CODE AIDED FRAME SYNCHRONIZATION FOR FREQUENCY SELECTIVE CHANNELS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

UMUT UTKU EKİNCİ

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

MAY 2010

Approval of the thesis:

CODE AIDED FRAME SYNCHRONIZATION FOR FREQUENCY SELECTIVE CHANNELS

submitted by **UMUT UTKU EKİNCİ** in partial fulfillment of the requirements for the degree of **Master of Science in Electrical and Electronics Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan Özgen
Dean, Graduate School of **Natural and Applied Sciences** __________

Prof. Dr. İsmet Erkmen
Head of Department, **Electrical and Electronics Engineering** ______

Prof. Dr. Temel Engin Tuncer
Supervisor, **Electrical and Electronics Engineering Dept., METU** ______

Examining Committee Members

Assoc. Prof. Dr. Ali Özgür Yılmaz
Electrical and Electronics Engineering Dept., METU __________

Prof. Dr. Temel Engin Tuncer
Electrical and Electronics Engineering Dept., METU __________

Assis. Prof. Dr. Çağatay Candan
Electrical and Electronics Engineering Dept., METU __________

Dr. Arzu Tuncay Koç
Electrical and Electronics Engineering Dept., METU __________

Dr. Bekir Ahmet Doğrusöz
ASELSAN __________

Date: 28.05.2010

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name : Umut Utku EKİNCİ

Signature :

ABSTRACT

CODE AIDED FRAME SYNCHRONIZATION FOR FREQUENCY SELECTIVE CHANNELS

Ekinci, Umut Utku

M. Sc., Department of Electrical and Electronics Engineering

Supervisor: Prof. Dr. Temel Engin Tuncer

April 2010, 91 pages

Frame synchronization is an important problem in digital communication systems. In frame synchronization, the main task is to find the frame start given the flow of the communication symbols. In this thesis, frame synchronization problem is investigated for both additive white Gaussian noise (AWGN) channels and frequency selective channels. Most of the previous works on frame synchronization consider the simple case of AWGN channels. The algorithms developed for this purpose fail in frequency selective channels. There is limited number of algorithms proposed for the frequency selective channels. In this thesis, existing frame synchronization techniques are investigated for both AWGN and frequency selective channels. Code-aided frame synchronization techniques are combined with the methods for frequency selective channels. Mainly two types of code-aided frame synchronization schemes are considered and two new system structures are proposed for frame synchronization. One of the proposed structures performs better than the alternative methods for frequency selective channels. The overall system for this new synchronizer is composed of a list synchronizer which generates the possible frame starts, a channel estimator, a soft output

MLSE equalizer, and a soft output Viterbi decoder. A mode separation algorithm is used to generate the statistics for the selection of the true frame start. Several experiments are done and the performance is outlined for a variety of scenarios.

Keywords: Optimum Frame Synchronization, channel estimation, frequency selective channel, convolutional coding, code aided frame synchronization, soft output Viterbi equalizer, soft output Viterbi decoder.

ÖZ

FREKANS SEÇİCİ KANALLARDA KODLAMA YARDIMLI ÇERÇEVE SENKRONİZASYONU

Ekinci, Umut Utku

Yüksek Lisans, Elektrik ve Elektronik Mühendisliği Bölümü
Tez yöneticisi: Prof. Dr. Temel Engin Tuncer

Nisan 2010, 91 sayfa

Sayısal haberleşme sistemlerinde, çerçeve senkronizasyonu önemli bir problemdir. Çerçeve senkronizasyonunda başlıca görev verilen haberleşme sembolleri akışında çerçeve başlangıcını bulmaktır. Bu tezde, çerçeve senkronizasyon problemi, hem toplanır beyaz Gauss gürültüsü kanallarda hem de frekans seçici kanallarda incelenecektir. Çoğu çerçeve senkronizasyonu üzerine olan geçmiş çalışmalarında basit toplanır beyaz Gauss gürültüsü kanal koşulları dikkate alınmıştır. Bu amaçla geliştirilen algoritmalar, frekans seçici kanallarda başarısız olmaktadır. Frekans seçici kanallar için önerilen sınırlı sayıda algoritma vardır. Bu tezde varolan çerçeve senkronizasyon teknikleri hem toplanır beyaz Gauss gürültüsü kanalları hem de frekans seçici kanallar için incelenmektedir. Kodlama yardımlı çerçeve senkronizasyon teknikleri frekans seçici kanallar için olan metodlarla birleştirilmektedir. Başlıca iki çeşit kodlama yardımlı çerçeve senkronizasyon tasarımı dikkate alınmaktadır ve çerçeve senkronizasyonu için iki yeni sistem yapısı önerilmektedir. Önerilen yapılardan bir tanesi, frekans seçici kanallarda alternatif metodlardan daha iyi sonuç vermektedir. Bu yeni tip senkronizasyon yöntemi için genel olarak sistem, olası çerçeve başlangıç noktalarını üreten liste yapısında senkronizasyon bloğu, kanal kestirici, yumuşak çıktılı denkleştirici ve yumuşak çıktılı Viterbi çözücüden oluşmaktadır. Bir kip ayırıcı algoritması doğru çerçeve başlangıcını seçmek için istatistik üretmekte kullanılmaktadır. Çeşitli deneyler yapılmıştır ve performans çeşitli senaryolar için ana hatlarıyla belirtilmiştir.

Anahtar Kelimeler : Optimum Çerçeve Senkronizasyonu , frekans seçici kanallar, kanal kestirimi, evrişimsel kodlama, kodlama yardımlı çerçeve senkronizasyonu, yumuşak çıktılı Viterbi denkleştirici, yumuşak çıktılı Viterbi çözücü.

ACKNOWLEDGEMENT

I would like to thank to my supervisor Prof. Dr. Temel Engin Tuncer for his guidance.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

BER : bit error rate

SW : synchronization word

SNR : signal to noise ratio

BPSK : binary phase shift keying

AWGN : additive white gaussian noise

ML : maximum likelihood

LLF : log-likelihood function

MSCEE : mean square channel estimation error

MAP : maximum a-posteriori

APP : a posteriori probability

PDF : probability density function

ISI : intersymbol interference

PSK : phase shift keying

QAM : quadrature amplitude modulation

FAP : false frame acquisition probability

MLSEEQ: maximum likelihood sequence equalizer

CSI : channel side information

HPL : high probability list

HCFS : high complexity frame synchronizer

CAFS1 : code aided frame synchronizer 1

CAFS2 : code aided frame synchronizer 2

CHAPTER 1

INTRODUCTION

Frame synchronization is the problem of locating the frame starting point at the receiver. It is one of the essential functions of a communication system that undertakes the establishment of a communications link. Most communication systems employ one of the two basic approaches to packet data transmission; burst or continuous. In each case the problem of frame synchronization needs to be handled differently. Two basic differences for frame synchronization for burst and continuous transmission modes are the available observation interval and the a-priori knowledge relevant to the frame start location. In continuous mode, it is assumed that the frame could start anywhere within the observation period and the length of the observation period is equal to the length of the whole frame which includes the length of both the training and data [4,5]. By processing these observations, the receiver can locate the frame boundaries between the frames for a better bit error rate (BER) performance. BER performance is directly related to the frame acquisition performance.

In burst mode [13], the observation interval is limited to the training sequence which is shorter than that in the continuous mode case [4,5]. For burst mode packet transmission finding the frame starting point can be even more critical because the cost of retransmitting a lost packet can be quite high. In either cases, frame synchronization is an essential component for a reliable digital communication system.

Conventional frame synchronization techniques have been quite adequate for most legacy communication systems, since the synchronization performance need only be as good as the BER performance of the system, and the synchronization overhead of conventional techniques might be negligible from a practical viewpoint. However, recent, powerful techniques for error control coding have radically improved realizable BER performance of modern communication systems, leaving synchronization performance as the immediate bottleneck. Since the improvement of coding techniques has enabled reliable communication at very low signal to noise ratio (SNR) regions, the necessity for better frame synchronization algorithms appears.

For practical cases the most widely used method for frame synchronization is the simple correlation rule. It is easy to implement and has a little complexity. However, it is not the optimum method for finding the frame starting point.

Massey has derived the optimum maximum likelihood (ML) rule for coherent binary phase shift keying (BPSK) over the AWGN channels [1]. The ML rule derived by Massey performs 3 dB better than the simple correlation rule. He also derived the low complexity approximations of the optimum rule for high SNR and low SNR regions. Nielsen [3] subsequently reported additional results on the performance of Massey's rule. He showed that the high SNR approximation of the ML rule yields a performance indistinguishable from the optimum rule. The high SNR rule is only slightly more complicated to implement than the correlation rule. These results show that the performance of the frame synchronizer can be improved for AWGN channels at the expense of very little increase in complexity.

Lui and Tan carried forward the Massey's pioneering work for coherent and non-coherent M-ary signaling over the AWGN channel. An analytical lower bound on synchronization probability performance for the coherent correlation rule was derived [2] for the purpose of performance evaluation. Both Nielsen and Lui and Tan used the random data limited upper bound to aid in performance evaluation. Random data limited upper bound gives the maximum performance that can be obtained for a given modulation order, packet length and synchronization word length regardless of the distortion and noise. Note that even when there is no channel noise and no distortion, the frame synchronization word can be seen in the random data sequence which leads the receiver to a false synchronization. The false synchronization due to random data limits the performance of the communication system.

Despite the fact that the problem of frame synchronization for frequency selective channels is different from that of AWGN channels, it has received much less attention than for AWGN channels. Some examples of these works are given in [6-8]. In [6], continuous mode algorithm for joint frame synchronization, channel and frequency offset estimation for frequency selective channels are derived based on the maximum likelihood criterion. In [7], data aided and non data aided algorithms that allow to estimate the frame starting point over a frequency selective channels are considered. In [8], an iterative frame synchronization technique for frequency selective channels has been proposed. The uncoded data at the output of the equalizer is utilized in order to have a better estimate of the frame start, leading to a better estimate of the channel taps. In simulation results it is shown that considerable gain is achieved in terms of false acquisition probability, mean square channel estimation error (MSCEE) and bit error rate (BER). These results obtained by Samad [8], give hope about better frame acquisition probability for frequency selective channels when coding is employed in the system. In frequency selective channels, if the estimation of the channel parameters is not successfull, the decoders' BER performance suffers because of the bad performance of the equalizer.

Robertson [9], proposed a general two stage structure which can be applied to various systems that employ coding. The first stage generates a list of most probable frame starting points. The function of the next stage following the frame sync unit is to distinguish incorrect from correct positions. Previously derived random data limited upper bound is extended for a list of several alternative frame starting positions. The results obtained by Robertson are derived for AWGN channels but the proposed structure is general. These results are used as a criterion for the evaluation of the performance of the frame synchronizers that are examined thorughout the thesis. In this thesis, the frame synchronizers which use this kind of structure, are called list synchronizers.

There are code aided frame synchronization methods in the literature that are developed based on this list synchronizer concept. We considered two of these list synchronizer type frame synchronizers which are employed coding in their system. The first method is proposed by Howlader in [10]. Generally the synchronization word is left uncoded and sent to the channel as the way it is. But Howlader's approach is slightly different from this traditional approach. He proposed to embed the sync word into the data and encode the sync word as it is a part of the data sequence. By doing this, the sync word is protected by the coding employed in the system. The second code aided frame synchronization method which has been considered in the scope of this thesis is proposed by Cassaro and Georghiades [11]. They also employed the list sychronization concept in their scheme. The synchronization word is left uncoded. In the receiver, after an initial synchronization is performed, mode separation algorithm that exploits the frame likelihood statistic is employed for the purpose of distinguishing synchronized frame from unsynchronized ones.

Throughout the thesis, existing frame synchronization methods are investigated which are developed for both AWGN channels and frequency selective channels. Cassaro's method of code aided frame synchronization and Howlader's method of frame synchronization method have been combined with the Wang's [6] and Samad's [8] method of frame synchronization methods which are developed for frequency selective channels. Mainly two new code aided frame synchronization structures are proposed. The performances of the proposed structures have been determined for various system parameters and channel conditions and compared with the existing frame synchronization methods.

The thesis is organised as follows: In Chapter 2 and in Chapter 3 frame synchronization techniques which are developed for additive white gaussian noise and frequency selective channels, respectively, have been introduced. In Chapter 4 mainly two types of code aided frame synchronization methods which are developed for AWGN channel are introduced. In Chapter 5 two new code aided frame synchronization methods are proposed for frequency selective channels based on the techniques which are discussed in Chapter 4. In Chapter 6, the results of numerical simulations of existing and proposed methods have been determined and the proposed methods have been compared with the existing methods. Finally, in Chapter 7, the thesis is concluded and possible future work is mentioned.

CHAPTER 2

FRAME SYNCHRONIZATION TECHNIQUES FOR AWGN CHANNEL

In this chapter, the optimum frame synchronization methods are derived for several modulation schemes. The principle of these synchronizers is to evaluate likelihood function for each possible starting position of the sync word [1]. The likelihood function is a measure for the probability that the true starting position of the sync word was at the corresponding position in the observed sequence. The derived inequalities can be approximated to derive sub-optimal likelihood functions.

The derived optimum rule in [1] outperforms the traditional correlation rule. But still both methods suffer from random data upper limit [3],[2]. In this chapter, the derivations of the optimum frame synchronizations rule are explained for AWGN. In Section 2.1, an introduction to detection is given. In Section 2.2, optimal frame synchronization for AWGN channels is explained.

2.1 Introduction to Detection

The optimal synchronizers are derived from the MAP principle. In this section the concepts of detection, estimation, maximum a-posteriori and maximum likelihood detection are briefly introduced.

Estimation and detection theory are based on the concept of risk or cost minimization. The goal is now to reduce the average risk, and the so-called Bayes test in a detection problem is a decision rule that minimizes the average risk.

Decision will be based on the observation *r* which is a realization of a random variable **r.** The **ML rule** can be written as

Select A, that maximizes

$$\Pr\{\hat{A} = A \mid \mathbf{r} = r\} \tag{2-1}$$

We should find the maximum of the probability of $\hat{A}$ given that we observed r . Using the Bayes rule, we can write,

$$\Pr\left\{\tilde{A} = A \mid \mathbf{r} = r\right\} = f_r(r \mid \tilde{A} = A)\frac{\Pr\{\tilde{A} = A\}}{f_r(r)}, \tag{2-2}$$

and ignoring the term $f_r(r)$ which will not affect our detection outcome, the ML rule can be rewritten as

$$\tilde{A} = \arg\max_{A} f_r(r \mid \tilde{A} = A) \tag{2-3}$$

There may be some other unknown event(s) that are present in our model which also influence observation **r**. For example, we might only be able to determine the PDF

$$f_r(r \mid \tilde{A} = A, \tilde{B} = B) \tag{2-4}$$

for all possible *B*. If we are interested in $\tilde{B}$ as well, then our search will become a two-dimensional one (over *A* and *B*). If we are not interested in detecting $\tilde{B}$ at all, and $\hat{A}$ and $\tilde{B}$ are statistically independent, one can simply integrate over all possible B, yielding

$$f_r(r \mid \tilde{A} = A) = \sum_{\forall B} f_r(r \mid \tilde{A} = A, \tilde{B} = B).\Pr\{\tilde{B} = B\} \tag{2-5}$$

In a technical environment we are often only interested in the outcome of the maximization process, so it is not necessary to maximize over the PDFs themselves, but instead over monotonically increasing function of the PDF. For example, it is possible to take the logarithm of the PDF; the resulting function A is often referred to as a log-likelihood function. It can be adopted to the simple notation of calling any monotone increasing function of the PDF a likelihood function, and it is going to be denoted by L(A) in the other sections.

2.2 Optimal Frame Synchronization for AWGN Channel

Massey showed that the optimum rule has a 3-dB improvement compared to a pure correlation rule. Lui and Tan extend this rule and its high SNR approximation for M-ary phase coherent and phase non-coherent signaling over the additive white Gaussian noise channel.

Derivation of the Optimal Sync-Word Locating Rule [2]

Data transmission is formatted in successive frames, each of which consists of **N M**-ary data symbols where the first N_s symbols are fixed frame synchronization sequence, and the remaining N - N_s symbols are random data. Let $\{W_j,\ 1 \le j \le M\}$ be the set of signal vectors; we may then equivalently consider each frame as consisting of N signal vectors

$\boldsymbol{S}_0$, $\boldsymbol{S}_1$, . . . , $\boldsymbol{S}_{Ns-1}$, $\boldsymbol{d}_{N\,s}$, . . . , $\boldsymbol{d}_{N-1}$ corresponding to the sequence of N data symbols. The first N_s vectors $\boldsymbol{S}_0$, $\boldsymbol{S}_1$, . . . , $\boldsymbol{S}_{Ns-1}$ represent the fixed frame synchronization sequence and the $N - N_s$ vectors $\boldsymbol{d}_{Ns}$, . . . , $\boldsymbol{d}_{N-1}$ represent the random data portion of the frame. It is assumed that each data vector is chosen equally likely from the M-ary signal vector set .

The noise vectors are mutually statistically independent, and the noise vector is independent zero mean Gaussian process with variance $N_0/2$ where N_0 is the one sided noise power spectral density.

Assuming that the symbol boundaries are known, it can be shown that the sufficient statistics are the sequence of demodulator output vectors $\boldsymbol{x} = (\boldsymbol{x}_0 , \boldsymbol{x}_1 , \ldots , \boldsymbol{x}_{N-1})$ corresponding to the arbitrarily selected span of N transmitted signal vectors. The observations for the frame synchronization problem can be thus taken to be $\boldsymbol{x}$. These N vectors comprising $\boldsymbol{x}$ are conditionally independent Gaussian vectors given the $N - N_s$ data vectors. The frame synchronization problem is to determine $\hat{\mu}$, $0 \leq \hat{\mu} \leq N - 1$, to maximize the likelihood that $(x_{\hat{\mu}}, x_{\hat{\mu}+1}, ..., x_{\hat{\mu}+Ns-1})$ are the corresponding N_s demodulator outputs to the frame synchronization pattern ($\boldsymbol{S}_0$, $\boldsymbol{S}_1$, . . . , $\boldsymbol{S}_{Ns-1}$).

2.2.1 Maximum Likelihood Estimate $\hat{\mu}$ for M-ary Coherent Phase Signaling [1]

Maximum likelihood estimate $\hat{\mu}$ is the integer μ in [0,1, ... , N -1] that maximizes the conditional probability density, $p(\mathbf{x}|\mu)$, of the coherent demodulator output sequence $\mathbf{x}$ given the μ th vector in $\mathbf{x}$ is the starting position of the synchronization pattern.

Let $\boldsymbol{d} = (\boldsymbol{d}_{Ns}, \ldots , \boldsymbol{d}_{N-1})$ be the $(N - N_s)$ signal vectors corresponding to the random data portion of the frame and denote by $p(\mathbf{x}|\mu,\boldsymbol{d})$ the conditional probability density of $\mathbf{x}$ given μ and $\boldsymbol{d}$. Since the $M^{(N - Ns)}$ possible sequence of $N - N_s$ data signal vectors are equally probable we have

$$
\begin{aligned}
p(\mathbf{x} \mid \mu) &= \sum_{all\ \mathbf{d}} p(\mathbf{x} \mid \mu, \mathbf{d}) M^{-(N-Ns)} \\
&= M^{-(N-Ns)} \sum_{all\ \mathbf{d}} \prod_{i=0}^{Ns-1} (\pi N_o)^{-K/2} \exp\left[\frac{-\| x_{i+\mu} - S_i \|^2}{N_o}\right] \qquad (2\text{-}6) \\
&\quad \cdot \prod_{i=Ns}^{N-1} (\pi N_o)^{-K/2} \exp\left[\frac{-\| x_{i+\mu} - d_i \|^2}{N_o}\right]
\end{aligned}
$$

where the sum is over all possible sequences $\boldsymbol{d}$ of $(N - N_S)$ signal vectors and $\| \,.\, \|$ denotes the Euclidean norm.

The maximum-likelihood approach is to choose μ to maximize $p(\mathbf{x}|\mu)$. Eliminating quantities independent of μ from (2-6) and denoting the inner product by $\langle , \rangle$, one can equivalently maximize

$$
L_1(\mu) = \prod_{i=0}^{Ns-1} \exp\left[\frac{2}{N_o}\left\langle x_{i+\mu}, S_i \right\rangle\right] \cdot \sum_{all\ \mathbf{d}} \prod_{i=Ns}^{N-1} \exp\left[\frac{2}{N_o}\left[\left\langle x_{i+\mu}, S_i \right\rangle - \frac{\| d_i \|^2}{2}\right]\right]. \qquad (2\text{-}7)
$$

Next, observe that the sum

$$
\begin{aligned}
&\sum_{all\ \mathbf{d}} \prod_{i=Ns}^{N-1} \exp\left[\frac{2}{N_o}\left[\left\langle x_{i+\mu}, d_i \right\rangle - \frac{\| d_i \|^2}{2}\right]\right] \\
&\quad = \prod_{i=Ns}^{N-1} \sum_{j=1}^{M} \exp\left[\frac{2}{N_o}\left[\left\langle x_{i+\mu}, W_j \right\rangle - \frac{\| W_j \|^2}{2}\right]\right],
\end{aligned} \qquad (2\text{-}8)
$$

since the signal vectors are chosen independently from the set $\{W_j,\ 1 \le j \le M\}$. Also note that the product

$$
\prod_{i=Ns}^{N-1} \sum_{j=1}^{M} \exp\left[\frac{2}{N_o}\left[\left\langle x_{i+\mu}, W_j \right\rangle - \frac{\| W_j \|^2}{2}\right]\right], \qquad (2\text{-}9)
$$

is independent of μ because of the periodic frame boundaries. It is therefore equivalent to maximize (2-7) divided by (2-9). A simple rearrangement of the resulting expression leads to the conclusion that the maximum-likelihood estimate $\hat{\mu}$ is the value of μ in [0, 1, . . . , $N-1$] which maximizes

$$L(\mu) = \sum_{i=0}^{Ns-1} \left\langle x_{i+\mu}, S_i \right\rangle - \frac{N_0}{2} \sum_{i=0}^{Ns-1} \ln\left[\sum_{j=1}^{M} \exp\left[\frac{2}{N_0} \left[\left\langle x_{i+\mu}, W_j \right\rangle - E_j / 2 \right] \right] \right], \tag{2-10}$$

where $E_j = \left\| W_j \right\|^2$ is the energy of the signal vector W_j, $1 \le j \le M$. The equation (2-10) is referred as the maximum likelihood decision rule for the frame synchronization problem for coherent phase M-ary signaling over the AWGN channel. The first term in (2-10) is the ordinary correlation rule and the second term compansates for the effect of random data surrounding the frame synchronization pattern. For equal energy signal vectors, the exponent $E_j/2$ may be deleted from (2-10) without affecting maximization. It must be noted that the ML decision rule reduces to Massey's ML rule for coherent BPSK signaling ($W_1 = +\sqrt{E}$ and $W_2 = -\sqrt{E}$) over AWGN channel [1].

2.2.2 High SNR Approximation of the ML rule [2]

In the limit of large average SNR = E_{avg}/N_0 where

$$E_{avg} = \frac{1}{M} \sum_{j=1}^{M} E_j , \tag{2-11}$$

E_{avg} is the average energy of the signal set {Wj, $1 \le j \le M$}. For a particular value of i, $0 \le i \le N_s - 1$, consider the sum in (2-10)

$$\sum_{j=1}^{M} \exp\left[\frac{2}{N_0} \left[\left\langle x_{i+\mu}, W_j \right\rangle - E_j / 2 \right] \right], \tag{2-12}$$

which can be rewritten as

$$\sum_{j=1}^{M} \exp\left[\frac{2E_{avg}}{N_0}\left[\frac{1}{E_{avg}}\left\langle x_{i+\mu}, W_j \right\rangle - E_j / 2E_{avg}\right]\right]. \tag{2-13}$$

And for large SNR, (2-9) can be approximated by

$$\exp\left[\frac{2E_{avg}}{N_0}\left[\left\langle x_{i+\mu}, W_{\hat{j}(i)} \right\rangle / E_{avg} - E_{\hat{j}(i)} / 2E_{avg}\right]\right], \tag{2-14}$$

where $\hat{j}(i)$ is the value of j in [1,2, . . . , M] which achieves the following maximum

$$\max_{i \le j \le M} \left\langle x_{i+\mu}, W_j \right\rangle - E_j / 2. \tag{2-15}$$

If the signal vectors are of equal energy, the term $E_j/2$ can be omitted from the maximization (2-15). Using the approximation (2-14) for (2-13) in (2-10) gives the high SNR decision rule where $\hat{\mu}$ is chosen to maximize

$$L_H(\mu) = \sum_{i=0}^{Ns-1} \left\langle x_{i+\mu}, S_i - W_{\hat{j}(i)} \right\rangle - E_{\hat{j}(i)} / 2. \tag{2-16}$$

Note that if the signal vectors are of equal energy, the term $E_{\hat{j}(i)} / 2$ is independent of i and can also be omitted from (2-16) without affecting maximization. For coherent BPSK signaling where M=2, K=1, $W_1 = +\sqrt{E}$ and $W_2 = -\sqrt{E}$, the maximum correlation $\left\langle x_{i+\mu}, W_{\hat{j}(i)} \right\rangle$ is simply the absolute value, or norm, of the demodulator output, $\left\| x_{i+\mu} \right\|$. In this case (2-16) reduces to the following simple form:

$$L_H(\mu) = \sum_{i=0}^{Ns-1} S_i x_{i+\mu} - \sum_{i=0}^{Ns-1} \left\| x_{i+\mu} \right\|. \qquad (2\text{-}17)$$

Equation (2-17) is the approximate optimum rule for the BPSK signaling and derived by Massey.

ML decision rule provides optimum correct synchronization probability performance but it is computationally complex to implement. The high SNR ML rule is as easily implementable as the simple correlation rule, and appears to perform nearly as well as the ML rule.

2.2.3 The Random Data Limited Upper Bound

Synchronization probability performance is affected not only by channel noise but is also fundamentally limited by random data. The fact that the frame synchronization sequence can be replicated in certain subsequences of the random data sequence ensemble implies that false synchronization on data can occur even when the channel is noiseless. Consider for example, an $\boldsymbol{M}$-ary data noise free channel in which a length $\boldsymbol{N_s}$ synchronization sequence $\boldsymbol{S}$ is inserted every $\boldsymbol{N}$ symbols and a frame synchronizer that selects one at random among all positions where $\boldsymbol{S}$ may occur in an $\boldsymbol{N}$ symbol received sequence. For given values of $\boldsymbol{N}$, $\boldsymbol{N_s}$ and $\boldsymbol{M}$, the resulting probability of correct synchronization depends only on the structure of the frame synchronization sequence S. A good sync pattern S should have the desirable property that its prefixes and suffixes of the same length are distinct, so that the number of replications in the random data is minimized . In [2], the following expression for the resulting synchronization probability performance, P_{RDL} , is shown:

$$P_{RDL} = \sum_{i=0}^{[N/Ns-1]} \frac{(-1)^i}{(1+i)} \binom{N-Ns-(Ns-1)i}{i} M^{-Nsi}. \qquad (2\text{-}18)$$

P_{RDL} is called the random data limited upper bound on synchronization probability of any decision rule used over an M-ary noisy channel, it can be used as a benchmark for comparing the performance of the frame synchronization decision rules developed before.

A random data limited upper bound [18] on synchronization probability performance is valid for any decision rule. In simulations, it is seen that performance curves can not exceed the corresponding upper bound.

CHAPTER 3

FRAME SYNCHRONIZATION TECHNIQUES FOR FREQUENCY SELECTIVE CHANNELS

Frame synchronization problem has been studied extensively for AWGN channels, but for frequency selective channel environment, it has received much less attention. In the literature there are techniques [19], which use ML-based frame synchronizer assuming a binary pulse modulated system and known static dispersive ISI (intersymbol interference) channels. However in many cases, the channel coefficients are unknown and the channel is time varying.

Generally, two possible approaches may be derived to estimate the frame boundary according to *ML criterion* in the presence of other unknown variables like channel coefficents. First one is the *Bayesian* approach which models other unknown variables as random variables with certain pdf and computing the average of joint likelihood function with respect to their pdfs to produce the marginal likelihood of the frame starting point, from which the ML estimate of frame starting point can be obtained [20]. Second method estimates the frame boundary and other unknown variables jointly.

In this chapter, a computationally efficient synhronization scheme for joint frame synchronization and channel acquisition by exploiting the *ML rule* will be explained [6]. In Section 3.1 system model is introduced. Section 3.2 describes an ML scheme for joint frame synchronization and channel estimation. In Section 3.3, the algorithm is extended for the multiple frame synchronization case. In Section 3.4, iterative frame synchronization concept is given for frequency selective channels.

3.1 System Model

Data is transmitted through a slow time-varying frequency-selective channel and the modulation type is linear (e.g., PSK or QAM). We assume throughout the formulation that the channel coefficients are constant during the observation interval.

$\mathbf{h} = [h_0, h_1, \ldots, h_{L-1}]^T$, L represents the channel memory.

The frame consists of N symbols total including data and synchronization word where the first N_s symbols form a synchronization pattern $\mathbf{s} = [s_0, s_1, \ldots, s_{Ns-1}]^T$ followed by $P := N- Ns$ data symbols $\mathbf{d} = [d_0, d_1, \ldots, d_{P-1}]^T$.

Data symbols d_k are assumed to be zero-mean i.i.d. with unit average energy per symbol, i.e., $\sigma_k^2 = E\{|d_k|^2\} = 1$, and the training symbols s_k are selected from the same set as that of data symbols d_k, so that no restriction is made on the frame structure to prohibit the replication of the frame synchronization pattern in the portion of random data..

It is generally desirable to choose a sync word with good autocorrelation property satisfying the condition $(s_0, s_1, \ldots, s_{j-1}) \neq (s_{Ns-j}, s_{Ns-j+1}, \ldots, s_{Ns-1})$, $j = 1, 2,\ldots,N_s - 1$ which ensures the number of replications of the sync word in the random data to be minimized.

The transmitted signal is passed through the channel $\mathbf{h}$, and sampled at the symbol period . The outputs of the channel corresponding to the i-th frame are modeled as

$$x_k^{(i)} = \sum_{l=0}^{L-1} w_{k-l} h_l, \qquad k = 0,1,\ldots,N-1, \tag{3-1}$$

where

$$w_k = \begin{cases} d_{P+k}^{(i-1)}, & if\ -(L-1) \le k \le -1, \\ s_k, & if\ 0 \le k \le N_s - 1, \\ d_{k-N_s}^{(i-1)}, & if\ N_s \le k \le N-1 \end{cases}$$

Based on (3-1), $x_0^{(i)}$ $(-\infty < i < \infty)$, is defined as the frame boundaries in the channel outputs, where the first training symbol s_0 is involved in the first path h_0 of the channel.

The location of x_0 is not known hence , the received signal is a linear shift of the x_k sequence with an unknown delay $\mu \in [0, 1, \ldots, N-1]$. The frame starting location may appear in any of the N positions with equal probability in a received sequence which has length N, $\mathbf{r} = [r_0, r_1, \ldots, r_{N-1}]^T$. It is shown in Figure 3.1.

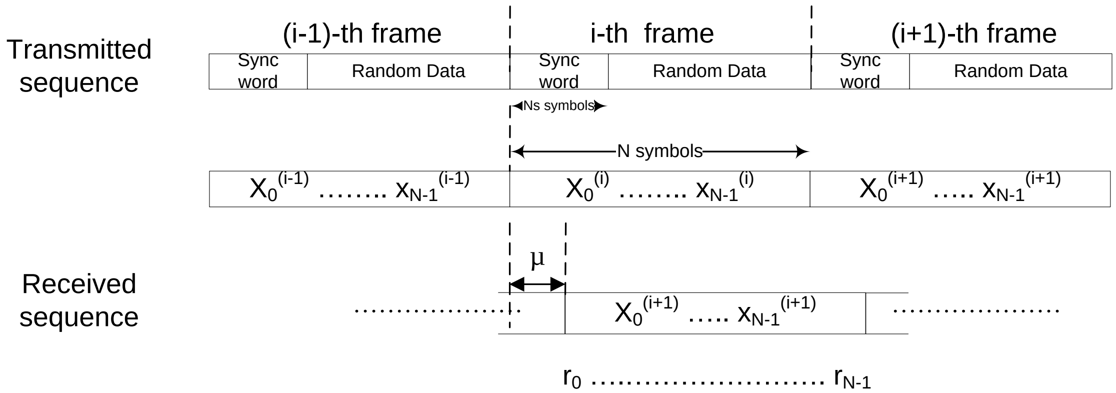


Figure 3. 1. Transmitted and Received sequence of the model

Let $T(\mathbf{x}^{(i)}) = [x_1^{(i)}, x_2^{(i)}, ..., x_{N-1}^{(i)}, x_0^{(i+1)}]^T$, be the linear shift operator. Then we can express the received segment as

$$\mathbf{r} = T^{\mu}(\mathbf{x}^{(i)}) + \mathbf{n}, \qquad (3\text{-}2)$$

Where $\mathbf{n} = [n_0, n_1, \ldots, n_{N-1}]^T$ and the components n_k are independent complex Gaussian random variables with zero-mean and variance N_0.

3.2 Joint Frame Synchronization and Channel Acquisition

The optimum MAP algorithm maximizes the *posterior* probability,

$$\Lambda_1 = p(m \mid r), \qquad 0 \le m \le N-1,$$

where

$$\mathbf{r} = T^{\mu}(\mathbf{x}^{(i)}) + \mathbf{n}$$

The Bayes' theorem becomes $\Lambda_1 = f(\mathbf{r}|m, d_m)\, p\,(m\,|r) / f(\mathbf{r})$, where $f(\mathbf{r})$ stands for the pdf of $\mathbf{r}$. Since p(m) = 1/N for all m, the MAP algorithm reduces to the ML estimator which maximizes

$$\Lambda_2 = f(\mathbf{r}\,|\,m) = \sum_{all\ d_m} f(\mathbf{r}\,|\,m,\mathbf{d}_m)\cdot p(\mathbf{d}_m) = \sum_{all\ d_m} \frac{1}{(\pi N_0)} e^{-\frac{1}{N_0}(\mathbf{r}-T^m(\mathbf{x}))^H.(\mathbf{r}-T^m(\mathbf{x}))} .p(\mathbf{d}_m), \quad (3\text{-}3)$$

'$\mathbf{d}_m$' represents the set of unknown data involved in the operation $T^m(x)$. The probability $p(\mathbf{d}_m)$ depends on the size of the symbol alphabet and the synchronization position '*m*'. The averaging over all possible data vectors of length varying with '*m*', whose complexity increases exponentially with the channel memory is complicated enough hence the optimal estimator (3-3) is computationally prohibitive. To circumvent this diffuculty, Wang and Serpedin [6] proposed a suboptimal algorithm, which does not necessitate the averaging over the unknown data.

One can observe that the subset $x_s := [x_{L-1}, x_L, \ldots, x_{Ns-1}]^T$ can be expressed as

$$x_s = \mathbf{S}\cdot\mathbf{h}, \qquad \mathbf{S} := \begin{pmatrix} s_{L-1} & s_{L-2} & \cdots & s_0 \\ s_L & s_{L-1} & \cdots & s_1 \\ \vdots & \vdots & \ddots & \vdots \\ s_{N_s-1} & s_{N_s-2} & \cdots & s_{N_s-L} \end{pmatrix} \quad (3\text{-}4)$$

It is affected by the sync word **s** only and not by the random data. Let the right cyclic shift operator of the observed signals **r** be

$$\mathbf{r}^{(m)} := T^m{}_{rc}(\mathbf{r}),$$

where

$$T_{rc}(\mathbf{r}) := \mathbf{r} = [r_{N-1}, r_0, r_1, \ldots, r_{N-2}]^T,$$

and choose a subwindow of length $N_s - L + 1$ of $\mathbf{r}^{(m)}$, namely, $\mathbf{r}_s^{(m)} := [r^{(m)}_{L-1}, r^{(m)}_L, r^{(m)}_1, \ldots, r^{(m)}_{Ns-1}]^T$. If '*m*' is the correct position of the frame starting location, $r_s^{(m)}$ can be expressed in terms of the synchronization word s as in (3-4).

A reduced-complexity ML-based estimator of μ that exploits the information provided by $r_s^{(m)}$ can be obtained by maximizing the following likelihood function.

$$\Lambda_3 = \frac{1}{(\pi N_0)^{N_s-L+1}} e^{-\frac{1}{N_0}(\mathbf{r}_s^{(m)}-\mathbf{x}_s)^H(\mathbf{r}_s^{(m)}-\mathbf{x}_s)}, \tag{3-5}$$

Equivalent log-likelihood function is

$$\hat{\mu} = \arg\min_{0\le m\le N-1} (\mathbf{r}_s^{(m)} - \mathbf{Sh})^H(\mathbf{r}_s^{(m)} - \mathbf{Sh}), \tag{3-6}$$

For a fixed m, the ML estimate of **h** is given by (see, e.g., Lambrette [7])

$$\hat{\mathbf{h}}(m) = (\mathbf{S}^H\mathbf{S})^{-1}\mathbf{S}^H\mathbf{r}_S^{(m)} \tag{3-7}$$

Substituting (3-7) into (3-6), we can obtain the following estimator equivalent to (3-6)

$$\mu = \arg\max_{0\le m\le N-1} \Lambda_4(m), \quad \Lambda_4(m) = r_s^{H(m)}(B-I)r_s^{(m)} \tag{3-8}$$

where **I** is the $(N_s - L + 1) \times (N_s - L + 1)$ identity matrix and **B** := $S(S^HS)^{-1}S^H$ denotes the projection matrix.

By using the metric explained in eq. (3-8), we can find a μ_{est} which achieves the maximum among the metrics for $m \in [0, N\text{-}1]$. Note that if $N_s - L = L - 1$ (i.e. ,$N_s = 2L - 1$), **S** is an L x L square matrix and nonsingular, then one can see that **B** = **I**. For this case $\Lambda_4(m)$ is always 0 and the estimator (3-8) is meaningless. Hence the length of the synchronization word has to be chosen as $N_s \geq 2L$. From this we see that the first $L - 1$ symbols of the sync word are guard symbols which prevent the remaining sync symbols from being affected by random data. At least L + 1 unaffected snyc symbols are required to estimate the L + 1 unknown parameters (μ, **h**).

3.3 Multiple Frame Synchronization

The synchronizer explained is based on one frame length of channel observations. If the processing delay is not a constraining factor, the performance can be improved by using multiple frames of channel observations to estimate the frame starting location. By using majority decision rule for K succesive frames based on individual estimates of each frame, a performance gain can be achieved. Another approach which will be explained, is to jointly exploit the K successive frames of observations $\mathbf{r} = [r_0, r_1, \ldots, ,r_{KN-1,}]^T$ to obtain a single estimate of μ.

Let us assume that the **h** (channel coefficients) remain constant during the consecutive *K* frames of channel observations. If we follow the procedure which is used for a single frame, we can obtain the K- frame based synchronizer, which is

$$\hat{\mu} = \arg \max_{0 \le m \le N-1} r_{Ks}^{(m)} (B_K - I_K) r_{Ks}^{(m)}, \tag{3-9}$$

where $\mathbf{I}_{\boldsymbol{K}}$ is the $K(\mathrm{Ns} - \mathrm{L} + 1) \times K(\mathrm{Ns} - \mathrm{L} + 1)$ identity matrix and

$$\begin{aligned} &B_K := S_K (S_K^H S_K)^{-1} S_K^H, \qquad S_K := [\underbrace{S^T, S^T, \ldots, S^T}_{K}]^T, \\ &r_{Ks}^{(m)} = [r_K^{(m)}(L-1), \ldots, r_K^{(m)}(N_s - 1), r_K^{(m)}(N + L - 1), \ldots, r_K^{(m)}(N + N_s - 1), \\ &\qquad \ldots, r_K^{(m)}((K-1)N + N_s - 1)]^T, \quad r_K^{(m)} := T_{rc}^m (r_K). \end{aligned}, \tag{3-10}$$

In [6], it is shown by simulations that multiple frame synchronization increases the synchronization performance and the effect of packet length N in the synchronization performance is negligible. The frame acquisition performance is mainly dependent on the synchronization word length. Employing multiple frame synchronization in the system has the same effect of using a longer synchronization sequence. So if the application is delay tolerant, multiple frame synchronization can be employed in the system.

3.4 Iterative Frame Synchronization for Frequency Selective Channels

Iterative receiver techniques provide a powerful tool to enhance receiver performance and have been used in receiver components like channel estimation, equalization and decoding. In this part an efficient iterative technique applied to frame synchronization is to be explained.

In [8], Samad proposed an iterative frame synchronization technique in which the hard-output, uncoded data at the output of the equalizer is utilized to improve the accuracy of synchronization. In simulations, it is shown that there is a large SNR gain in terms of false aquisition probability (FAP), mean square channel estimation error (MSCEE), and uncoded bit error rate (BER). The technique is attractive because it is applicable to various receiver structures like the ones which are using coding.

In classical Bayesian approach, unknown variables like channel coefficients, are treated as random variables with known probability density functions (pdf's) [1],[2],[6].

The Samad's approach other than classical Bayesian approach is to jointly estimate the frame boundary along with the other unknown variables. In [8], the frame boundary and the channel coefficients are jointly estimated. Other unkowns , like carrier frequency offset, can also be estimated following the Samad's approach.

System Model

The transmission of linearly modulated data through an L tap frequency selective channel given by $\mathbf{h} = [h_1 , h_2 , . . . , h_L]^T$ is considered, where L is the channel memory. It is assumed that throughout the formulation the channel is constant during the observation interval. The symbol-sampled received signal at time k is given by

$$r_k = \sum_{i=1}^{L} s_{k-i+1} h_i + n_k \tag{3-11}$$

where s_k's are the transmitted symbols and n_k is additive, i.i.d. complex Gaussian noise with variance N_0. Let us assume that $\mathbf{E}\{|s_k|^2\}=1$ for simplicity. Let us define $(N - L + 1)\times 1$ vectors, $r_k = [r_k, r_{k+1}, \ldots, r_{k+N-L}]^T$, and $n_k = [n_k, n_{k+1}, \ldots, n_{k+N-L}]^T$ and finally $(N - L + 1)\times L$ matrix, S_k as

$$S_k = \begin{bmatrix} s_k & s_{k-1} & \cdots & s_{k-L+1} \\ s_{k+1} & s_k & \cdots & s_{k-L+2} \\ \vdots & \vdots & \ddots & \vdots \\ s_{k+N-L} & s_{k+N-L-1} & \cdots & s_{k+N-2L+1} \end{bmatrix}, \tag{3-12}$$

Now we can rewrite the equation (3-11) as

$$\mathbf{r}_k = \mathbf{S}_k\mathbf{h} + \mathbf{n}_k. \tag{3-13}$$

Joint Frame Synchronization and Channel Estimation

In this section the derivation of the joint likelihood function for frame synchronization and channel estimation is shown. Let $\mathbf{S}_m$ denote the matrix of the known N_s training symbols $\{t_1, \ldots, t_{Ns}\}$ arranged according to equation (3-12) with $N = N_s$. Note that by restricting the size of $\mathbf{r}_k$ to $(N - L + 1) \times 1$ and under the assumption that k is the correct starting position of the training sequence, $\mathbf{r}_k$ only depends on the training sequence. Thus, the conditional pdf of $\mathbf{r}_k$ does not depend on unknown random data.

The conditional pdf of $\mathbf{r}_k$ given $\mathbf{h}$ and m (the correct starting position) is given by

$$P(r_k \mid \mathbf{h}, m) = \frac{e^{-\frac{1}{N_0}(\mathbf{r}_k - \mathbf{S}_m\mathbf{h})^{\mathrm{H}}(\mathbf{r}_k - \mathbf{S}_m\mathbf{h})}}{(\pi N_0)^{N_s - L + 1}}, \tag{3-14}$$

Since the training sequence is assumed equally likely to start anywhere in the search interval, the ML estimate , $\hat{m}$, of where the traning sequence starts is given by

$$\hat{m} = \arg\max_{k} \; -(\mathbf{r}_k - \mathbf{S}_m\mathbf{h})^H(\mathbf{r}_k - \mathbf{S}_m\mathbf{h}). \tag{3-15}$$

Since the channel **h** is unknown, we use the ML channel estimate, assuming that the training sequence starts at *k*, which is given by

$$\hat{\mathbf{h}}(k) = (S_m^H S_m)^{-1} S_m^H \mathbf{r}_k \,, \tag{3-16}$$

Finally, the ML synchronization is obtained as

$$\hat{m} = \arg\max_{k} \quad \mathbf{r}_k^H (P - I)\mathbf{r}_k \tag{3-17}$$

where $P = \mathbf{S}_m(\mathbf{S}_m^H\mathbf{S}_m)^{-1}\mathbf{S}_m^H$ is a projection matrix.

The formulation so far accommodates both burst and continuous-mode frame synchronization. Indeed, the only difference between this metric and that of [6] for continuous mode is that the elements of $\mathbf{r}_k$ are obtained here by linear shifts of the received data while they are obtained by circular shifts in the case of continuous mode synchronization, due to the periodicity of occurrence of the training sequence.

Iterative Synchronization

After obtaining the initial synchronization, $\hat{m}^{(0)}$, using equation (3-17) the data is equalized. Then the new matrix $\mathbf{S}_{\tilde{m}}^{(0)}$ is formed using equation (3-12) with $N = N_s + N_d$, with the N_s training symbols appended by the N_d estimated data symbols $\hat{\mathbf{d}}^{(0)}$, i.e. , $\left\{t_1,...,t_{N_s},\hat{\mathbf{d}}_1^{(0)},...,\hat{\mathbf{d}}_{N_d}^{(0)}\right\}$, in scenarios where the training sequence precedes the data (preamble). If the training sequence falls in the middle of the data (midamble) then we have $N = N_{d1} + N_s + N_{d2}$ and the matrix $\mathbf{S}_{\tilde{m}}^{(0)}$ is formed using the sequence

$\left\{\hat{\mathbf{d}}_1^{(0)},...,\hat{\mathbf{d}}_{N_{d1}}^{(0)},t_1,...,t_{N_s},\hat{\mathbf{d}}_1^{(0)},...,\hat{\mathbf{d}}_{N_{d2}}^{(0)}\right\}$. Using $\mathbf{S}_{\tilde{m}}^{(0)}$ in place of $\mathbf{S}_m$ in equation (3-17), we can obtain $\hat{m}^{(1)}$ and so on. The recursion could be written as

$$\hat{m}^{(i+1)} = \arg\max_k \quad \mathbf{r}_k^H (P^{(i)} - I)\mathbf{r}_k , \tag{3-36}$$

where $P^{(i)} = \mathbf{S}_m^{(i)}(\mathbf{S}_m^{(i)H}\mathbf{S}_m^{(i)})^{-1}\mathbf{S}_m^{(i)H}$ and i = 1, 2, ... , K. 'K' is the chosen iteration number. During that process new and improved channel and data estimates $\hat{\mathbf{h}}^{(i)}$ and $\hat{\mathbf{d}}^{(i)}$ are obtained.

Unlike the case of training based synchronization in (3-17), the matrix $P^{(i)}$ involves data that is not known before hand and thus the inversion of the matrix $(\mathbf{S}_m^{(i)H}\mathbf{S}_m^{(i)})$ has to be done in real time.

It is a common practice in frame synchronization to assume that the channel is constant throughout the observation interval. But this assumption only holds for slow fading channels.

CHAPTER 4

CODE AIDED FRAME SYNCHRONIZATION FOR AWGN CHANNELS

The use of concatanated convolutional codes in conjuction with iterative decoding, known as turbo coding, has enabled previously unachievable SNR levels. These improvements in the BER performance of the systems have resulted in a new problem. The steady state performance of a coded system is able to exceed the ability of the system to achieve initial synchronization. Initial synchronization must now be done at much lower SNR levels. It is possible to improve the frame synchronization performance, simply by increasing the synchronization word length. The drawback of this approach is the resulting increase in the overhead. This overhead may nullify the performance improvement obtained through coding. Another solution for increasing initial synchronization performance is to use the power of coding techniques used in the system. The decoder can assist in the frame synchronization stage and thus the frame synchronization performance can be improved.

In this chapter, first, a generalized frame synchronizer for coded systems will be outlined and some derivation results in the existing literature will be explained [9]. This may be used as a basis for practical code aided frame synchronization. In Section 4.2 , Howlader’s method [10] of decoder assisted frame synchronization is going to be investigated. In Section 4.3, a code aided frame synchronizer proposed by Cassaro [11], is going to be investigated.

4.1 A Generalized Frame Syncronizer for Coded Systems

A frame synchronizer utilizes a-priori knowledge of the synchronization word to produce an estimate of the frame starting position at the reciever, usually at a resolution of modulation symbol period or less. However, in most cases there can be many candidates that pass the initial estimation testing.

If the frame synchronization algorithm somehow can distinguish incorrect from correct syncs from several alternative frame starting positions, performance of the synchronizer can be improved. A frame sync unit followed by a decoder type system is assumed. A decoder can extract that kind of information from a coded sequence.

When decoding convolutionally encoded data, soft output decoders can supply the a-posteriori probability for each symbol, providing a “soft” confidence value about the probability that the decision reached for that symbol is correct , or provide several alternative decisions (paths through trellis) for single symbols or groups of symbols [21].

In [9] a strategy had been proposed analogous to the latter, for the problem of frame synchronization. The proposed strategy can be applied to all strategies which perform evaluation of a likelihood function across all possible frame starting positions.

In Figure 4.1, the value of a likelihood function for each possible start of the frame generated by a frame synchronizer is shown. For example it can be the output of the Massey’s optimum rule for frame synchronization for each possible frame start.

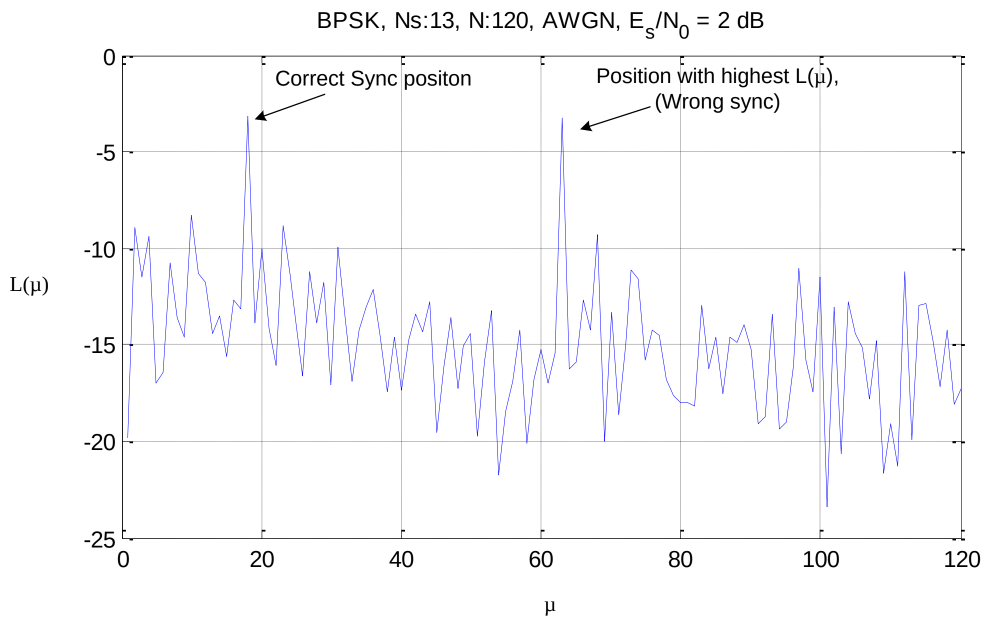


Figure 4.1. Output of a likelihood function

In the case shown in Figure 4.1, the frame synchronizer chooses the most likely frame starting point and makes a wrong decision. In this scenario the second most likely frame starting point according to the frame synchronizer in the receiver is actually the correct one . In this situation Robertson's approach makes the two or three most likely positions available to the next stage in the receiver and can give better results than the traditional approach of simply picking the apparently most likely one. In order to have some improvement, the next stage must have some means of eliminating those positions which are wrong, using data independent, or at least uncorrelated to that used by the initial frame synchronizer stage that produces the most likely positions. If this can be achieved without additional overhead, inserted specifically for frame synchronization, the price of the improvement of frame synchronization performance will only be the computational complexity.

As an example of a typical application it can be a block of data is secured by an outer error detecting block code as well as a forward error correcting inner convolutional code.

Assume a frame length N, of which $N - N_s$ symbols are information, and N_s symbols belong to the known synchronization word with special spectral properties. For example the systems which use synchronization words with impulsive autocorrelation have better frame synchronization performance. The frames are transmitted as a continuous stream across a channel. Receiver faces the task of determining the start of the received sequence $\mathbf{x} = (x_0, x_1, ..., x_{N-1})$ which must lie somewhere within this stream of received sequence. A likelihood function *L(μ)* is calculated for each possible position *μ* in [*0, 1, . . . , N – 1*], as shown in Figure 4.1. The frame sync unit outputs the most probable starting position that has the highest *L(μ)* to the next processing stage (in our case a decoder). If the next processing stage accepts this position sync is achieved. All the list elements will be searched until a sync is achieved or all the elements are tried.

4.1.1 Optimal Synchronizer When Data is Coded

In [9], the conditional probability which the synchronizer has to evaluate for each possible frame start location μ, is shown as

$$p(\mathbf{x} \mid \mu) = const \cdot \prod_{i=0}^{Ns-1} \exp(-\frac{\left\| x_{i+\mu} - S_i \right\|^2}{N_0}) \cdot \sum_{all\ \mathbf{d}} \prod_{i=Ns}^{N-1} \exp(-\frac{\left\| x_{i+\mu} - d_i \right\|^2}{N_0}) \tag{4-1}$$

The synchronization word **S** = (S_0, S_1, ... , S_{Ns-1}) and AWGN channel are considered where each component of the noise has variance $N_0/2$. If the data symbols are independent (uncoded data), then it is possible to derive a simple likelihood function. Maximizing this likelihood function is equivalent to maximizing (4-1) [2]. In that case the steps in the derivation simplify sum over **d**. “**d**” is the data sequence of the packet. If the data is coded, the last term in (4-1) has to be evaluated for each **d**.

$$\prod_{i=Ns}^{N-1} \exp(-\frac{\left\| x_{i+\mu} - d_i \right\|^2}{N_0}). \tag{4-2}$$

Realizing equation (4-2) is equivalent to what is performed efficiently by an ML decoder. But still evaluating it for all *μ* imposes considerable computational overhead. A small number $v << N$ of likely frame starting positions can be selected. The final choice is made by second stage by performing (4-2) just for these v positions. The value of (4-1) will be dominated by that of (4-2), and will be large only for one μ and one **d**. Frame synchronization and decoding must therefore be seen as a joint estimation.

In Figure 4.2, the general structure of the list synchronizer is shown.

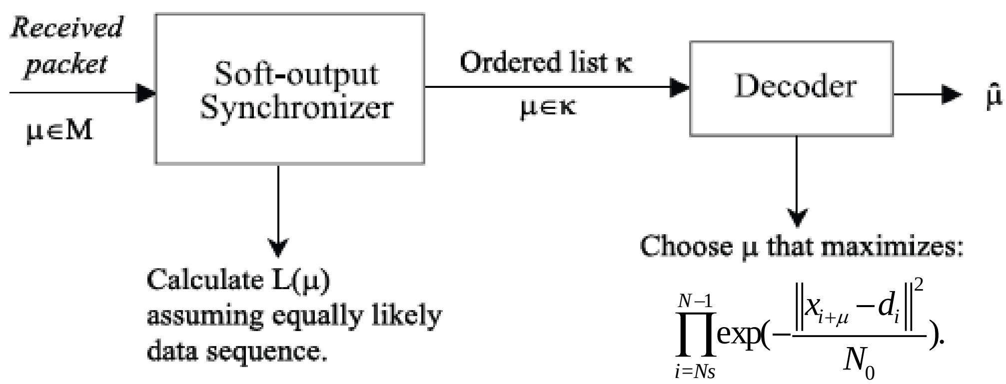


Figure 4.2. List Synchronizer

4.1.2 Performance in the Noiseless Case

Nielsen [3] derived the performance of a frame synchronizer in the noiseless case . Robertson [9] extended the results by defining correct sync to be achieved when the correct position is within top v likelihood functions. It is denoted by $P_{RDL}(v)$ (RDL: Random Data Limited). Using the restrictions made on the structure of the sync word in [3], it is easy to show that

$$P_{RDL}(v) = \sum_{j=0}^{M} \min(1, \frac{v}{j+1}) D_j Q^{-(N-Ns)}, \qquad (4\text{-}3)$$

where $M = (N - N_s)/ N_s$. D_j is the number of possible data sequences in which exactly j occurences of the sync word occur. $Q^{-(N - Ns)}$ is simply the probability of any one data sequence, and $\min(1, \frac{v}{j+1})$ is the probability that the correct position is in the set of v positions chosen out of j + 1 competing positions. Note that in the case of j + 1< v, the correct position will be found. D_j can be defined recursively by:

$$D_j = \binom{N - Ns - (Ns-1)j}{j} Q^{(N-Ns)-Ns.j} - \sum_{i=j+1}^{M} D_i \binom{i}{j} \qquad (4\text{-}4)$$

Let $P_F(v)$ be the probability that the correct sync word position does not correspond to the best 'v' likelihood function outputs and P_{EC} denote the probability of declaring a correctly synchronized frame incorrect. Assuming $P_F(v)$ and P_{EC} to be independent, we can express the probability that any one frame will eventually not be received correctly as,

$$P_{FE} = 1 - \left[(1 - P_F(v))(1 - P_{EC})\right]. \qquad (4\text{-}5)$$

As noted earlier the price to be paid for this improved synchronization performance is the requirement of overhead in a subsequent processing stage.

When evaluating the code aided frame synchronizer's performance, we take the list synchronizer performance curve derived in [9] as a lower bound. Whatever coding technique or decoder is used, it is not possible to obtain a better result than the ones obtained in [9] for the case of a list synchronizer. Because the second stage of the synchronizer can only find the correct position among the list provided by the first stage, and not succeed, unless the correct position is among those in the list.

4.2 Decoder Assisted Frame Synchronization

Howlader [10] proposed a frame synchronization methodology for packets with convolutionally encoded data. The sync bits are placed in a midamble and encoded as part of the data sequence rather than by placing bits in a separate header. The method is developed for using the error correction capability to resolve time ambiguities. The scheme is based on the principle that an error in trellis termination may result in decoding a wrong information sequence, while the synchronization portion of the sequence can be decoded as error free.

In [10] it is claimed that it is possible to eliminate the need for a synchronization word entirely in a coded system by inclusion of a coded midamble sequence. In previous section it is mentioned that the appeareance of the sync word pattern in the random data sequence can degrade the performance of synchronization. It is expected to overcome this degradation by using a coded midamble sequence. By making some assumptions, the performance of the proposed system appears to be better than traditional methods. Also the real frame synchronization performance is investigated.

4.2.1 System Model

It is assumed that the data is random, independently distributed, BPSK modulated, and that perfect symbol and carrier synchronization has already been achieved. The total packet length is N-symbol, of which N_s symbols comprise the known SW $\mathbf{s} = (s_0, s_1, . . ,s_{Ns-1}) \in \{1, -1\}^{Ns}$, and the remaining $(N - N_s)$ symbols represent data sequence $\mathbf{d} = (d_{Ns}, d_{Ns+1}, . . ,d_{N-1}) \in \{1, -1\}^{N-Ns}$. The complex baseband received signal, sampled at symbol intervals $t = kT$, is given by

$$r_k = a_k + n_k \tag{4-6}$$

where $\{a_k\}$ denotes the transmitted encoded bit sequence and $\{n_k\}$ are samples of AWGN with two sided power spectral density $N_0/2$. T is the symbol period, k is an integer.

In Figure 4.3, the transmitter of a traditional frame synchronization method is shown. Even if the convolutional coding is used, the SW portion is not protected by the coding and vulnerable to channel impairments. After receiving the N symbols from the channel, correlation rule or optimum frame synchronization rule can be applied for detecting the frame starting point.

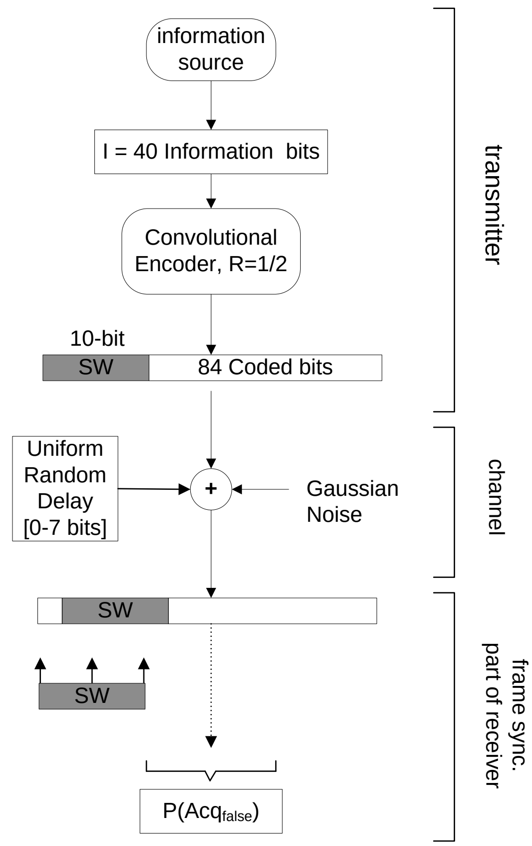


Figure 4.3. Traditional Frame Synchronization Method

4.2.2 Frame Synchronization Method with Coded SW

Sync word is embedded in the information sequence, as a midamble at time t = P, rather than as a separate header. Sync word will be encoded by the convolutional encoder and will influence the path followed through the convolutional code trellis. For example , if the sync word "00110" is inserted into the data stream, then in the four time intervals leading up to time t = P, the code trellis will visit states "00", "01", "11" and "10" in sequence. It is known a-priori that the trellis should be at a certain state at time t = P, and the unknown frame starting point can be distinguished by looking at the trellis states.

But it is assumed that the packet delay will be in between 0-3 trellis steps. For a BPSK modulated and ½ rate encoded data packet it means that the receiver accepts only 0,2,4 and 6 bits delay. The problem is called *node synchronization* . There is a proposed solution for this problem in [10]. It is claimed that by using two decoders, offset from each other by a single coded bit, the problem can be resolved. A Viterbi decoder that is misaligned by one coded bit will produce with high probability a set of path metrics that is far worse than that of the correctly aligned decoder.

In Figure 4.4 it is seen that the sync word is appended in the middle of the data and encoded with the information sequence. Information length is chosen 40 bits for illustrative purpose.

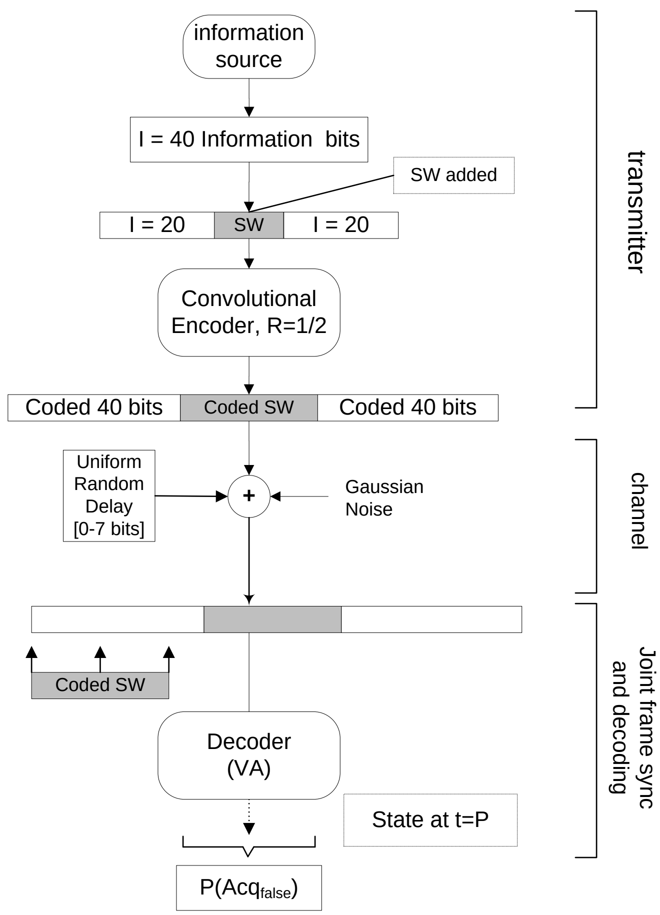


Figure 4.4 . Frame Synchronization proposed by Howlader

For determination of the initial state for the Viterbi decoder m zeros are appended at the beginning of the sync word. And N+m-P (the length between the end of the packet) is chosen long enough, like several constraint lengths so that with high probability the Viterbi iteration will have converged to the correct state sequence by the time it reaches the sync word.

4.2.3 Soft Synchronization Using List Synchronizers

In the previous section, hard synchronization is considered where the frame synchronizer estimates a packet starting position by calculating the ML function or estimating the ML state of the Viterbi algorithm. In the list syncronization tecnique, two stages of synchronizations jointly estimate the packet starting position. The first stage synchronizer produces an ordered list of v best estimates of frame starting positions in the descending order of the probabilities. The encoded preamble structure dictates the starting state of the Viterbi algorithm. In the list-synchronization technique, soft and iterative estimates can be passed between synchronization modules to enhance overall synchronization performance. In the scheme explained here, the decoding operation aids the synhronization operation. Assuming $P_{SN1}(v)$ is the probability that the frame starting position is not in the list provided by the first stage of the structure, the probability that the frame starting position is in the list increases monotonically as the list length v increases. And let P_{SN2} be the probability of incorrect frame starting decision of the second synchronizer. These two synchronization stages are independent, so the overall probability of the synchronization failure can be written as

$$P_{snT} = [1-(1-P_{SN1}(v))(1-P_{SN2})] \cong [P_{SN1}(v)+P_{SN2}] \qquad (4\text{-}7)$$

The probability $P_{SN1}(v)$ can be kept very low for a sufficiently high value of v and is comparatively easier to achieve.

For the first stage of the synchronizer, Massey's optimum frame synchronization (high SNR approximation) rule can be applied. It is simple to implement and a time efficient algorithm. For the second stage, the decoder aided synchronization scheme explained in the previous section can be implemented.

In Figure 4.5, a synchronizer is shown which has a capability of resolving any time ambiguity.

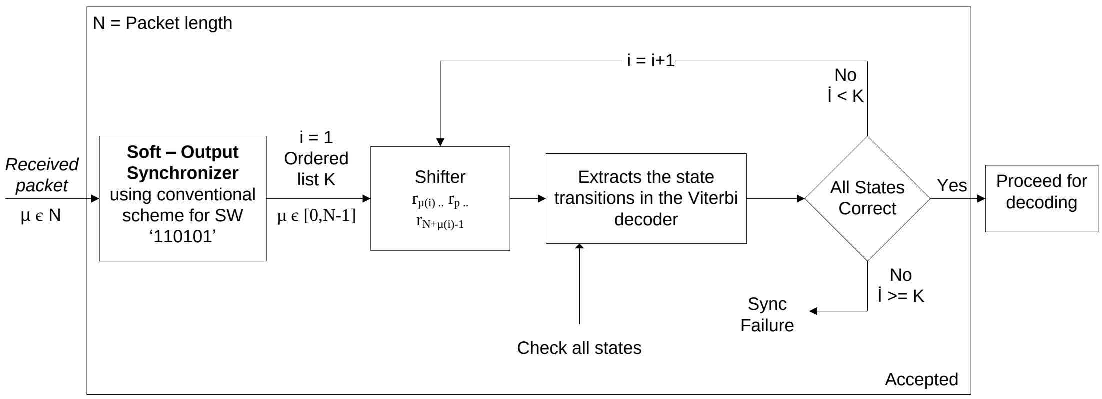


Figure 4.5. List Synchronizer for resolving up to N ambiguity of the packet

Note that the decoder assisted scheme discussed previously can resolve up to four trellis steps. The first module named with soft-ouput synchronizer is a conventional synchronizer and the SW "110101" is 6 bits long. In the previous example, the uncoded synchronization word which was inserted in the middle of the data was "00110" . The first m = 2 bits "00" of the uncoded sync word is for terminating trellis. So that it is guaranteed that encoder state is at "00" at time t = P. And after encoding the last portion of the uncoded sync word we have "xxxx110101" at the output of the encoder (x's represent the unknown output). The shifter module shifts the received packet according to the delay length which comes from the first module. And the decoder assisted synchronizer takes the most probable packet starting position from the ordered list and verifies the positions of the midamble bits using a one to one correspondance with state of the decoder. The last verification stage can be considered as the decoding operation. If the estimated midamble bits are not correct, this unit rejects the delay estimate by the first module and proceeds to the next most likely estimate on the ordered list. These cycles of operation can be continued until the correct midamble bits are obtained or a designated number of iterations are considered. After exhausting all iterations, if a correct midamble bit can not be obtained, the decoder can announce a synchronization failure.

In Figure 4.6, an example of a convolutional encoder used in the transmitter part of the system is shown. It has memory length of m = 2 bits and so the code has a constraint length of 3, i.e., memory length plus one.

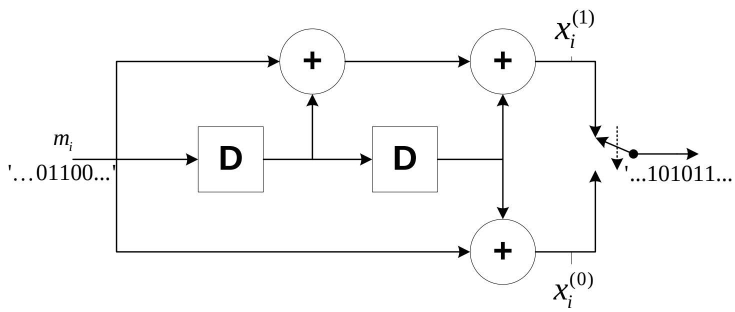


Figure 4.6. Convolutional encoder with memory length = 2

4.2.4 Choosing Synchronization Word for Howlader's scheme

A good sync pattern should have the desirable property that its autocorrelation has an impulsive shape and its prefixes and suffixes of the same length are distinct, so that the number of replications in the middle of random data is minimized [2].

In the list synchronization technique, first stage of the synchronization process consists of correlating the known coded sync pattern with the received symbols (or Massey's optimum rule can be used instead of correlation). In order to find the best possible uncoded sync pattern, a searching process is performed among all the possible uncoded sequences. The coded sync pattern which has the lowest sidelobe power of the autocorrelations is chosen.

Let $\mathbf{S}_u$ be the uncoded sync pattern and $\mathbf{S}_c$ be the coded sync pattern. Search result for an uncoded sync word length of 7 is given by

$\mathbf{S}_u = [0, 1, 1, 0, 0, 1, 0]$,

$\mathbf{S}_c = [1,1,-1, -1, 1, -1, 1, -1, -1, -1, -1, -1, -1, 1]$,

In Figures 4.7 and 4.8, the autocorrelations of coded synchronization word are shown. N_s is the length of coded synchronization word.

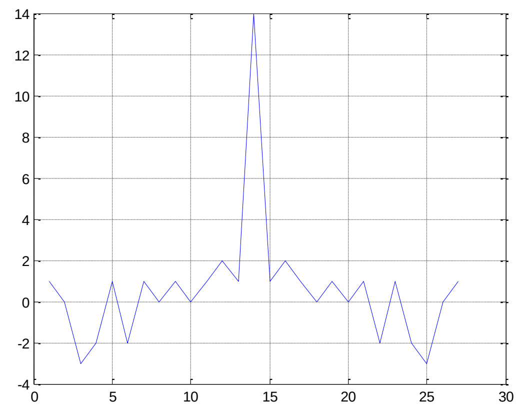


Figure 4.7. Autocorrelation of coded sync word, Ns = 14

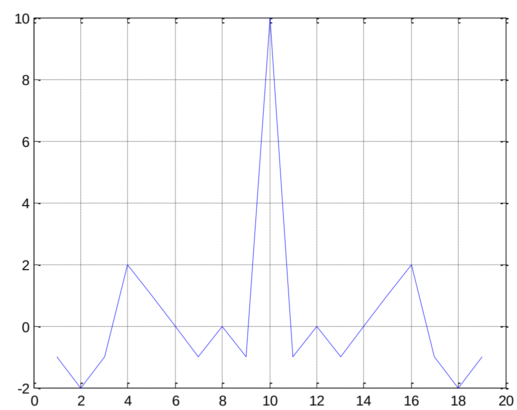


Figure 4.8. Autocorrelation of sync word length Ns = 10

In Howlader's scheme, synchronization word is coded with a convolutional encoder. In the Viterbi decoder the fixed states that are caused by uncoded sync pattern are searched to ensure that sync is achieved. In order to have fix states caused by uncoded synchronization word, the trellis must start at a known state. So ‘0’s are appended at the beginning of the uncoded synchronization word. There are m (m: memory length of the convolutional encoder) number of ‘0’s that are appended at the beginning of the uncoded synchronizer.

4.3 Frame Synchronization for Coded Systems Over AWGN Channel

In this section, an algorithm that takes advantage of soft information provided by a soft decoder to produce an enhanced estimate of the frame boundary is explained for additive white Gaussian noise channel. In [11], a method had been proposed to reduce complexity, that is a hybrid of the optimal uncoded frame synchronizer introduced by Massey and the list synchronizer introduced by Robertson. The whole system is investigated with and without complexity reduction.

4.3.1 System Model

It is assumed that BPSK modulation is used over an AWGN channel with two sided power spectral density ($N_0/2$). Other channel effects like ISI are assumed to be negligible and receiver functions like symbol timing, carrier recovery are assumed to be functioning perfectly. The frame of transmitted data consists of N symbols, where the first Ns symbols form a fixed frame synchronization pattern $\mathbf{s} = [s_0, s_1, \ldots, s_{Ns-1}]^T$ followed by a code word $\mathbf{c} = [c_{Ns}, c_{Ns+1}, \ldots, c_{N-1}]^T$. Accordingly, transmitted sequence is defined as an N×1 vector generated by concatenated **s** and **c** to form $\mathbf{sc} = \{s_0, s_1, \ldots s_{Ns-1}, c_{Ns}, \ldots, c_{N-1}\}$. The received sequence is the arbitrary phase shifted version of the transmitted sequence added noise.

Let $T(\mathbf{x}^{(i)}) := [x_1^{(i)}, x_2^{(i)}, \ldots, x_{N-1}^{(i)}, x_0^{(i+1)}]^T$, be the linear shift operator. Then we can express the received segment as

$$\mathbf{r} = T^m(\mathbf{sc}) + \mathbf{n}\,. \tag{4-8}$$

The corresponding log-likelihood function (LLF) can be written as

$$\begin{aligned}\ell(m=\mu\,|\,\mathbf{r}) = &\sum_{k=0}^{Ns-1}\Re(r_{(k+\mu)\oplus N}s_k^*)\\ &+\sigma_v^2\log\left[\sum_{i=1}^{|\mathbf{C}|}q(\mathbf{c}^i)\exp\left\{\frac{1}{\sigma_v^2}\sum_{a=0}^{\mu-1}\Re(r_a c_{N-\mu+a}^{i*})\right\}\right]\ ,\\ &+\sigma_v^2\log\left[\sum_{j=1}^{|\mathbf{C}|}q(\mathbf{c}^j)\exp\left\{\frac{1}{\sigma_v^2}\sum_{d=Ns+\mu}^{N-1}\Re(r_d c_{d-\mu}^{j*})\right\}\right]\end{aligned} \tag{4-9}$$

where |**C**| is the cardinality of the code space, q($\mathbf{c}^j$) is the probability of the *j*th codeword, $\oplus$ is modulo addition, $\Re()$ is the real operator, and c_j^k is the *j*th element of the *k*th codeword. Note that for practical systems the LLF is too complex, because of the summation over all possible codeword sequences. Hence another approach is needed.

In [11], the work of Lui and Tan (ML estimation) was taken as a basis, but also information contained in the coded data employs for finding frame starting point.

Log-likelihood ratio of the coded sequence can be used for distinguishing between a synchronized frame and an unsynchronized frame. For binary transmission, the ratio of the a posteriori probabilities (APP) of the information data sequence **d** can be written as

$$LLR(d_k) = \log\left(\frac{\Pr(d_k = 1\,|\,r)}{\Pr(d_k = 0\,|\,r)}\right), \tag{4-10}$$

where **r** is the received sequence, d_k = 1 is the null hypothesis, and d_k = 0 is the alternative. The magnitude of the LLR provides a measure of reliability of the corresponding symbol. The distribution of the LLR sequence is bimodal, and the corresponding distance between the modes (mode separation) contains information concerning the reliability of the decoded sequence $\hat{\mathbf{d}}$. If the decoded sequence is highly erroneous, the separation between the modes will be small and the distribution will appear about zero. This situation can happen because of a low SNR channel or an unsynchronized sequence.

Mode Seperation Algorithm is based on the following observation: the messages that are computed by the decoder $\mu_{c_k \to \varphi}(c_k)$ will have a different distribution depending on whether or not the receiver is synchronized.

The mode separation algorithm [11,12] for estimating the frame starting point μ,

$$
\begin{aligned}
& \textit{for} \;\; \mu = 0 \;\; \textit{to} \;\; N-1 \\
& \quad \mathbf{y} = T^{\mu}(r) \\
& \quad * \lambda_k = \log(\mu_{c_k \to \varphi}(c_k = 1)) - \log(\mu_{c_k \to \varphi}(c_k = 0)) \\
& \quad * M_+ = E[\lambda \mid \lambda > 0] \\
& \quad * M_- = E[\lambda \mid \lambda < 0] \\
& \quad * MS = M_+ - M_- \\
& \textit{endfor}
\end{aligned}
$$

The Mode-seperation statistic is used to estimate the frame boundary in the expense of increase in complexity. Mode separation algorithm can be computed for all possible frame starting points (from 0 to N-1, which N is the packet length) and an estimate can be obtained. Note that without the use of a synchronization word it is possible to estimate the frame starting point. But this approach will greatly increase the complexity. If a high probability list (HPL) can be provided from a previous stage, the mode separation algorithm can distinguish the synchronized frame without much of a complexity increase. A high probability list contains the most probable frame starting positions.

Moreover the estimator’s performance can be improved by randomly interleaving the transmitted codewords. At the receiver, an unsynchronized frame is still random for the Viterbi decoder even if it is deinterleaved. This assures that only the true synchronized frame gives a good mode-separation statistic. Therefore, the estimator’s ability of discriminating between a true boundry and a false boundry is significantly increased.

In Figure 4.9 , the transmitter of the code aided frame synchronization scheme is shown.

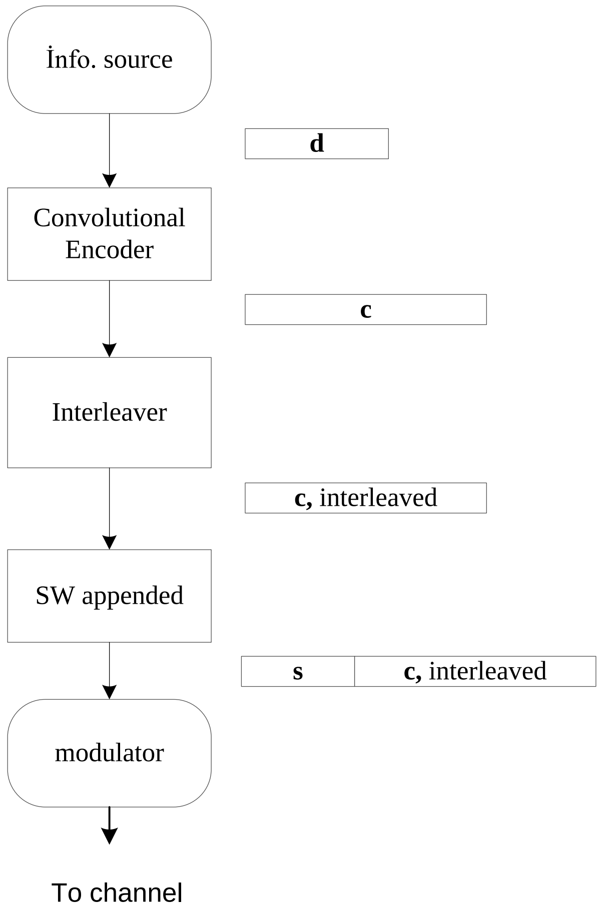


Figure 4.9. Transmitter part of the System, d : data , c: coded data , s: sync. word

The data sequence is convolutionally encoded and interleaved by a random interleaver. Then, an uncoded synchronization word is appended at the beginning of the interleaved and coded sequence.

4.3.2 Complexity Reduction

The high probability list (HPL) which was previously mentioned is one way of reducing the complexity of the coded frame synchronization scheme. In [11], another way of complexity reduction is discussed by Cassaro and Georghiades. A test before computing the mode separation of the received coded data determines if the most probable frame starting point that is provided by the uncoded frame synchronization is reliable enough or whether list synchronization is required. Uncoded frame synchronization can be a simple correlation rule or Massey's optimum rule. This approach reduces the average complexity by relying on the first stage of the scheme. The second stage can be called a high complexity frame synchronizer (HCFS). HCFS has ability to distinguish synchronized frame from unsynchronized ones by using the mode separation algorithm. HCFS is constrained by a list of probable frame starting points. The list is denoted by **Q** and referred to as a high-probability list (HPL). **Q** can be obtained by evaluating the Massey's optimum frame synchronization rule or by evaluating a simple correlation rule. The list is composed of '*v*' items. The final estimate of the frame starting point is obtained by the HCFS after evaluating all *v* possible sequences.

The test which was designed to choose in favor of relying on the decision of the first stage or to use the HCFS. The test can be stated as

$$\delta(\mathbf{Q}) = \begin{cases} 0, & T(\mathbf{Q}) \le \tau \\ 1, & otherwise \end{cases}, \tag{4-11}$$

where $\delta(\mathbf{Q}) = 0$ decides using the relatively simple first stage and $\delta(\mathbf{Q}) = 1$ is the alternative for using the list synchronizer. T(**Q**) is the normalized distance between the two largest statistics in **Q** and is defined as

$$T(\mathbf{Q}) = \left| \frac{Q_{\max} - Q_{\max 2}}{Q_{\max}} \right| \quad , \tag{4-12}$$

where Q_{max} is the largest statistic in **Q**, $Q_{max\ 2}$ is the next largest statistic and τ is the threshold which determines the rate of the usage of HCFS. Basically if the largest Q_{max} is much larger than the second one $Q_{max\ 2}$ it is assumed that the decision of the uncoded synchronizer is reliable. Otherwise HCFS is employed to find a more reliable estimate.

In Figure 4.10, code aided frame synchronization scheme with complexity reduction is shown.

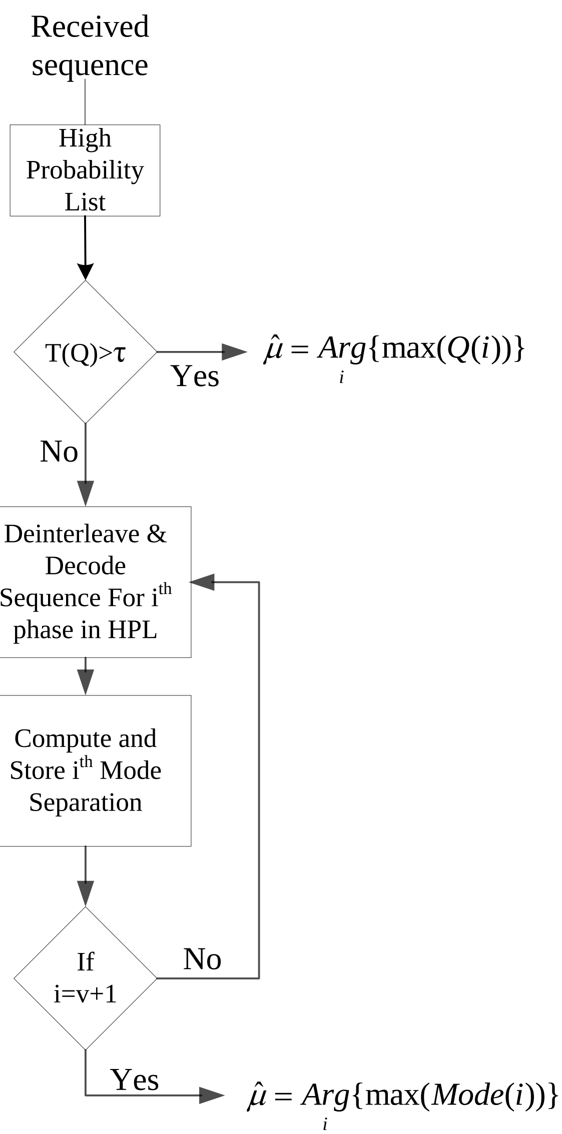


Figure 4.10. Block Diagram of the Cassaro&Georghiades' method of frame synhronization algorithm for AWGN channel

First, a list of most probable frame starting points is generated from the received data by employing a simple correlation or using the Massey's rule of estimating the frame boundary. Then a reliability test is run to decide either use the result of the first stage or activate the second stage of the synchronizer. The second stage of the synchronizer consists of a deinterleaver, a soft output Viterbi decoder and a block that computes the mode separation statistics of the received and shifted frames. When the second stage is activated, all the possible frame starting points which are involved in the list are processed and among them the one with the best mode separation statistic is chosen as the final estimate.

In [11], the performance results are not demonstrated when complexity reduction algorithm is employed. We have been investigated the simulation results of the system when complexity reduction is employed. And it is appeared that the performance results are no better than the Massey's high SNR rule [1] which is employed for stage 1 for high SNR region. Because the system suffers from the random data limited upper bound. The repetition of the sync word in the random data can easily deceive the test. But the HCFS which is a Viterbi decoder employes mode separation algorithm in our case, can eliminate this repetition effect. If a wrong location is taken as a frame starting point because of a repetition effect, the frame will be unsynchronized and the separation between the modes appears about zero. A synchronized frame causes a relatively large separation between the modes. Unlike in the case of Howlader's method [10] which extracts the trellis states and searches a certain state sequence caused by a sync word, Cassaro's method [11] is not affected by the random data limited upper limit.

CHAPTER 5

CODE AIDED FRAME SYNCHRONIZATION TECHNIQUES FOR FREQUENCY SELECTIVE CHANNELS

Many algorithms have been developed that take advantage of soft information provided by a soft decoder to produce an enhanced estimate of frame boundary for AWGN channels. But for frequency selective channels, there are not much studies which take into account the power of coding when dealing with the problem of frame synchronization. It is possible to develop a frame synchronization technique for frequency selective channels which uses the power of coding by using the existing methods in the literature. In Section 5.1 a code aided frame synchronization method has been proposed for frequency selective channels which is based on Howlader's method [10]. In Section 5.2 another method has been proposed for the same purpose which is based on Cassaro's method of frame synchronization [11].

5.1 Modification of the Howlader's Scheme for Frequency Selective Channel (CAFS1)

In [10], Howlader proposed a frame synchronization technique for the systems which use convolutional coding. It uses the Viterbi states at the decoder to ensure that the first stage soft-output synchronizer estimates frame boundary correctly. In [6] Wang and Serpedin proposed a joint channel estimation and frame synchronization for frequency selective channels. By combining Wang's method of frame synchronization for frequency selective channels and Howlader's decoder assisted frame synchronizer for

AWGN channels, we have been proposed a new decoder assisted frame synchronizer for frequency selective channels. We call this proposed frame synchronizer Code Aided Frame Synchronizer 1 (CASF1).

5.1.1 System Model

A BPSK modulated signal is assumed to be transmitted through a slow time-varying frequency- selective channel, whose coefficients $\mathbf{h} = [h_0, h_1, \ldots, h_{L-1}]^T$ are assumed to remain constant over the duration of the observed sequence, L represents the channel memory. The frame of transmitted data consists of N symbols, where the first Ns symbols form a fixed frame synchronization pattern $\mathbf{s} = [s_0, s_1, \ldots, s_{Ns-1}]^T$ followed by a code word $\mathbf{c} = [c_{Ns}, c_{Ns+1}, \ldots, c_{N-1}]^T$. The transmitted signal is passed through the channel $\mathbf{h}$ and is sampled at the symbol period.

The transmitter part of the system is exactly the same as explained in Section 4.2.1.

5.1.2 Adaptation of the Howlader's Scheme to the Frequency Selective Channels

Two types of receiver structures are proposed for CAFS1. The blocks used in the receiver structures are almost the same. For both of the proposed receiver structures a joint frame synchronization for frequency selective channel and channel estimation algorithm is used. This part is the first stage of the list synchronizer concept which is composed of two stages. First stage outputs an ordered list of most possible frame starting locations. The second stage is the Viterbi decoder which extracts the state sequence of the received sequence and searches for the known sync state sequence. But before employing the Viterbi decoder the ISI effect introduced by channel must be eliminated. A Viterbi equalizer [14,15] is used for this purpose. If any sync state sequence can not be found in the Viterbi decoder, the second most probable frame starting point will be tried by shifting the received sequence. This process will be repeated until a sync is achieved or K number of possible locations tried.

The difference between the receiver structures is the first stage of them. The first one of the two proposed receiver structures is generally based on the Howlader's structure which is discussed in the Section 4.2 and used Wang's method for generating the possible frame starts and estimating the channel taps '**h**'. In the second proposed receiver structure Samad's iterative frame synchronization [8] scheme is adopted.

The proposed receiver structures (Receiver Structure 1 & 2) are the variations of the proposed frame synchronizer method 1 which is entitled as CAFS1.

Receiver Structure 1

In the receiver structure 1, first stage ,which is labeled as 'soft-output synchronizer' in Figure 5.1, is employed for generating a '**K**' size ordered list of possible frame starting points. Then a shifter block shifts the received sequence according to the list. Channel estimation algorithm is run over the shifted received sequence. A maximum likelihood sequence equalizer (in our case a hard-output Viterbi equalizer) is used to compansate the ISI effect. Finally Viterbi decoder checks the sync state sequence in the coded data. This process is repeated as shown in Figure 5.1, until a sync is achieved or the **K** possible frame starts tried.

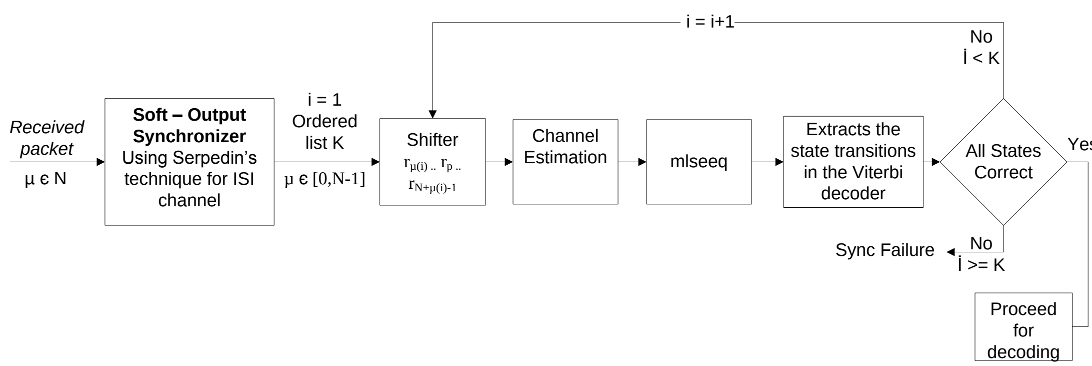


Figure 5.1. Proposed Frame Synchronizer for Frequency Selective channel, receiver structure 1

Receiver Structure 2

In the receiver structure 2, the synchronizer block which is the first block of the structure, is based on the Samad's [8] method of frame synchronization for frequency selective channels. Samad's method is given in Section 3.4. It produces a possible frame starting point unlike the receiver sructure 1 where a list of frame starts is generated. The received sequence is shifted. The shifted received sequence is equalized and decoded. At first the frame synchronization and channel estimation algorithms run with the known sync word. Then an iteration is employed and in this case not only the synchronization word is used for estimating the frame start but also the known data sequence is employed. In the numerical results, it is shown that iteration brings a considerable performance gain for high SNR region. Other blocks of the receiver structure 2, are the same as in the receiver structure 1. As the iteration count increases, the complexity increases linearly.

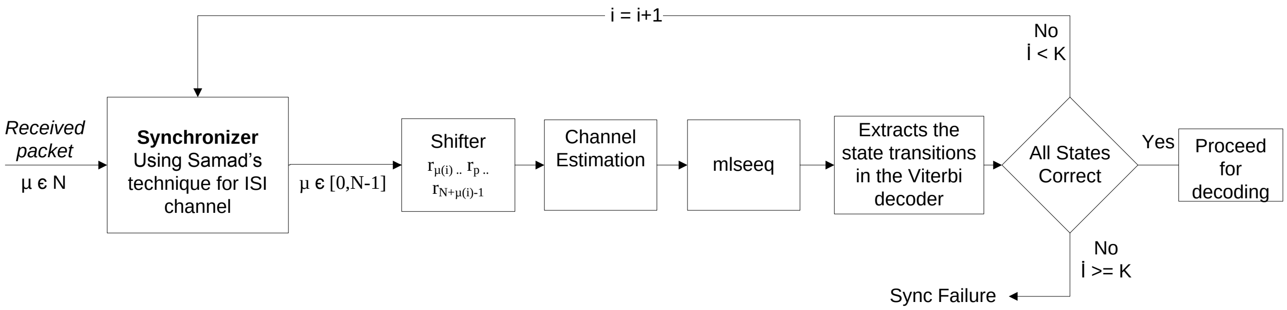


Figure 5.2. Proposed Frame Synchronizer for Frequency Selective channel, receiver structure 2

5.2 Modification of the Cassaro's Scheme for Frequency Selective Channel (CAFS2)

In this section a new code aided frame synchronization method is proposed for frequency selective channel and it is based on the Cassaro's method which is described in the previous chapter. The proposed method has a two stage structure like in the Cassaro's method. First stage outputs a list of possible frame starting points by using the method explained in Section 4.3. In Section 3.2, a frame synchronization algorithm which was proposed by Wang [6] is introduced. This algorithm can estimate the frame boundary without equalizing the channel for frequency selective channels means channel parameters are unknown. Wang's synchronization rule is based on the ML rule and is the optimal frame synchronization method for frequency selective channels. Also the channel parameters are estimated in the first stage. Second stage is able to distinguish synchronized frame from unsynchronized ones. But before activating the second stage an equalizer is employed. In order to combat ISI effects introduced by the channel a soft output Viterbi equalizer [16,17] is used. Equalizer needs the channel parameters to work properly. The channel parameters are provided from the first stage. In the second stage of the synchronizer, a soft output Viterbi decoder is used for computing the mode separation algorithm like in the case of AWGN channel. We call this proposed frame synchronizer Code Aided Frame Synchronizer 2 (CASF2).

5.2.1 System Model

A BPSK modulated signal is assumed to be transmitted through a slow time-varying frequency- selective channel, whose coefficients $\mathbf{h} = [h_0, h_1, \ldots, h_{L-1}]^T$ are assumed to remain constant over the duration of the observed sequence and L represents the channel memory. The frame of transmitted data consists of N symbols, where the first Ns symbols form a fixed frame synchronization pattern $\mathbf{s} = [s_0, s_1, \ldots, s_{Ns-1}]^T$ followed by a code word $\mathbf{c} = [c_{Ns}, c_{Ns+1}, \ldots, c_{N-1}]^T$.

The transmitted signal is passed through the channel **h** and is sampled at the symbol period. The transmitter part of the proposed method is exactly the same as Cassaro's scheme which has been described in Section 4.3, Figure 4.9.

Let $T(\mathbf{x}^{(i)}) := [x_1^{(i)}, x_2^{(i)}, ..., x_{N-1}^{(i)}, x_0^{(i+1)}]^T$, be the linear shift operator. Then we can express the received segment as

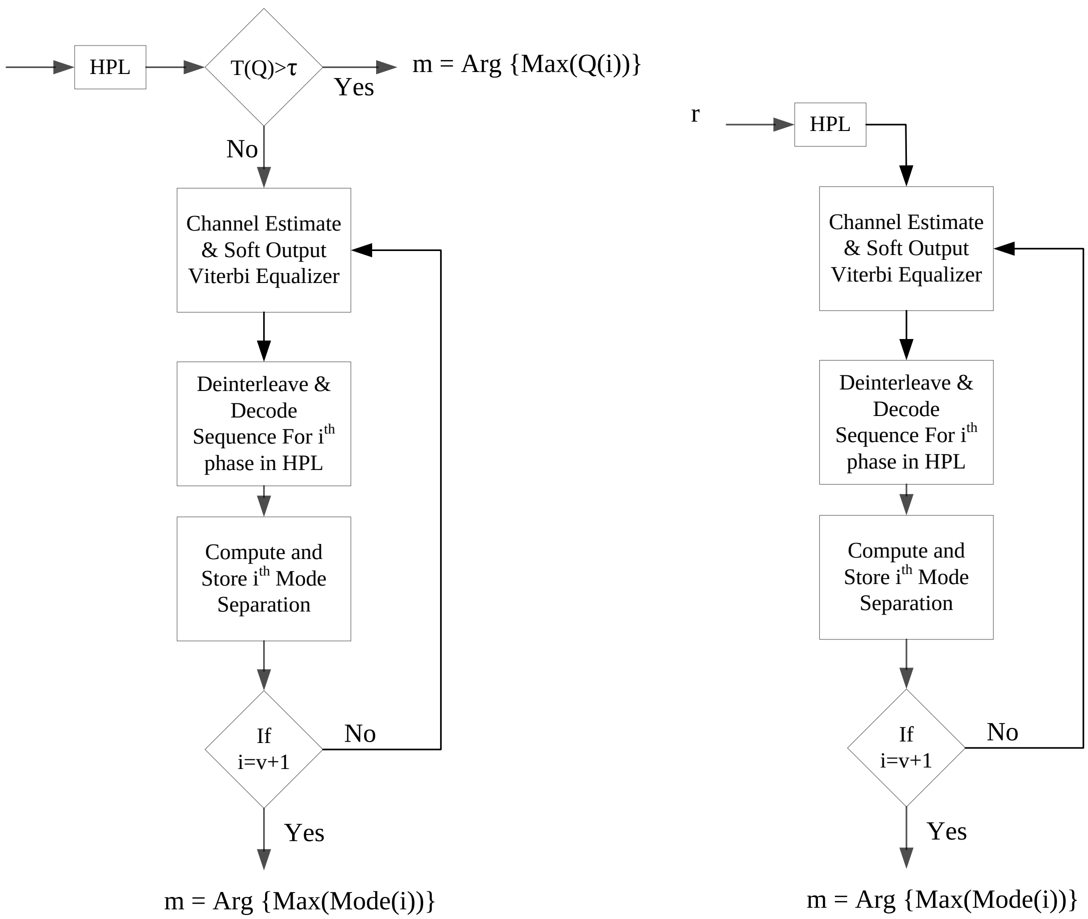
$$\mathbf{r} = T^m(\mathbf{sc} \cdot \mathbf{h}) + \mathbf{n}, \tag{5-1}$$

5.2.2 Receiver Structure of the Proposed Scheme

In Figure 5.3, two receiver structures are shown for the proposed frame synchronizer for the frequency selective channel. In Figure 5.3 (a), complexity reduction is employed in the frame synchronizer. The frame synchronizer which is shown in Figure 5.3 (b) is not using the complexity reduction.

If we skip the test for the complexity reduction of the proposed structures, the overall system operation can be summarized as

1. A list of most probable frame starting points (the number of elements in the list is '*v*') is generated by the HPL block.
2. Received sequence is shifted according to the estimate of the HPL block.
3. Channel is estimated and a soft output Viterbi equalizer is employed among the shifted received sequence in order to compansate the ISI effect introduced by the channel.
4. Equalized received sequence is deinterleaved.
5. Deinterleaved sequence is decoded by the soft output Viterbi decoder and mode separation algorithm is employed to extract the statistic.
6. Return to step 2, until for all the probable frame starting points, received sequence is processed.
7. Choose the frame starting point among the list which has the best mode separation statistic.



(a) **(b)**

Figure 5.3 a) Block Diagram of the proposed frame synchronization algorithm with complexity reduction
b) Block Diagram of the proposed frame synchronization algorithm without complexity reduction

Figure 5.4 illustrates the Mode Separation Statistics for three different systems.

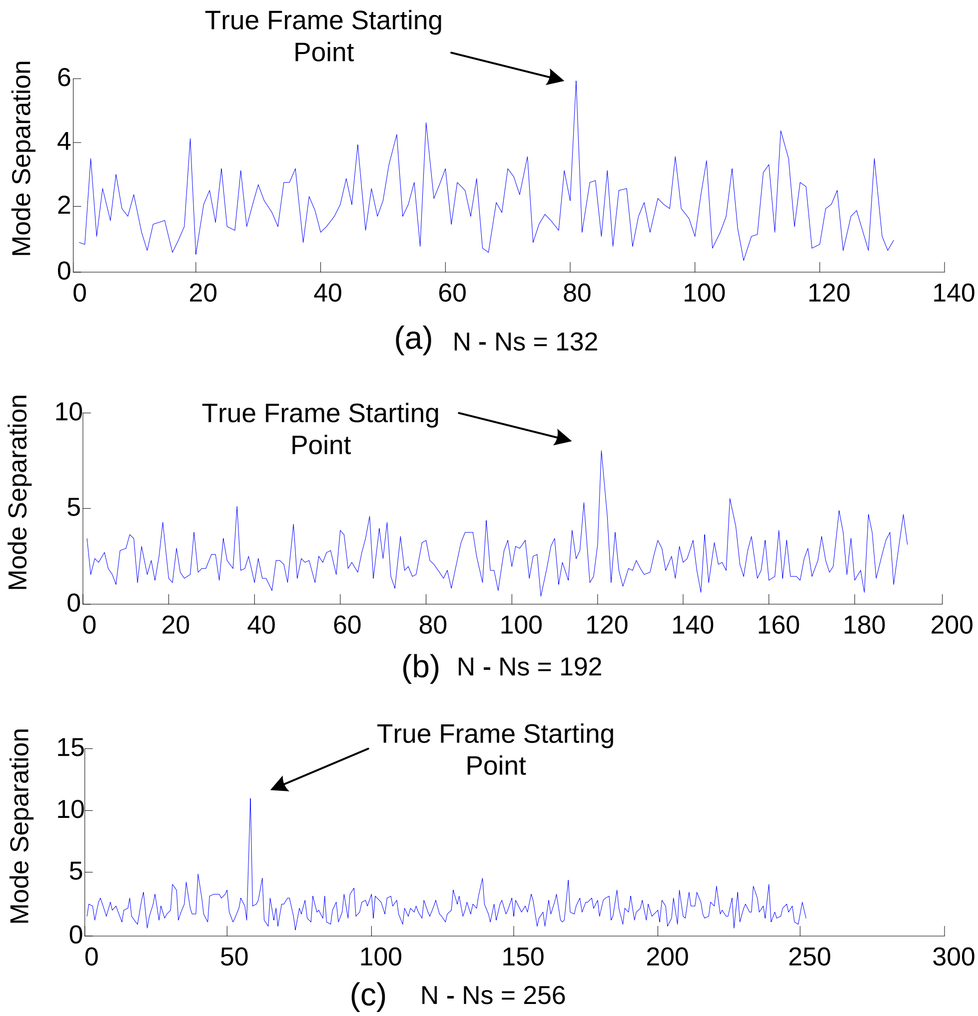


Figure 5.4. Mode Separation Statistics for different frame lengths (synchronization word length Ns is constant for all the tree graphs), SNR = 10 dB

For all three systems shown in the figure, the systems parameters are exactly the same except for the coded data length. Synchronization word lengths are constant for all three systems. Mode Separation algorithms are run for all the possible frame starting points from ‘0’ to ‘N-1’, and the output of the mode separation algorithms are plotted as seen in Figure 5.4. We can think the output of the mode separation algorithm as reliability value for the corresponding frame starting location. From Figure 5.4 (a) through (c) the packet length increases. As the packet length increases we have a more reliable estimate of the frame starting location because we have more coded symbol statistics to compute the frame statistics. Hence we expect better results from the systems which have relatively longer packet length compared to synchronization word length. The simulation results in Chapter 6, support this conjecture.

In this thesis, our main concern is the performance of the frame synchronizer. Hence we mainly emphasized on the scheme which is not using complexity reduction method proposed by Cassaro. Employing complexity reduction in the system reduces the performance of the system significantly. Because the system suffers from random data limited upper bound as mentioned previously in this Chapter. In Chapter 6 (Numerical Results), most of the frame acquisition performance curves are evaluated without employing the complexity reduction.

CHAPTER 6

NUMERICAL RESULTS

In this chapter, results of the simulations which have been performed in this thesis are presented. In Section 6.1, the performance of the optimum frame synchronizer (Massey's optimum rule) is shown. In Section 6.2, the performance of the optimum frame synchronizer for the frequency selective channel is shown along with the mean square channel estimation error (MSCEE) vs SNR curves. In Section 6.3, the performance of the iterative frame synchronization method over a frequency selective channel is shown. In Section 6.4 , the false acquisition probability vs SNR curves are shown for the concept of the list synchronizer introduced by Robertson. In Section 6.5, the performance results of the proposed method 1 which is based on the Howlader's scheme [10] are shown. In Section 6.6, the performance results of the proposed method 2 which is based on the Cassaro's scheme [11], are shown.

General Simulation Settings

In this thesis, we consider BPSK modulation for all the simulations for simplicity. The additive noise **n** is generated as white Gaussian noise with variance N_0, and SNR is defined as SNR := $10\log_{10}(1/N_0)$. Computer simulations were performed to assess the false acquisition probability (FAP, i.e., $\hat{\mu} \neq \mu$) , where '$\hat{\mu}$' is the estimated frame starting location and 'μ' is the true frame starting location. In all figures, N represents the packet length, Ns represents the synchronization word length and L represents the number of channel taps for frequency selective channels.

6.1 Performance of the Optimum Frame Synchronization for AWGN channel

The performance of the optimum and suboptimum frame synchronization rules for AWGN channel are shown along with the simple correlation rule. Suboptimum rules are high SNR and low SNR approximation of the optimum rule. The simulation is done using 100,000 Monte Carlo trials.

In Figure 6.1 simulation results on false frame acquisition probability are shown for packet length N = 28, synchronization word length Ns = 7.

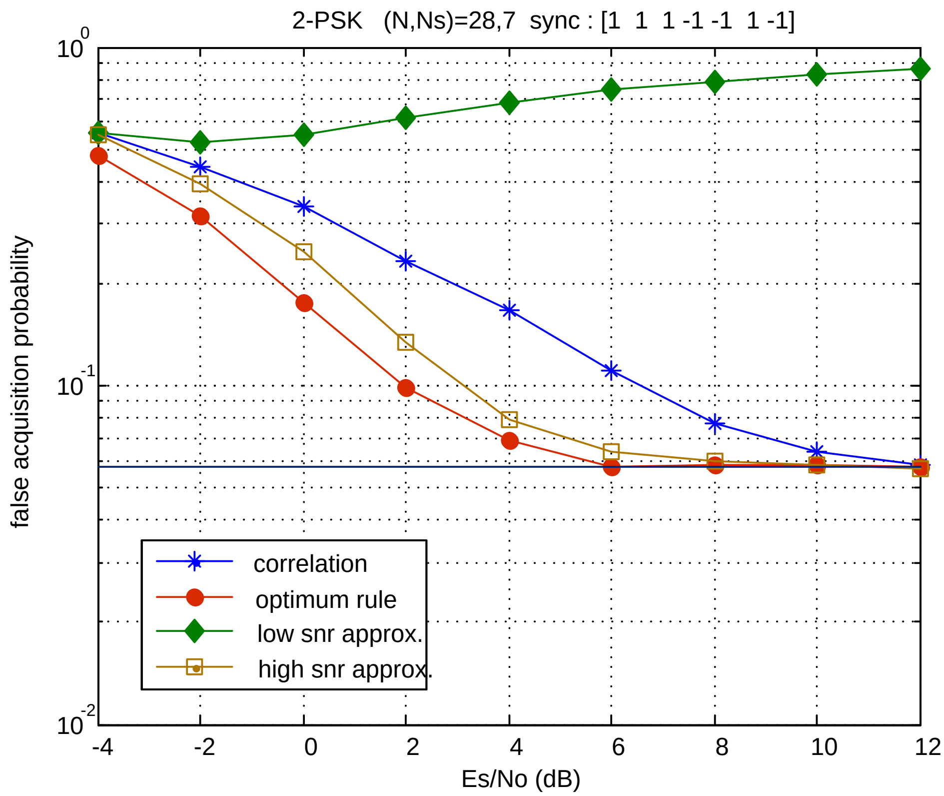


Figure 6.1. Probability of false frame acquisition vs SNR in dB for each of the four frame synchronization methods

For these system parameters, the random data limited upper bound is calculated as (1-P_{RDL}) = 0.0579, where [2]

$$P_{RDL} = \sum_{i=0}^{[N/Ns-1]} \frac{(-1)^i}{(1+i)} \binom{N - Ns - (Ns-1)i}{i} M^{-Ns \cdot i}$$

Remember that random data limited upper bound P_{RDL} limits the synchronizer performance for every system where the synchronization word can be replicated in the random data.

In Figure 6.2, the packet length is chosen as N=130 and the synchronization word length is chosen as Ns = 13.

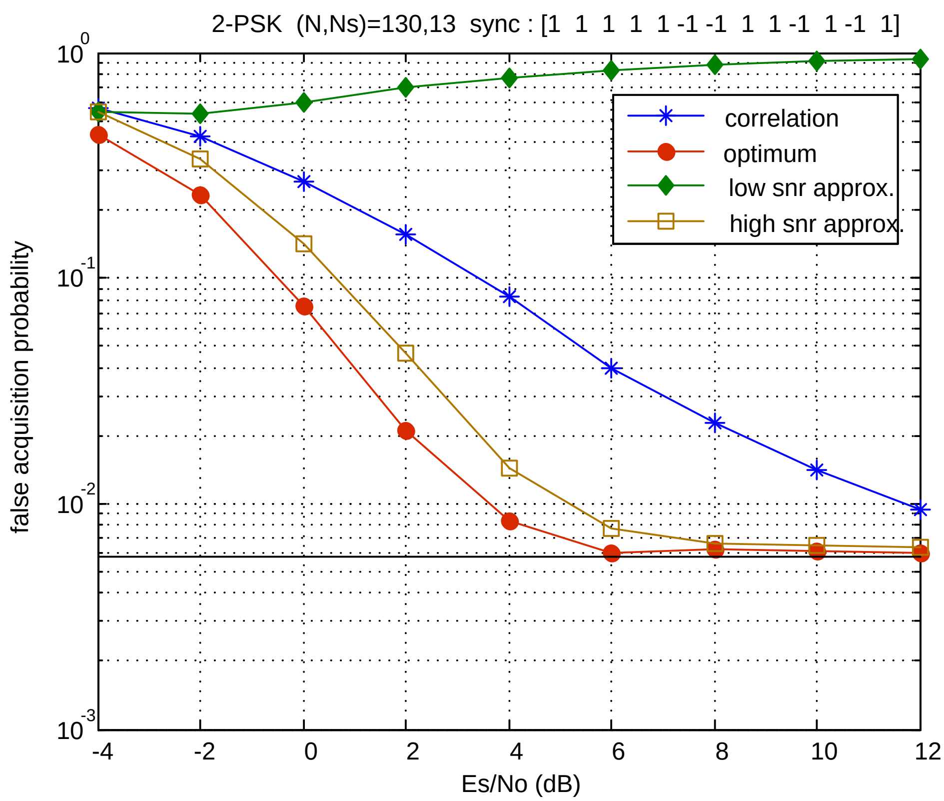


Figure 6.2. Probability of false frame acquisition vs SNR in dB for each of the four frame synchronization methods.

It is observed that performance of the correlation rule degrades from that of the optimum rule by as much as 6 dB over a range of SNR from 0 to 12 dB.

Note that for high SNR's the asymptotic performance is achieved with either of the frame synchronization rules, optimum, correlation or sub-optimum ones for both figures.

The performance of the high SNR approximation rule is very close to the performance of the optimum rule and its complexity is very low compared to the optimum rule.

6.2 Performance of Joint Frame Synchronization and Channel Acquisition for Frequency Selective Channels

In Figures 6.3 and 6.4, the mean square channel estimation error (MSCEE) $E=\{\|\hat{\mathbf{h}}-\mathbf{h}\|^2\}$ and the false acquisition performance of the joint frame synchronization and channel estimation algorithm are shown, respectively. The frequency selective channel coefficients are modeled as i.i.d. complex Gaussian random variables with zero mean and variance 1/L. 'L' is the number of the channel taps for the frequency selective channel. There are two different channels considered in the simulation which have channel taps length L=2 and L=4. The synchronization sequences are chosen as 13-symbol {1,-1,1-1,1,1,-1,-1,1,1,1,1,1} and the 18-symbol {1,1,-1,-1,1,1,1,-1,1,-1,-1,1,-1,1,1,1,1}. The frame synhronization patterns are derived from the computer search method.

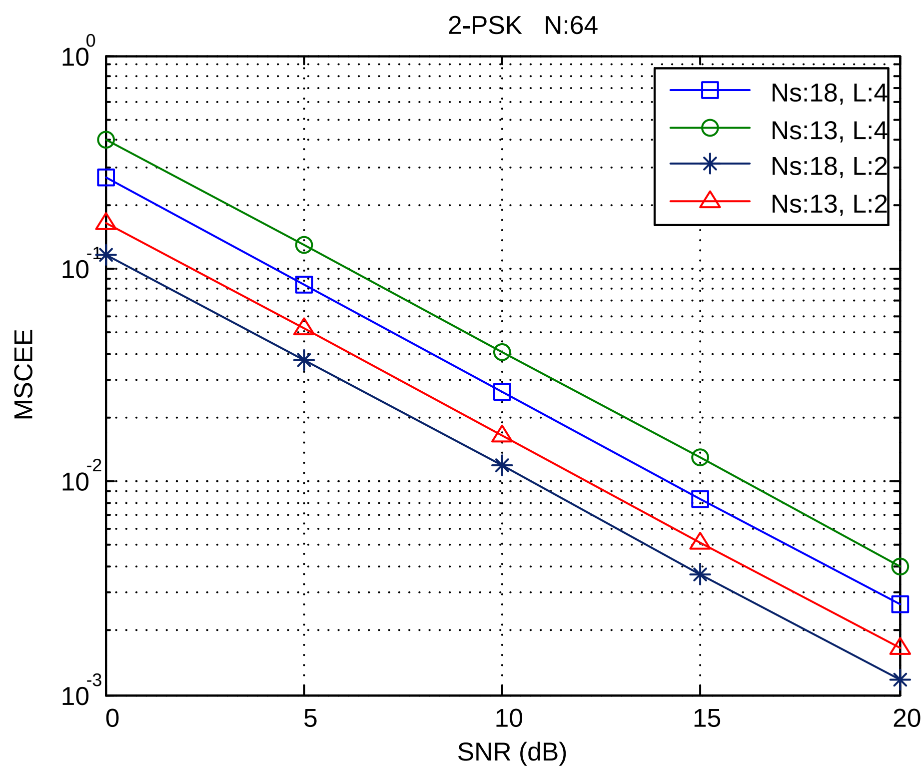


Figure 6.3. MSCEE vs SNR (dB)

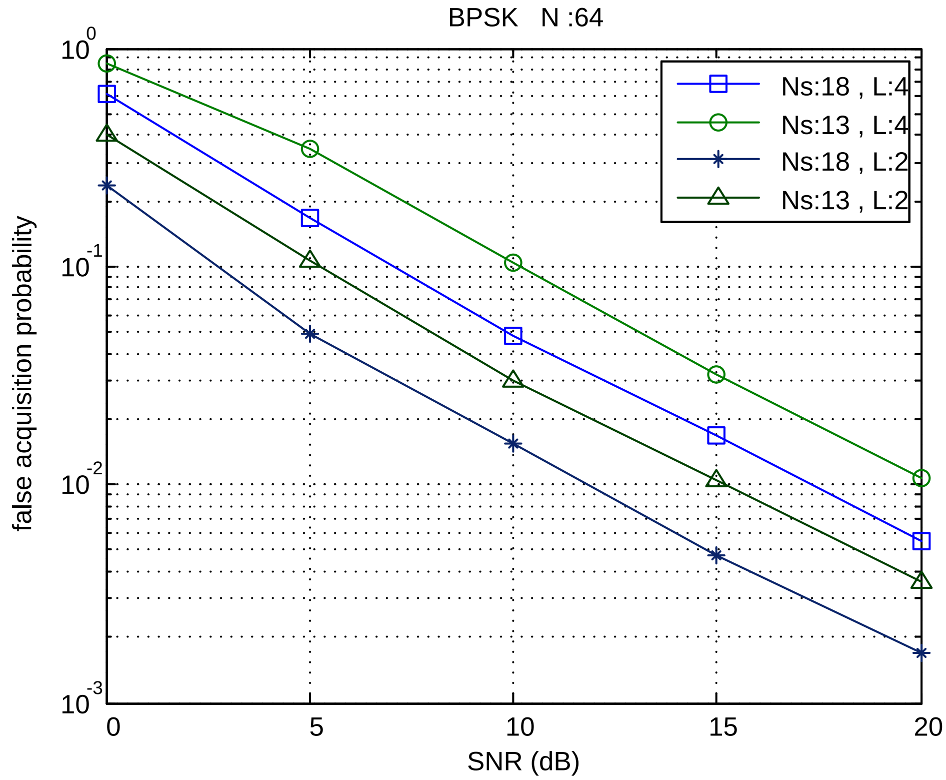


Figure 6.4. FAP vs SNR (dB)

Both the MSCEE and the FAP performance of the joint channel estimation and frame synchronization algorithm increase when the channel tap number decreases and the synchronization word length increases as seen in the figures.

6.3 Performance of Iterative Frame Synchronization and Channel Acquisition for Frequency Selective Channels

The frequency selective channel coefficients are modeled as i.i.d. complex Gaussian random variables with zero mean and variance 1/L. ‘L’ is the number of the channel taps for the frequency selective channel and chosen as L=4. In the receiver a hard-output Viterbi equalizer is used to estimate the data symbols which are used in the iteration of the frame synchronization algorithm. Remember that iterative frame synchronization method uses the hard-output uncoded data at the output of the equalizer to improve the accuracy of the synchronizer. The packet length is chosen as N=96 and the synchronization word length is chosen as Ns = 16. The synchronization sequence is derived from the computer search method as in the previous case and it is inserted in the middle of the data sequence unlike the case in Section 6.2. This simulation setting is exactly the same as in the referenced paper [8].

In Figures 6.5 and 6.6 the frame synchronization and the MSCEE performance of the iterative frame synchronization method are shown, respectively.

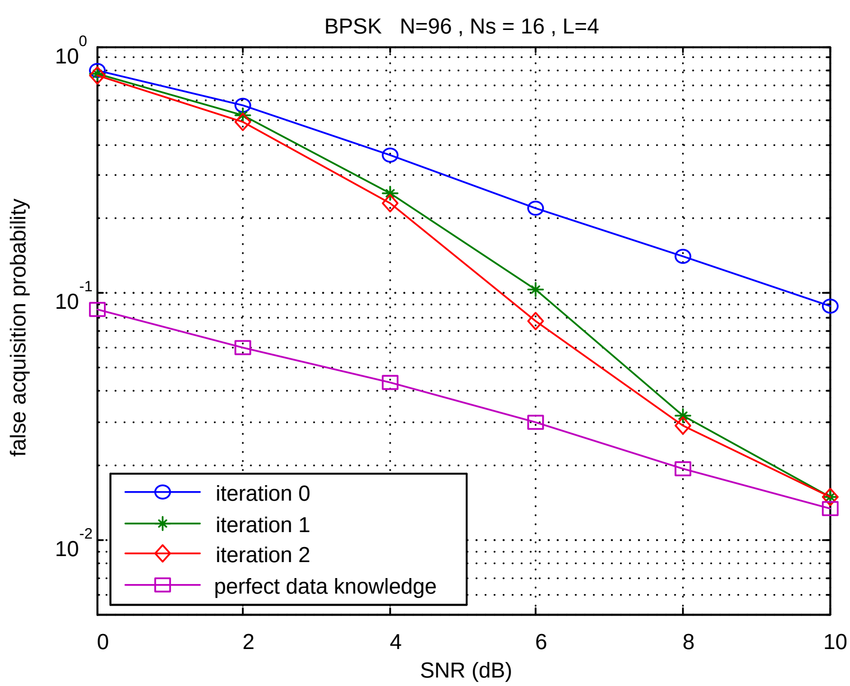


Figure 6.5. Iterative Frame False acquisition performance vs SNR (dB), L:4, BPSK

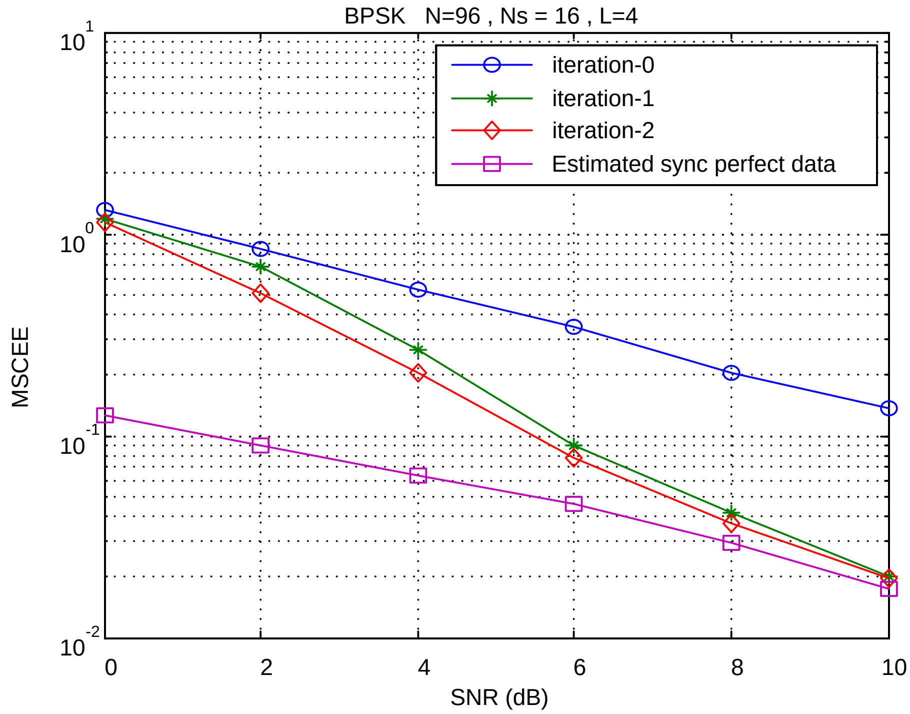


Figure 6.6. Mean square channel estimation error versus SNR, L:4, BPSK

The performance curve which is labeled as 'iteration-0' shows the frame synchronization performance when only the training sequence is used for synchronization. This situation is the same as in Section 6.2. It is also shown when iterative synchronization is used (iteration 1-2) in addition to the reference case in which the transmitted data is perfectly known and used for synchronization. When iterative synchronization is used both the MSCEE and the frame synchronization performance increase significantly for high SNR levels. Consequently the performance of the iterative frame synchronization method approaches the case of perfect data knowledge as the SNR increases.

Note that, as the iteration number increases there is not much increase in the performance of the system, but complexity increases linearly.

6.4 Simulation Results of the Generalized Frame Synchronizer

In this simulation, the performance of the generalized frame synchronizer which is proposed by Robertson in [9] is shown. In [9], the list synchronization concept was introduced. A list synchronizer has a two stage structure. First stage generates a list of most possible '*v*' frame starting points. Second stage distinguishes the true frame starting point from the wrong ones among the list. The measure of performance is the probability that the correct sync word position does not correspond to the best *v* likelihood functions. Also the random data limited upper bound is calculated for the system parameters used in the simulations.

$$P_{RDL}(v) = \sum_{j=0}^{M} \min(1, \frac{v}{j+1}) D_j Q^{-(N-L)},$$

In Figure 6.7, the performance of the list synchronizer is shown for which the system has a packet length of N = 42 and a synchronization word length of N_s = 7 using BPSK modulation on the AWGN channel.

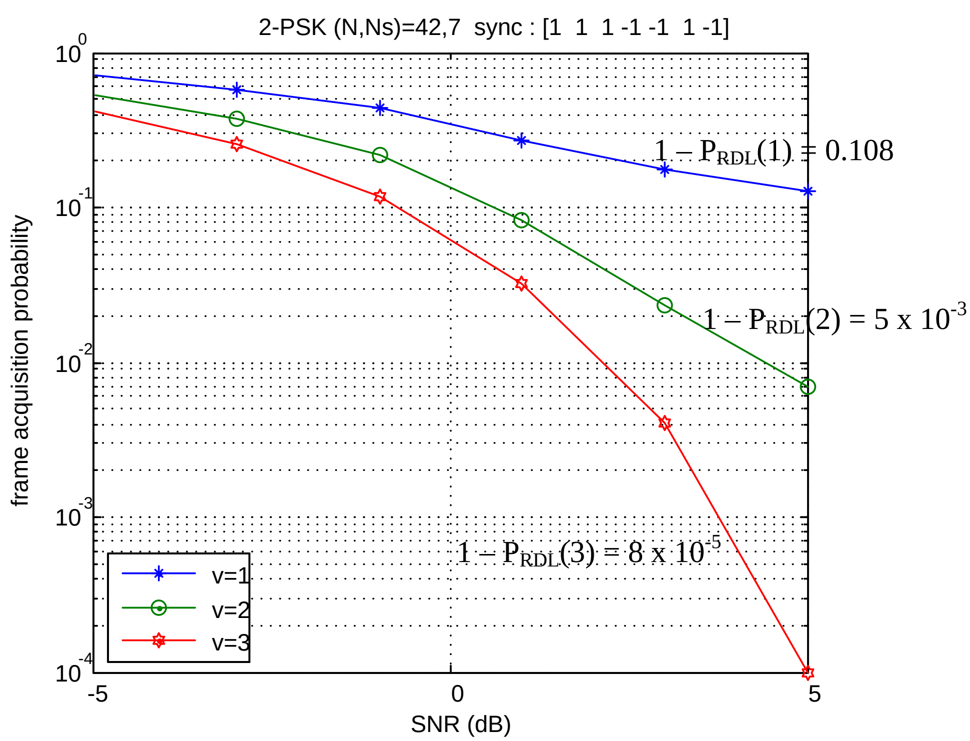


Figure 6.7. Frame acquisition probability vs SNR

It can be easily seen that the gain is high for higher SNR levels. This is because of the increase in P_{RDL}.

In Figure 6.8, the packet length is chosen as N = 133 and synchronization word length is chosen as N_s = 13. Similar results are obtained as in Figure 6.7. Note that for higher values of v and N_s random data limited upper bound is higher and the system performance is less affected by the limit.

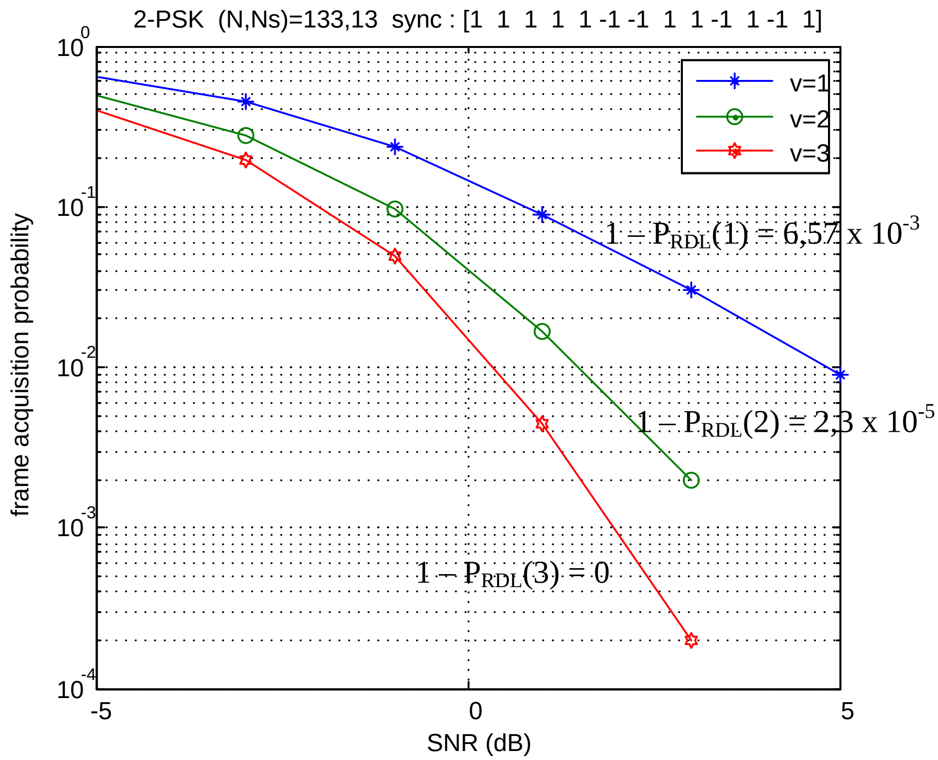


Figure 6.8. Frame acquisition probability vs SNR

6.5 Performance of the Decoder Assisted Frame Synchronizer

In Section 6.5.1 performance of the decoder assisted frame synchronization method proposed by Howlader [10] for AWGN channels (Section 4.2) and the modified version of it (Section 5.1) for frequency selective channels are evaluated. The system parameters are coded packet length ‘*N*’, synchronization word length ‘*Ns*’, and the list synchronization length ‘*K*’. In the figures, the curve which is labeled as ‘Wang’, is the performance curve of the method proposed by Wang, Shi, for frequency selective channels [6].

6.5.1 Performance of Howlader’s Frame Synchronizer for AWGN Channels

In Figure 6.9, performance of the Howlader’s frame synchronizer for AWGN channels is shown. In figures, the curve labeled as ‘detection’ shows the performance of the system without making any assumption, hence it shows the real performance of the system. The curve labeled as ‘hypothesized’ is the performance of the system when the assumption in [10] is made. The total packet size of the encoded data and encoded sync sequence is N=102. The convolutional coding employed in the system has a rate r = 1/2 and constraint length 3. There are 40 bits of uncoded information in the packet, 7 bits of uncoded sync sequence and 4 bits of flush bits (‘0’s that bring the Viterbi states at a known state at the receiver) 2 at the beginning of the sync word and 2 at the end of the packet. Hence there are total of 51 uncoded bits in the packet.

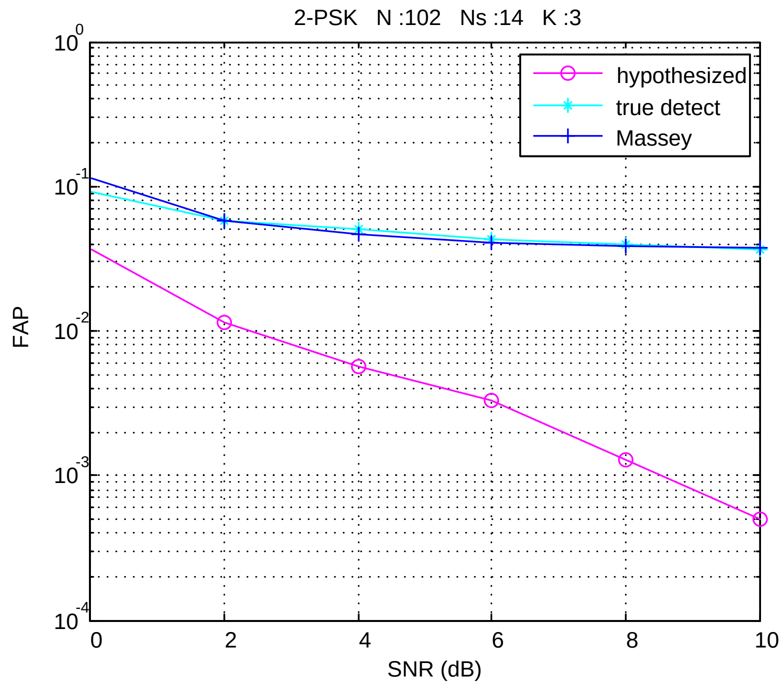


Figure 6.9. Decoder assisted frame synchronizer's performance for AWGN channel

From the difference between the 'true detection' and the 'hypothesized' curves, it is appeared that the false estimation of the frame starting location significantly increases due to the repetition of the sync state sequence in the random data state sequence. But the receiver assumed that the sync is achieved which is not the case in the realistic sense.

6.5.2 Performance of the Modified Howlader's Frame Synchronizer for Frequency Selective Channels (CAFS1)

False frame acquisition performance of the proposed method 1 (CAFS1) is evaluated in this section for different packet lengths, synchronization word lengths and the channel taps length. From Figures 6.10 through 6.14, three performance curves are evaluated in each figure, hypothesized, detection and the wang3 curves. The curve labeled as 'hypothesized' is the performance of the system when the assumption in [10] is made. In [10], it is assumed that the sync is achieved when the synchronization state sequence is found at the Viterbi decoder. Thus the frame acquisition probability is excluded from the repetition effect of the sync state sequence in the random data state sequence at the Viterbi decoder. The curve labeled as ‘detection’ shows the performance of the system without making any assumption, hence it shows the real performance of the system. The ‘Wang3’ curve shows the probability that the correct sync word position does not correspond to the best ‘K’ likelihood functions. ‘K’ is the length of the list in the first stage of the frame synchronizer.

The frequency selective channel coefficients are modeled as i.i.d. complex Gaussian random variables with zero mean and variance 1/L. ‘L’ is the number of the channel taps for the frequency selective channel and chosen as L = 4 and L = 2. For the MLSEEQ block, hard-output Viterbi equalizer is used in the simulations.

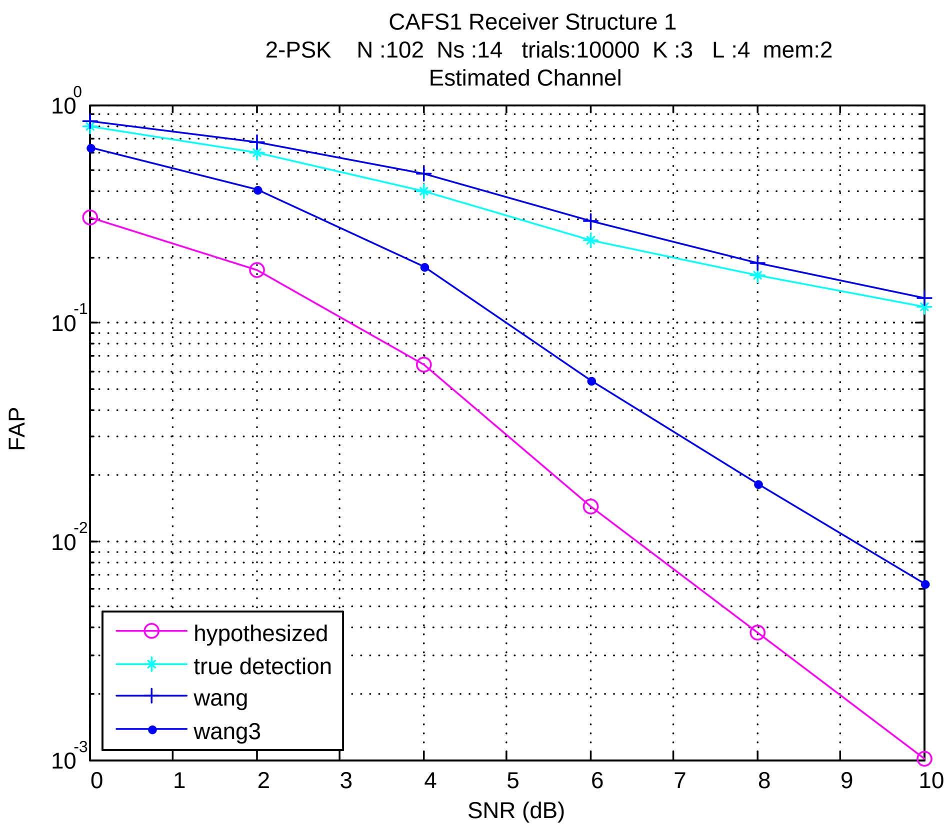


Figure 6.10. FAP vs SNR (dB) for CAFS1

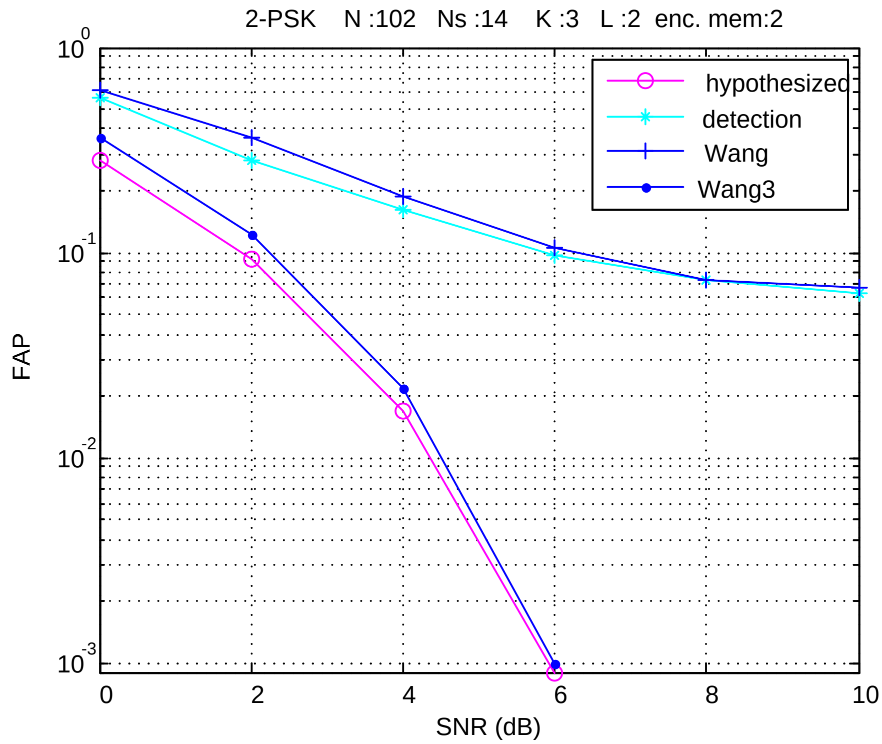


Figure 6.11. FAP vs SNR (dB) for CAFS1

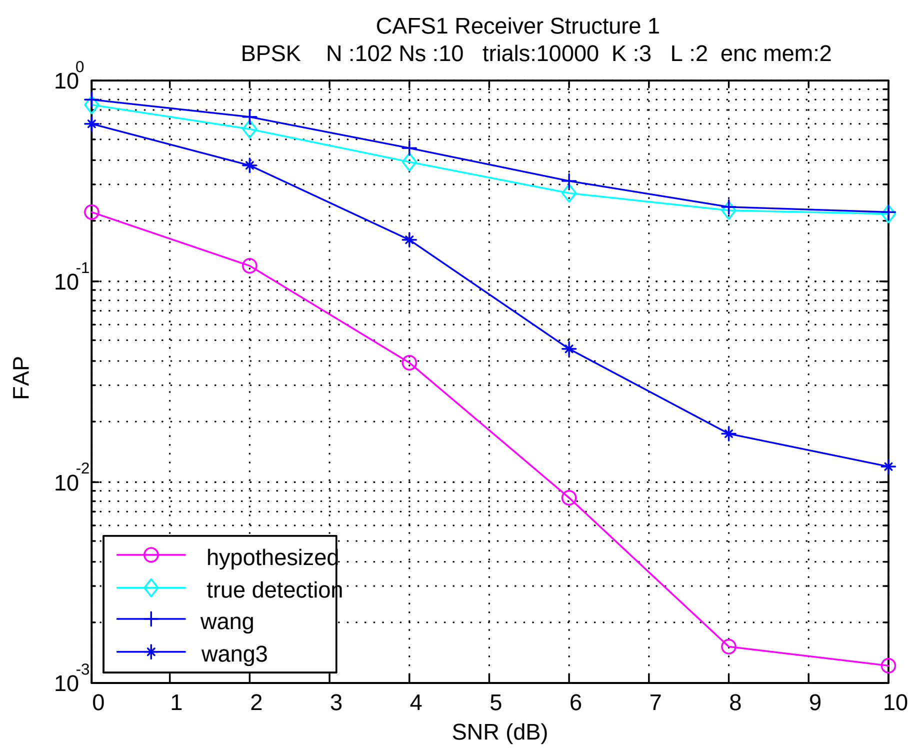


Figure 6.12. FAP vs SNR (dB) for CAFS1

From Figures 6.10 through 6.12 the FAP performance of the systems are shown for different synchronization word lengths and channel tap lengths. It is easy to see that the true detection rate can not exceed the Wang3 curve, because the first stage of the synchronizer only outputs the best possible K-ordered list, which K is 3 here. This means that the false alarm rate is very high.

The error in the estimated channel parameters results a poor performance of the hard-output Viterbi equalizer. Therefore false alarm increases beyond the limit. The limit is the wang3 curve in the figures.

In the case of perfect channel estimation:

In Figure 6.13, the FAP performance is shown for CAFS1 receiver structure 2 in the case of perfectly known channel at receiver.

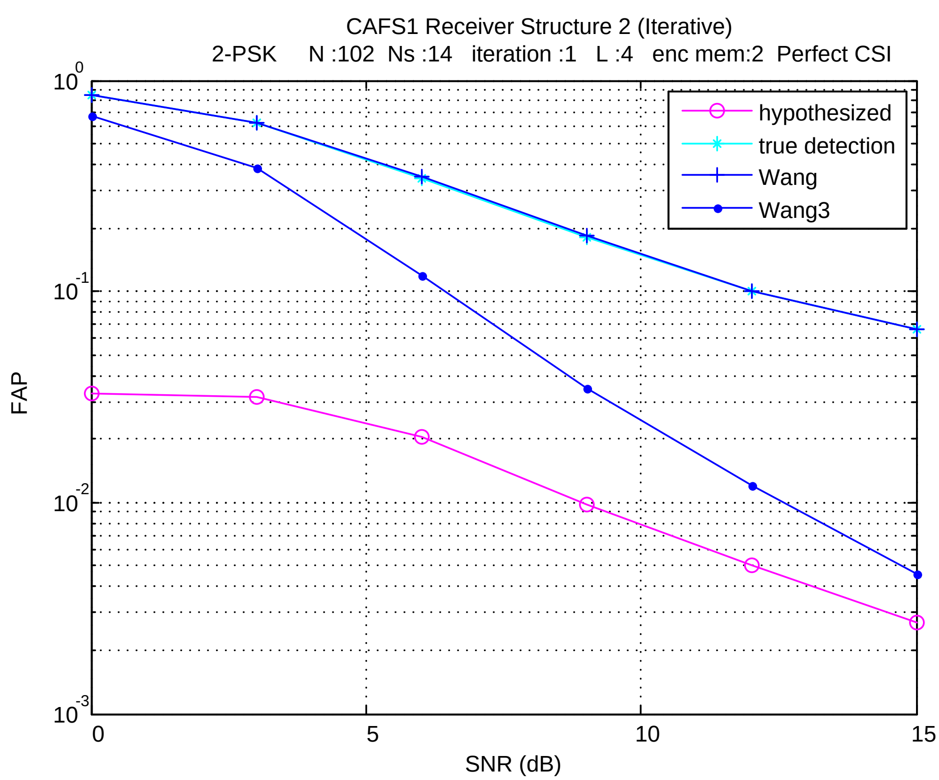


Figure 6.13. FAP vs SNR(dB) for CAFS1, Perfectly Known Channel at Receiver

Even in the case of a perfect channel side information, the FAP performance of the proposed system (CAFS1, receiver structure 2) is not better than the Wang's method of frame synchronization. Also we can see from the figure that false alarm rate , the difference between the 'hypothesized' curve and the 'true detection' curve , is very high. In conclusion we can say that the receiver structure 2 for CAFS1 is not practical.

In Figure 6.14 FAP performance of the proposed structure (CAFS1 receiver structure 1) is shown with perfect channel side information at receiver.

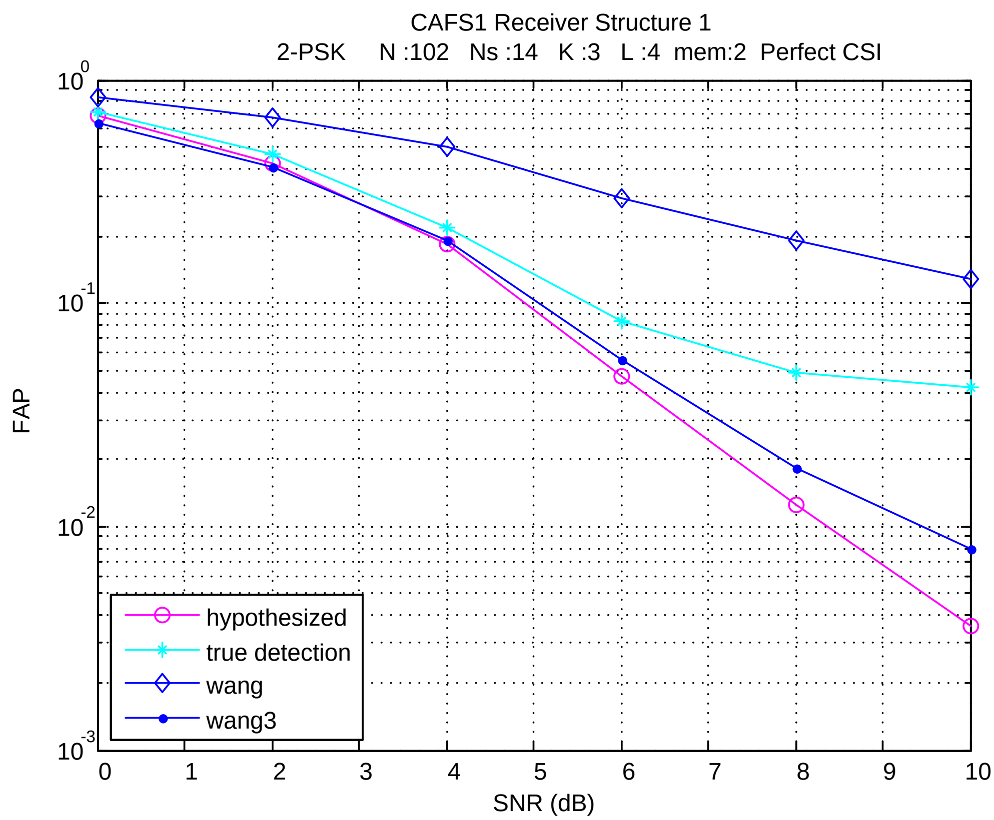


Figure 6.14. FAP vs SNR (dB), for CAFS1 Perfectly Known Channel at Receiver

If the channel is known perfectly at the receiver side MLSEEQ block gives better hard decision outputs to Viterbi block. From Fig. 6.14, it is seen that false alarm rate decreases in comparison with the estimated channel situation. But still in the high SNR region false alarm rate continues increasing that means the difference between the 'hypothesized' curve and the 'true detection' curve increases.

6.6 Performance of the Frame Synchronization Algorithm for Coded Systems

In Section 6.6.1 performance of the code aided frame synchronization method proposed by Cassaro [11] for AWGN channels (Section 4.3) and the modified version of it (Section 5.2) for frequency selective channels are evaluated. The system parameters like coded packet length 'N', synchronization word length 'Ns', list synchronization length 'v' and the complexity reduction threshold ' τ ' are shown at the title part of the figures.

6.6.1 Performance of Cassaro's Frame Synchronizer for AWGN Channels

In Figures 6.15 through 6.16 there are 3 false frame acquisition performance curves in each of the figures, proposed method (CAFS2) performance curve, Massey's optimum method of frame synchronization performance curve and the 'v = 4' curve which is introduced in [9]. 'v = 4' curve is a FAP performance limit for a system which uses two stage receiver structure to estimate frame starting position. The performance of the Massey's rule is included in the figures for the purpose of performance comparison with the proposed method. In this section all simulations are run for 10.000 Monte Carlo simulations.

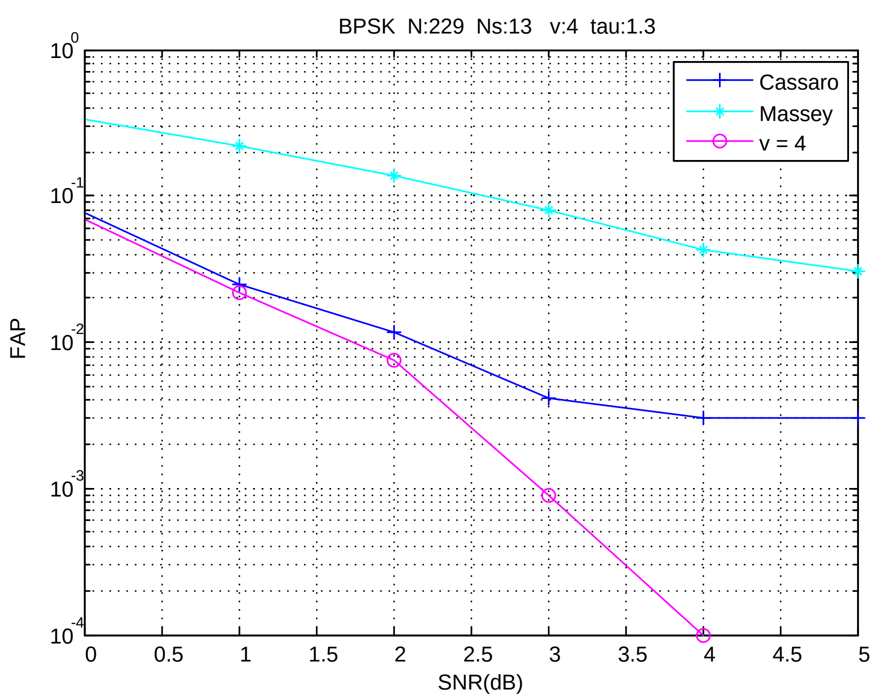


Figure 6.15. FAP vs SNR, with complexity reduction

Remember that the Cassaro's structure is composed of two stages. First stage outputs a list of most probable frame starting locations and the second stage chooses the true frame starting location among that list. The performance curve which is labeled as 'Massey' is the false frame acquisition performance of the first stage. It gives an idea about the performance improvement when employing decoding in the process of frame synchronization. The 'v = 4' is the probability that the correct sync word position does not correspond to the best *v* likelihood functions. So the best expected performance of the system is this 'v = 4' curve.

In Figures 6.15 and 6.16 similar system parameters are used, where the synchronization word length is Ns=13 and the list length is *v*=4 for both of the simulations. The only difference between the simulations is that the complexity reduction method is employed in Figure 6.15.

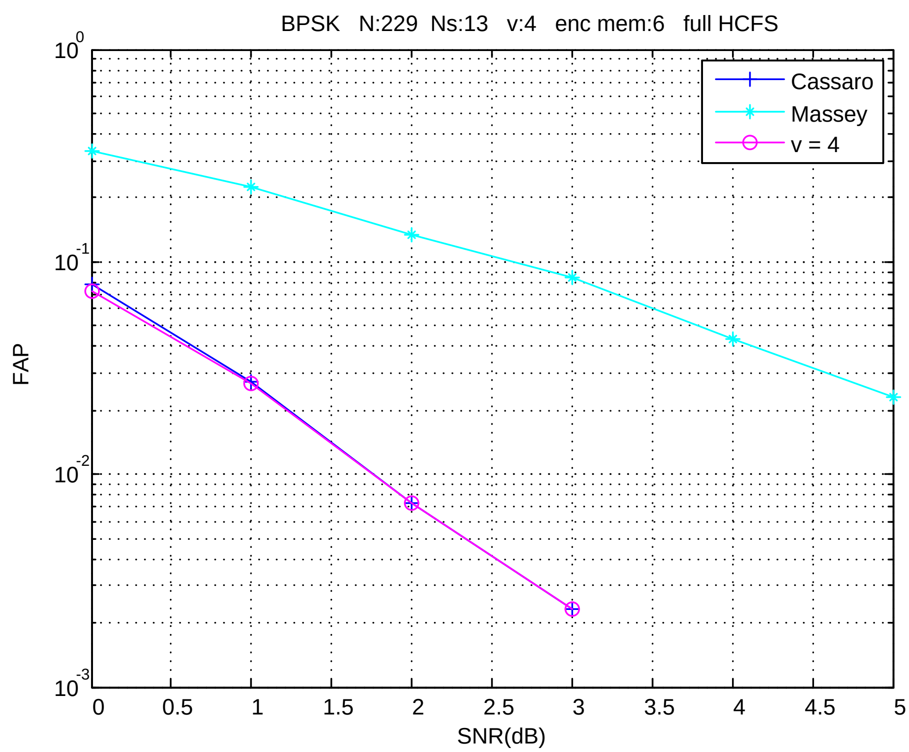


Figure 6.16. FAP vs SNR, without complexity reduction

Observing the performance difference in Figures 6.15 and 6.16, the effect of complexity reduction can be seen. It can be easily seen that complexity reduction is worsen the performance for high SNR region. Because as the SNR increases, usage of High Complexity Synchronizer decreases, and the performance comes close to Massey's frame synchronizer which is limited by the random data upper limit.

6.6.2 Performance of the Modified Cassaro's Frame Synchronizer for Frequency Selective Channels (CAFS2)

In computer simulations, the false frame acquisitions of the proposed joint channel estimation, frame synchronization and decoding algorithms are evaluated. It is the second proposed frame synchronization method which is called CAFS2 (Section 5.2). The frequency-selective channel coefficients are modeled as i.i.d. complex Gaussian random variables with zero mean and variance 1/L. In this section all simulations are run for 10.000 Monte Carlo simulations. In Figures 6.17 through 6.30 the curve labeled with 'CAFS2' is the performance curve of the proposed structure, the curve labeled with 'Wang' is the performance curve of the Wang's method for the corresponding system parameters, and the curve labeled with 'v=4' (best 4 frame starting point candidates, in all simulations this parameter is chosen as 4) is the performance curve that the proposed structure can achieve theoretically. Remember that the proposed structure has two stages. First one produces the possible frame starting points and the second stage chooses among them. So second stage can not be able to find if the true frame boundary is not outputted from the first one. By examining the 'Wang' curve and the 'v=4' curve, we can have an idea about the FAP performance of the proposed structure.

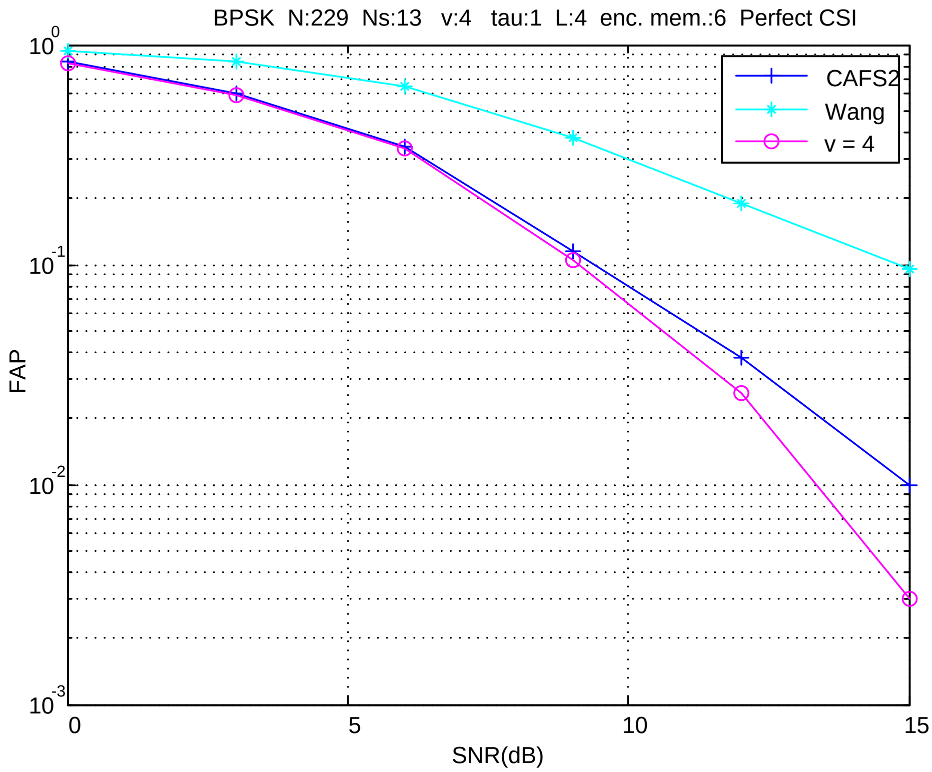


Figure 6.17. FAP vs SNR with Perfect CSI at Receiver

The effect of the complexity reduction algorithm for CAFS2 is shown in Figure 6.17. The system parameters are shown in the title part of the figure. In the simulation, ‘tau’ is chosen as 1 and its meaning is explained in Section 4.3.2 . All the system parameters are the same for the simulations for which their results are shown in Figures 6.17 and 6.18. It is understood from the performance difference between the figures, complexity reduction reduces the FAP performance of the system.

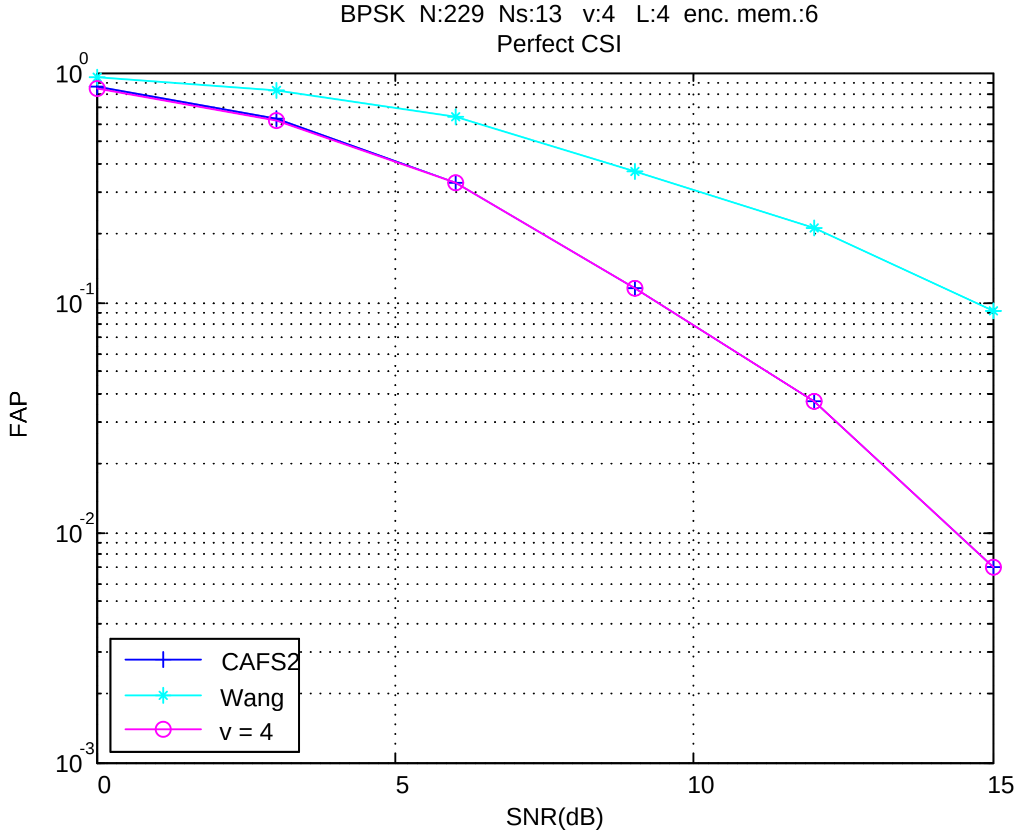


Figure 6.18. FAP vs SNR with Perfect CSI at Receiver without Complexity Reduction

In Figure 6.18 , the frame error rate performance of the proposed frame synchronizer is shown under frequency selective channel with perfect Channel Side Information (CSI), which means the channel taps are perfectly known at the receiver side. N is the total symbols in a transmitted packet, Ns is the number of synchronization symbols. It is observed that the gain in the performance of the frame synchronizer is significant such that it is indistinguishable from the 'v = 4' labeled curve.

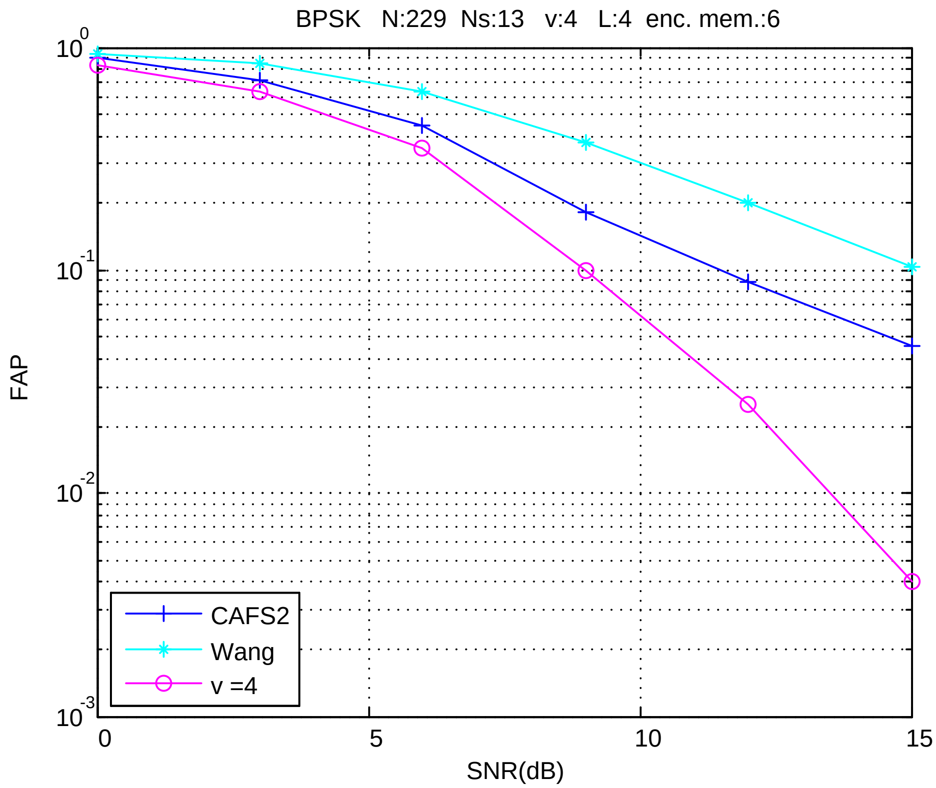


Figure 6.19. FAP vs SNR (dB) for CAFS2

In Figure 6.19, the simulation is performed with a channel estimation algorithm. The estimated channel is taken as input by the soft output Viterbi equalizer. Equalizer performance is affected by the channel estimation error and this is reflected to the final estimation of the frame boundry. This result can be seen from the performance difference between Figures 6.18 and 6.19.

In Figures 6.19 and 6.20, the only difference between the simulations parameters is the packet length ‘N’. It is seen from the figures, FAP performance degrades when the packet length ‘N’ increases because there are more candidates for the frame starting point.

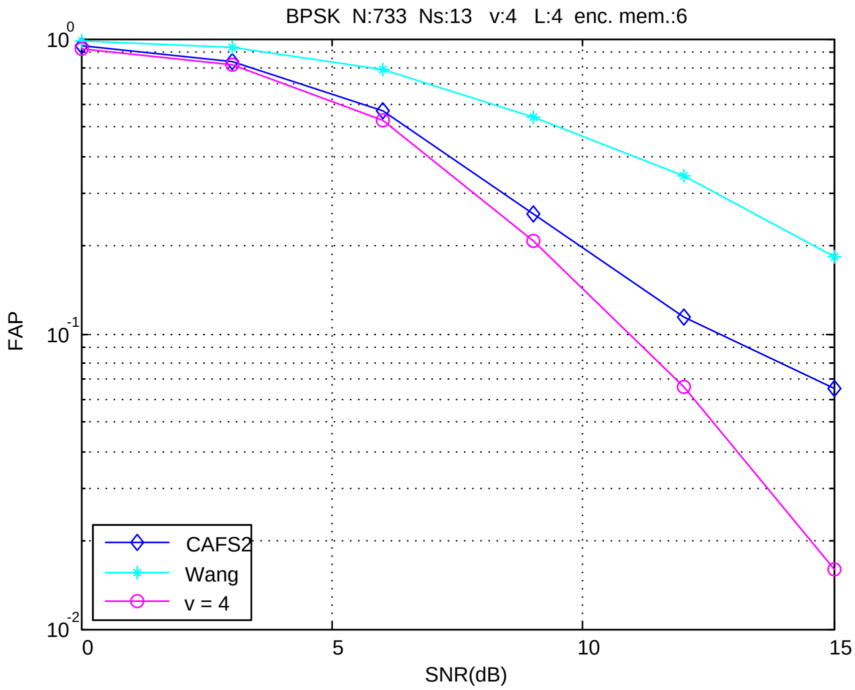


Figure 6.20. FAP vs SNR(dB) for CAFS2

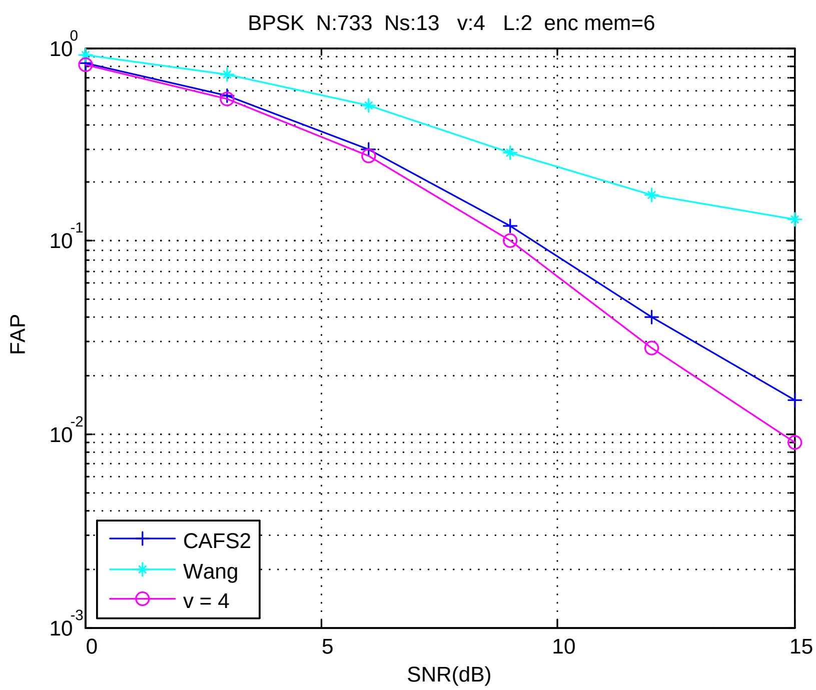


Figure 6.21. FAP vs SNR(dB) for CAFS2

The effect of different channel tap number 'L' to the FAP performance can be seen from Figures 6.20 and 6.21. As it is expected the FAP performance increases as 'L' decreases.

In Figure 6.22, FAP performance of the CAFS2 is shown for different encoder memory. As the encoder memory increases we have more reliable mode separation statistics at the receiver. So the FAP performance of the system increases.

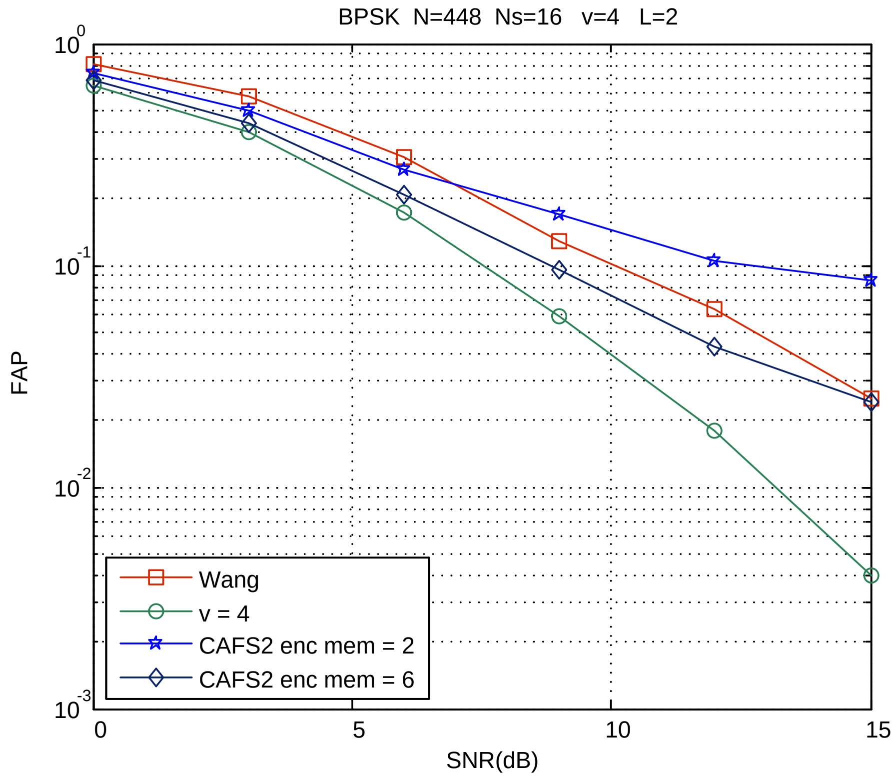


Figure 6.22. FAP vs SNR(dB) for CAFS2

QPSK vs BPSK

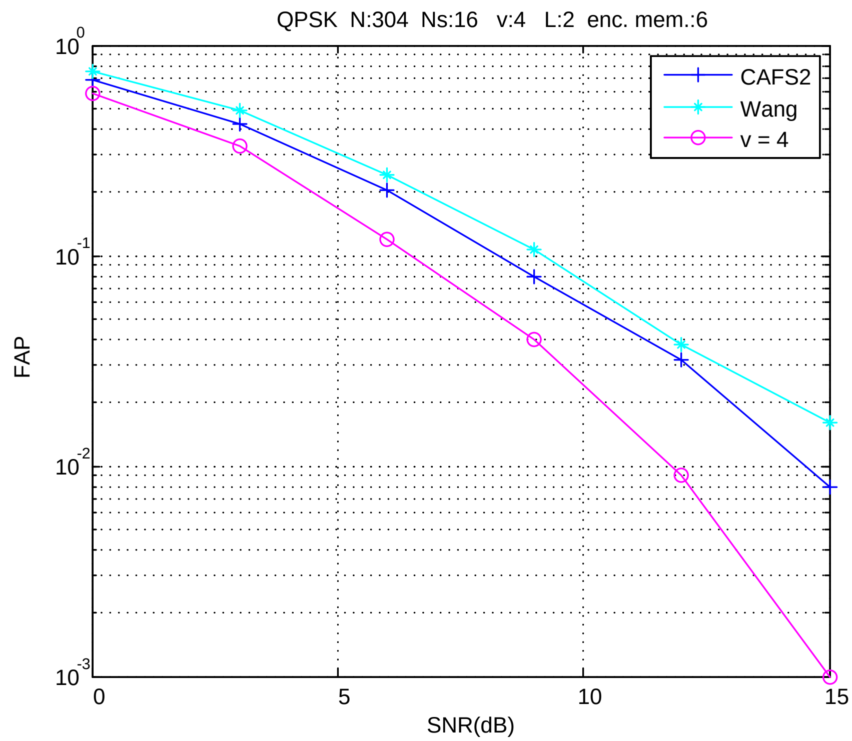


Figure 6.23. FAP vs SNR(dB) for CAFS2

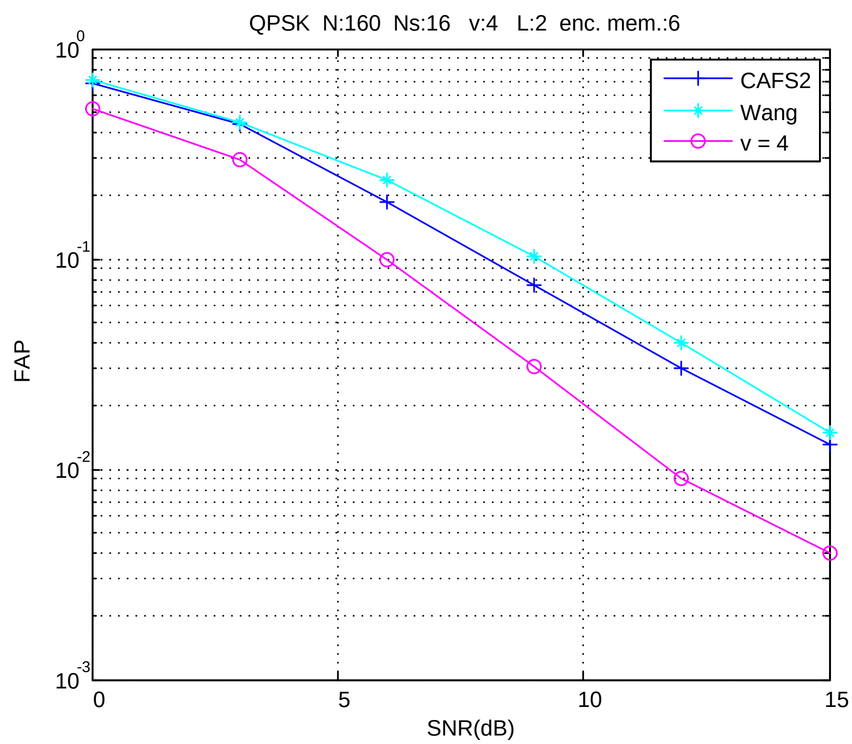


Figure 6.24. FAP vs SNR(dB) for CAFS2

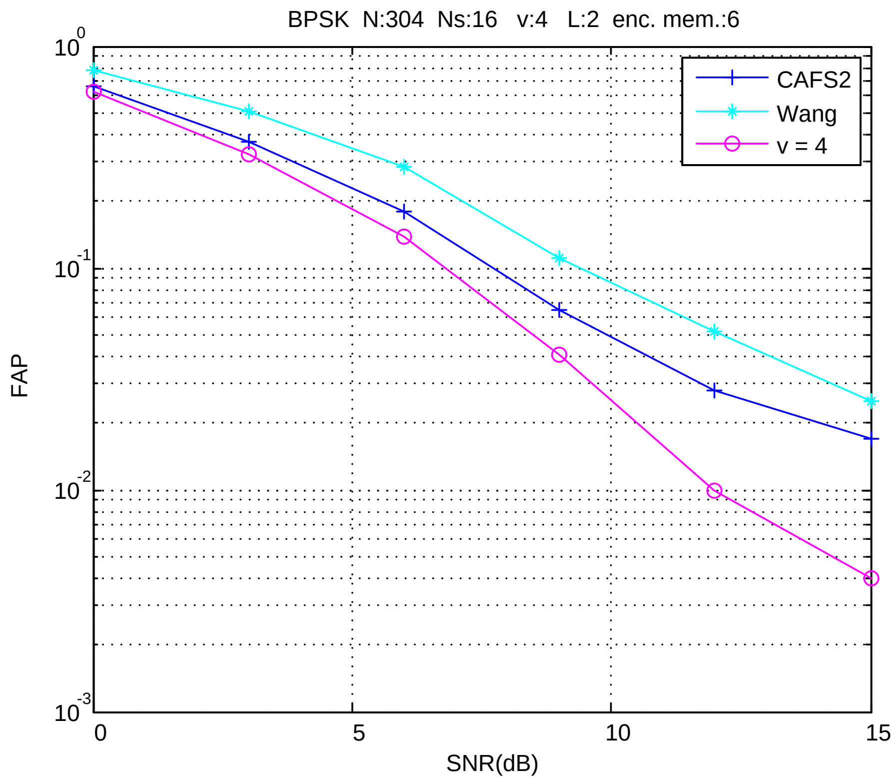


Figure 6.25. FAP vs SNR(dB) for CAFS2

The purpose of Figures 6.23 through 6.25 is to show the effect of modulation on the system. In Figure 6.23 the packet length 'N' is chosen as 304, that is 304 symbols sent through the channel for 1 packet. In Figure 6.24 'N' is chosen as 160. If we compare these two figures, we can see that the FAP performance in Figure 6.23 is slightly better. As 'N' increases we have more candidates to estimate the frame boundary but also we have more statistics about frame boundary because of increase in coded bits (synchronization word length 'Ns' is the same for both figures). The combination of these effects yields a result that can be seen from the figures.

We can see from Figures 6.24 and 6.25, the FAP performance of the systems are very close to each other. The reason of this result lies behind the mode separation algorithm. In Figure 6.24, modulation type is QPSK and the frame length 'N' is 160. This is equivalent to 288 coded information bits. In Figure 6.25, modulation type is BPSK and the frame length 'N' is 304. This is equivalent to 288 coded information bits. The coded information bits are used to compute the frame starting point statistics by the mode separation algorithm.

Multiple Frame Processing

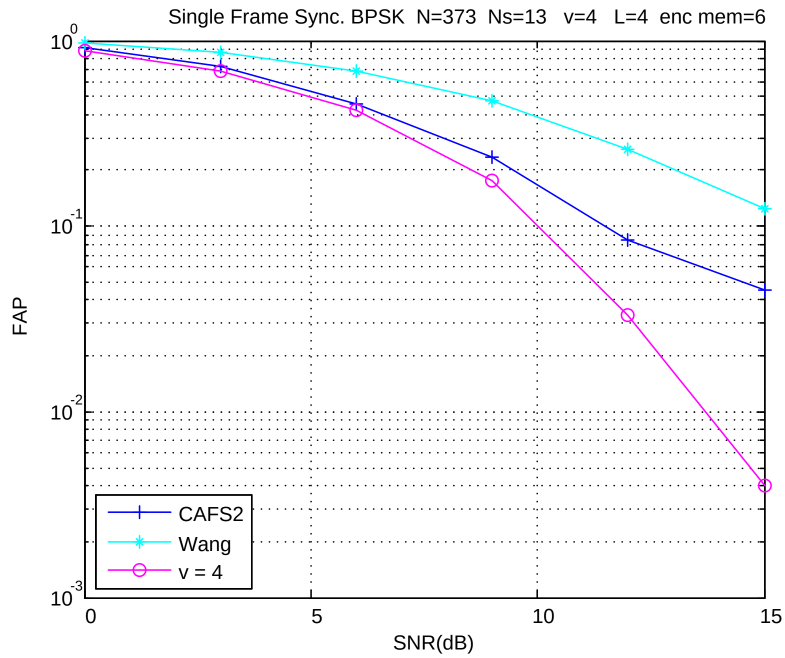


Figure 6.26. FAP vs SNR(dB) for CAFS2

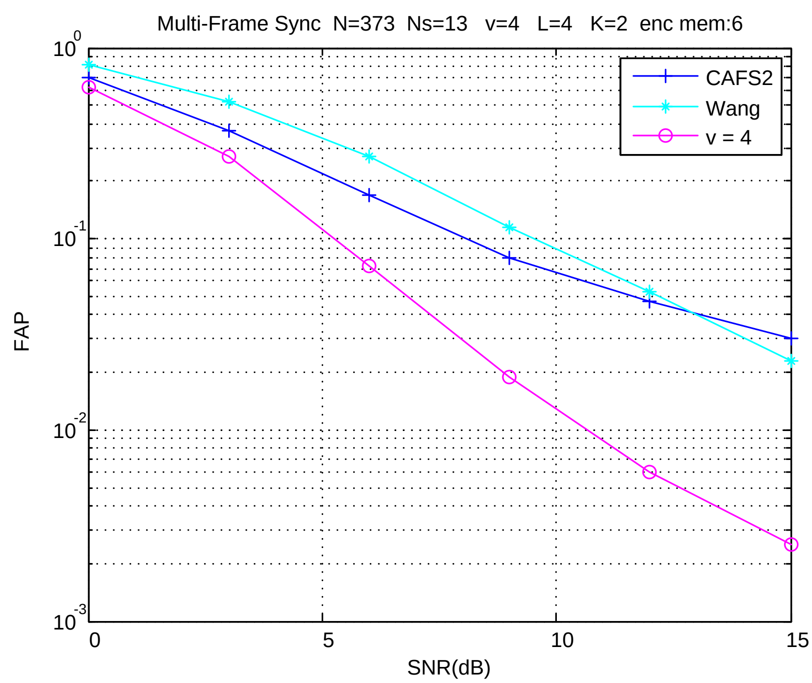


Figure 6.27. FAP vs SNR(dB) for CAFS2

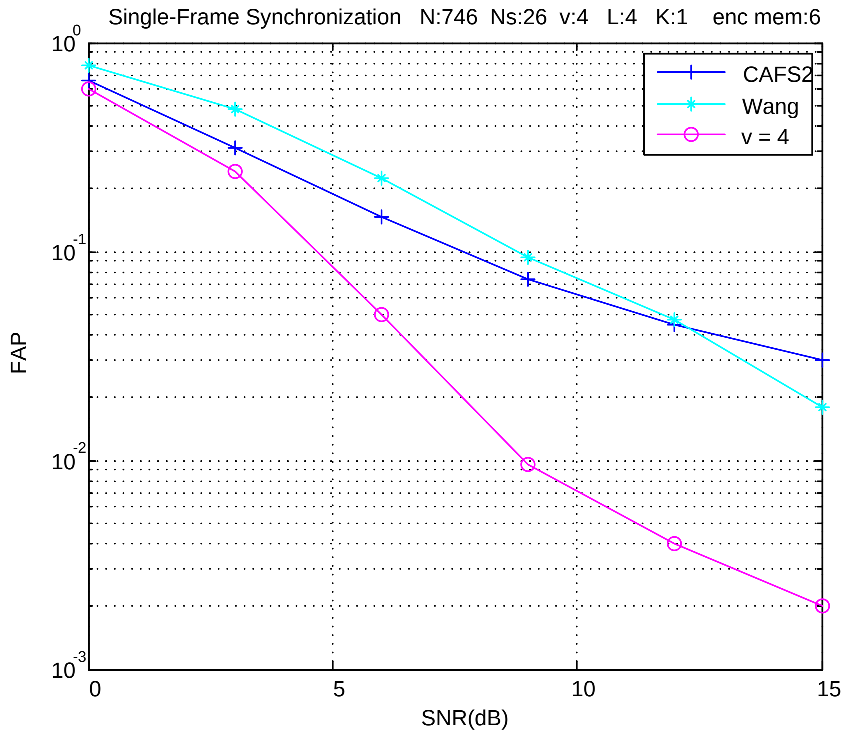


Figure 6.28. FAP vs SNR(dB) for CAFS2

The purpose of Figures 6.26 through 6.28 is to show the effect of the multiple frame processing to the FAP performance of the proposed structure. In Figure 6.27 K-frames (K=2) are processed together to find the frame starting point and it is assumed that the channel taps are constant for K-frame duration. Packet length 'N' is chosen as 373 and synchronization word length 'Ns' is chosen as 13. In the first stage of the receiver when extracting the possible frame starting points, not only one synchronization sequence is used but also the system takes advantage of both of the synchronization sequences (from the first and the second frame). This method is explained in Section 3.3. Also in the second stage mode separation algorithm uses both of the frames to extract the statistics. This approach is similar to process one frame but with parameters (N and Ns) K times larger than a multi-frame processing structure that of a single fram. In Figure 6.28, the simulation results of a single frame processing sructure is shown. This structure has a packet length N=746 and synchronization word length Ns=26 which are twice the simulation parameters which results are shown in Figure 6.27. It is seen from Figures

6.27 and 6.28, both structures have very similar performance results. If we compare Figures 6.26 and 6.27 we can see that FAP performance of the proposed method increases if we employ multiple frame processing to the system. Nevertheless the proposed system loses its advantages over Wang's method and gives poor results at higher SNR levels.

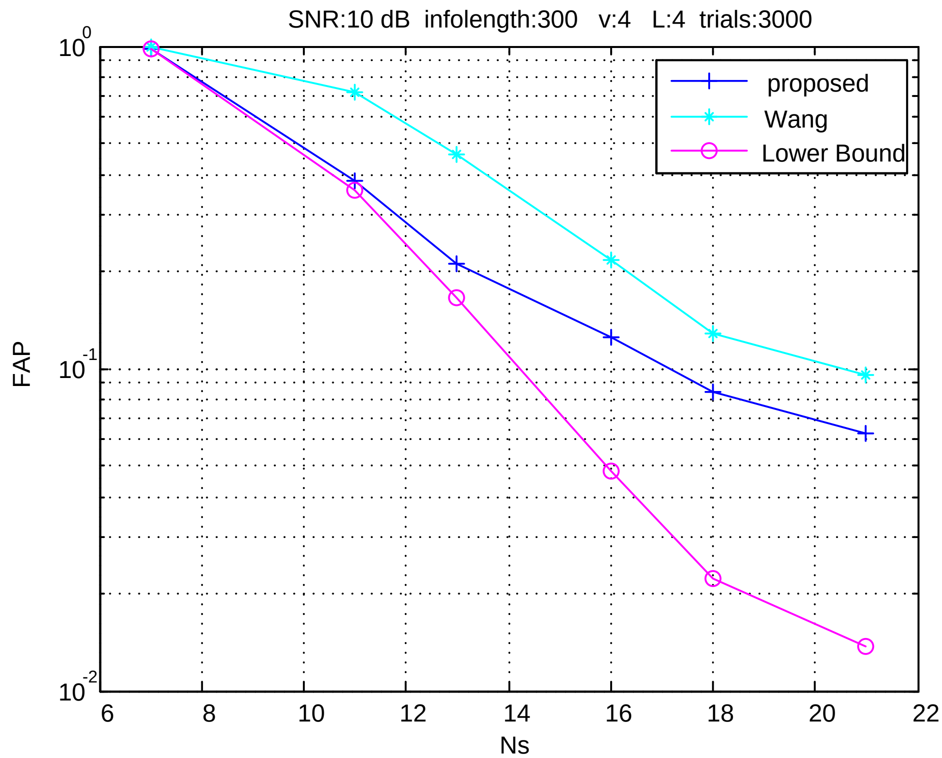


Figure 6.29. FAP vs Ns at SNR = 10 dB

In Figure 6.29, the frame error rate of the proposed frame synchronizer method (CAFS2) is evaluated against the synchronization word length '*Ns*'. In simulations information length kept constant and concentrated on the effect of '*Ns*' to the FAP performance of the system. It is observed that as the synchronization word length increases the performance of the system increases but it loses its advantage over the traditional method of frame synchronization for frequency selective channels [6]. The FAP performance difference between the the Wang's method and the proposed method (CAFS2) decreases as the synchronization word length increases.

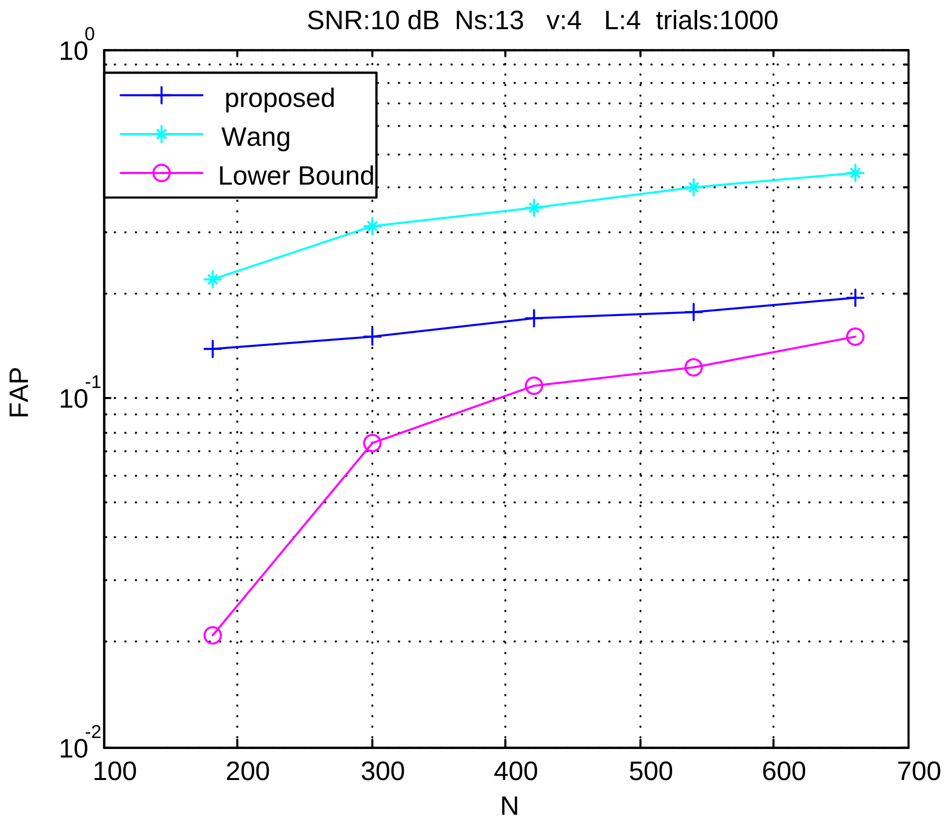


Figure 6.30. FAP vs N at SNR = 10 dB

In Figure 6.30, the frame error rate of the proposed frame synchronizer method (CAFS2) is evaluated against the coded packet length '*N*'. It is observed that as the coded packet length increases the performance of the system decreases little compared to the traditional method of frame synchronization for frequency selective channels [6]. Hence we can say that the proposed method (CAFS2) is noticeable if the packet length is relativelylong.

In the figures we observed that as the length of the synchronization word getting shorter, Wang's frame synchronization method in the ISI channel estimates the frame starting location worse. So we have a worse HPL (high probability list) and the total performance degrades. But the proposed method's power comes from the statistics obtained from the mode separation algorithm. If we left N-Ns unchanged as Ns decreases, lower bound and the Wang's curve degrade sharper than the proposed method's curve. In conclusion, if the packet length is large and the synchronization word is small compared to packet length, then the proposed method has a considerable performance in the frequency selective channel condition in an expense of complexity increase.

CHAPTER 6

CONCLUSION

In this thesis, several frame synchronization methods have been examined both for AWGN channels and frequency selective channels. A review of frame synchronization methods has been given. By combining some of the efficient existing methods two new frame synchronization methods for frequency selective channels are proposed. The proposed methods use the power of coding that exists in the frame structure to aid in frame synchronization.

The classical problem of frame synchronization is widely investigated in the literature and optimum rules based on the maximum likelihood are derived for AWGN channels and for frequency selective channels. Most of the methods use periodically inserted known sequence (called synchronization word or preamble) into the data. In the receiver, the process of searching synchronization word in the noisy data is performed. This process is called correlation. There exist methods which use not only the synchronization word but also the power of coding to enhance the synchronization performance. Although the classical correlation rule is attractive due to its simplicity, there is a need for improved frame synchronization performance due to recent improvement in BER performance of the coded systems.

In the literature, there exist frame synchronization techniques designed for AWGN channels which use the power of coding utilized for both transmission reliability and frame synchronization. Existing work shows that the code aided syncronization methods further enhanced the synchronizer performance. Some of proposed methods are very powerful in the aspect of frame acquisition probability performance. But there is not much work on code aided frame synchronization for frequency selective channel. The proposed methods are principally developed for the purpose of filling this gap.

One of the code aided frame synchronizer technique which was developed for AWGN channel is the Howlader's frame synchronizer [10]. The first one of the methods proposed in this thesis for code aided frame synchronizer over a frequency selective channel is based on Howlader's method. By making modifications, the method was adopted to the frequency selective channel case. With the help of Wang's method of frame synchronization in a frequency selective channel environment and a maximum likelihood sequence equalizer, Howlader's structure has been adopted to the frequency selective channel condition. The proposed system performance seems to be adequate and better than that of Wang's optimum rule under the assumption that the synchronization word repetition in the random data is excluded from the simulation results that this assumption is exactly the same as Howlader's asumption in his work. The frame acquisition process suffers from the appearance of the synchronization word in a random data sequence, as does our proposed scheme. Additional simulations have been run by taking into account this effect. There was a considerable performance improvement if the channel parameters were perfectly known at the receiver. But if the channel taps are unknown at the receiver, the proposed algorithm seems to have poor results due to the estimation error of the channel taps.

The second method proposed in this thesis for code aided frame synchronization over a frequency selective channel is based on Cassaro and Georghiades' method [11]. Cassaro and Georghiades' frame synchronization algorithm achieves a performance approaching the limit that is introduced by Robertson [9]. Robertson proposed a general structure for frame synchronization and gave insight about the list synchronization method which can be applied to any coded systems. Cassaro and Georghiades' showed that Robertson's list synchronization concept is realizable. With modifications to Cassaro's structure, similar to that which was made for the first proposed method, the second method for code aided frame synchronizer over frequency selective channel was devised. Wang's method of frame synchronization was used for the first stage and soft output Viterbi equalizer was used to equalize the channel. For the second stage of the synchronizer, a soft output Viterbi decoder was used for extracting the mode separation statistics. The proposed system has a very good performance when the channel is known perfectly at the receiver. The performance of the system is considerably improved for some system parameters for the case of an unknown channel. As the packet length increases while the sync word length is kept constant, the second proposed frame synchronizer gives better performance results over a frequency selective channels with unknown channel taps. If the channel taps are perfectly known, the second proposed frame synchronizer achieves a performance approaching the limit that is introduced by Robertson [9].

Out of the two proposed frame synchronizers, the second one (CAFS2) seems to perform better because of its better frame acquisition performance in the case of unknown channel parameters. The performance difference between the first and second proposed methods is due to the random data limited upper bound. The second proposed method does not suffer from random data limited upper bound unlike the first one. The second proposed method does not search a known sequence in the receiver unlike the first one. The first proposed frame synchronizer which is based on Howlader's scheme searches a known sequence of state transitions.

The power of coding is used more and more in the different parts of the receiver to enhance the overall performance of the system. In our case coding is employed to enhance the performance of the frame synchronization. Considerable improvement has been achieved for relatively longer packet lengths for the second proposed method (CAFS2).

REFERENCES

[1] *James L. Massey, "Optimum Frame Synchronization" IEEE Transactions on Communications, vol. com-20, no. 2, pp. 115-119, April 1972*

[2] *Gee L. Lui, Harry H. Tan, "Frame Synchronization for Gaussian Channels" IEEE Transactions on Communications, vol com-35, no. 8, pp. 818-829, Aug. 1987*

[3] *P. Tolstrup Nielsen, "Some Optimum and Suboptimum Frame Synchronizers for Binary Data in Gaussian Noise " IEEE Transactions on Communications, pp. 770-772, June 1973*

[4] *P. Robertson, "Optimal Frame Synchronization for Continuous and Packet Data Transmission " Phd. Thesis, June 1995*

[5] *B. K. Levitt, "Maximum Likelihood vs Threshold Frame Synchronization" JPL Deep Space Network Progress Report, pp. 84-87*

[6] *Yan Wang, Kai Shi, Erchin Serpedin, "Continuous Mode Frame Synchronization for Frequency Selective Channels" IEEE Transactions on Vehicular Technology, vol. 53, no.3, May 2004*

[7] *U. Lambrette, J. Horstmannshoff, H Meyr, "Techniques for Frame Synchronization on Unknown Frequency Selective Channels"*

[8] *Ayman Abdel-Samad, "Iterative Frame Synchronization for Frequency Selective Channels," pp 539-542, 2005*

[9] *P. Robertson, "A generalized frame synchronizer," Proc. Global Telecommunications Conf., vol. 3, pp. 1606-1611, Dec 1993*

[10] *M. Mostafa K. Howlader "Decoder Assisted Frame Synchronization for Packet Transmission," IEEE Journal on Selected Areas in Communications., vol. 19, no. 12, Dec 2001*

[11] *Thomas M. Cassaro, Costas N. Georghiades, "Frame Synchronization for Coded Systems Over AWGN Channels" IEEE Transactions on Communications, vol. 52, no. 3, pp. 484-489, March 2004*

[12] *H. Wymeersch, H. Steendam, H. Bruneel, M. Moeneclaey, "Code Aided Frame Synchronization and Phase Ambiguity Resolution" IEEE Transactions on Signal Processing, vol. 54, no. 7, July 2006*

[13] *W. Suwansantisuk, M. Chiani, M. Z. Win, "Performance Measure for Frame Synchronization Techniques"*

[14] *Linda M. Davis, Peter Hoeher, "Joint MAP Equalization and Channel Estimation for Frequency and Frequency-Flat Fast-Fading Channels " IEEE Transactions on Communications, vol. 49, no. 12, pp. 2106-2114, Dec 2001*

[15] *J. G. Proakis, "Digital Communications", 3 ed. 1995*

[16] *G. David Forney, "Maximum Likelihood Sequence Estimation of Digital Sequences in the Presence of Intersymbol Interference", IEEE Transactions on Information Theory, vol. IT-18, no. 3, May 1972*

[17] *J. Hagenauer, P. Hoeher, "A Viterbi Algorithm with Soft Decision Outputs and its Applications" IEEE pp. 1680-1686, 1989*

[18] *G. L. Lui and H. H. Tan, "Frame synchronization for direct detection optical communications systems" IEEE Transactions on Communications, vol. COM-34, pp. 227-237, Mar. 1986*

[19] *B. H. Moon, and S. S. Soliman, "ML frame synchronization for the Gaussian channel with ISI," in Proc. ICC'91, Denver, CO, 1991, pp. 1698-1702*

[20] *J. A. Gansman, M. P. Fitz and J. V. Krogmeier, "Optimum and Suboptimum Frame Synchronization for Pilot-Symbol-Assisted Modulation" IEEE Transaction on Communications, vol. 45, no. 10, pp. 1327-1337, October 1997*

[21] *N. Seshadri and C. E. W. Sundberg, "Generalized Viterbi algorithms for error-detection with convolutional codes," in Proc. GLOBCOM '89, pp. 1534-1538, December 1989*