau_1 = \tau_2 = \dots = \tau_k = \dots = \tau_K. \quad (1)$$

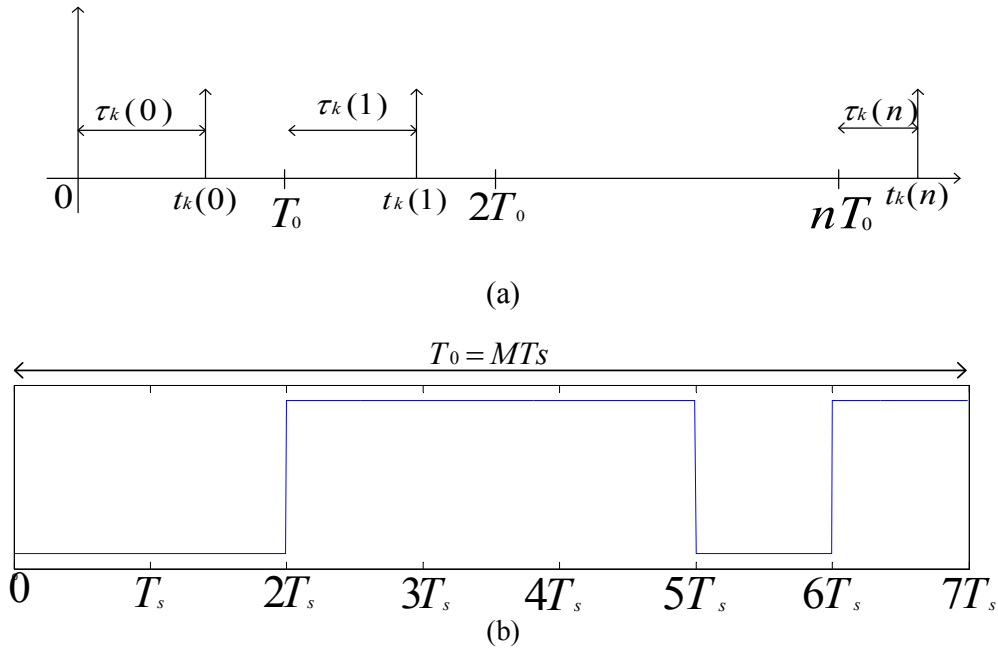


Figure 1: (a) Clock $t_k(n)$ of k -th node; (b) sample PN code generated at $t_k(n)$ for $M = 7$.

To achieve synchronization, each node transmits a PN code at time $t_k(n)$. The signal received at the k th receiver at the n th time slot is

$$\mathbf{y}_k(n) = \sum_{i=1}^K \sqrt{E_{ki}} \cdot \beta_{ki} \cdot \mathbf{PN}_k(n - (t_i(n) - t_k(n))) + \mathbf{w}(n) \quad (2)$$

where $t_i(n)$ and $t_k(n)$ are the clock time of i -th node and k -th node, respectively, at the n -th observation interval, the power during a symbol period $E_{ki} = C/(d_{ki}^\gamma)$, C is an appropriate constant depending on transmitted power, $d_{ki} = d_{ik}$ is the distance between node i and node k , γ is the path loss exponent, β_{ki} denotes the shadow fading coefficient, $w(n)$ is the additive Gaussian noise with zero mean and power N_0 , and $\mathbf{PN}_k(n)$ is the PN code transmitted by the k -th user for the n -th observation interval $\mathbf{y}_k(n)$, $\mathbf{w}(n)$, and $\mathbf{PN}_k(n)$ are vectors of length $N_s M$, where N_s is the number of samples taken for one symbol interval.

CHAPTER 3

EXISTING TIME SYNCHRONIZATION ALGORITHM

3.1 Synchronization Algorithm

This section reviews the synchronization algorithm proposed by Simeone and Spagnolini [17]. Assume that the network has K nodes, each with a common clock period T_0 . This synchronization procedure assumes that any node, say the k -th node, is able to measure exactly the timing differences $t_i(n) - t_k(n)$ and the power $E_{ki}(n)$ of all other nodes using the received signal. In the next section, this assumption will be eliminated and a practical implementation that synchronizes all nodes is provided.

At the n th period, the k th node updates its clock $t_k(n)$ according to a weighted sum of timing differences $\Delta t_k(n + 1)$:

$$t_k(n + 1) = t_k(n) + \varepsilon \cdot \Delta t_k(n + 1) + T_0 \quad (3a)$$

$$\Delta t_k(n + 1) = \sum_{i=1, i \neq k}^K \alpha_{ki}(n) \cdot (t_i(n) - t_k(n)) \quad (3b)$$

where ε is the step size ($0 < \varepsilon < 1$), and the coefficients $\alpha_{ki}(n)$ are written as

$$\alpha_{ki}(n) = \frac{E_{ki}(n)}{\sum_{j=1, j \neq k}^K E_{kj}(n)}. \quad (4)$$

Theorem 1 in the work of Simeone and Spagnolini [17] proves that this algorithm converges if all nodes are strongly connected.

3.2 Implementation using Discrete-Time Oscillators

This section reviews the discrete-time oscillators algorithm proposed by Simeone and Spagnolini [17]. Here, each node, say k -th, transmits band-limited waveform $g(t)$ centered at $t_k(n)$ with symbol period $1/F_s$. The symbol period $1/F_s$ defines the timing resolution of the system. Each node works in a half-duplex mode and measures the received signal over the

interval $(t_k(n) - \frac{T_0}{2}, t_k(n) + \frac{T_0}{2})$. The received signal is oversampled with frequency LF_s ($L \geq 1$).

Based on the $N = LF_s T_0$ samples received in the n th observation window, the sampling index ranges from $-N/2$ to $N/2$ with $m = 0$ corresponds to the firing time $t_k(n)$. The received signal at the k th receiver is given by

$$y_k(n, m) = \sum_{i=1, i \neq k}^K \sqrt{E_{ki}} \cdot \beta_{ki} \cdot g\left(\frac{m}{LF_s} - (t_i(n) - t_k(n))\right) + w(n, m). \quad (5)$$

A sample sketch of the received signal is shown in Figure 2.

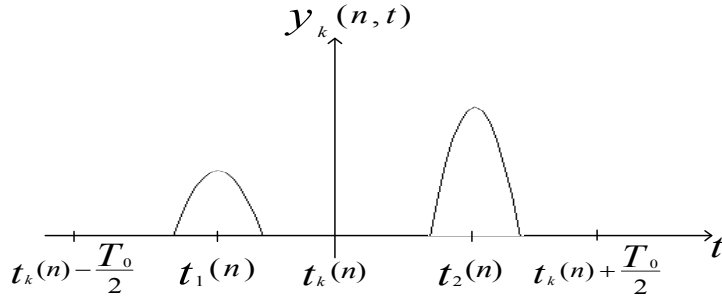


Figure 2: Sample sketch of received signal at k -th node when discrete-time oscillator implementation is used.

The timing update is calculated as

$$\Delta t_k(n+1) = \sum_{i \in I} \bar{\alpha}_{km} \cdot \frac{m}{LF_s} \quad (6a)$$

$$\bar{\alpha}_{km} = \frac{|y_k(n, m)|^2}{\sum_{i \in I} |y_k(n, i)|^2} \quad (6b)$$

where I is the subset of time instants for which received signal is above a given threshold, and $m \in \left(-\frac{N}{2}, -\theta LF_s\right) \cup \left(\theta LF_s, \frac{N}{2}\right)$. The sample in the interval $(-\theta LF_s, \theta LF_s)$ is not measured due to the half-duplex constraint. Equation (6a) provides an estimate of equation (3b), and the next firing time can be calculated using equation (3a).

CHAPTER 4

PROPOSED PN CODE-BASED TIME SYNCHRONIZATION ALGORITHM

This chapter provides a practical algorithm to find the values of α_{ki} . The correlation property of the PN code is used to find the values of α_{ki} . Consider $c(t)$ as a pseudo noise code signal; then the normalized auto correlation is defined as

$$R(\tau) = \frac{1}{T_0} \int_{-\frac{T_0}{2}}^{\frac{T_0}{2}} c(t)c(t - \tau)dt. \quad (7)$$

A sample plot of $R(\tau)$ is shown in Figure 3. As shown, the correlation is maximum when $\tau = 0$. Since all nodes are transmitting the same PN code, it can be said that the nodes are synchronized when the correlation between any two transmitted signals is maximum.

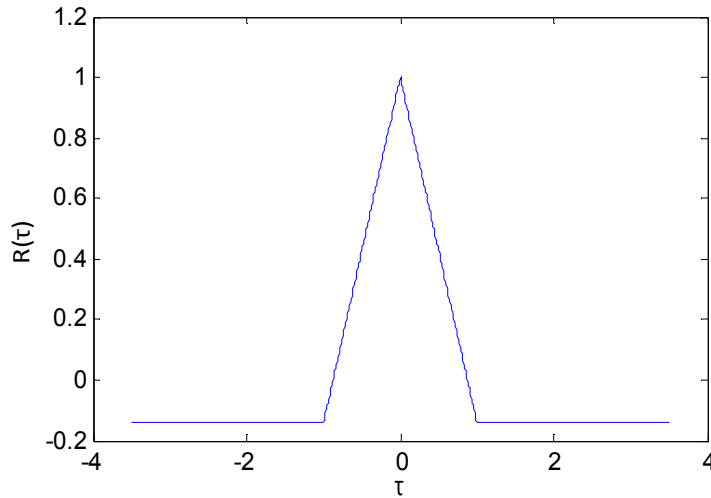


Figure 3: Sample plot of correlation vector of PN code.

In this model, since a PN code is used, instantaneous power cannot be found, but the values of $\alpha_{ki}(n)$ can be determined using instantaneous correlation values instead of instantaneous power because the correlation is directly proportional to the received power.

The value of α_{ki} can be estimated by examining the received signal over the interval $-T_0/2$ to $T_0/2$. Because this system is a full duplex, the received signal at the k -th user will contain the signal transmitted by the k -th node. Therefore, the local PN code signal is subtracted from the received signal in order to find the sum of transmitted signals from other users. By sliding the local PN code at the k th receiver over the observation interval $-T_0/2$ to $T_0/2$, the value of correlation at each timing interval can be determined:

$$r_k(n, m) = \langle \mathbf{PN}_k(n - m), y_k(n) - \mathbf{PN}_k(n) \rangle \quad (8)$$

where m is a sampled value between $(-\frac{T_0}{2}, \frac{T_0}{2})$, the interval $(-\frac{T_0}{2}, \frac{T_0}{2})$ is sampled into L values, $m = 0$ corresponds to the timing value $t_k(n)$, and $\langle \mathbf{a}, \mathbf{b} \rangle$ is the dot product of the two vectors \mathbf{a} and \mathbf{b} . Figure 4 shows an example of the correlation vector in equation (8).

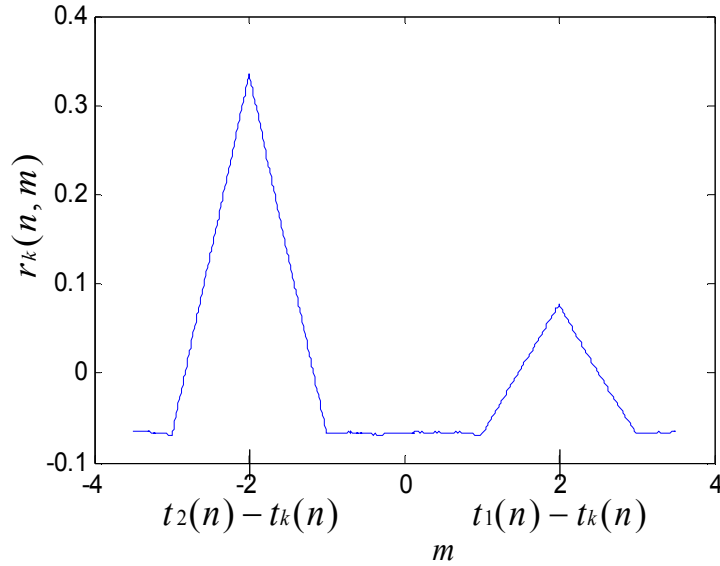


Figure 4: Sketch of sample correlation vector for AWGN channel.

As shown in Figure 4, the correlation value will have a peak at all $t_i(n) - t_k(n)$, the value of which depends on the power $E_{ki}(n)$, which in turn depends on the path loss between nodes k and i .

Now the value of $\Delta t_k(n + 1)$ can be found by

$$\Delta t_k(n + 1) = \sum_{\substack{m=-\frac{T_0}{2} \\ m \in G}}^{T_0/2} \bar{\alpha}_{km} \cdot m \quad (9)$$

where

$$\bar{\alpha}_{km} = \frac{r_k(n,m)}{\sum_{\substack{m=-\frac{T_0}{2} \\ m \in G}}^{T_0/2} r_k(n,m)} \quad (10)$$

and G is a subset of time instances for which the correlation $r_k(n, m)$ is above a given threshold. This scheme does not need the exact information of $t_i(n) - t_k(n)$ but instead uses the distributed consensus control scheme. Each node can share only its output with its neighbors through which the consensus is achieved [18]. Here, each m value corresponds to the timing difference from the k -th node firing time $t_k(n)$, and the value of m at a peak corresponds to the value of $t_i(n) - t_k(n)$. Also, the value of correlation is used instead of instantaneous power, and hence, equation (9) can provide an estimate of $\Delta t_k(n + 1)$ in equation (3b).

CHAPTER 5

SIMULATION RESULTS

In order to demonstrate the efficiency of the proposed algorithm, the demonstrated algorithm is simulated using Matlab and presented results for four different network topologies. The simulation environment is as follows:

Length of PN sequence = 7

PN sequence pattern = [0 0 1 1 1 0 1]

T_s (symbol time) = 1 (i.e., $T_0 = 7$)

N_s (number of samples per T_s) = 10

Local PN code slides from -3.5 to 3.5 in steps of 0.1

Threshold = 0

γ (path loss exponent) = 3

SNR = $(\frac{E_{12}}{N_0}) = 5\text{dB}$

Channel is AWGN ($\beta_{ki} = 1$).

Simulation results show the expected value of standard deviation $\xi(n)$ of the timing vector $\mathbf{t}(n)$ versus n , where $\xi^2(n) = E[\sum_{k=1}^K ((t_k(n) - 1/K \sum_{k=1}^K t_k(n))^2]$, and expectation $E[\cdot]$ is taken with respect to noise.

5.1 Triangular Network Topology

Figure 5 shows the triangular network topology used.

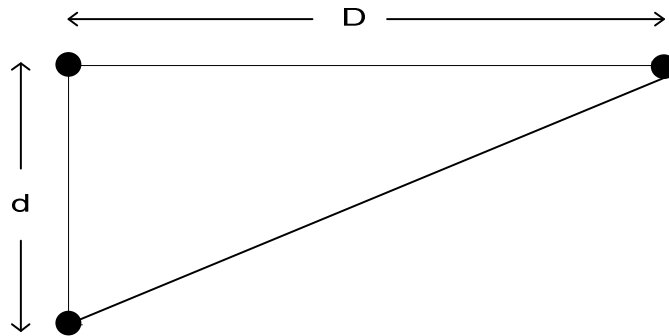


Figure 5: Triangular network topology.

Figure 6 shows the standard deviation of timing vectors for the triangular network topology shown in Figure 5, with initial timing vector = $[0, -3, 3]$, $D = 8$, $d = 4$.

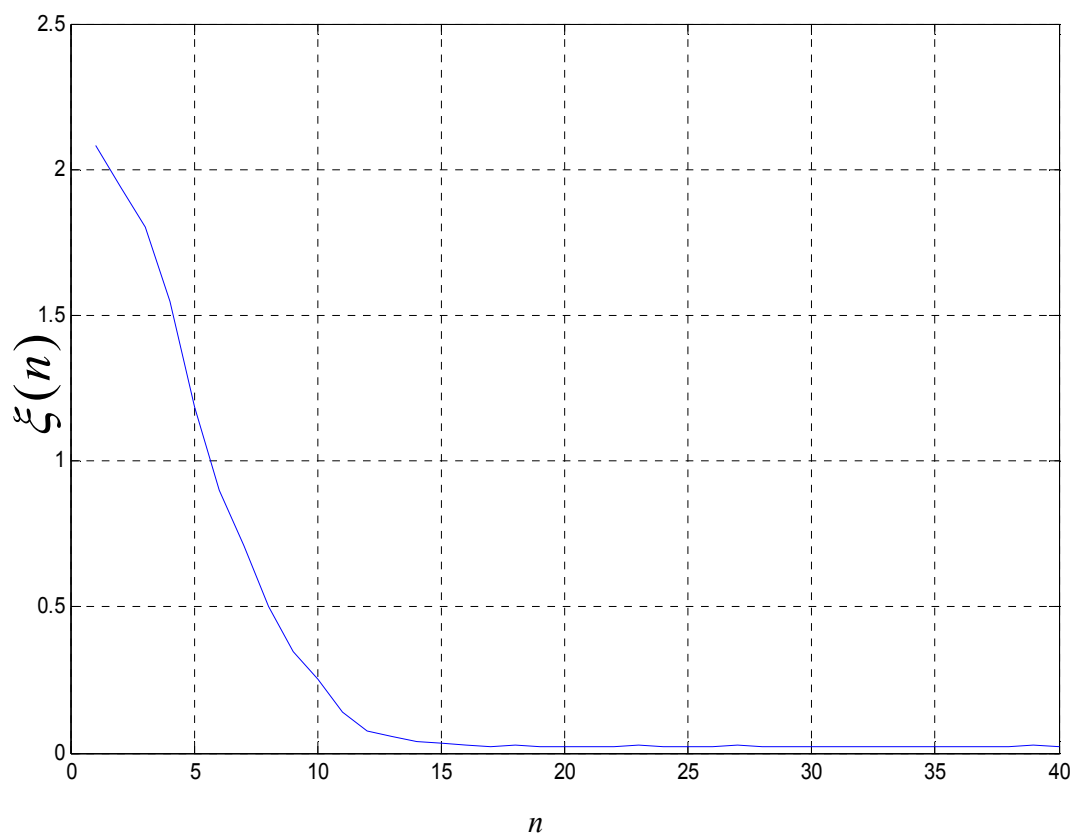


Figure 6: Standard deviation of timing vectors for triangular network shown in Figure 5.

5.2 Linear Network Topology

Figure 7 shows the linear network topology used.

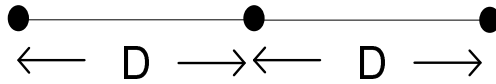


Figure 7: Linear network topology.

Figure 8 shows the standard deviation of timing vectors for the triangular network topology shown in Figure 7, with initial timing vector = $[0, -3, 3]$, $D = 2$.

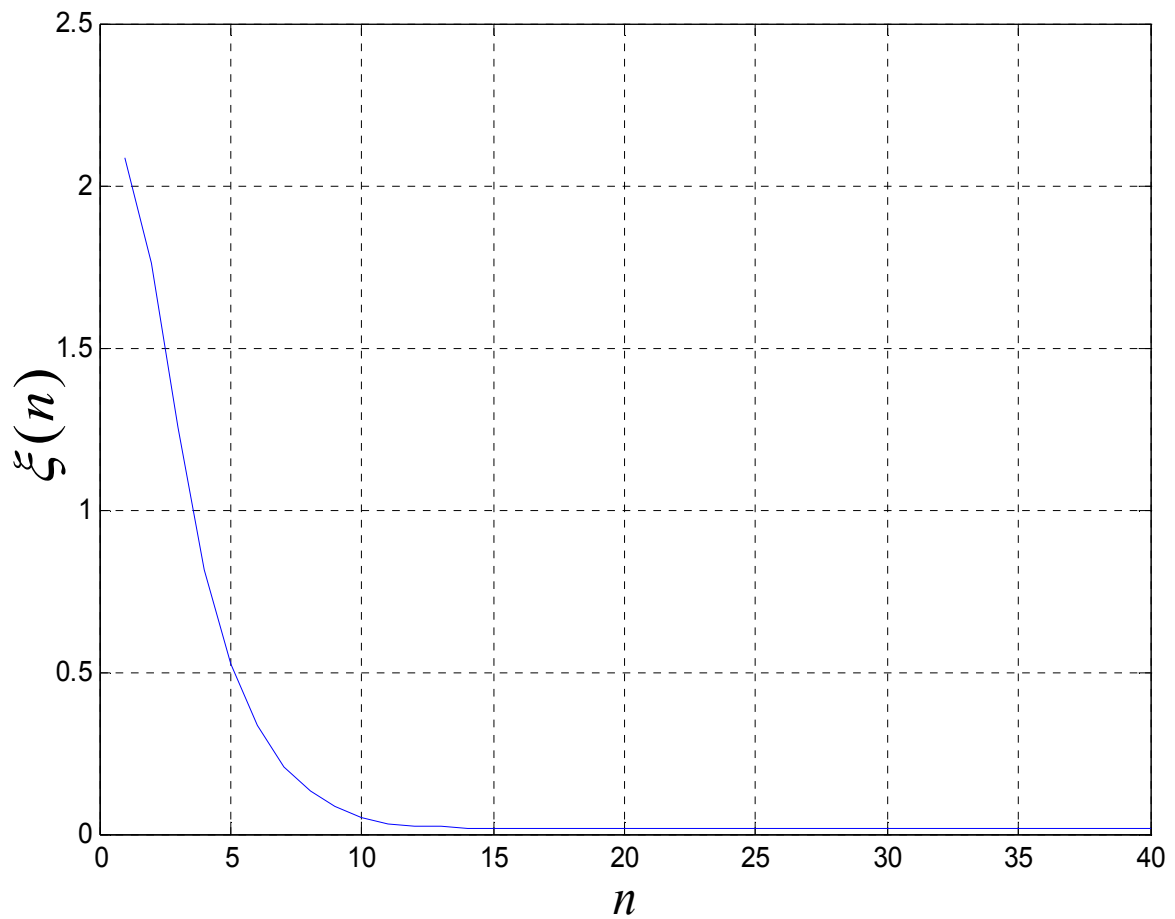


Figure 8: Standard deviation of timing vectors for linear network shown in Figure 7.

5.3 Rectangular Network Topology

Figure 9 shows the rectangular network topology used.

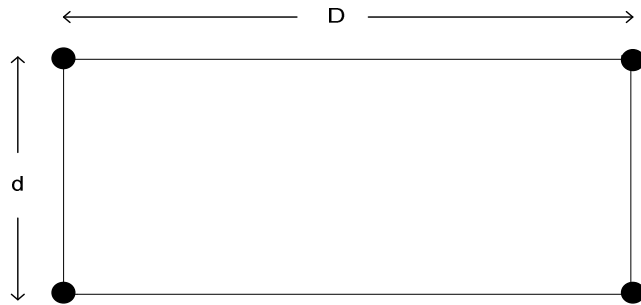


Figure 9: Rectangular network topology.

Figure 10 shows the standard deviation of timing vectors for the triangular network topology shown in Figure 9, with initial timing vector = $[-3, -1, 1, 3]$, $D = 4$, $d = 2$.

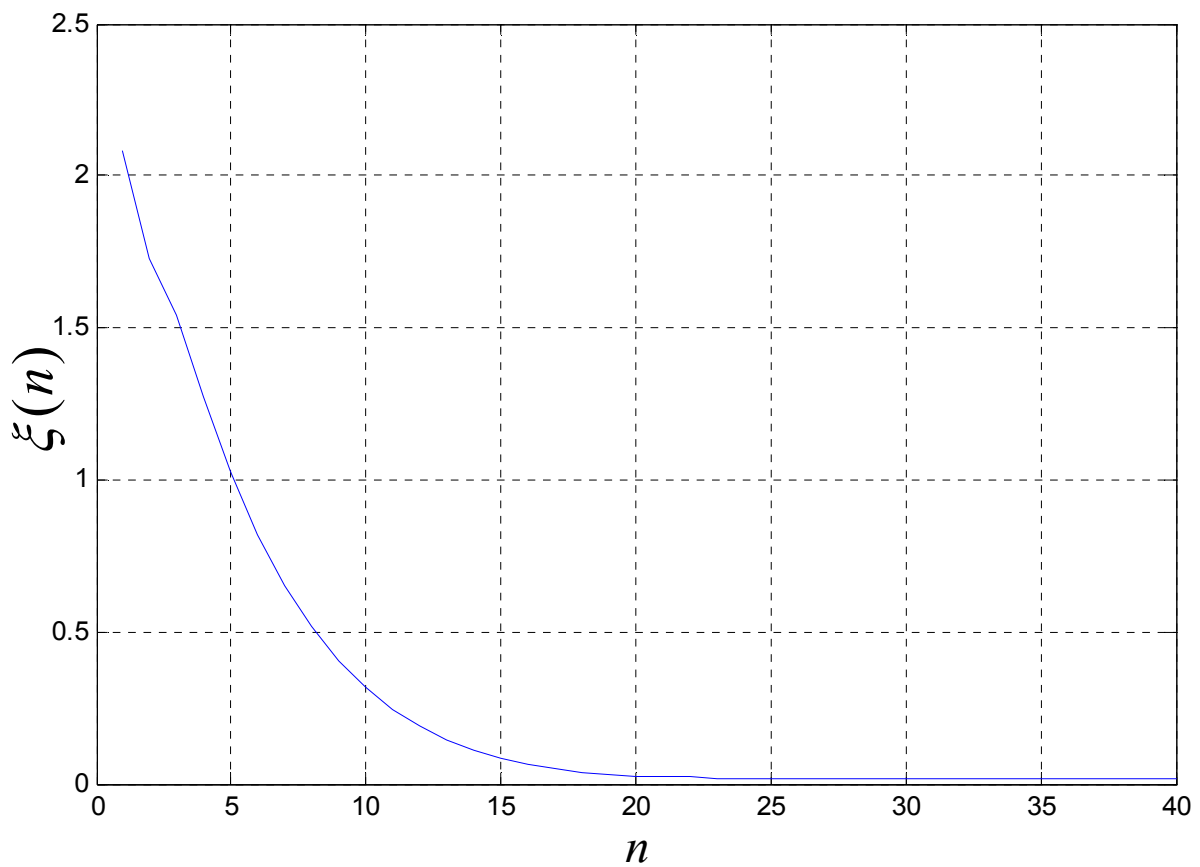


Figure 10: Standard deviation of timing vectors for rectangular network shown in Figure 9.

5.4 Star Network Topology

Figure 11 shows the star network topology used.

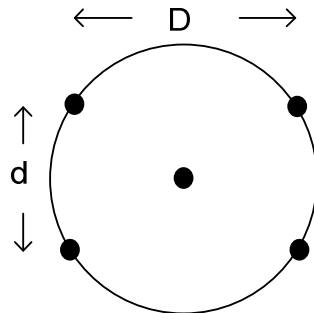


Figure 11: Star network topology.

Figure 12 shows the standard deviation of timing vectors for the triangular network topology shown in Figure 11, with initial timing vector = $[-3, -1.5, 0, 1.5, 3]$, $D = 4$, $d = 2$.

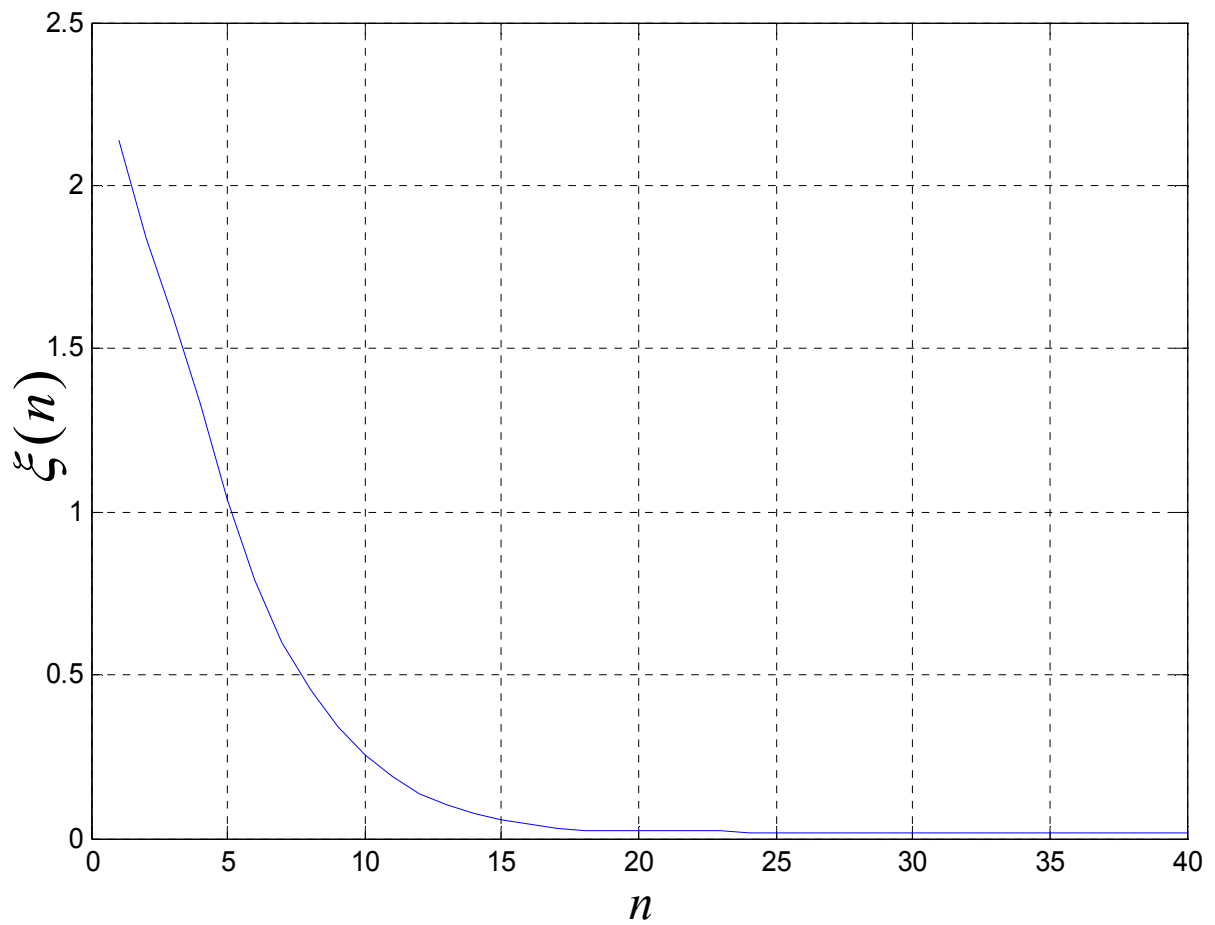


Figure 12: Standard deviation of timing vectors for star network shown in Figure 11.

5.5 Effect of Fading

Figure 13 shows that the proposed algorithm works even under fading. The absolute value of the PN code correlation value is used for the fading channel to avoid negative peaks in the correlation vector. The fading coefficients $\beta_{ki} = \beta_{ik}$ for $i \neq k$ and β_{ki} is constant for each trial simulation time, and expectation is taken with respect to the distribution of fading. The parameters of Jakes' fading coefficients are sampling frequency of 1,000 Hz and Doppler frequency of 100 Hz. Because it would be difficult to implement the Jakes' fading model for some realistic parameters, the above parameters are considered for generating fading coefficients.

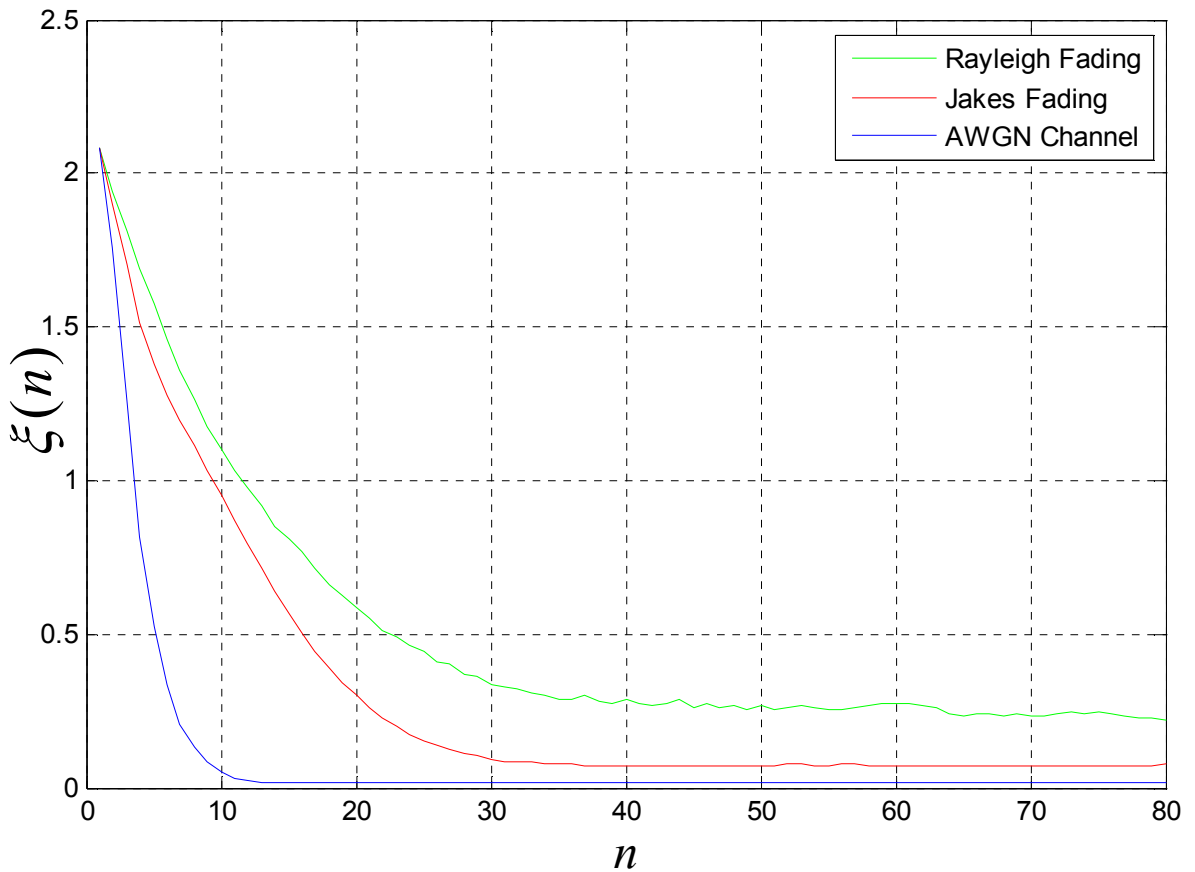


Figure 13: Comparison of standard deviation of timing vectors of different fading channels for linear network shown in Figure 7.

5.6 Comparison of Number of Samples per Symbol

Figure 14 compares the effect of a varying number of samples per symbol. As shown, it can be seen that the time to reach steady-state error is independent of the number of samples taken per symbol. The value of steady-state error increases marginally when the number of samples per symbol is decreased. The value of steady-state error is 0.002 when the number of samples per symbol is 100, and the value of steady-state error increases to 0.032 when the number of samples per symbol is decreased to 5. From these results, we can say that the proposed algorithm works even if the number of samples per symbol is small.

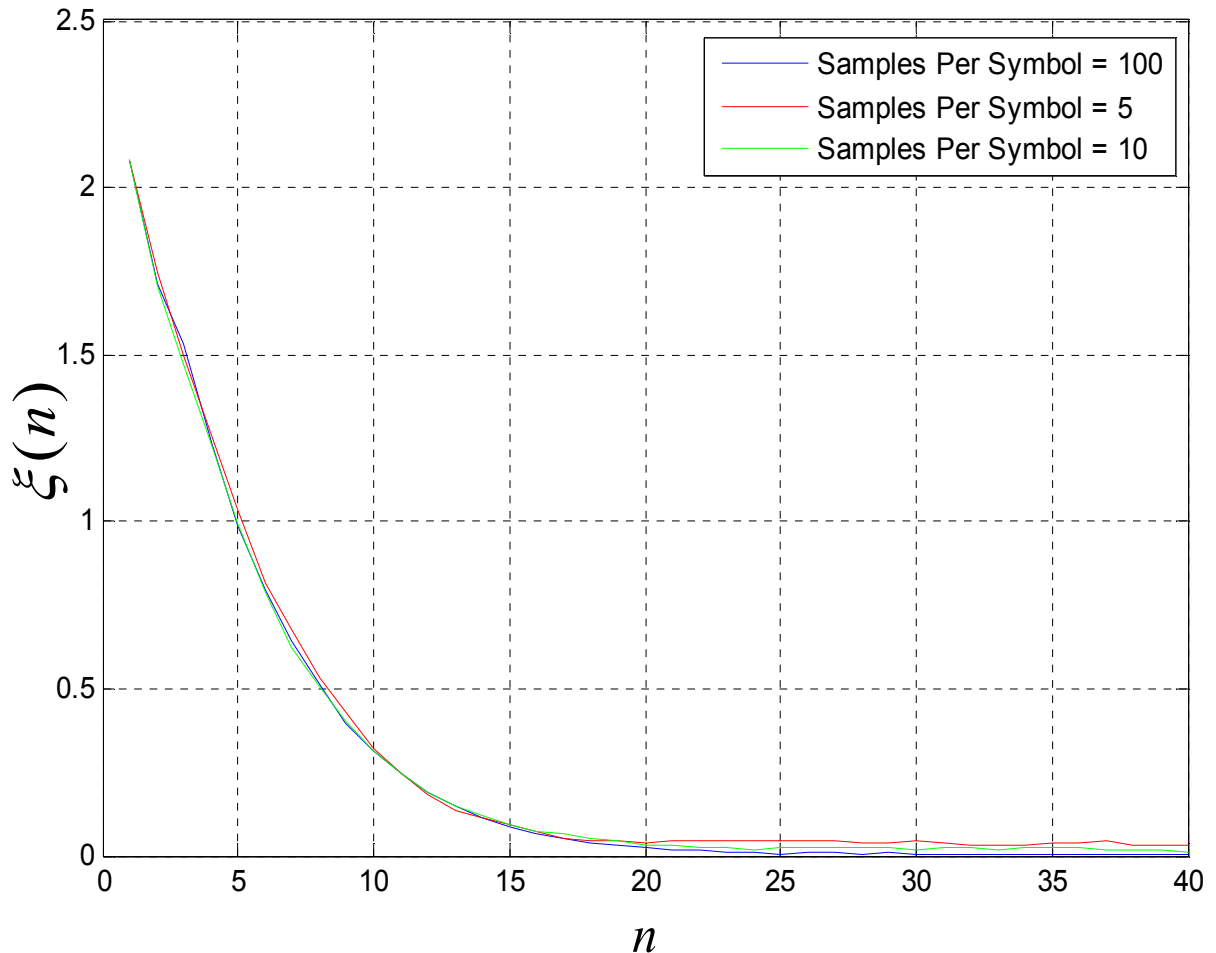


Figure 14: Comparison of number of samples taken per symbol for rectangular network shown in Figure 9 and AWGN channel.

5.7 Comparison of Discrete-Time Oscillator and PN Code Algorithms

In this section, comparisons are made to the existing discrete-time oscillator algorithm in the work of Simeone and Spagnolini [17] and the proposed PN code algorithms. Here, both algorithms are compared for AWGN channel and Rayleigh fading channel to show the effectiveness of the proposed algorithm over Simeone and Spagnolini's algorithm [17].

For comparison purposes, the simulation environment of the PN code algorithm was changed to make the observation interval of both algorithms equal. The simulation environment for the proposed algorithm is as follows:

Length of PN sequence = 7

PN sequence pattern = [0 0 1 1 1 0 1]

T_s (symbol time) = $\frac{1}{7}$ (i.e., $T_0 = 1$)

N_s (number of samples per T_s) = 14

Local PN code slides from -0.5 to 0.5 in steps of 1/98

Threshold = 0

γ (path loss exponent) = 3

SNR = $\left(\frac{E_{12}}{N_o}\right) = 5\text{dB}$.

The simulation environment for Simeone and Spagnolini's algorithm [17] is as follows:

$F_s = 0.01$

$L = 15$

$g(t)$ is raised cosine waveform with roll-off = 0.2

Threshold = 0

SNR = $\left(\frac{E_{12}}{N_o}\right) = 5\text{dB}$

$T_0 = 1$.

Comparison of Standard Deviation for AWGN Channel

Figure 15 compares the standard deviation of timing vectors when discrete-time oscillators and pseudo noise code are used and when the channel is AWGN. It can be seen that the number of iterations required to achieve a standard deviation of 0.1 when using the PN code algorithm is 16, whereas the number of iterations required to achieve the same standard deviation when using Simeone and Spagnolini's algorithm [17] is 32.

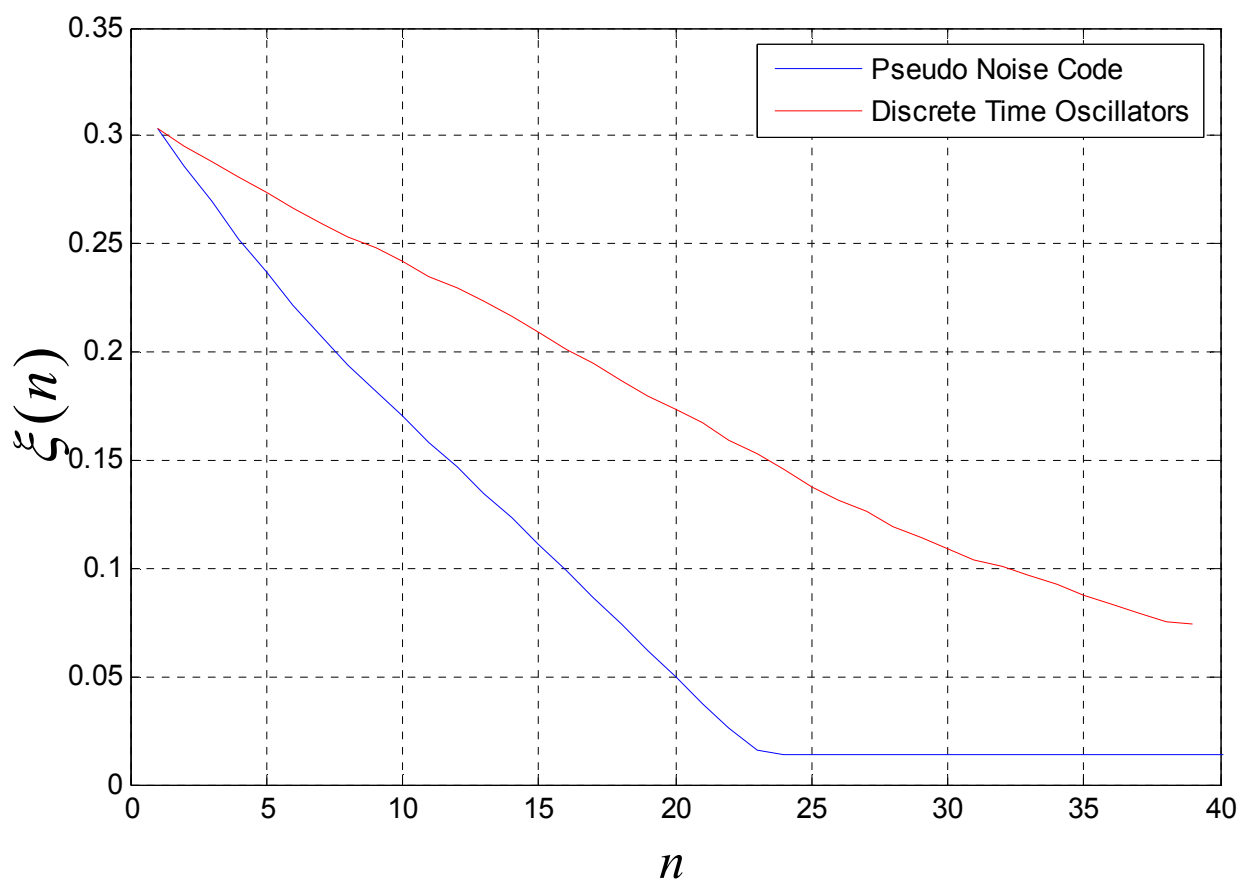


Figure 15: Comparison of standard deviation of timing vectors of discrete-time oscillator and PN code algorithms for rectangular network and AWGN channel.

Comparison of Standard Deviation for Rayleigh Fading Channel

Figure 16 compares the standard deviation of timing vectors when discrete-time oscillators and PN code algorithms are used for a Rayleigh fading channel. As shown, the number of iterations required to achieve a standard deviation of 0.25 when using the PN code algorithm is 6, whereas the number of iterations required to achieve the same standard deviation when using Simeone and Spagnolini's algorithm [17] is 49.

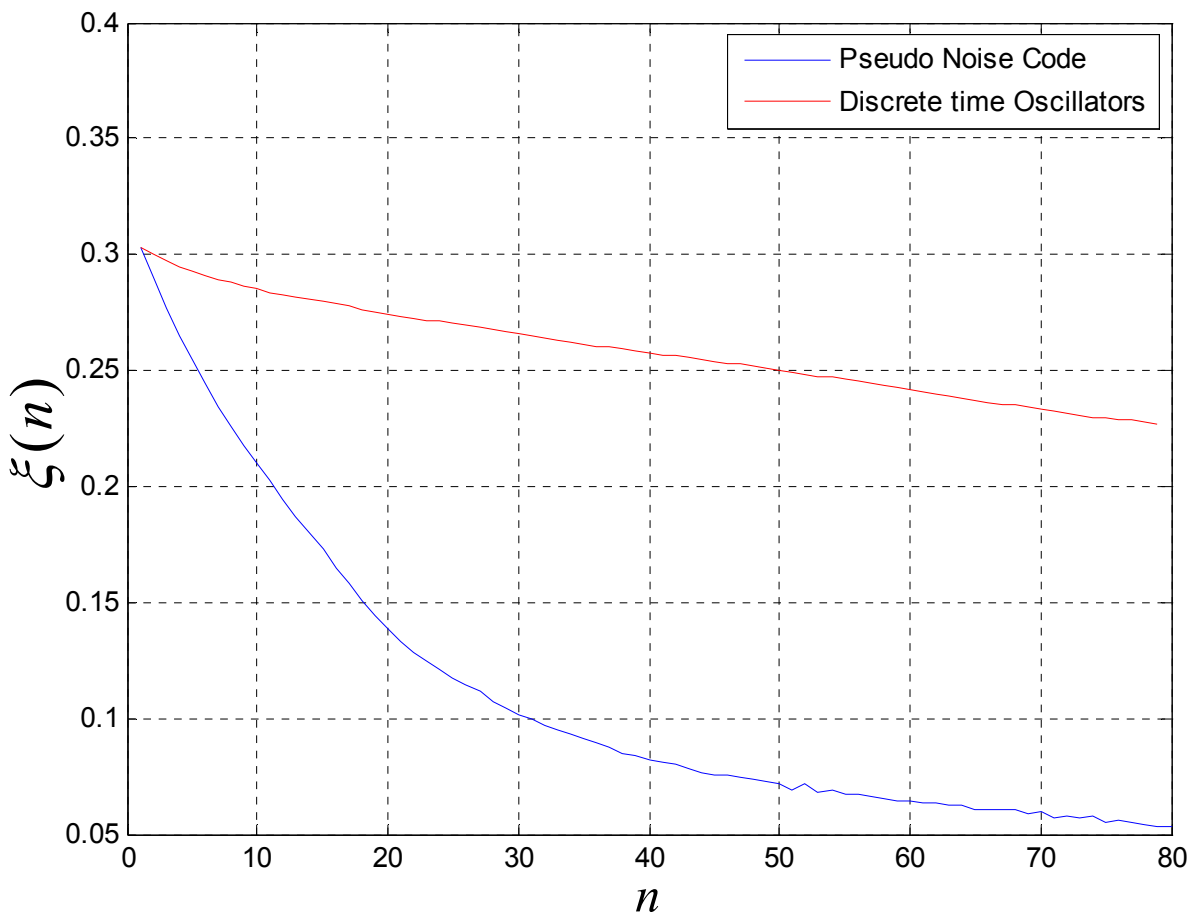


Figure 16: Comparison of standard deviation of timing vectors of discrete-time oscillator and PN code algorithms for rectangular network and Rayleigh fading channel.

CHAPTER 6

DISCUSSION

The advantages of this algorithm over the discrete-time-oscillator algorithm proposed by Simeone and Spagnolini [17] are as follows: (i) the pulse-shaping filter can be designed over a one-bit interval in the proposed algorithm, whereas the pulse-shaping filter for Simeone and Spagnolini's scheme [17] needs to be designed over a smaller interval, which is much less than the bit interval; (ii) the proposed scheme can even work under lower SNRs when compared to Simeone and Spagnolini's algorithm [17] because the correlation is taken by accumulating many samples over multiple bits or symbols, which reduces the effect of random noise, whereas Simeone and Spagnolini's algorithm [17] uses a single sample, and the effect of noise will be very high; (iii) the number of iterations required to reach a standard deviation of 0.1 in the PN code scheme is much less than Simeone and Spagnolini's algorithm [17]; (iv) the steady-state error is less in the PN code scheme when compared to Simeone and Spagnolini's algorithm [17]; and (v) the number of samples required per one update interval is much less in the proposed algorithm than Simeone and Spagnolini's algorithm [17]. The disadvantage of this algorithm over the one in the work of Simeone and Spagnolini [17] is that the computation complexity of the proposed algorithm increases because the correlation is calculated many times by sliding the local PN code at the k -th receiver.

CHAPTER 7

CONCLUSION

This thesis presented a PN code-based time synchronization algorithm and provided simulation results for different network topologies. This approach provides acceptable performance for different network topologies. For programming convenience, only three-node, four-node, and five-node scenarios are considered, but this can be extended easily to a greater number of nodes. This approach of synchronization is stable for different types of network topologies and can be used under different fading environments. From the simulation results, it can be seen that the proposed distributed network synchronization method performs significantly better than the one in the work of Simeone and Spagnolini [17].

REFERENCES

LIST OF REFERENCES

- [1] Fumiuyuki 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] 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.
- [3] H. Q. Ngo, E. G. Larsson, and T. Marzetta, "Energy and Spectral Efficiency of Very Large Multiuser MIMO Systems," arXiv:1112.3810, 2011.
- [4] S. K. Mohammed and E. G. Larsson, "Per-Antenna Constant Envelope Precoding for Large Multi-User MIMO Systems," arXiv:1201.1634v1, 2011.
- [5] C. Studer and E. G. Larsson, "PAR-Aware Large-Scale Multi-User MIMO-OFDM Downlink," arXiv:1202.4034, 2012.
- [6] 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.
- [7] 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.
- [8] 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.
- [9] 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.
- [10] Ram Zamir, Shlomo Shamai (Shitz), and Uri Erez, "Nested Linear/Lattice Codes for Structured Multiterminal Binning," *IEEE Transactions on Information Theory*, vol. 48, no. 6, pp. 1250–1276, June 2002.
- [11] 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.

List of References (continued)

- [12] Sumanth Jagannathan, Hamid Aghajan, Andrea Goldsmith, “The Effect of Time Synchronization Errors on the Performance of Cooperative MISO Systems,” *Transactions of IEEE GLOBECOM Workshops*, pp. 102–107, 2004.
- [13] R. C. Palat, A. Annamalai, and J. H. Reed, “Upper Bound on Bit Error Rate for Time Synchronization Errors in Bandlimited Distributed MIMO networks,” *Proceedings of IEEE Wireless Communications and Networking Conference WCNC 2006*, vol. 4, April 3–6, 2006, pp. 2058–2063.
- [14] F. Sivrikaya and B. Yener, “Time Synchronization in Sensor Networks: A Survey,” *IEEE Network*, vol. 18, no. 4, pp. 45–50, 2004.
- [15] Yutaro Kida, Keiichi Mizutani, Gia Khanh Tran, Takehiro Miyamoto, Kei Sakaguchi, and Kiyomichi Araki, “Network Synchronization Scheme for Scalable Two-Way Multi-Hop Network Employing MIMO Network Coding,” *IEEE 75th VTC*, Spring, 2012.
- [16] 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.
- [17] 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.
- [18] Guanghui Wen, Zhongkui Li, Zhisheng Duan, and Guanrong Chen, “Distributed Consensus Control for Linear Multi-Agent Systems with Discontinuous Observations,” *International Journal of Control*, vol. 86, no. 1, pp. 95–106, January 2013.