IMPROVING THE BIT ERROR RATE PERFORMANCE OF A GSM SYSTEM USING A INDEPENDENT COMPONENT ANALYSIS BASED NON-LINEAR FILTER

by

KUMARABHIJEET SINGH

Presented to the Faculty of the Graduate School of

The University of Texas at Arlington in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

THE UNIVERSITY OF TEXAS AT ARLINGTON

May 2008

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my thesis advisor Dr. Venkat Devarajan for involving me in this project and helping me in each step during the course of my work. This thesis originated as an idea from Dr. Devarajan. I have tried to pursue it to its logical conclusion under his constant guidance. His encouragement and advice kept me highly motivated. It has been a great endeavoring experience to work with Dr. Devarajan.

I would like to sincerely thank my thesis committee faculty members Dr. Stephen Gibbs and Professor Bernard Svihel for their time and cooperation.

I would like to thanks all my colleagues for their timely advice and support. I would also like to thank my roommates for their constant support.

Finally, I would like to thank my family who have always been supportive and given me the best possible education.

Above all, I would like to thank GOD because of whose blessing I have reached so far in life.

March 5, 2008

ABSTRACT

IMPROVING THE BIT ERROR RATE PERFORMANCE OF A GSM SYSTEM USING A INDEPENDENT COMPONENT ANALYSIS BASED NON-LINEAR FILTER

Kumarabhijeet Singh, MS

The University of Texas at Arlington, 2008

Supervising Professor: Dr. Venkat Devarajan

Presently, call drop out is considered a major problem in wireless communication. A large percentage of call drops take place due to network congestion, handover failure and high bit error rate.

Our main objective is to reduce call drops due to high bit error rate using a newly proposed signal processing block (SPB) at the receiver section of a mobile handset. The SPB helps improve the performance of the wireless system, particularly Global Systems for Mobile (GSM) by providing a coding gain over the existing system.

In this work, we carried out a simulation of the physical layer of a GSM system and a performance analysis in the presence of our new signal processing block. More

specifically, we developed and incorporated the subsystems to generate the two mixtures of signal and noise, centering and whitening. This was our contribution to the development of the SPB. In addition, we also incorporated in the simulation diversity, channel coding, phase estimation and channel compensation. The simulation results prove that an improvement in the BER is achieved over many typical existing configurations of the physical layer a GSM wireless system.

The BER is considered to be one of the parameter during the handoff initiation procedure. One of conditions for hard handoff is that the bit error rate of the serving base station should be below a defined threshold. Hence, improving the BER that we have achieved could help reduce call drops during hard handoff.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

Table Page

CHAPTER 1

INTRODUCTION

1.1 Background

The performance of any wireless system can be measured in terms of the bit error rate (BER). Every service provider has a certain threshold set for the bit error rate (Quality of service) for which it will provide the service. If the BER increases above the set threshold, this may lead to discontinuous of the service for some time or, call drop.

Call drop out is one of the most annoying and common problems in mobile communications. A call is said to be drop when the active phone call is unexpectedly terminated. The main causes for call drop are,

- Handover (Handoff) failure
- Network congestion
- High bit error rate
- Co-channel and Adjacent channel interference
- Lack of coverage in the particular area

Anecdotal evidence suggests that a large portion of call drop happens particularly due to handover failure, network congestion and high bit error rate. Many strategies have been proposed to solve this problem particularly due to network congestion and handover failure [1,2,3,4].

1.1.1 Previous Work

Previous efforts to reduce call drop due to congestion were primarily based on,

- Dynamic channel assignment [1].
- Handoff prioritization [3].
- Call admission control [2, 4].

The previous efforts to reduce bit error rate were based on,

- Diversity (chapter 6)
- Channel coding (chapter 7)
- Channel equalizer [10]

1.2 Overview

The received signal at the handset is often corrupted by noise. This corrupted version of signal can lead to call drop outs. When the noise is dominant, it becomes necessary to reduce the noise to improve the performance by reducing the BER. In this work, we try to reduce call drop out due to high BER by introducing a new signal processing block (SPB) at the receiver section of the mobile handset.

The decrease in the BER has the effect on the handoff initiation procedure. The handoff decision is based on the BER of the serving base station and the signal strength of the serving and the neighboring base stations [5]. The handoff gets initiated if

- the bit error rate of serving base station increases above a certain threshold and
- the received signal strength indicator (RSSI) of the serving base station is below a certain threshold and the received signal strength of one of the neighboring base

station is greater than serving base station by *h* dB (h is the hysteresis level) for a predefined interval of time.

As shown in Figure 1.1, a handoff will be initiated if the bit error rate of the serving base station, in this case BS_1, is below the defined threshold T_1 and, the received signal strength of the neighboring base station, in this case BS_2, is greater than BS_1 by a hysteresis of *h,* for a predefined interval of time. Label C in Figure 1.1 denotes the stage, where handoff is to be initiated.

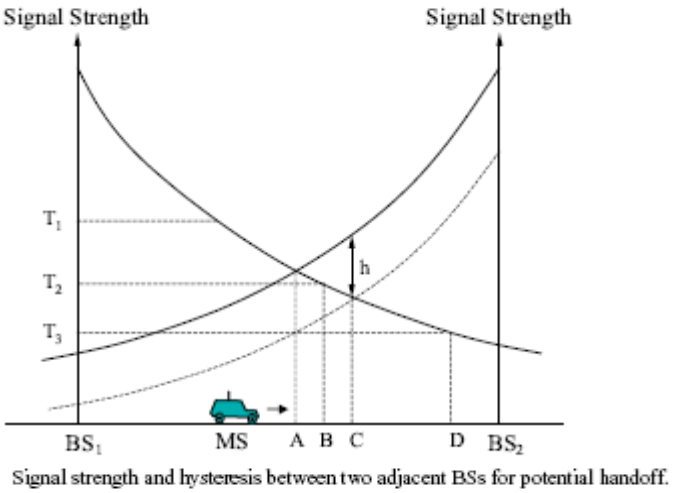


Fig.1.1 Illustration of handoff initiation

The handoff procedure will not be initiated till the BER increases above the set threshold even though the signal strength of neighboring base station is greater than the serving base station. Thus, the reduction of the BER could decrease the frequency of handoff in a cell. The reductions in the number of handoff procedure will decrease in call drops during this scenario.

We had two objectives at the start of our project. One was to fully develop the concept of the preliminary SPB developed by Dr. Devarajan (Supervisor on this thesis). We accomplished this in collaboration with Mr. Sudeep Sharma. The second major objective was to incorporate the SPB in to a realistic simulation of the physical layer of a GSM wireless communication system. Once again we accomplished this objective in collaboration with Mr. Sudeep Sharma. In this thesis, we describe specifically, the subsystems that we developed and simulated within the overall simulation and the results of that simulation. The specific subsystems were: additive white Gaussian channel (AWGN), Ricean fading channel, lock phase estimator and channel compensation and, diversity and channel coding.

The thesis is organized as follows: The details of the new signal processing block (chapter 2 and 5), physical layer simulation of a wireless system (chapter3 and 5), channel (chapter 4), diversity (chapter 6) and channel coding (chapter 7), Performance analysis of the physical layer simulation of a wireless system with the newly introduced signal processing block in terms of bit error rate (chapter 8).

CHAPTER 2

INITIAL APPROACH TO ICA BASED SIGNAL PROCESSING BLOCK

2.1 Introduction

The purpose of the signal processing block is to reduce the noise in the received signal and thereby improving the bit error rate. The received signal can be considered to be a mixture of signal and noise. While traditional approaches to reducing noise are to design linear filters, we have attempted to design a non-linear filter using the well known independent component analysis (ICA) system [7,8]. ICA is traditionally used to separate two signals from each other when two mixtures of these two signals are provided as inputs to the ICA. This is called Blind Source Separation (explained in some detail in the next section). To our knowledge no one has attempted to use ICA to separate signal from noise, possibly because it was assumed that only mixture (of the two signals – in this case signal and noise are the two "signals"). We resolved this issue by synthetically creating a second mixture of signal and noise and provided the two mixtures to the ICA system. We hypothesized that the ICA would have significant success in separating the signal and noise at which point we would still be left with the task of automatically identifying which of the two separated "signals" was truly signal and which the noise. Our initial attempt to create a signal processing block is shown in Figure 2.2 and will be explained in some detail later.

2.1.1 Blind Source Separation

The problem of recovering the original signal from two mixtures of two signals is similar to *Blind Source Separation* [7,8]. The mixture can be represented as $X=AS$, where S is the original sources signal = $[s_1 \ s_2]$, X is the mixture of two sources, A is the mixing matrix. This mixture of independent voice signals need to be separated by estimating the "un-mixing matrix", which in principle is the inverse of the mixing matrix originally used to mix the two sources.

Figure 2.1 gives the details of blind source separation. Suppose that in a room there are two people who are simultaneously speaking. The microphones give two recorded time signals, which can be denoted by $x_1(t)$ and $x_2(t)$, where x_1 and x_2 are the amplitudes which are functions of time. Each of these recorded signals is a weighted sum of the speech signals emitted by the two speakers which we denote by $s_1(t)$ and $s_2(t)$. The mixture of the recorded signal can be represented as a linear equation

$$x_1(t) = a_{11}s_1(t) + a_{12}s_2(t) \quad (2.1)$$

$$x_2(t) = a_{21}s_1(t) + a_{22}s_2(t) \quad (2.2)$$

In the matrix form, the equation 2.1 and 2.2 can be represented as

$$X = AS. \quad (2.3)$$

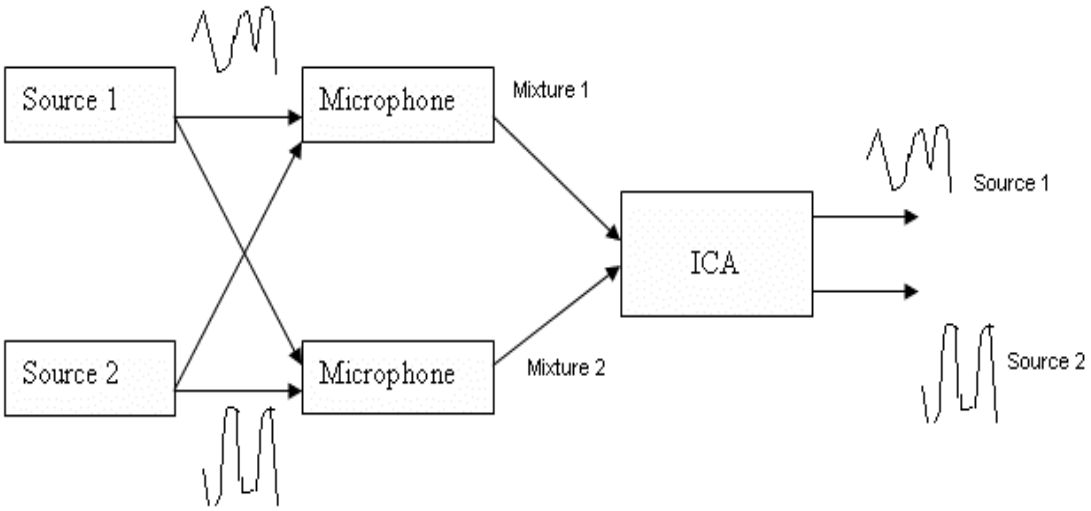


Fig. 2.1 Blind source separation

In order to estimate the original voice signal, the mixing matrix *A* needs to be estimated. Independent Component Analysis (ICA) [8] is one of the approaches used to solve this problem, which uses information-maximization principle [9] to estimate the unknown mixing matrix *A*. In ICA, the original source signals are considered to be statistically independent. ICA helps in the estimation of the mixing

2.2 Design Rationale for the Initial Approach to ICA Based Signal Processing Block

The initial idea for the ICA based signal processing block simulation is shown in Figure 2.2. The rationale for this figure is as follows:

At the handset, the received signal is a mixture of voice signal plus noise. When noise is dominant it becomes necessary to use some filtering technique to reduce the noise in the signal to avoid call drops. This was the overall goal.

The core idea was to store samples of voice when its quality was good and then use that information to identify the signal when the signal and noise are separated by the ICA-based filtering system. It is the decision process to identify the signal from noise after their separation that makes this filter a non-linear system. The identification process was based initially on pitch calculations. This approach led to the flow chart shown in Figure 2.2.

First, we defined a threshold which determines the quality of voice. If the received signal is greater than the threshold, it indicates that the quality of voice is good. Samples of good quality signal are stored and the pitch for this signal is calculated (Figure 2.2). On the other hand, if the received signal is less than the defined threshold, this may leads to discontinuity of the service leading to call drop. It then becomes important to reduce the noise from the signal and then boost the separated weak signal. To achieve this objective, we pass the signal below the threshold through ICA. As ICA requires two inputs of mixture and, in our case we have only one input of mixture of voice and noise. So we synthetically generate the second mixture by adding the same nature of noise to the first noisy signal (Figure 2.4).

ICA decorrelates the signals and reduces the higher-order statistical dependencies, attempting to make the signals as independent as possible. Thus, at the output of ICA, we have the reasonably noise-free audio signal and noise. To identify the desired signal versus the noise, we used pitch detection techniques, since one of the outputs signal is voice which is characterizes by its pitch.

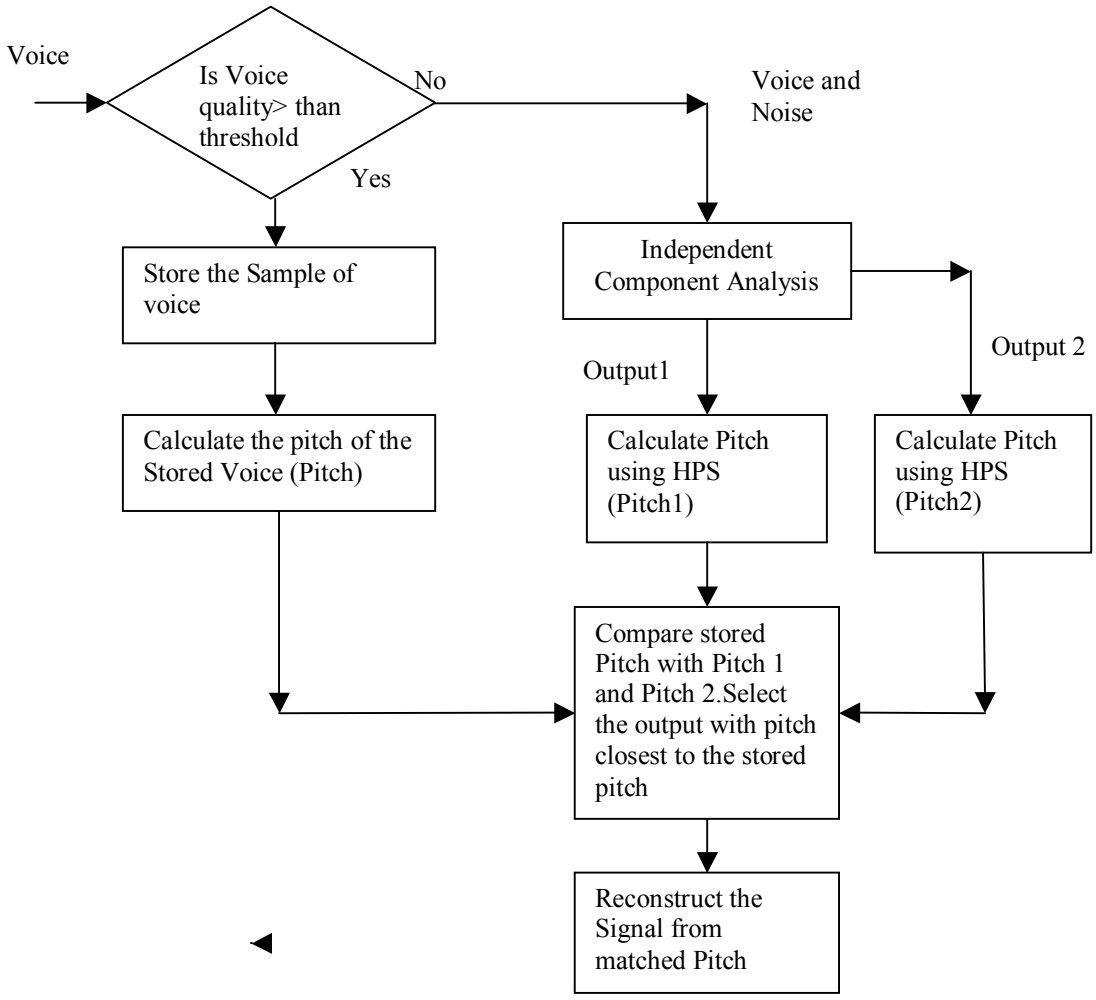


Fig. 2.2 Initial approach to ICA based signal processing block for call drop reduction

Pitch detection is accomplished using Harmonic product Spectrum (HPS) [10]. The detection of pitch in case of audio signal is the detection of the frequency at which maxima occurs. The HPS algorithm is described below.

The input signal is divided into segments by applying a Hanning window, using the desired window and hop size. The short time Fourier Transform (STFFT) is evaluated for each window to convert the signal from the time domain to the frequency domain.

After the windowed frame is taken into the frequency domain, the magnitude of the spectrum is calculated. The spectrum is down-sampled to create more compressed versions of it. The higher harmonics of the frequency get aligned with each other in the down sampled spectra. The multiplication of the aligned frequencies gives a maximum at a frequency which will represent the pitch of the signal. Hence, the pitch of the signal is determined. Figure 2.6 gives the details of HPS algorithm.

The calculated pitch of the two outputs is compared to the stored pitch of the audio signal. The pitch for the noise is random compared to the stored pitch of the audio signal while the signal pitch at the output of ICA will be close to the stored audio signal pitch (Figure 2.2). This helps to distinguish between the desired signal and the noise.

Thus, at the output we get a reasonably noise-free signal thereby improving the received signal quality. Figure 2.5 shows the output of ICA.

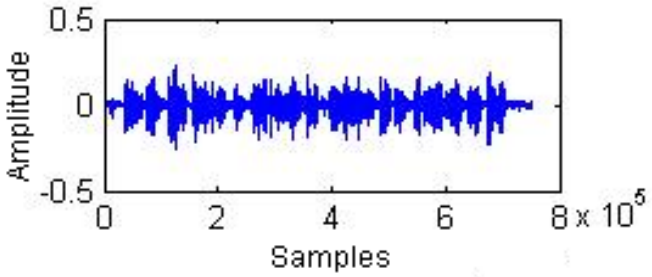
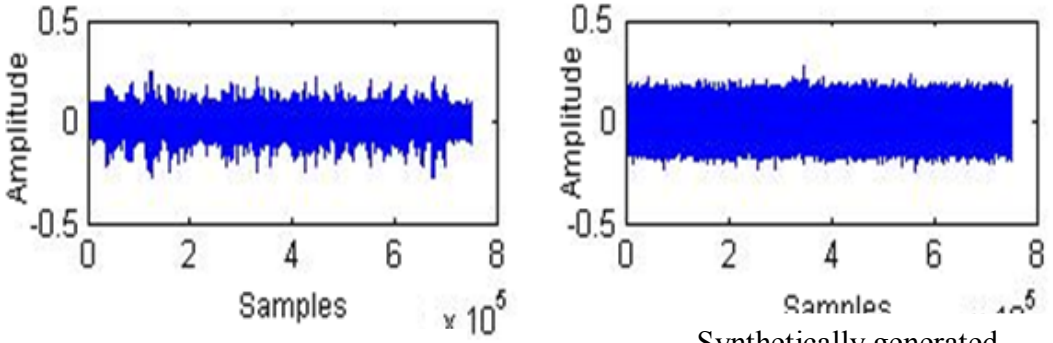


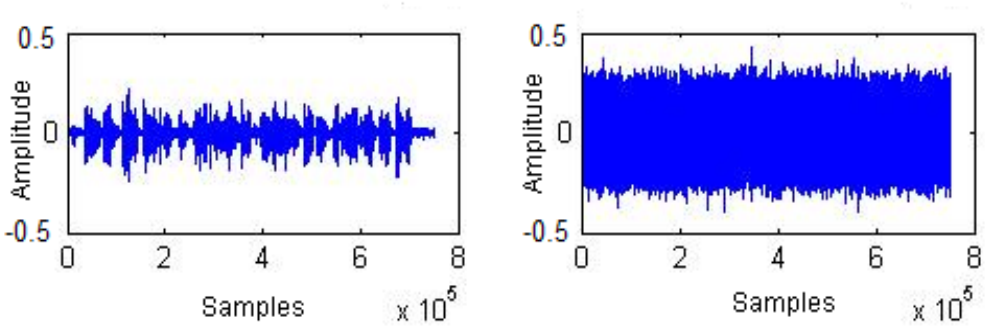
Fig. 2.3 Recorded voice sample



Noisy audio signal

Synthetically generated noisy audio as second input

Fig. 2.4 Input mixtures to ICA



Recovered audio signal

Noise

Fig. 2.5 Recovered source signals at the output of ICA

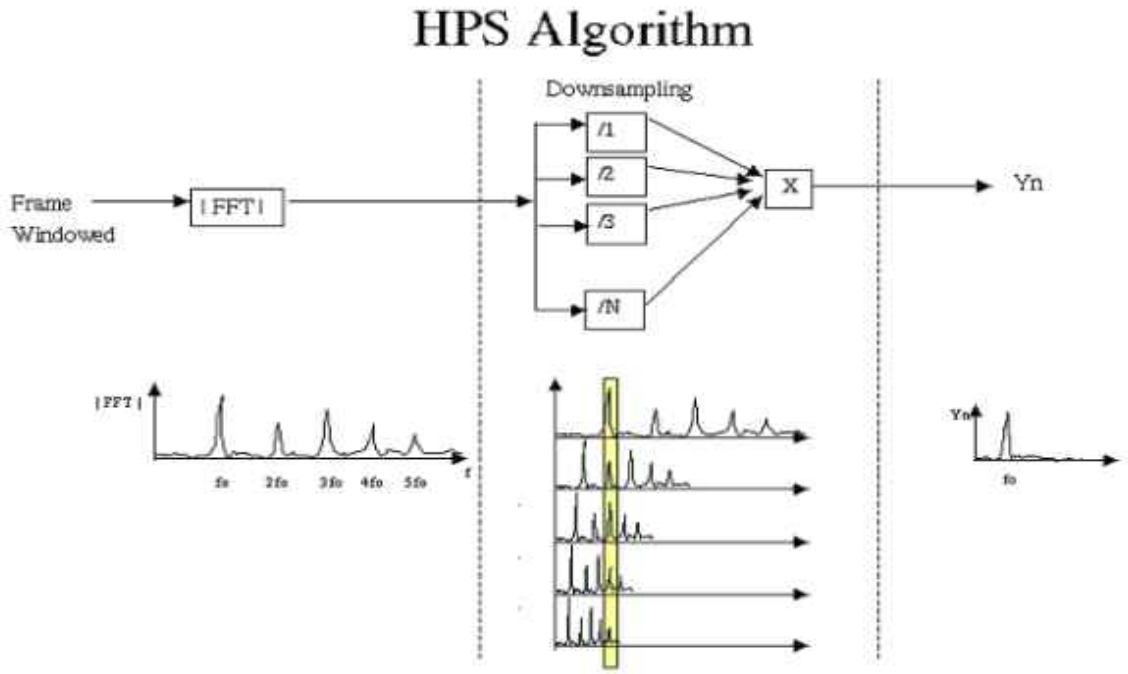


Fig. 2.6 Pitch detection using harmonic product spectrum

2.2.1 Rationale for Modifications to the Initial Approach

The initial proposed system was designed for audio (analog) signal. As our main objective is to design a signal processing block (SPB) applicable to the wireless system (in particular GSM), where the received signal at the physical layer is digital in nature, it was necessary to modify the SPB to make it applicable to it.

In the initial SPB, we judged the quality of our approach by simply listening to the output audio. We do not have the luxury of this perceptual evaluation within the physical layer simulation, for many reasons. Therefore, some quantitative measure of improvement was needed. In the wireless technology, a BER plot is one standard way of measuring quality. Therefore, we adopted it for our purposes as well. We also simulated

the physical layer block (PLB) of the wireless system with the signal processing block to allow evaluation and validation of the SPB under different conditions: AWGN channel, Ricean channel, phase estimation and channel compensation, diversity and channel coding.

We also modified the signal and noise identification process since, in the initial approach it was based on the signal pitch detection, which is not possible in the Physical Layer signal flow.

The new signal processing block (SPB) is thus developed by modifying the ICA algorithm and by introducing a new preprocessing and a new identification block. The SPB is to be inserted at the front end of the receiver as it is important to reduce noise from the received signal at the initial stage before the matched filter and demodulator.

Figure 2.7 shows the physical layer simulation of the wireless system in presence of the SPB.

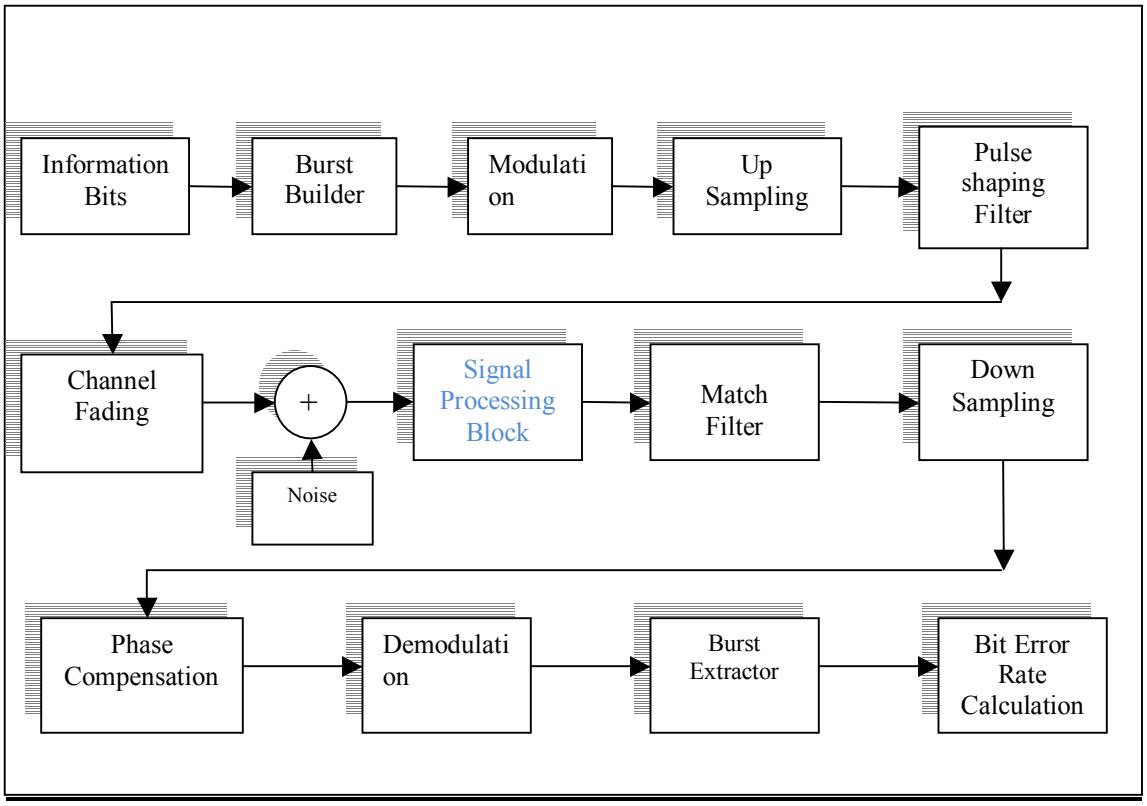


Fig. 2.7 Physical layer simulation of a wireless system with the proposed signal processing block (SPB)

CHAPTER 3

SIMULATION OF THE PHYSICAL LAYER OF THE TRANSMITTER

3.1 Introduction

In the physical layer, the generated information (data) bits is input to the burst builder block, which attaches unique words and guard bits to form a frame structure. The generated burst is input to the modulation block, which performs the mapping of bits to symbols. The modulated signal is passed through an up-sampler and a pulse shaping filter to give a continuous and band limited signal, which is transmitted through a high frequency carrier.

3.2 Burst Builder

The variable length information bits (data) are generated as ones and zeros (which simulates the "signal" arriving at the input of the burst builder in a base station), which is input to the burst builder to form a frame to be transmitted. The burst builder generates unique words and guard bits, which are attached to the information bits [11]. A unique word is a known unique sequence of bits used in the synchronization at the handset. During the phase estimation in the handset, the unique word helps evaluate the initial phase rotation which in turn helps evaluate the phase rotation of entire burst. The calculated phase is used in channel compensation.

Guard (Tail bits)	Unique Word	Data (Information)	Guard (Tail bits)

Fig. 3.1 Transmitted frame structure

3.3 Modulation

The modulation techniques we have used is Quadrature Phase Shift Keying (QPSK) modulation [12], where every symbol consists of two information bits and the modulation is being carried out using the following mapping:

Table 3.1 Mapping of bits to symbol for QPSK

Bits	Symbol
00	1
01	j
11	-1
10	-j

In case of PSK, the modulated signal can be represented as,

$$S_m(t) = \sqrt{\frac{2E}{T}} \cos(\omega_0 t + 2\pi m / M), \text{ for m = 1, 2.....M}$$

where, E is the symbol energy, T is the symbol time duration, M is the number of constellation points. For QPSK $M = 4$.

For PSK modulation, the number of bits/symbol = $k = log_2 M$, where M is the constellation of PSK modulation used. In case of QPSK, $M = 4$, therefore $k = 2$.

In the PSK modulation, the phase of the signal is varied with constant amplitude. This property is exploited in phase estimation and channel compensation, where only the phase needs to be compensated for rotation due to fading. The change in amplitude will not have any effect in the detection, since it depends only on the phase of the received signal.

3.4 Up-sampling

Up-sampling is performed by sampling the signal with the desired sampling frequency. This is accomplished by introducing zeroes in between the samples. If sampling frequency of 16 samples/symbols is used then, 15 zeroes are introduced in between the samples. Introducing the zeroes and interpolating it with the other samples helps in better signal representation.

3.5 Pulse Shaping Filter

The output of the up-sampler is input to the pulse shaping filter. If the rectangular pulses are passed through a bandlimited channel, the pulses will spread in time and, the pulse for each symbol will smear into the time intervals of succeeding symbols. This causes intersymbol interference (ISI) and leads to an increased probability of the receiver making an error in detecting a symbol. Hence, pulse shaping techniques are needed to simultaneously reduce the intersymbol effects and the spectral width of the modulated digital signal. The most popular pulse shaping filter is the raised cosine filter.

3.5.1 Raised Cosine Roll-off Filter

A raised cosine filter belongs to the class of filters which satisfy the Nyquist criterion. The transfer function of a raised cosine filter is given by [12]

$$H_{RC}(f)=\begin{cases}1 & 0 \leq | f | (1-\alpha)/2T_s \\ \frac{1}{2}\left[1+\cos\left[\frac{\pi | f | 2T_s - 1+\alpha}{2\alpha}\right]\right] & (1-\alpha)/2T_s \leq f_s | (1+\alpha)/2T_s \\ 0 & | f | > (1+\alpha)/2T_s \end{cases} \quad (3.1)$$

where, α is the roll-off factor between 0 and 1 and T_S is the symbol period. When $\alpha = 0$, the raised cosine filter corresponds to a rectangular filter of minimum bandwidth.

The impulse response of the filter is,

$$h_{RC}(t)=\frac{\sin\left(\frac{\pi t}{T_S}\right)}{\pi t}\frac{\cos\left(\frac{\pi\alpha t}{T_s}\right)}{1-\left(\frac{4\alpha t}{2T_s}\right)^2} \quad (3.2)$$

The cosine transfer function is achieved by using identical $\sqrt{H_{RC}(f)}$ filters at the transmitter and receiver, which provide a matched filter at the receiver end for optimum performance in a flat-fading channel.

After matched filter, the signal is passed through the channel to introduce the effect of channel fading. Details of the channel simulation are in the next chapter.

CHAPTER 4

CHANNEL SIMULATION

4.1 Introduction

Since we are simulating an end-to-end path of the signal from the transmitter (base station) to the receiver (the handset) via the channel, it is necessary to properly simulate the noise introduced in the channel. The noise falls in to two broad categories: fading and additive white Gaussian noise.

4.2 Channel Fading

Fading is used to describe the rapid fluctuations of the amplitude, phases or multipath delays of a radio signal over a period of time or travel distance. Fading is caused by the interference between two or more versions of the transmitted signal (multipath) which arrive at the receiver at slightly different times. These multipath waves combine at the receiver antenna to give a resultant signal, which can be widely vary in amplitude and phase depending on the distribution of the intensity and relative propagation time of the waves and the bandwidth of transmitted signal.

The factors influencing fading are multipath propagation, speed of the mobile (Doppler shift), speed of the surrounding objects and the transmission bandwidth of the signal. The multipath delay spread leads to time dispersion and frequency selective fading.

Time dispersion due to multipath causes the transmitted signal to undergo either flat or frequency selective fading. In the simulated wireless system, we have simulated the most common type of observed fading, which is flat fading [12].

4.2.1 Flat Fading

If a mobile radio channel has a constant gain and linear phase over a bandwidth which is greater than the bandwidth of the transmitted signal, the received signal will undergo flat fading [12].

4.2.1.1 Rayleigh Flat Fading

Rayleigh is commonly used to describe the statistical time varying of the received envelope of a flat fading signal. The Rayleigh channel only has the scattered component of the signal with no direct path [10].

The received signal is represented as, $r(u,t)=g(u,t)s(t)+n(u,t)$, where *g(u,t)* is the channel gain, *s(t)* is the transmitted signal and *n(u,t)* is white noise. For $t=t_0$, the gain $g(u) = g_I(u)+g_Q(u)$. If $g_I(u)$ and $g_Q(u)$ are Gaussian random variables with zero mean and variance σ^2, the magnitude of the received envelope follows a Rayleigh distribution i.e. $r(u)=|g(u)|=\sqrt{g^2{}_I(u)+g_Q^2(u)}$ has Rayleigh distribution. The pdf is given as,

$$p(r)=\frac{r}{\sigma^2}\exp\left[\frac{-r^2}{2\sigma^2}\right] \qquad \text{for } r\geq 0 \tag{4.1}$$

Jakes model is the widely used deterministic method for simulating Rayleigh fading [14, 15]. It assumes that the *N* equal strength rays arrive at the moving receiver with the uniformly distributed arrival angle α_n such that the N^{th} ray experiences Doppler

shift $\omega_n = \omega_M \cos(\alpha_n)$, where $\omega_M = 2\pi fv / c$ is the maximum Doppler shift, v is the vehicle speed, f is the carrier frequency and c is the speed of the light.

The in-phase and quadrature phase component of the channel can be represented as the sum of sinusoids which by Jakes model is given by,

$$g_I(t) = \sqrt{2}\{2\sum_{n=1}^{M} \cos\beta_n \cos\omega_n t + \sqrt{2}\cos\alpha\cos\omega_m t\} \quad (4.2)$$

$$g_Q(t) = \sqrt{2}\{2\sum_{n=1}^{M} \sin\beta_n \cos\omega_n t + \sqrt{2}\sin\alpha\cos\omega_m t\} \quad (4.3)$$

The set up parameters used for simulating a Rayleigh channel using Jakes model are

N = multipath (number of sinusoids)

$\omega_m = 2\pi f_m$*=maximum Doppler shift*

$\omega_n = \omega_m \cos(2\pi n / N)$*= Doppler shift*

M = 1/2 (N/2 – 1) = low frequency oscillators to generate the sinusoids

In the design (equation 4.2 and 4.3) the parameters β_n and α are given as,

$\beta_n = n\pi / M$, $\alpha = 0$

The total channel gain g*(t) = $g_I(t)+jg_Q(t)$*. The total channel coefficient is normalized to avoid any amplification or attenuation effects on the transmitted signal amplitude.

4.2.1.2 Ricean Flat Fading

Some scattering environments have a line-of-sight (LOS) or direct path component. In such cases, $g_I(t)$ and $g_Q(t)$ are Gaussian variables with non-zero mean m_I

and m_Q and a variance = σ^2. The magnitude of the received envelope $r(u)=|g(t)|=\sqrt{g_I^2(t)+g_Q^2(t)}$ follows the Ricean distribution. Its pdf is given as [10]

$$p(r)=\frac{r}{\sigma^2}\exp\left[-\frac{r^2+s^2}{2\sigma^2}\right]I_0\left(\frac{rs}{\sigma^2}\right), \qquad r\geq 0 \tag{4.4}$$

where, $s^2=m_I^2+m_Q^2$, I_0 is the zero order Bessel function.

The Ricean channel can be characterized by the Doppler shift (f_d) and Ricean fading factor *K = (direct path power) / (scattered path power)*

4.2.1.2.1 Simulation of a Ricean Channel

A Ricean channel has a direct path component (LOS) and the scattered path components of the transmitted signal. Hence, a Ricean channel can be modeled as a Rayleigh channel with a direct path (LOS). The software module for this step in the simulation was provided by [11]. Figure 3.1 shows the simulation of a Ricean channel.

Ricean fading factor = K = *direct power/scattered power* = $1/A^2$ (4.5)

$K_{dB} = 10\log_{10}(1/A^2)$

B, used to normalize the total power = $\frac{1}{\sqrt{total\ power}}=\frac{1}{\sqrt{1+(10^{-K/20})^2}}=\frac{1}{\sqrt{1+10^{-K/10}}}$

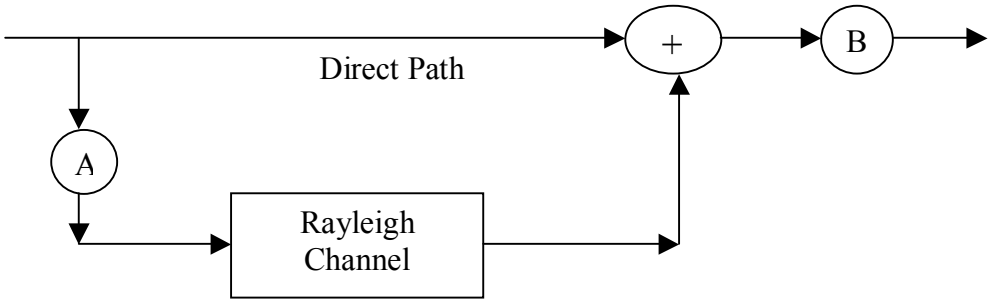


Fig. 4.1 Simulation of a Ricean channel

4.3 Additive White Gaussian Noise

The faded transmitted signal is passed through the AWGN block, where the random white Gaussian noise is added to it.

4.3.1 Calculation of Noise Power

The signal to noise ratio (SNR) and bit energy to noise (E_b/N_0) are related by the equation [12],

$SNR = E_b/N_0 + log_{10}(k)$

where, k is the bits/symbols which in case of QPSK is 2.

$$SNR = E_b/N_0 + log_{10}(2) = 3 + E_b/N_0 \tag{4.6}$$

Signal Power = variance of the signal.

$$Signal\ to\ Noise\ Ratio = SNR = 10 * \log_{10}\left(\frac{Signal\ Power}{Noise\ Power}\right) \tag{4.7}$$

Hence, we can calculate the noise power using equation 4.6 and 4.7. Therefore,

$$Noise\ Power = \frac{Signal\ Power}{10^{SNR/10}} \tag{4.8}$$

The white Gaussian noise *n(t)* with the calculated noise power given by equation (4.8) is generated and added to the faded signal as shown in equation 4.9. Figure 4.2 shows the total channel noise. The received signal at the receiver end can be represented as,

$$r(t) = s(t) * g(t) + n(t) \tag{4.9}$$

where, $s(t)$ is the transmitted signal, $n(t)$ is the additive white Gaussian noise with a

power spectral density $N_0/2$, $g(t)$ is the channel coefficient $= g_I(t) + jg_Q(t)$. $g_I(t)$ and $g_Q(t)$ are in-phase and quadrature components of the channel.

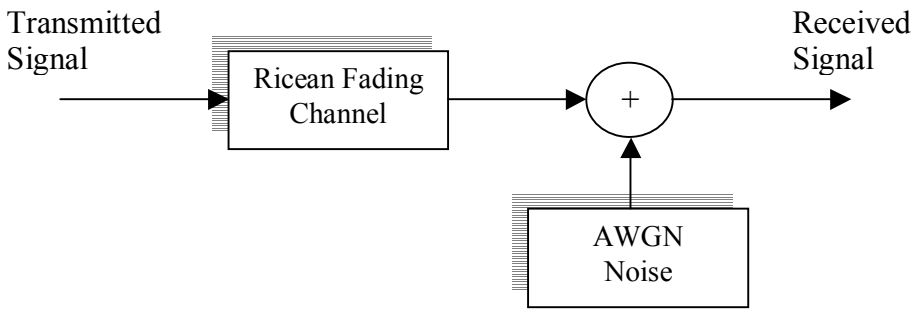


Fig. 4.2 Total channel noise

CHAPTER 5

SIMULATION OF THE PHYSICAL LAYER OF THE RECEIVER

5.1 Overview

The faded received signal is passed through the newly introduced signal processing block (SPB) at the front end of receiver (Figure 2.7). The signal at the output of the SPB is input to the matched filter. After filtering, the signal is down-sampled and input to the phase estimator and channel compensation block to reduce the effect of channel fading. The phase-compensated signal is input to the demodulator, which is simulated using a hard slicer. The demodulated signal is compared with the transmitted information signal and number of bits in error is evaluated. The following section details each block.

5.2 Signal Processing Block (SPB)

The SPB is our new proposed block to be simulated at the front end of the receiver. The SPB mainly consist of following sub blocks,

- Generation of two signal and noise mixtures
- Centering
- Whitening
- Estimation of the un-mixing matrix
- Identification of signal and noise

Figure 5.1 provides the details of the SPB (also see Figure 2.7)

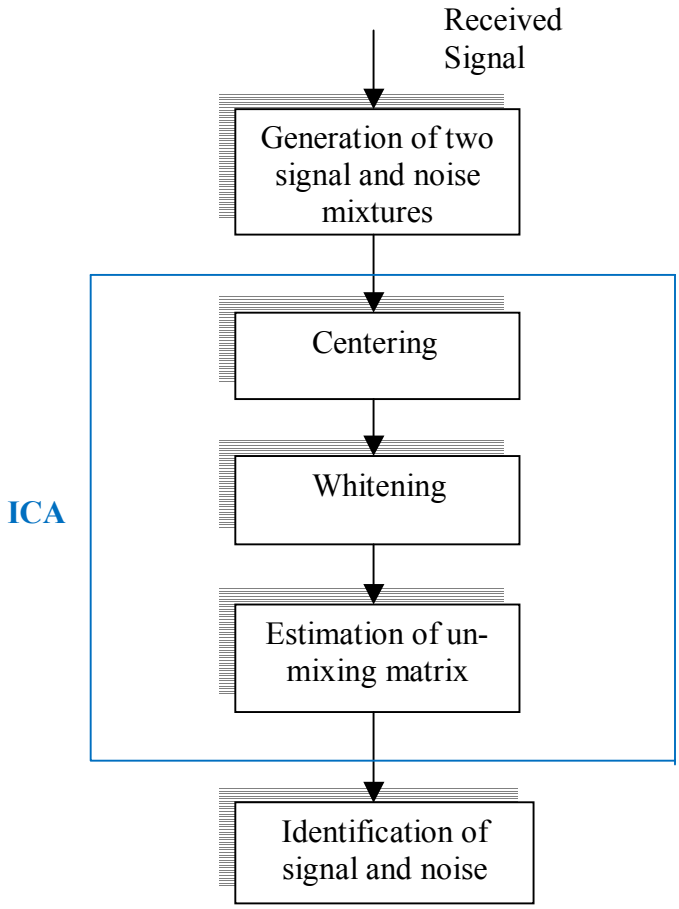


Fig. 5.1 Proposed signal processing block at the receiver section

5.2.1 Generation of Two Signal and Noise Mixtures

The received signal can be considered to be a mixture of signal and noise. Since the ICA works with two signal mixtures, the received signal needs to be preprocessed to get two mixtures of signal and noise. The second mixture is synthetically generated by adding more noise to the received signal using a noise model similar to the one that the channel introduced in to the received signal. The two mixtures X is input to the next stage of the SPB.

The two mixtures of signal and noise help estimate the un-mixing matrix to be able to recover the original source signal.

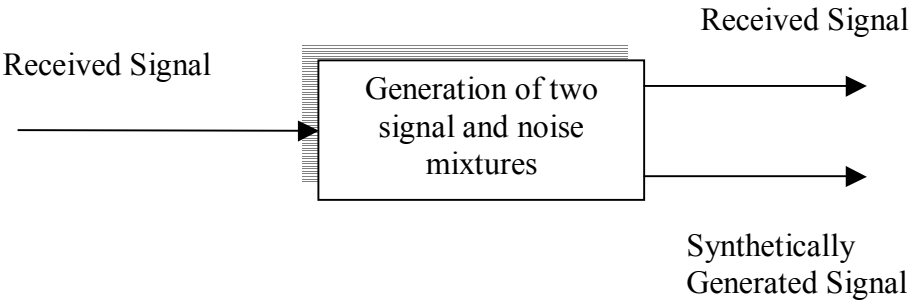


Fig. 5.2 Generation of the second mixture of signal and noise

The mixture $X = AS = [received\ signal, synthetically\ generated\ signal]$ (5.1)

where, A is the non-singular mixing matrix. Therefore,

$S = A^{-1}X = WX$, where, W is the un-mixing matrix that need to be estimated in order to recover the original signal.

The mixtures obtained after preprocessing are input to the next stage of sub blocks to estimate the un-mixing matrix W .

5.2.2 Centering

The next step is to center the random mixture X obtained from equation (5.1) by subtracting the mean vector $m = E(X)$. This makes X a zero-mean variable. This also implies that S will be zero-mean. This step simplifies the un-mixing algorithm.

After the estimation of the mixing matrix A with the centered data, the estimation is completed by adding the mean vector of S back to the centered estimates of S . After estimating the mixing matrix, its mean is evaluated by $A^{-1}m$.

5.2.3 Whitening

Whitening reduces the number of the parameters to be estimated. Instead of having to estimate the n^2 parameters that are the elements of the original mixing matrix A, we only need to estimate a new orthogonal mixing matrix $\overset{\Lambda}{A}$, obtained after whitening.

We transform the observed vector X linearly so that we get a new vector $\hat{X}$ which is white i.e.

$$E\left\{ \hat{X}\ \hat{X}^{T} \right\} = KI \tag{5.2}$$

The method used for whitening is the Eigen-value decomposition (EVD) of the covariance matrix [8, 9]. The covariance of vector X using EVD can be decomposed as,

$$E\left\{XX^{T}\right\} = \mathrm{E}\,\Lambda\,\mathrm{E}^{\mathrm{T}} \tag{5.3}$$

where, E is the orthogonal matrix of the eigenvectors of $E\left\{XX^{T}\right\}$ and Λ is the diagonal matrix of its Eigen values. $\Lambda = diag(\Lambda_1, \Lambda_2 ....., \Lambda_n)$. We whiten the random vector X by using the transformation,

$$\overset{\Lambda}{X} = \mathrm{E}\,\Lambda^{-1/2}\,\mathrm{E}^{\mathrm{T}} X \tag{5.4}$$

such that the covariance matrix $K_{xx} = E\left\{ \overset{\Lambda}{X}\ \overset{\Lambda}{X}^{T} \right\} = \mathrm{I}$

Whitening transforms the mixing matrix into a new $\overset{\Lambda}{A}$. So the transformed vector is

$$\overset{\Lambda}{X} = (\ \mathrm{E}\,\Lambda^{-1/2}\,\mathrm{E}^{\mathrm{T}}\ )\ \mathrm{A}X = \overset{\Lambda}{A}\,S \tag{5.5}$$

The new orthogonal matrix obtained from whitening is orthogonal

$$E\left\{ \hat{X}\ \hat{X}^T \right\} = \hat{A}\, E(SS^T)\, \hat{A}^T = \hat{A}\, \hat{A}^T = I$$

5.2.4 Estimation of Un-mixing Matrix

The estimation of un-mixing matrix is achieved using the gradient ascent rule [15]. The software module for this step in the simulation was provided by [11].

Initially, we assume the un-mixing (W) to be an identity matrix (I). Then by using the gradient ascent rule the un-mixing matrix is evaluated iteratively as,

$$W_{new} = W_{old} + l[I - f(U)U^T]W_{old} \tag{5.6}$$

where, $U = WX$ is the intermediate output, where X is the input mixture of signal and noise after whitening, l is the learning rate typically chosen to be 0.001 but can be varied to get the optimum result , *f(u)* is the logistic transfer function given as [11],

$$f(u) = \frac{1}{(1+e^{-u})}$$

Thus, the gradient algorithm helps to evaluate the un-mixing matrix(W) from which we can detremine the original source signals. At the output, we get a reasonably noise-free signal (henceforth simply called as signal) and noise.

5.2.5 Identification of Signal and Noise

The output of the un-mixing matrix estimator gives the signal and noise. At this point, our system needs to distinguish between the signal and noise. This module was also provided by [11]

The identification is achieved by exploiting the autocorrelation property of white Gaussian noise. If n is the white Gaussian noise then the autocorrelation is given as, [17]

$$R_{XX}(n,n+m) = \sigma^2 \delta(m) \tag{5.7}$$

The signal as compared to noise will be more correlated due to up-sampling and interpolation performed at the transmitter section even though the initial data (information bits) of ones and zeros which is the output of the source encoder are uncorrelated. Hence, the autocorrelation plot of the signal is more widely spread with peak at the mean.

5.3 Matched Filter

The signal processing block gives the desired information signal after separation from noise. This signal is input to the matched filter at the receiver section to maximize the signal to noise ratio [12].

The received signal can be represented as $r(t) = s(t) * g(t) + n(t)$, where $s(t)$ is the transmitted signal, $g(t)$ represents the gain of the Ricean channel and $n(t)$ is the additive white Gaussian noise. The filter impulse response is matched to the input signal to maximize signal to noise ratio at the output. In our simulation, the transfer function of the matched filter is designed to be the same as that of the pulse shaping filter used at the transmitter side.

The output (maximum) SNR with the matched filter is given as [12]

$$SNR_0 = \frac{2}{N_0} \int_0^T s^2(t)\, dt$$

$$SNR_0 = \frac{2E}{N_0} \tag{5.8}$$

The output SNR depends on the energy of the signal *s(t)*. $N_0/2$ is the power spectral density of the white noise.

The signal at the output of the filter is the convolution of the input signal and the transfer function i.e. $r'(t) = \int_0^T r(\tau)h(t-\tau)\,d\tau$. The output of the filter is input to the down-sampler to convert the sample into symbols.

5.4 Down-sampling

Down-sampling by a factor 'J' is a process of keeping every J^{th} sample and discarding the rest. J is called as the sampling factor or the sampling rate. In the simulation, the sampling frequency of 16 samples per symbol is used. The length of the output after down-sampling is given as

$L = F/J$

where, L = length of the output after down-sampling.

F = Length of the down-sampled input.

J = Sampling factor.

The output of the down-sampler is input to the Phase Estimator.

5.5 Phase Estimation and Channel Compensation

The block phase estimation and channel compensation (Figure 5.3) is used at the receiver end to combat the effects of channel fading.

Since QPSK modulation is used, the detection at the handset receiver is based on the phase. Therefore, the change in the amplitude will not affect the detection process. This property is exploited by the phase estimation block, in which only the phase rotation caused by the channel needs to be estimated.

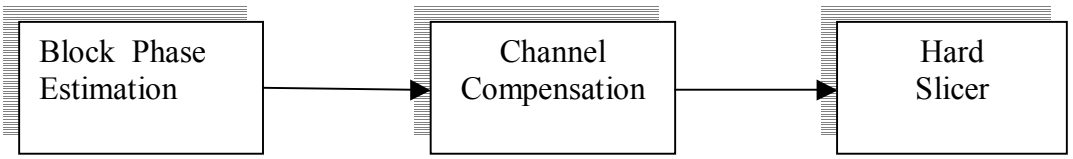


Fig. 5.3 Phase estimation and channel compensation

5.5.1 Block Phase Estimation

The received vector is given as $r(t) = s(t) * e^{j\theta} + n(t)$, where, $s(t)$ is the transmitted signal, $e^{j\theta}$ is the phase term introduced due to fading, $n(t)$ is AWGN.

In block phase estimation, we estimate the phase term θ introduced by the channel. The phase estimation is simulated based on [18]. The simulated steps are:

a) Nonlinear Transformation: The received signal $r(t)$ after down sampling is raised to the power of four to give, $y = (r(t))^4$

b) To calculate the phase rotation, a window size of length N (in our case 60) and a step size of M (10 in our case) are selected

c) The phase angle is calculated over the window size using the equation,

$$\theta_{ph} = \frac{1}{4}\tan^{-1}\left(\frac{\sum imagpart(y)}{\sum realpart(y)}\right) \tag{5.9}$$

d) The calculated phase can have values $\theta_{ac} = \theta_{ph} + \frac{k\pi}{4}$ where, k is the ambiguity, which varies from *-4 to +4*. So the possible phase candidates are,

$$\theta_{ac} = [\ \theta_{ph}\ ,\ \theta_{ph} + \frac{\pi}{4}\ ,\ \theta_{ph} - \frac{\pi}{4}, \theta_{ph} + \frac{\pi}{2}, \theta_{ph} - \frac{\pi}{2}, \theta_{ph} + \frac{3*\pi}{2}, \theta_{ph} - \frac{3*\pi}{2}, \theta_{ph} + \pi\ ]$$

e) The ambiguity is resolved by calculating the phase, using the unique word in the burst structure over its length *L*.

$$\theta_{UW} = angle\left(\frac{1}{L}\sum UW_R * UW_T^*\right) \tag{5.10}$$

where, UW_R = unique word at the receiver side and UW_T = unique word at the transmitter side.

f) The actual value of the phase is calculated from θ_{ac}, by comparing it with θ_{UW} and is termed θ_1. Now, θ_1 will be the reference angle for the calculation of the next angle θ_2. The calculated θ_1 corresponds to the phase rotation of the sample at the midpoint of interval *[0, N-1]*

g) The block window is incremented to *[M, M+N-1]* and the possible phase candidate (θ_{ac}) is obtained as in step *d*.

h) The next phase (θ_2) is calculated from the above phase candidate (θ_{ac}) as in *step f*, by choosing an angle closest to θ_1. It is then assigned to the sample at a midpoint of interval *[M, M+N-1]*

i) The above steps, from *c-g* are repeated till the length of the burst is covered.

This will give us the phase rotation of the symbol for every step size *M*. To get the phase rotation for every symbol in the burst we have used a *linear interpolation technique*.

5.5.2 Channel Compensation

The received signal, $r = s * e^{j\theta} + n$, where θ accounts for phase rotation. If $\hat{\theta}$ is the estimated phase of a sample (obtained from block phase estimation) then, the signal is phase compensated by multiplying the corresponding sample with the negative $\hat{\theta}$. The phase compensated signal,

$$r' = r * e^{-j\hat{\theta}} \tag{5.11}$$

5.6 Hard Slicer

The phase compensated signal is the input to the hard slicer. The detector is simulated as a hard slicer for PSK modulation, where detection is based only on the phase of the received signal. The hard slicer can be best described from Table 5.1 and Figure 5.4.

Table 5.1 Decision logic for QPSK demodulation

Decision Logic for PSK demodulation	Decoded Bits
$-\pi/4 \leq$ If angle of received symbol $\leq \pi/4$	00
$\pi/4 \leq$ If angle of received symbol $\leq 3\pi/4$	01
$3\pi/4 \leq$ If angle of received symbol $\leq -3\pi/4$	11
$-3\pi/4 \leq$ If angle of received symbol $\leq -\pi/4$	10

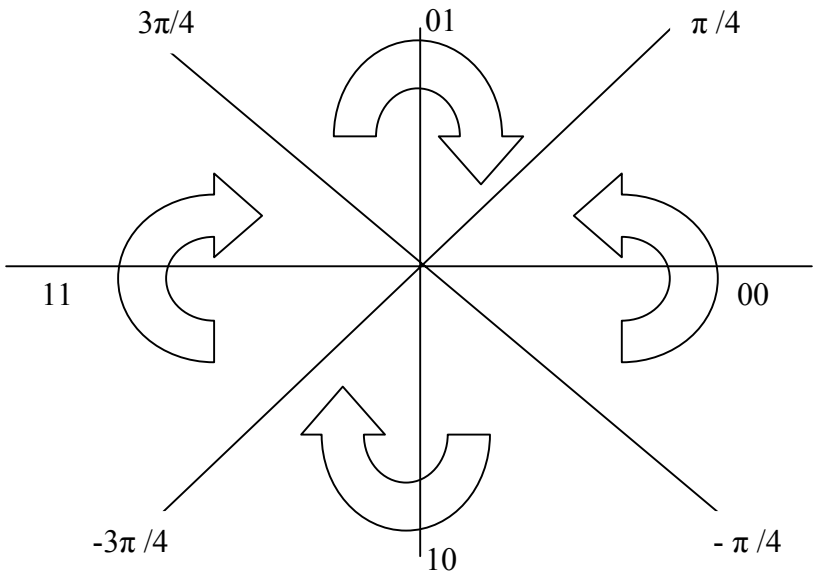


Fig. 5.4 Hard slicer

5.7 Burst Extractor

The demodulated signal consists of information bits, unique word, and guard bits. So, it is necessary to extract the information bits and discard the other bits. The extracted information bits are compared to the transmitted information bits for BER calculation.

5.8 Bit Error Rate Calculation

The extracted information bits at the receiver end needs to be compared to the transmitted information bits. This helps to evaluate the number of bits in error.

The Bit Error Rate is the ratio given by,

$$\text{BER} = \frac{number\ of\ bits\ in\ error}{total\ number\ of\ transmitted\ bits}$$

The performance analysis (BER calculation) of the physical layer simulation of a wireless system with our SPB is described in chapter 8.

Diversity and Channel coding are some of the techniques used to combat channel fading and improve the quality of the received signal. So it is important to test the performance of our SPB in presence of diversity or channel coding. Hence, we simulated diversity and channel coding. This is described in the next chapter.

CHAPTER 6

SIMULATION OF THE DIVERSITY

6.1 Introduction

Diversity techniques are used to combat multipath channel fading. If the handset receives several replicas of the same information signal transmitted over independently fading channels, the probability that all the signal components will fade simultaneously is considerably reduced. If p is the probability that any one signal will fade below some critical value then p^L is the probability that all L independently fading replicas of the same signal will fade below that critical value. There are several ways in which the receiver is provided with L independently fading replicas of the same information-bearing signal [12].

6.2. Diversity Techniques

- Frequency diversity: The same information bearing signal is transmitted on L carriers, where the separation between successive carriers equals or exceeds the coherence bandwidth $(\Delta f)_c$ of the channel.
- Time diversity: Transmit the signal in L different time slots to achieve L independently fading version of the same information-bearing signal. The separation between the successive time slots equals or exceeds the coherence time $(\Delta t)_c$ of the channel.

- Space diversity (antenna diversity): In space diversity, a single transmit antenna and multiple receiving antennas are employed. The received antenna must be spaced sufficiently apart so that the multipath components in the signal have significantly different propagation delays at the antennas.

Of the above three techniques, we have simulated the space diversity.

6.3 Diversity Combining

Maximum Ratio Combining is the popular diversity combining technique used at the receiver section [12] [20]. It is based on the principle of maximum likelihood detection in which, the probability $p(r/S_m)$ is maximized where, r is the received signal, S_m is the transmitted signal, $m = 1, 2, 3 \ldots\ldots M$. M is the number of constellation points. For QPSK, $m = 1, 2, 3, 4$ and $S_m = \{1, j, -1, -j\}$.

If diversity order used is L, then we have L copies of the received signals. The channel coefficient $g = [g_1, g_2, \ldots\ldots, g_L]$, received signal $r = [r_1, r_2 \ldots\ldots\ldots r_L]$. g and r are complex vectors. The probability $p(r/S_m)$ *is* maximized by simulating the term [19],

$$maximum(\mathrm{Re}\,al(S_m^* g^H r)) \tag{6.1}$$

Figure 6.1 gives the block diagram representation of equation (6.1) for QPSK modulation, where the transmitted signal $S_m = (1, j, -1, -j)$. r^i is the received signal, g is the estimated channel gain.

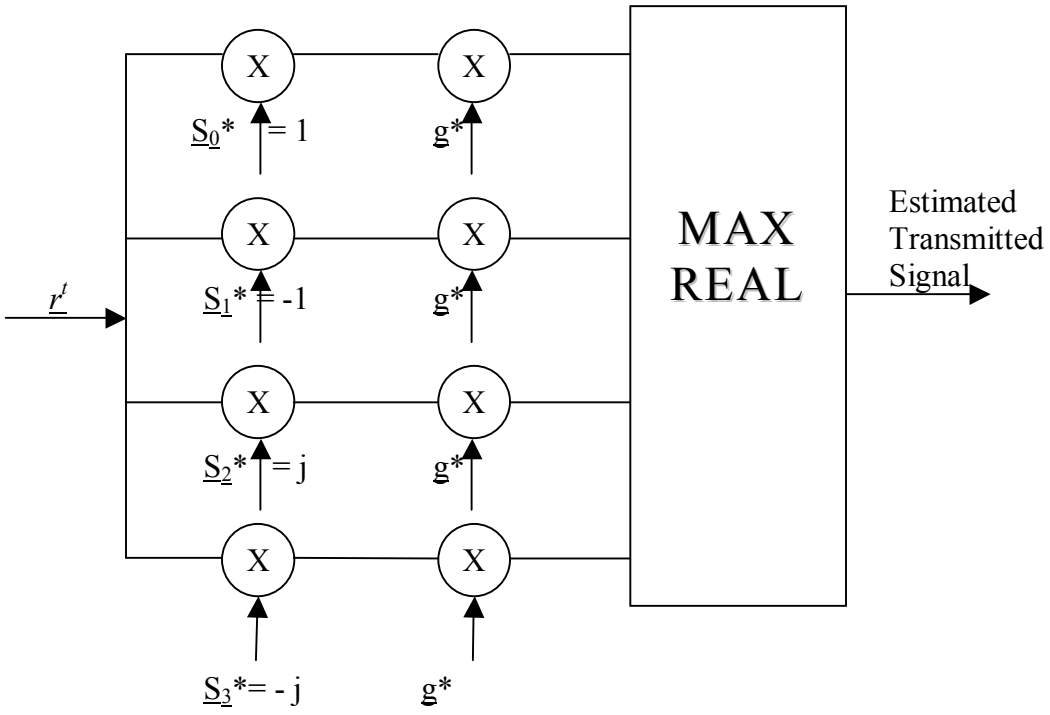


Fig. 6.1 Logic diagram for maximum ratio combiner for QPSK modulation

We have simulated diversity using space diversity technique of order 3.The 3 copies of the received signals are combined using MRC. Figure 6.2 shows the physical layer simulation of the wireless system with MRC.

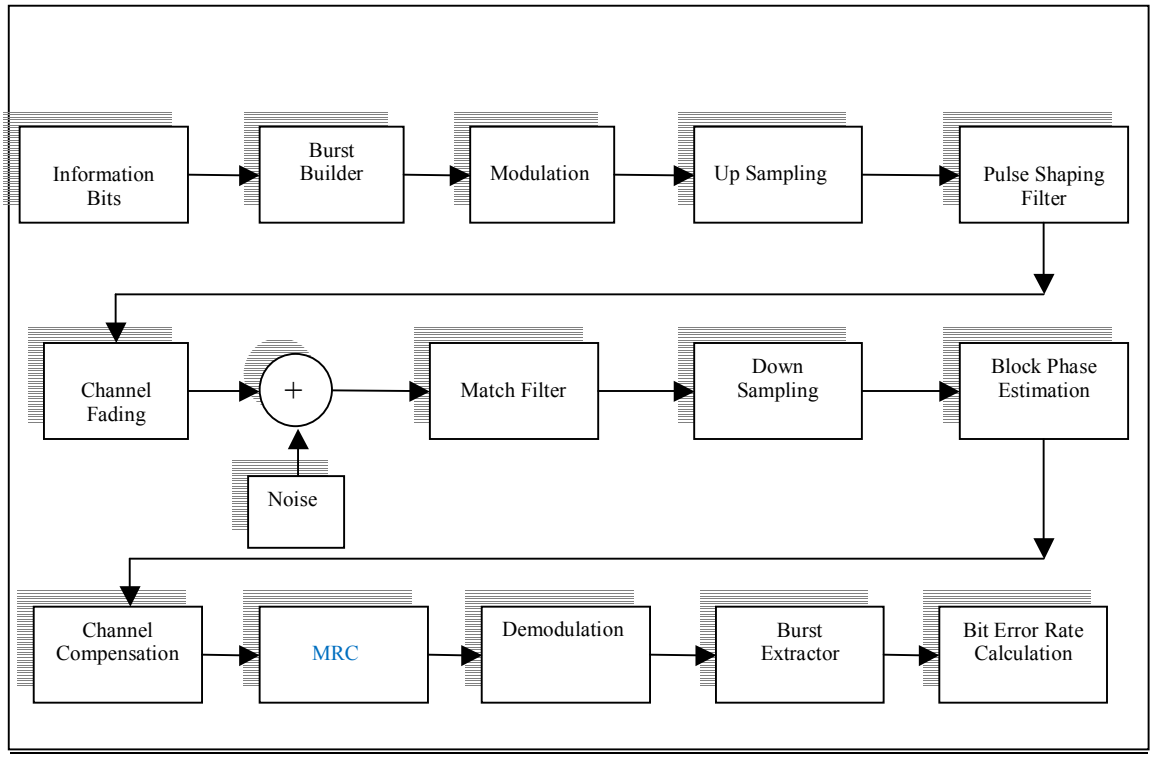


Fig. 6.2 Physical layer simulation of a wireless system with diversity using MRC

CHAPTER 7

SIMULATION OF THE CHANNEL CODING

7.1 Introduction

Channel coding improves the link performance (reduces the BER) by adding redundant bits in the transmitted message so that if an instantaneous fade occurs in the channel, the data may still be recovered at the receiver. At the baseband portion of the transmitter, a channel coder maps the digital message sequence into another specific code sequence containing a greater number of bits than originally contained in the message. The coded message is then modulated for the transmission in the channel.

Channel coding is used by the receiver to detect or correct some of the errors induced by the channel. The addition of coding bits lower the raw data transmission rate through the channel but improves the performance of the system by providing coding gain.

7.1.1 Types of Channel Coding

There are three general types of channel coding [10],

- Block codes
- Convolutional coding
- Turbo codes

Of the above techniques, we have simulated convolutional coding with coding rate *1/2*.

7.2 Convolutional Encoding

Convolution encoder contains memory (shift register), and the encoder outputs at any point of time not only depend on the present input but also on the previous inputs.

The convolution codes is represented with three parameter *(n,k,m)*

n= number of outputs

k= number of inputs

m= number of shift registers.

The constraint length $L = k\,(m-1)$.

The constraint length *L* represents the number of bits in the encoder memory that sffects the generation of the n output bits.

A rate *R=k/n,* convolution encoder with memory (shift register) order *m* can be realized as *k*-input, *n*-output linear sequential circuit with input memory *m.* That is, inputs remain in the encoder for an additional *m* time units after entering. Encoders for convolutional codes fall into two general categories of *feed forward (non-recursive)* and *feedback (recursive)*. Further, within each category, encoders can be either *systematic* or *non-systematic*.

For an encoder with memory *m*, we can write the convolution operation using the following mathematical model stated below.

$$v^{j}{}_{l} = \sum_{i=0}^{m} u_{l-i} g_{i}{}^{j} \qquad (7.1)$$

where $v^j{}_l$ signifies output l^{th} bit of j^{th} output, u corresponds to the input bit sequence and g_i^j corresponds to the i^{th} bit of j^{th} generator sequence. The output code word's length $N = n\ (K + m)$, where, K is the length of input sequence, m is the number of shift register and n is the number of outputs.

Figure 7.1 shown below is for rate $R = 1/2$ binary non-recursive, nonsystematic convolution encoder with memory order $m = 3$, $g^0 = [1\ 0\ 1\ 1]$, $g^1 = [1\ 1\ 1\ 1]$. The coded output is given by $v = [v_0{}^0 v_0{}^1,\ v_1{}^0 v_1{}^1,\ v_2{}^0 v_2{}^1 \ldots\ldots]$.

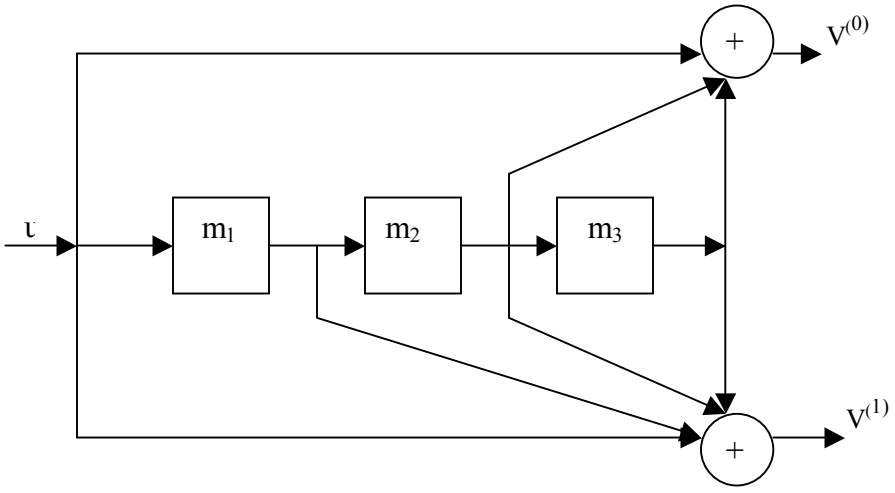


Fig. 7.1 Rate 1/2 non-recursive, non-systematic convolutional encoder

We have simulated Figure 7.1 at the transmitter end for encoding the data bits.

7.3 Convolutional Decoding

The function of the decoder is to estimate the encoded input information. There is a one-to-one correspondence between the information sequence and the code sequence.

Any information and the code sequence pair are uniquely associated with a path through a trellis. Thus, the main purpose of the convolution decoder is to estimate the path through the trellis that was followed by the encoder.

There are a number of techniques for decoding convolutional codes, of which Viterbi algorithm [23] is a commonly used technique.

7.3.1 Generation of a Trellis

To simulate the Viterbi algorithm, first a trellis is generated [23]. A trellis diagram is the expanded form of a *state diagram* of encoder in time i.e., to represent each time unit with a separate state diagram. A trellis diagram shown in Figure 7.2 is for (3, 1, 2) i.e. *(n, k, m)* code with a generator sequence G (D) = [1+D, $1+D^2$, $1+D+D^2$] and an information sequence of length $L = 5$. A trellis contains $L+m+1$ time unit levels, and these values are labeled from 0 to $L+m$ in Figure 7.2. Assuming the encoder always start in state S_0, the first m time units correspond to the encoder's departure from state S_0 and the last m time units correspond to the encoder's return to state S_0. It follows that not all the states can be reached in the first m or the last m time units. However, in the center portion of a trellis, all the states are possible and each time unit corresponds to a state diagram. There are two branches leaving and entering each state. The upper branch leaving each state at the time unit i represent the input $u_i = 1$, while the lower branch represents $u_i = 0$.Each branch is label with n corresponding outputs v_i, and each 2^L code words of length $N = n\,(L + m)$ is represented by a unique path through the trellis.

Assume that an information sequence $u = (u_0, \ldots\ldots, u_{L-1})$ of length kL is encoded into a code word $v = (v_0, v_1, \ldots\ldots, v_{L+m+1})$ of length $N = n(L+m)$ and the received

sequence over a binary input is $r = (r_0.r_1,\ldots\ldots,r_{L+m-1})$. The decoder produces an estimate $\hat{v}$ of the received sequence r. A maximum likelihood decoder chooses $\hat{v}$ which maximizes the log-likelihood function log P(r/v).

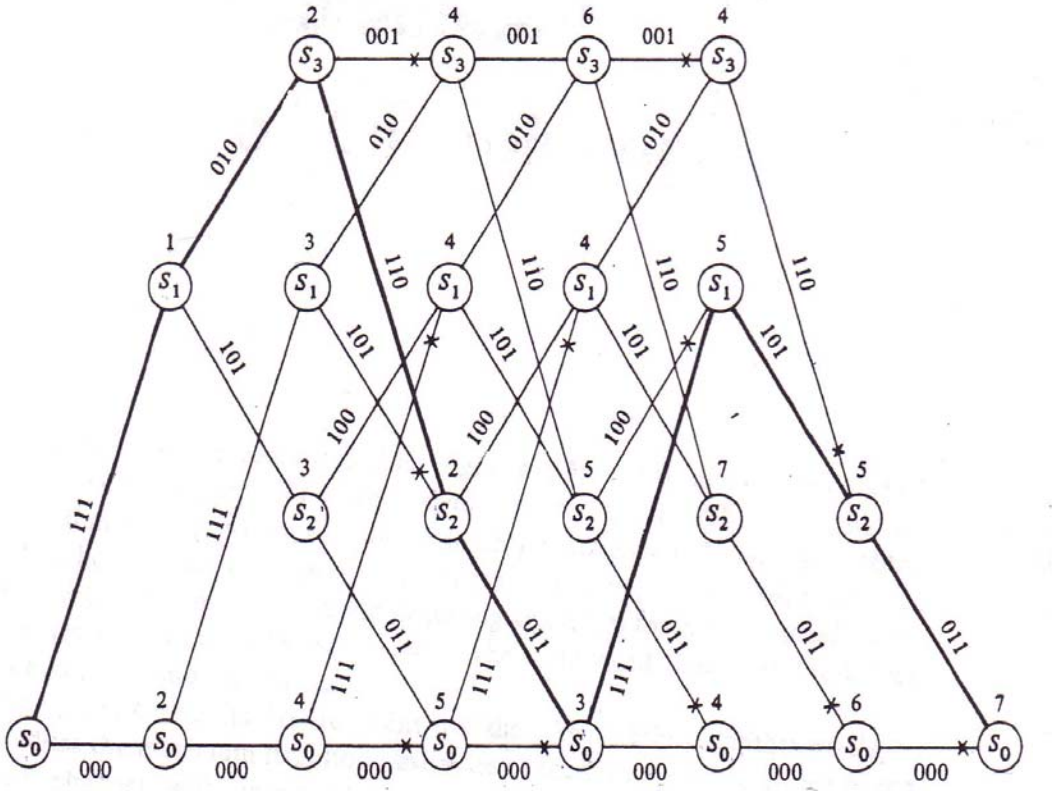


Fig. 7.2 Trellis diagram for code (3, 1, 2)

In the simulation, we have used the encoder sequence G = [1+D^2, 1+D+D^2]. Based on this sequence a trellis is generated. The Viterbi algorithm is simulated based on the generated trellis.

7.3.2 *Simulation of Viterbi's Algorithm*

Let a trellis node corresponding to state S_j at time i be denoted $S_{j,\,i}$. Each node in a trellis is to be assigned a value $V(S_{j,\,i})$ based on metric. When the hard decision is

performed, the metric used is the Hamming distance (as in our case), while the Euclidean distance is used for soft decision decoding. The node values are computed in the following manner [12] [23],

1. Initially, V ($S_{0,0}$) = 0 and i = 1.
2. At time i, the partial path metrics for all paths entering each node is computed.
3. V ($S_{j,i}$) equal to the smallest partial path metric entering the node corresponding to state S_j at time i. The non surviving branches are deleted from the trellis. In this manner, a group of minimum paths is created from $S_{0,0}$.
4. If $i < L+m$, where L is the input length sequence and m is the number of shift register,

   let $i = i+1$ and go back to step 2.

Once all the node values have been computed, start at state S_0, time $i = L+m$, and follow the surviving branches backward through a trellis. The path thus defined is unique and corresponds to the decoded output. The decoded output is then compared to the transmitted bits to compute the BER.

Figure 7.3 shows the physical layer simulation of a wireless system with channel coding and decoding.

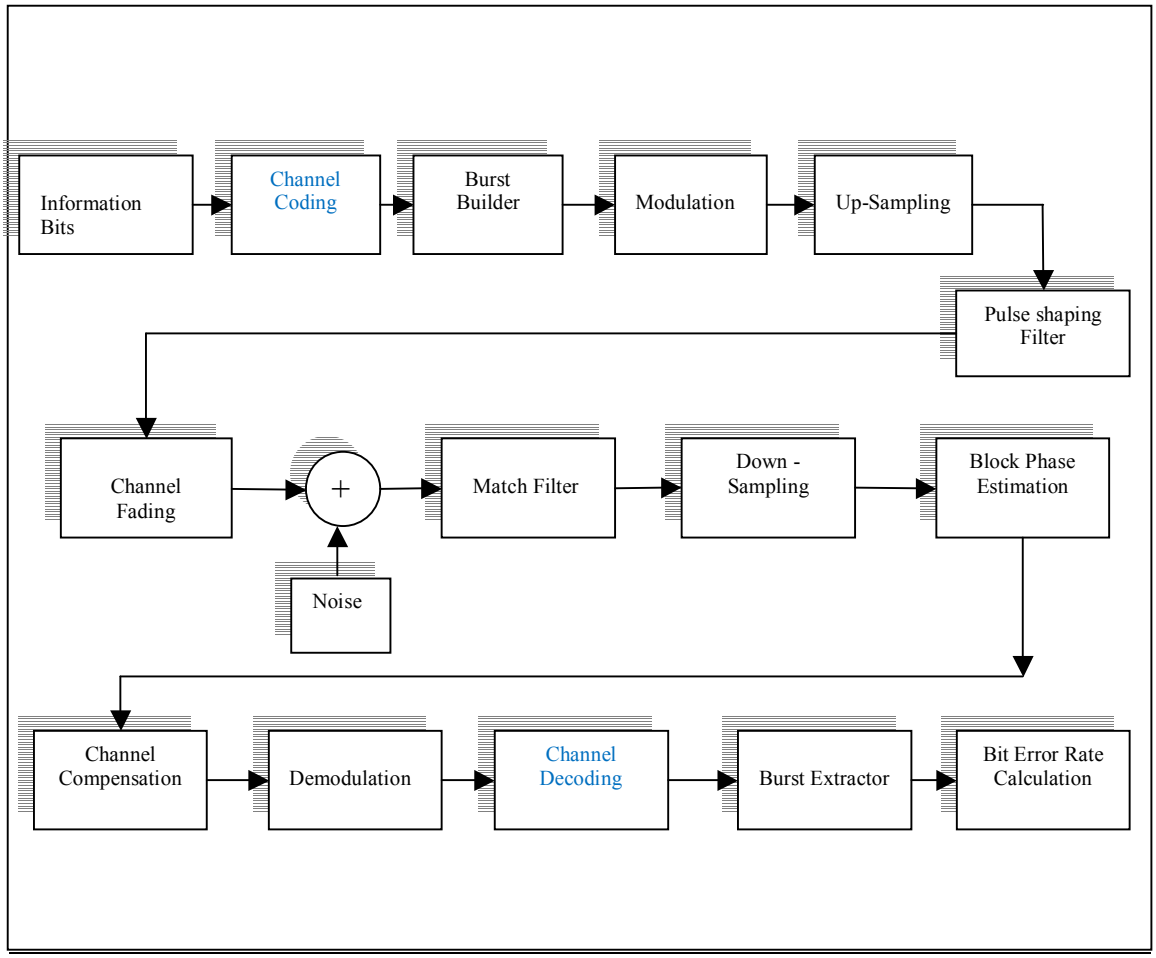


Fig. 7.3 Physical layer simulation of a wireless system with channel coding and decoding

CHAPTER 8

SIMULATION RESULTS

8.1 Overview

We have simulated the physical layer block (PLB) of a wireless system (Figure 2.7) with our SPB at the receiver end, in the presence of

- additive white Gaussian channel (AWGN)
- Ricean fading channel
- block phase estimator and channel compensation
- diversity and
- channel coding

The received information bits are compared with the transmitted bits, to calculate the bit error rate for different values of the signal to noise ratio.

Simulations are carried out using MATLAB version 7.0.4. In the following sections we present the BER obtained with and without the SPB and different configurations of the physical layer.

8.2 Physical Layer Simulation of a Wireless System with the SPB for AWGN Channel

Figure 8.1 is the plot of BER v/s E_b/N_0, obtained by simulating the physical layer of a wireless system in the presence of AWGN and without channel fading.

In the plot (Figure 8.1), 'Simulated with SPB' is obtained in presence of the SPB and 'Simulated without SPB' is obtained in the absence of the SPB. 'Theoretical' is the plot of theoretical BER with AWGN which, in our case served as the baseline to compare the performance. In case of QPSK modulation, the theoretical BER is obtained by the following formula [13],

$$\text{BER} = Q(\sqrt{2Eb/N_0})[1 - 2Q(\sqrt{2Eb/N_0})] \quad (8.1)$$

The simulated 'BER Simulated' and theoretical BER 'Theoretical BER with AWGN' (Figure 8.1) are almost coinciding with each other therefore *validating* our simulation of the physical layer of a wireless system.

The values of BER for different values of E_b/N_0 are shown in the Table 8.1.

Table 8.1 BER and E_b/N_0 for AWGN channel

E_b/N_0 in dB	0	1	2	3	4	5
Theoretical	0.0756	0.0547	0.0368	0.0226	0.0124	0.0059
Simulated without SPB	0.0789	0.0568	0.0358	0.0210	0.0125	0.0050
Simulated with SPB	0.0215	0.0133	0.0053	0.0021	0.0006	0.0002

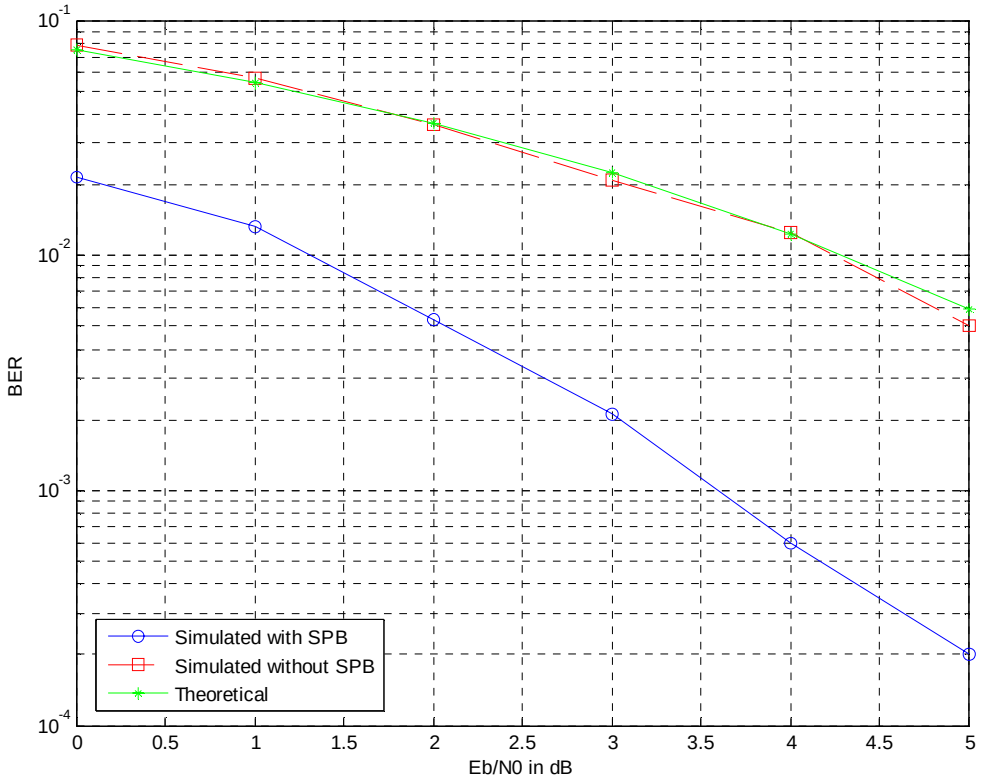


Fig. 8.1 BER vs. E_b/N_0 for additive white Gaussian channel

It can be seen from the plots in Figure 8.1 that coding gain is achieved with the introduction of the SPB.

8.3 Physical Layer Simulation of a Wireless System with the SPB for Ricean Channel

Figure 8.2 is the plot of BER v/s signal to noise ratio (E_b/N_0) obtained by simulating the PLB in the presence of channel fading and AWGN (Figure 2.7).

Parameters used for simulating a Ricean channel:

Fading factor (K_{dB}) = 7dB, Doppler shift (f_d) = 20 Hz.

In the plot (Figure 8.2), 'Simulated with SPB' is obtained in presence of the SPB and 'Simulated without SPB' is obtained in absence of the SPB. The values of BER for different values of E_b/N_0 are shown in the Table 8.2.

Table 8.2 BER and E_b/N_0 for Ricean fading

E_b/N_0 in dB	0	1	2	3	4	5
Simulated without SPB	0.1045	0.0895	0.0668	0.0486	0.0339	0.0236
Simulated with SPB	0.0464	0.0363	0.0239	0.0139	0.0088	0.0059

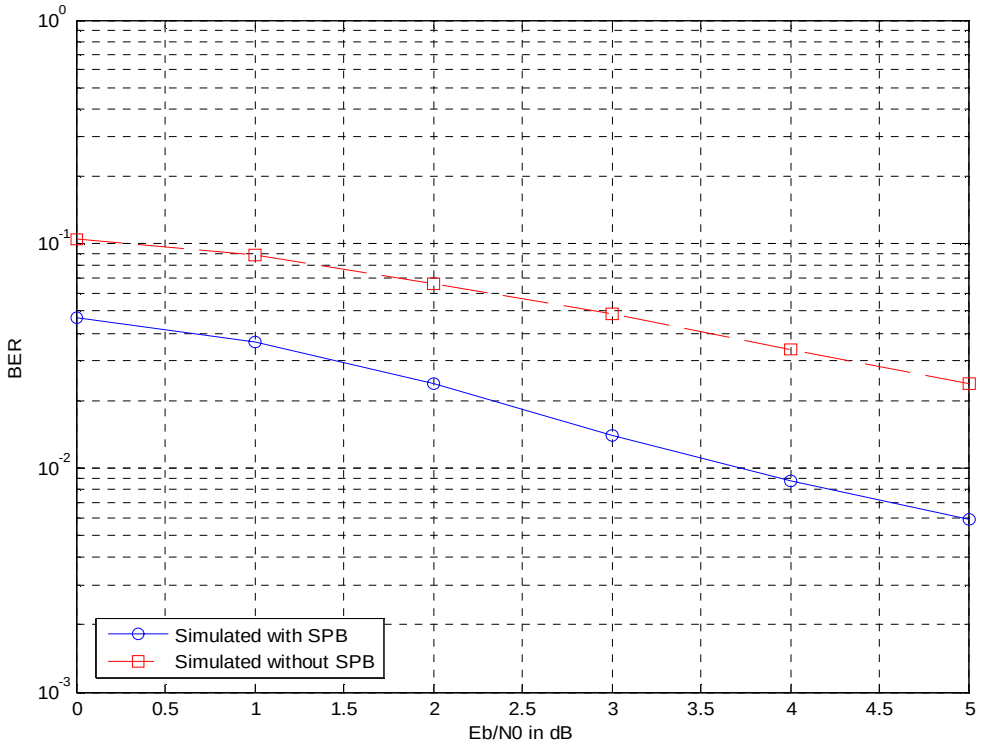


Fig. 8.2 BER vs. E_b/N_0 for Ricean channel

It can be seen from the plots in Figure 8.2 that coding gain is achieved with the introduction of the SPB.

8.4 Physical Layer Simulation of a Wireless System with the SPB for Ricean Channel and Diversity

Figure.8.3 is the plot of BER v/s E_b/N_0 obtained by simulating the PLB of a wireless system in presence of channel fading and diversity (Figure 6.2).

Parameters used for simulation:

1. Ricean channel parameters: Fading factor (K_{dB}) = 7dB, Doppler shift (f_d) = 20 Hz.
2. Diversity Combining: MRC
3. Diversity order(L): 3

The values of BER for different values of E_b/N_0 are shown in the Table 8.3.

Table 8.3 BER and E_b/N_0 for Ricean fading and diversity

E_b/N_0 in dB	0	1	2	3	4	5
Simulated without SPB	0.0846	0.0476	0.0208	0.0118	0.0056	0.0041
Simulated with SPB	0.0236	0.0048	0.0024	0.0010	0.0004	0.0003

It can be seen from the plots in Figure 8.3 that coding gain is achieved with the introduction of the SPB.

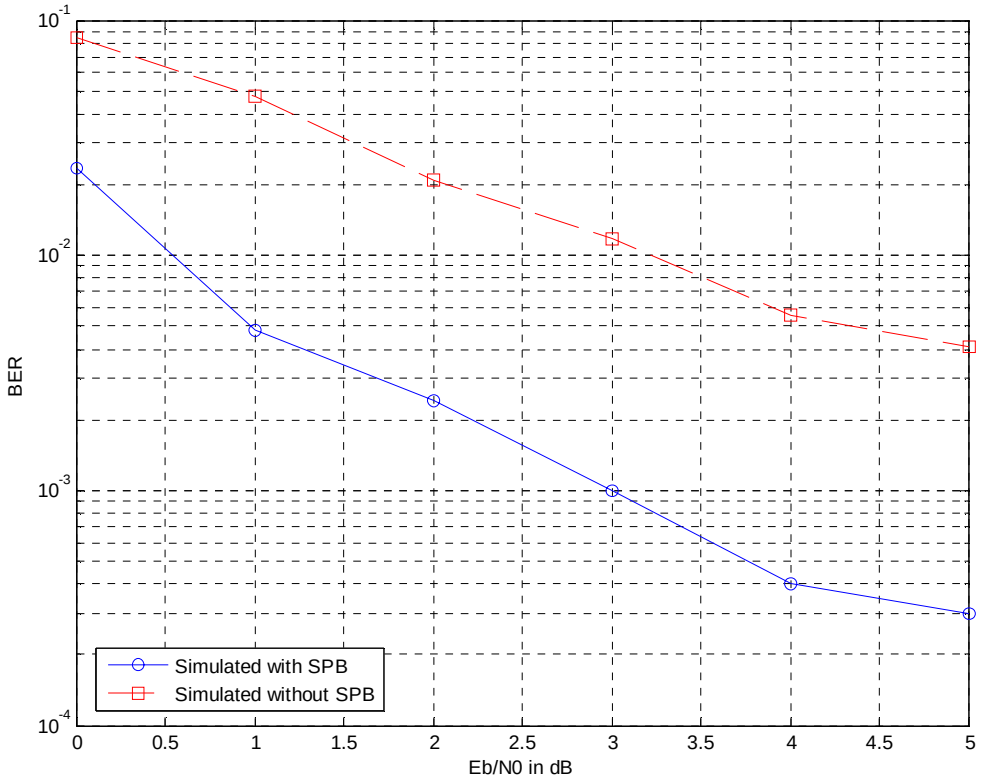


Fig.8.3 BER vs. E_b/N_0 for Ricean channel and diversity

8.5 Physical Layer Simulation of a Wireless System with the SPB for Ricean Channel and Channel Coding

Figure.8.3 is the plot of BER v/s E_b/N_0 obtained by simulating the physical layer of a wireless system in presence of a Ricean fading and channel coding (Figure 7.3).

Parameters used for simulation:

1. Ricean channel parameters: Fading factor (K_{dB}) = 7dB, Doppler shift (f_d) = 20 Hz.
2. Channel Coding: Convolutional coding and Viterbi decoding
3. Coding Rate (R) = 1/2
4. Generator sequence for encoder(g) : g^0 = [1 1 1] , g^1 = [1 0 1]

Table 8.4 BER and E_b/N_0 for Ricean fading and channel coding

E_b/N_0 in dB	0	1	2	3	4	5
Simulated without SPB	0.1038	0.0772	0.0348	0.0194	0.0066	0.0033
Simulated with SPB	0.0386	0.0154	0.0062	0.0014	0.0006	0.0043

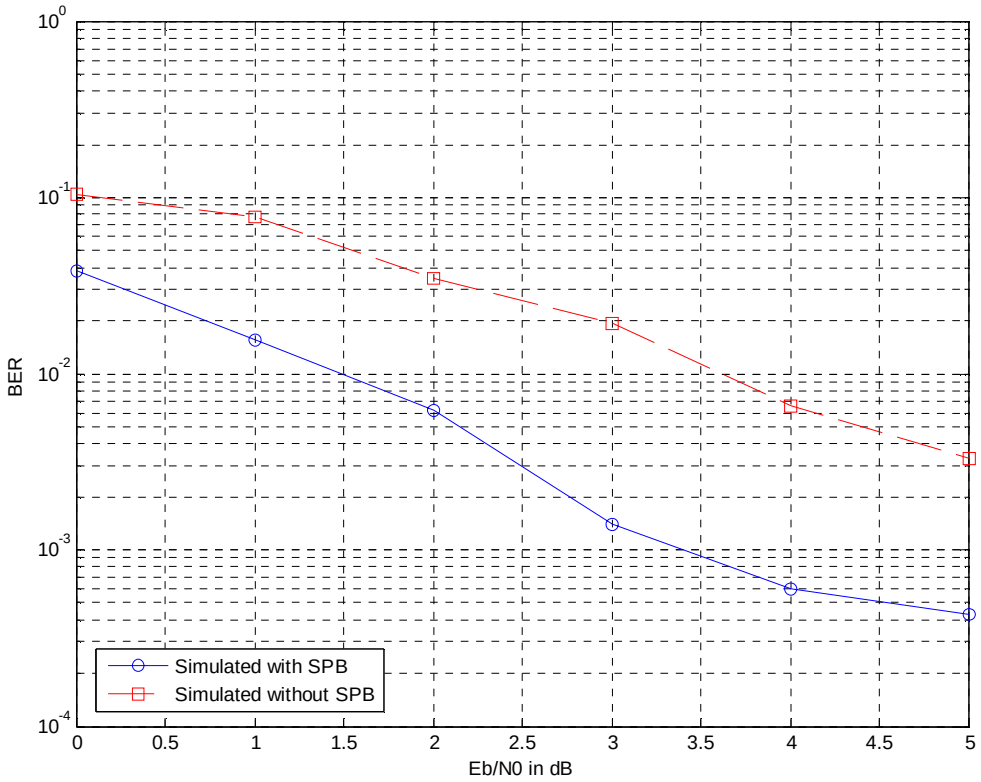


Fig. 8.4 BER vs. E_b/N_0 for Ricean channel and channel coding

It can be seen from the plot in Figure 8.4 that coding gain is achieved with the introduction of the SPB.

CHAPTER 9

CONCLUSIONS AND SUGGESTIONS FOR FUTURE WORK

9.1 Conclusions

In this work, we have introduced the SPB block at the receiver end and then simulated the physical layer of a wireless system. The simulation results prove that the introduction of the SPB helps in the reduction of BER, thereby improving the received signal quality.

9.1.1 Advantages of the Proposed SPB

- Improving the GSM system performance by reducing the BER and thereby potentially reducing call drops due to high bit error rate.
- Coding gain over an existing GSM system.
- Potentially help to extend the range of coverage area.

9.2 Suggestions for Future Work

- The simulation of the PLB of the wireless system with SPB takes approximately 4.5 minutes in MATLAB. Optimization is required for real time application.
- The simulations were carried out for the GSM system which can be extended to the CDMA system.

REFERENCES

[1] David Grace, Tim C.Tozer and Alister G.Burr "Reducing Call Dropping in Distributed Dynamic Channel Assignment by Incorporating Power Control in Wireless Ad Hoc Networks", *IEEE Journal on selected areas in communications,vol.18 no.11, pp. 2417-2419, November 2000.*

[2] Shigeki Shiokawa and Michihiro Ishizaka "Call Admission Scheme based on Estimation of Call Dropping Probability in Wireless Networks", *The 13th IEEE International Symposium on Personal, Indoor and Mobile Radio Communication, 2002, pp. 2185-2188.*

[3] Felipe A.Cruz-Perez, Lauro Ortigoza-Guerrero "Regulated Call Dropping Scheme for Handoff Prioritization in Wireless Communication Systems with Heterogeneous Platforms Types", *International Symposium on Personal, Indoor and Mobile Radio Communication Proceedings, 14th IEEE, pp. 1451-1455, 2003.*

[4] Youssef Iraqi and Raouf Boutaba, "Handoff and Call Dropping Probabilities in Wireless Cellular Networks", *International Conference on Wireless Networks, Communications and Mobile Computing, pp. 209-213, 2005.*

[5] P. Sarath Kumar and Jack Holtzman, "Analysis of Handoff Algorithm using both Bit Error Rate (BER) and Received Signal Strength'', *Wireless Information Network Laboratory (WINLAB), Rutgers University, pp.1-5, 1998.*

[6] Qing-AN Zeng and Dharma P. Agarwal, "Handoff in Wireless Mobile Networks" *Department of Electrical Engineering and Computer Science, University of Cincinnati, pp 1-26, 2001.*

[7] Cardoso, J.-F, "Blind Signal Separation: Statistical Principles", *Proceedings of the IEEE, vol. 86, pp. 2009-2025, Oct., 1998.*

[8] Aapo Hyvarinen and Erkki Oja, "Independent Component Analysis: Algorithms and Applications'', Neural *Networks Research Centre, Helsinki University of Technology, pp. 411-413, 2000.*

[9] Anthony J. Bell**,** Terrence J. Sejnowski**,** "An Information-Maximization Approach to Blind Separation and Blind Deconvolution**,**" *Howard Hughes Medical Institute, Computational Neurobiology Laboratory, The Salk Institute, pp. 1129-1159,1995.*

[10] M. Noll, "Pitch Determination of Human Speech by The Harmonic Product Spectrum, The Harmonic Sum Spectrum, and a Maximum Likelihood Estimate", *Proceedings of the Symposium on Computer Processing Communications, pp. 779–797, 1969.*

[11] Sudeep Sharma, "On Reducing Call Drop Outs by Improving Bit Error Rate Performance in the GSM Mobile communication System", *Master's Thesis, The University of Texas at Arlington, March, 2008.*

[12] Theodore S. Rappaport, "Wireless Communication: Principle and Practice", *Second Edition, Pearson Education, 2002.*

[13] John G. Proakis, "Digital Communications", *Third Edition McGraw-Hill International Editions, 1998.*

[14] William C. Jakes, "Microwave Mobile Communication", *Wiley-IEEE press, 1994*.

[15] P.Dent, G.E Bottomley, T. Croft, "Jakes Fading Model Revisited", Electronics *Letters, Vo1.29. No. 13, pp.1162-1 163, 1993.*

[16] Cardoso, J.-F, "Infomax and maximum likelihood for Blind Source Separation", *Signal Processing Letters, IEEE vol 4, pp. 112-114, Apr 1997.*

[17] Alberto Leon-Garcia, "Probability and Random Process for Electrical Engineering", *second edition, Pearson Education, 2004*.

[18] Andrew J. Viterbi and Audrey M. Viterbi, "Nonlinear Estimation of PSK-Modulation Carrier Phase with Application to Burst Digital Transmission", *IEEE Transactions on Information Theory, vol.IT-29, No.4, pp.543-551, 1983.*

[19] Henk Wymeersch and Marc Moeneclaey, "Iterative Code-Aided ML Phase Estimation and Phase Ambiguity Resolution", *EURASIP Journal on Applied Signal Processing, pp. 981–988, 2005:6.*

[20] Liangwei Ge and Lijuan Wang "Performance Analysis of MPSK System using MRC Diversity Reception in Rayleigh Fading Channel with Impulsive Noise" *Proceedings of IEEE TENCON, pp.1101-1112, 2002.*

[21] N. Ekanayake, "Performance of M-ary PSK Signals in Slow Rayleigh Fading Channel", IEEE *Electronics Letters, vol. 26, pp. 618-619, 1990.*

[22] R. Clarke, “A Statistical Theory of Mobile Radio Reception”, *Bell System Technology, vol. 47, pp. 957-1000, 1968.*

[23] A.J. Viterbi, “Error Bounds for Convolutional Codes and Asymptotically Optimum Decoding Algorithm,” *IEEE Transaction Information Theory*, pp. 260–269, 1967.

[24] B. Sklar, “Digital Communications - Fundamentals and Applications”, *Prentice Hall, 1988.*

BIOGRAPHICAL INFORMATION

Kumarabhijeet Singh was born on 26th June, 1982 in Bhagalpur, India. He completed his schooling from St.Fransis D'Assisi High School in Mumbai, India in 1999. He did his high school from Sathaye College, Mumbai in 2001. He received his Bachelors degree in Electronics and Telecommunication from Mumbai University in June 2005.He received his Master of Science degree in Electrical Engineering from The University of Texas at Arlington in May 2008. During his master's, he worked as an Engineering Intern in Qualcomm Inc, San Diego. He will be joining Qualcomm Inc, San Diego as a System Engineer in May 2008.