Sampling Time Jitter

by

Seyed Mohammad Seifi

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Applied Science
in
Electrical and Computer Engineering

Waterloo, Ontario, Canada, 2013

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners. I understand that my thesis may be made electronically available to the public.

Abstract

Electrical systems which use voltage transitions to represent timing information suffer from a degrading phenomenon called timing jitter. Sampling time jitter is the deviation of sampling clock from its ideal position. As transmission rates raise above couple of GHz, deviations become significant comparing to signalling interval, jitter becomes a fundamental performance bottleneck. Especially in band-limited communication systems that imperfect sampling times result in Inter-Symbol Interference (ISI) jitter is a very limiting factor to decode correct transmitted data. In this case, jitter timing error translates into amplitude error.

At first, the effect of sampling time jitter at the received signal is modelled as an additive noise .This additive noise signal which we call it jitter noise is a coloured noise that also depends on input signal. Expressions for jitter noise correlation factors, its cross-correlation with input signal are derived. These correlations depend on input spectral density, timing jitter characteristic function (Fourier transform of jitter probability density function) and whether timing jitter is white or coloured. In case of first order Gauss-Markov process for sampling time jitter it is observed that in high memory regime (highly correlated timing jitter) the spectral density of additive jitter noise is concentrated around higher frequencies. Exploiting this quality, a spectral shaping scheme is used to improve the performance in terms of Bit Error Rate (BER) for an AWGN channel with discrete input corrupted by sampling time jitter. Simulation results comparing the proposed scheme performance with a minimum distance decoder are provided.

As another approach the well-known Viterbi Algorithm is used for decoding same AWGN channel suffering from ISI terms due to sampling jitter. The Viterbi algorithm, which basically is a dynamic programming algorithm, finds the most likely input data and jitter times based on observed output sequence. A quantized version of jitter times is used to be able to work with a finite state trellis and to find likelihoods along the paths of the diagram. Then Bit Error Rate curves for different jitter quantization levels and different impulse response lengths of channel are presented.

Acknowledgements

I would like to express my sincere thanks and appreciation to my supervisor, Professor Amir.K. Khandani, for his mentorship, guidance, ideas, encouragement and full support. His great personality made this research a pleasant and wonderful journey. I would not have been able to accomplish all that I have over the past two years without his supervision. Thanks to Kamyar Moshkar for his great personality, his depth of knowledge, and his enduring support;and Hamid Ebrahimzad for always being available to listen and provide guidance. My warm wishes and thanks go as well to my dear friends, Hojat Abdollahnezhad, Babak Mamandipoor, Ershad Banijamali, Pouya Mahboubi and Akbar Ghasemi who have enriched my life beyond my studies. And most importantly my deepest appreciation goes to my family, whom I love and cherish dearly.I would like to express my deepest gratitude to my parents,Zahra Golzadeh, Alireza Seifi and my brother Mohsen Seifi , for their infinite love and continuous support and dedication.I would like to dedicate this work to them to show my appreciation for their unconditional love and sacrifice.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

One of the most important limiting factors degrading the performance of a communication system whether it is wireless,optical or storage medium is the absence of a common clock between the transmitter and the receiver.This asynchronism can have different corrupting effects in communication systems.In the following,we will first review works which has been done on information theoretic aspects of such channels corrupted by asynchronism.Then a review on timing jitter which is the main focus of this thesis is presented.

1.1 Literature review on information theoretic aspects of asynchronous channels

Depending on the transmission medium wide variety of asynchronous channel models has been introduced to tackle the problem of imperfect alignment of transmitter and receiver clocks.For example,consider the case where a sensor is emitting alarm messages to a centre on some random occasions.In order the centre detect the alarm message there should be a detection mechanism so as to find the start time of emission by sensor.This kind of asynchronism is modelled in the category of block asynchronous channels.On the other hand there are situations where the transmitter and receiver are block-wise synchronous(the time at which the transmission begins is known to the receiver),however, the receiver clock might suffer from a jitter in sampling the continuous time data.The following is a brief review of different asynchronous channel models studied in the literature and particularly the one which is considered in this thesis.

1.1.1 Frame asynchronous channel

In [17, 2, 16] the problem of frame asynchronous channel is considered.In this model the messages are encoded into codewords of fixed length and this codeword starts being transmitted at some random time instance unknown to receiver within a prescribed window.The windows is exponentially scaled with codeword length(scaling parameter "α", also referred as asynchronism exponent).Then pairs of achievable rates and asynchronism exponent (R, α) for a reliable communication is derived.Please note that in the proposed model, there is synchronism between transmitter and receiver in terms of time slots.However,the ambiguity is at what time slots the actual codeword is transmitted and what time slots pure noise is received.

1.1.2 Timing error

A somewhat different model is synchronization error channels suffering from irregularity in a communication medium and transceiver hardware.In many communication system continuous time data is sampled to produce discrete time samples.In an idle sampling system,the continuous time signal is sampled periodically at rate $1/T = 2W$, W being the highest frequency component of signal.Due to Nyquist-Shannon theorem [13, 5, 15]the continuous time data can be completely recovered by discrete time samples as long as the sampling rate is greater or equal to $2W$.However,in real life systems timing error is introduced in the sampling mechanism which is called "time jitter".Hence,sampling epochs are not any more the idle times, $T_s = nT = n/2W$.

In the case where these sampling jitters are comparable to duration of a symbol (symbol period),the asynchronism effect on the received data would be insertion and/or deletion of some symbols.Hence,each time a codeword is transmitted a string of variable random length codeword is received. This channel model of Dobrushin[8]has recently gained a lot of interest.Extensive studies[1, 19, 7, 10, 9, 11] has been done to derive upper and lower bound on the capacity of such channels.Furthermore,a family of error correcting codes has been presented by Davey and Mackay [6] for insertion deletion and substitution channels.

In most practical cases the above mentioned sampling jitter has a variance much less than duration of a symbol(symbol period).In such a case the length of the codeword received is the same as the length of the codeword transmitted.Moreover,the receiver is able to perfectly detect when a sequence of codeword is being started to receive. Transmission media in practice has a limited finite bandwidth, hence,any pulse transmitted through such a medium will have (theoretically) infinite time duration.We have to keep this in mind the pulses we use in practice might not actually go on and back in time for ever,but would have a duration considerably much more than data rate transmission $1/T$.This will cause a phenomenon called inter-symbol interference (**ISI**), where the data conveyed with one symbol instance cause the data on other symbol instances be misinterpreted.To avoid such misinterpretation, pulse shaping techniques are used, one must carefully use pulses which has a zero value at sampling time instances of other symbols,which means:

$$P(nT) = \begin{cases} 1 & n = 0 \\ 0 & \forall n \in \mathbf{Z} - \{0\} \end{cases} \tag{1.1}$$

Common pulses satisfying the above constraint are sinc pulse and raised cosine.Even if such pulses are used,in presence of sampling jitter the sampled data values at integer multiples of sampling times ,iT, will suffer from **ISI**.

In[3] to simplify effect of timing jitter on the amplitude of the received signal it is assumed that jitter is small enough such that

$$z(nT) = x(nT + \zeta_{\mathbf{n}}) - x(nT) \simeq \dot{x}(nT) * \zeta_{\mathbf{n}} \tag{1.2}$$

where $z(nT)$ is the error in amplitude due to jitter, ζ_n is the jitter value at nth symbol and $\dot{x}(nT)$ is derivative of the signal at nth sampling time.Then based on this simplified model upper and lower bounds on the capacity of channel with jitter is derived .

1.2 A review on timing jitter and its characteristics

As previously being said sampling jitters are the deviation of sampling clocks from their ideal values.The following is a summary of sampling jitter types, metrics and characteristics.For more detailed information refer to [12, 20]. 3 metrics used to characterize jitter are:

- **Time interval error (TIE)** measures the difference between the clock edge and its absolute value.TIE is the measure which is used in this thesis.
- **Cycle jitter** is the difference between the actual clock period and the ideal clock period, which is discrete time derivative of TIE .
- **Cycle to cycle jitter** measures how much the clock period changes between any two adjacent cycles which can be thought of discrete time derivative of cycle jitter.

Timing jitter is composed of two types of jitter,random jitter and deterministic jitter.Random jitter is timing noise which can not be predicted.It is assumed to have a Gaussian distribution.Random jitter is often caused by thermal effects of semiconductors and requires a deeper understanding of device physics.That along with central limit theorem is the reason why a Gaussian distribution is assumed for random jitter.The unit used for standard deviation of the Gaussian jitter is Unit Interval (abbreviated UI)which quantifies the jitter in terms of a fraction of the ideal period of a symbol .On the other hand,deterministic jitter is timing jitter that is repeatable and predictable.Unlike random jitter deterministic jitter is bounded.

Electronic circuits used to mitigate the level of jitter are called anti-jitter circuits.Examples of these circuitries are phase-locked loops and delay-locked loops.Even these circuits can not remove timing jitter completely.

In this thesis the focus is on random jitter model for channel asynchronism.For such timing jitter model,Balakrishnan in [4] has studied the effect of jitter on the received signal.Balakrishnan also estimated various error measures due to jitter and considered optimal use of jittered samples.Using and extending his work we have modelled the effect of timing jitter on signal modelled as an additive noise.Some characteristics of this ”additive jitter noise” is studied;These characteristics include its spectral density and cross-correlations with input signal.Based on these characteristics spectral shaping and decoding rules are proposed to improve the performance of the discrete input AWGN channel suffering from sampling time jitter in terms of bit error rate.

1.3 Thesis organization

In chapter 1 of this thesis the effect of sampling time jitter at received signal is modelled as an additive noise .Then characteristics of this additive jitter noise is studied including its spectral density and cross correlation factors with input signal.These characteristics have been derived for both cases when timing jitter is white or is coloured and correlated in time.Then numerical results of spectral densities are provided.These spectral are derived for a Gauss-Markov model for timing jitter which is the case in most practical systems.Case of high memory in timing jitter is of more interest in this thesis.

In chapter 2 exploiting the result and observations in chapter 1 a spectral shaping scheme is used for an *AWGN* channel with discrete input corrupted by sampling time jitter with Gauss-Markov distribution and a strong memory correlation factor, to better its performance in terms of *BER*.At the end of the chapter simulation results comparing the proposed scheme performance with a minimum distance decoder are provided.

In chapter 3 the well-known *Viterbi Algorithm* which is basically a dynamic programming algorithm is used for decoding same AWGN channel suffering form *ISI* terms due to sampling jitter.The algorithm finds the most likely input data and jitter times based on observed output sequence.A quantized version of jitter times is used to find likelihood probabilities along the paths of the trellis diagram.Then *BER* curves for different jitter quantization levels and different impulse response lengths of channel are presented.

Chapter 2

Jitter noise characteristics

In[4, 3] properties of output jittered samples are studied.In this chapter we follow the same formulations, modelling the amplitude error as an additive noise,$z_n = z(nT)$, we present characteristics of z_n which we also call it "**jitter noise**".These characteristics include "jitter noise" autocorrelation and spectral density and its cross-correlation with input signal for both white and coloured timing jitter.We will see how timing jitter memory will affect jitter noise spectral density.While in [4] the output process characteristics has been studied,following same steps we do formulations for output process minus input (additive jitter noise) and also we find cross-correlations of this additive jitter noise with input.We also numerically compute some of these qualities to get a deeper insight.For example, as we will see,although memory in timing jitters has a huge effect on the spectral density of additive jitter noise it won't change it's cross-correlations with input at all.Another observation is that in case of first order Gauss-Markov process for jitter the more memory in timing jitter the more the spectral density is concentrated around higher frequencies.

Symbols $x(n)$ provide a discrete parameter process,which can present a codeword intended to be transmitted through a communication system.These discrete samples are transmitted through the communication channel by means of continuous time signals.A typical example of such continuous time signals is ”Sinc” function :

$$x(t) = x_n a_n(t) \tag{2.1}$$

Where,

$$a_n(t) = \frac{sin\pi(2Wt - n)}{\pi(2Wt - n)} \tag{2.2}$$

and $2W$ being the bandwidth. The timing error introduced in sampling mechanism errors in amplitude of the output process.The samples are now taken at times $(nT + \zeta_{\mathbf{n}})$ the corresponding samples now become,

$$y_n = x(nT + \zeta_{\mathbf{n}}) \tag{2.3}$$

Hence we can consider the sampling error as an additive noise z_n

$$z_n = y_n - x_n = x(nT + \zeta_{\mathbf{n}}) - x(nT) \tag{2.4}$$

Let signal x_n be stationary random with zero mean.Then $x(t)$ in (2.1) will also be stationary,zero mean with correlation R(t) such that,

$$R(t) = \int_{-\infty}^{\infty} e^{2\pi ift} p_x(f) df \tag{2.5}$$

Where $p_x(f)$ is the spectral density of the random process $x(t)$. Then using conditional expectation

$$R_{zz}(n, m) = E[z_n z_m] = E[(x(nT + \zeta_{\mathbf{n}}) - x(nT))(x(mT + \zeta_{\mathbf{m}}) - x(mT))]$$

$$= E[x(nT + \zeta_{\mathbf{n}})x(mT + \zeta_{\mathbf{m}}) + x(nT)x(mT) - x(nT + \zeta_{\mathbf{n}})x(mT) - x(nT)x(mT + \zeta_{\mathbf{m}})]$$

$$= E_{\zeta_{\mathbf{n}},\zeta_{\mathbf{m}}}[E\{x(nT+\zeta_{\mathbf{n}})x(mT+\zeta_{\mathbf{m}})+x(nT)x(mT)-x(nT+\zeta_{\mathbf{n}})x(mT)-x(nT)x(mT+\zeta_{\mathbf{m}})\} \mid \zeta_{\mathbf{n}}, \zeta_{\mathbf{m}}]$$

$$\begin{aligned} &= \int_{-\infty}^{\infty} e^{2\pi i f(n-m)T} C_{n,m}(f,-f) p_x(f) df \\ &+ \int_{-\infty}^{\infty} e^{2\pi i f(n-m)T} p_x(f) df \\ &- \int_{-\infty}^{\infty} e^{2\pi i f(n-m)T} C_n(f) p_x(f) df \\ &- \int_{-\infty}^{\infty} e^{2\pi i f(n-m)T} C_m(-f) p_x(f) df \end{aligned} \tag{2.6}$$

Where

$$C_{n,m}(f,-f) = E[exp\ 2\pi i f(\boldsymbol{\zeta}_{\mathbf{n}} - \boldsymbol{\zeta}_{\mathbf{m}})] \tag{2.7}$$

and

$$C_n(f) = E[exp\ 2\pi i f \boldsymbol{\zeta}_{\mathbf{n}}] \tag{2.8}$$

in which the independence of $x(t)$ and $\boldsymbol{\zeta}_{\mathbf{n}}$ has been used.

If jitter time $\boldsymbol{\zeta}_{\mathbf{n}}$ considered to be strict sense stationary then $C_{n,m}(f,-f)$ depends only on $(n-m)$, so the z_n process has stationary covariance.Please note that $C_{n,m}(f,-f)$ depends on the characteristics of timing jitter and whether it is white or has memory. Lets first have a closer look on the case when the jitter process $\boldsymbol{\zeta}_{\mathbf{n}}$ is white.

2.1 White jitter

If $\zeta_\mathbf{n}$is independent of $\zeta_\mathbf{m}$ and process $\zeta_\mathbf{n}$ be strict sense stationary , we have for any (n, m), $n \neq m$,

$$C_{n,m}(f, -f) = \mid E[e^{2\pi i \zeta_\mathbf{n} f}] \mid^2 = \mid C(f) \mid^2 \tag{2.9}$$

Hence we can now simplify (2.32) as follows :

$$\begin{aligned} E[z_n z_m] &= \int_{-\infty}^{\infty} e^{2\pi i f(n-m)T}(\mid C(f) \mid^2 - C(f) - C^*(f) + 1)p_x(f)df \\ &= \int_{-\infty}^{\infty} e^{2\pi i f(n-m)T}(\mid C(f) \mid^2 - 2Re\{C(f)\} + 1)p_x(f)df \end{aligned} \tag{2.10}$$

Now if we consider a symmetric pdf (probability density function) for jitter $\zeta_\mathbf{n}$, which is reasonable, then $Re\{C(f\}) = C(f)$. Hence,

$$E[z_n z_m] = R_{zz}(n, m) = R_{zz}(n - m) = \int_{-\infty}^{\infty} e^{2\pi i f(n-m)T}(C(f) - 1)^2 p_x(f)df \tag{2.11}$$

equation (2.11) demonstrates how the jitter noise correlation in time is related to the input signal $x(t)$ autocorrelation and the Fourier transform for probability density function of timing jitter. Taking $1/T = 2W$ Nyquist sampling rate ,

$$R_{zz}(n) = E[z_{2n} z_n] = \int_{-1/2}^{1/2} e^{2\pi i f n}\psi(f)df \tag{2.12}$$

$$\psi(f) = \sum_{k=-\infty}^{\infty} (C(2Wf + 2kW) - 1)^2 p_x(2Wf + 2kW) \tag{2.13}$$

In general,the input process $x(t)$ can have arbitrary spectrum in frequency domain.However,random process $x(t)$ as defined in (2.1) is band-limited to $(-W, W)$.Thus, equation (2.13) is simplified to :

$$\psi(f) = (C(2Wf) - 1)^2 p_x(2Wf) \tag{2.14}$$

Therefore, we can rewrite the autocorrelation $R_z(n)$ for $n \neq 0$:

$$R_{zz}(n) = \int_{-1/2}^{1/2} e^{2\pi ifn}(C(2Wf) - 1)^2 p_x(2Wf)df \tag{2.15}$$

For $n = 0$ $C_{n,n}(f, -f) = 1$ and,

$$R_{zz}(0) = 2\int_{-1/2}^{1/2}(1 - C(2Wf))p_x(2Wf)df \tag{2.16}$$

$R_{zz}(0)$ is the power of jitter noise which is linearly scaled with input power $R_{xx}(0)$ for a white input.Using both equations (2.15) and (2.16),spectral density function for jitter noise is found .Please note that spectral density for discrete process x_n, $P_x(f)$, is a frequency scaled version of spectral density for interpolated continuous-time process $x(t)$, $p_x(f)$, which means,

$$P_x(f) = p_x(f/2W) \quad -1/2 < f < 1/2 \tag{2.17}$$

$$P_z(f) = \sum_{\infty}^{\infty} R_{zz}(n)e^{-2\pi ifn} \tag{2.18}$$

To find spectral density function for jitter noise, instead of counting above summation we do the same trick used in [4] :

$$P_z(f) = (C(2Wf) - 1)^2 p_x(f) + c \tag{2.19}$$

Whatever the constant term c is in equation (2.19), $P_z(f)$ satisfies (2.15) for all n.Yet, it must satisfy (2.16).Without loss of generality lets assume input power is equal to unity. Thus we obtain,

$$c = R_{zz}(0) - \int_{-1/2}^{1/2}(C(2Wf) - 1)^2 p_x(2Wf)df \tag{2.20}$$

For a Gaussian jitter of variance σ^2, $C(f)$ is

$$C(f) = e^{-2\pi^2 f^2 \sigma^2} \tag{2.21}$$

Again without loss of generality,lets take $2W = 1$,for a Gaussian jitter the power spectral density for jitter noise is

$$P_z(f) = (e^{-2\pi^2 f^2 \sigma^2} - 1)^2 p_x(f) + R_{zz}(0) - \int_{-1/2}^{1/2} (e^{-2\pi^2 f^2 \sigma^2} - 1)^2 p_x(f) df \tag{2.22}$$

The calculated spectral density above, for white input is depicted in figure (2.1) . As one can notice the spectral density for jitter noise is almost flat for white timing jitter.

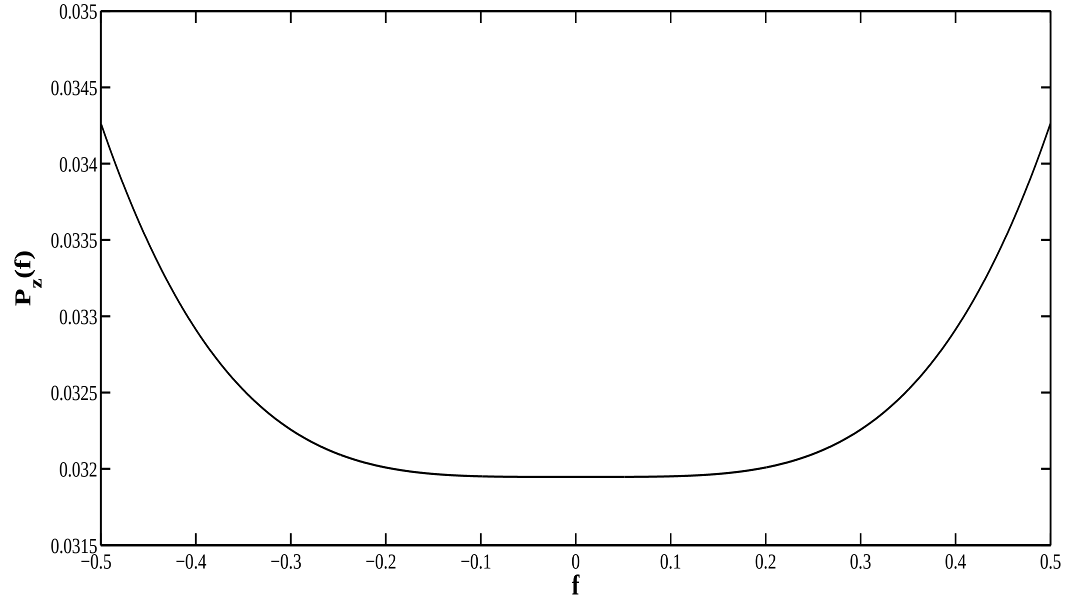


Figure 2.1: Jitter noise spectrum for white Gaussian timing jitter with $\sigma^2 = 0.01$

	$\rho_{zz}(0)$	$\rho_{zz}(1)$	$\rho_{zz}(2)$	$\rho_{zz}(3)$	$\rho_{zz}(4)$	$\rho_{zz}(5)$
$\sigma^2 = 0.001$	1	-0.0001190	0.0000643	-0.0000314	0.0000182	-0.0000118
$\sigma^2 = 0.01$	1	0.0545	-0.0133	0.0059	-0.0033	0.0021
$\sigma^2 = 0.02$	1	0.0442	0.0234	-0.0111	0.0064	-0.0041
$\sigma^2 = 0.05$	1	-0.2469	0.1261	-0.0576	0.0326	-0.0210

Table 2.1: Normalized jitter noise correlation coefficients for different values of timing jitter variance,timing jitter process is Gaussian variance σ^2

2.2 Coloured jitter

For the general case when the stationary jitter is coloured,(2.32) still holds.Although

$$C_{n,m}(f,-f) = C_{n-m}(f) = E[exp2\pi i f(\zeta_{\mathbf{n}} - \zeta_{\mathbf{m}})] \tag{2.23}$$

Lets investigate the effect of memory on correlation factors in equation 2.16 .Similar to [4],

$$R^m_{zz}(n) = R_{zz}(n) + D(n) \tag{2.24}$$

Where,

$$D(n) = \begin{cases} \int_{-1/2}^{1/2} [C_n(f) - \mid C(f) \mid^2] P_x(f) df & ,\forall n \neq 0 \\ 0 \quad , n = 0 \end{cases} \tag{2.25}$$

We call $D(n)$ perturbation in jitter noise correlation due to timing jitter memory.Consequently we can find the perturbation in jitter noise spectral density.

$$P^m_z(f) = P_z(f) + P_D(f) \tag{2.26}$$

Where,

$$P_D(f) = \sum_{-\infty}^{\infty} D(n) e^{-2\pi i n f} \tag{2.27}$$

Considering a Gaussian process for timing jitter, $C_n(f)$ is fully characterised by the covariance matrix of the timing jitter.Otherwise, higher moments of the jitter process is needed to describe $C_n(f)$.For the Gaussian process with variance σ^2 and covariance

$$E(\zeta_{\mathbf{n}} \zeta_{\mathbf{n+k}}) = r(k) \tag{2.28}$$

Then in (2.25),

$$C_n(f) - \mid C(f) \mid^2 = (e^{-4\pi^2 f^2 \sigma^2})(e^{4\pi^2 f^2 r(n)} - 1) \tag{2.29}$$

For the first order Gauss-Markov process we can replace r(n) with :

$$r(n) = r^{-|n|}\sigma^2 \qquad 0 < r < 1 \tag{2.30}$$

In (2.30) a closer r to one implies more correlation (memory) in timing jitter process.Thus,

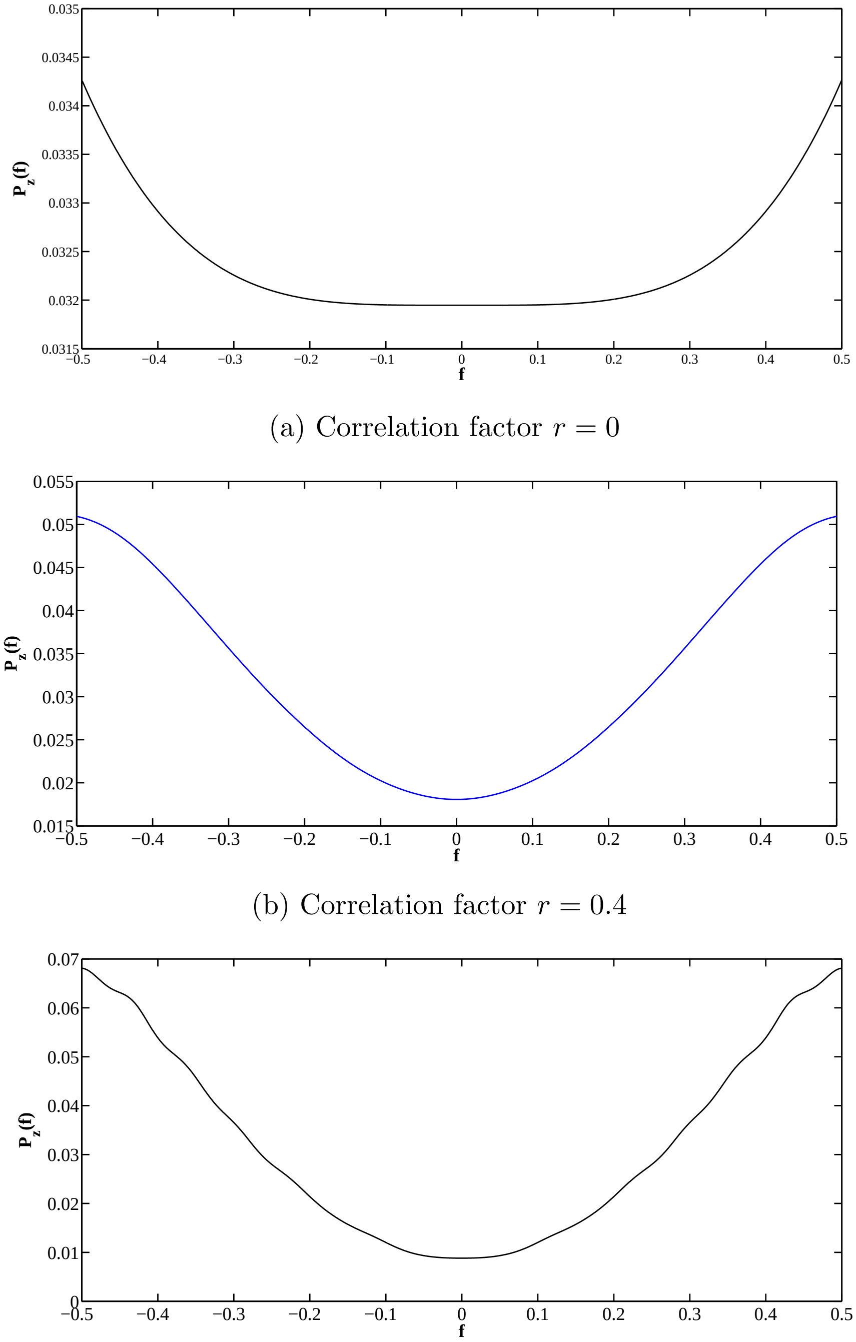
$$P_D(f) = 2\sum_{1}^{\infty} cos(2\pi n f) \int_{-1/2}^{1/2} (e^{-4\pi^2 f^2 \sigma^2})(e^{4\pi^2 f^2 r^{-|n|}\sigma^2} - 1)P_x(f)df \tag{2.31}$$

Unfortunately, finding a close form for equation (2.31) seems to be hard.That being said, for large n the integrand term in (2.31) converges to zero and so the whole integral.Consequently, we can numerically compute $P_D(f)$ as accurately as desire by truncating the summation to large n.Using this argument and MATLAB software the spectral density for some values of jitter memory r, are depicted in figure 2.2 .

The first important observation is that for a fixed jitter variance,the total power of jitter noise does not depend on jitter memory. Important observations that can be extracted from formulas and figures for a first order Gauss-Markov process timing jitter :

- For a fixed jitter variance,the total power of jitter noise does not depend on jitter memory.
- For a white jitter the spectral density is almost flat.
- The more memory in jitter process (in other words the bigger r), the more concentrated the jitter noise spectral density is around higher frequencies.

Table below shows how correlation coefficients change with memory in first order Gauss-Markov timing jitter.



(a) Correlation factor $r = 0$

(b) Correlation factor $r = 0.4$

(c) Correlation factor $r = 0.7$

Figure 2.2: Spectrum density for jitter noise for $AR(1)$ timing jitter with different correlation factors and $\sigma^2 = 0.01$

	$\rho_{zz}(0)$	$\rho_{zz}(1)$	$\rho_{zz}(2)$	$\rho_{zz}(3)$	$\rho_{zz}(4)$	$\rho_{zz}(5)$
$r = 0$	1	-0.0115	0.0061	-0.0030	0.0017	-0.0011
$r = 0.2$	1	-0.1266	0.0113	-0.0033	0.0018	-0.0011
$r = 0.4$	1	-0.2436	0.0272	-0.0066	0.0025	-0.0010
$r = 0.6$	1	-0.3624	0.0544	-0.0155	0.0055	-0.0027
$r = 0.8$	1	-0.4830	0.0941	-0.0335	0.0152	-0.0080
$r = 0.9$	1	-0.5440	0.1191	-0.0474	0.0239	-0.0138
$r = 0.99$	1	-0.5993	0.1449	-0.0635	0.0353	-0.0223

Table 2.2: Normalized jitter noise correlation factors for different values of jitter memory r , AR(1) model for timing jitter with variance $\sigma^2 = 0.01$

2.3 Jitter noise cross-correlation with input process

In this section formulations for jitter noise cross-correlation with input process as well as some numerical results are presented.These numerical results can provide insights on how we can use the dependency with input along with the self-correlation to enhance the performance of an additive white Gaussian noise channel suffering from jitter,which is studied in upcoming chapters. Steps of deriving cross-correlation are same as before :

$$R_{zx}(n,m) = E[z_n x_m] = E[(x(nT+\zeta_{\mathbf{n}}) - x(nT))(x(mT))]$$

$$= E[x(nT+\zeta_{\mathbf{n}})x(mT) - x(nT)x(mT)]$$

$$= E_{\zeta_{\mathbf{n}}} E[x(nT+\zeta_{\mathbf{n}})x(mT) - x(nT)x(mT) \mid \zeta_{\mathbf{n}}]$$

$$= \int_{-\infty}^{\infty} e^{2\pi i f(n-m)T}(C(f)-1)p_x(f)df \tag{2.32}$$

One can immediately conclude from (2.31) that unlike correlation factors,cross-correlation factors do not depend on timing jitter memory.In the table 2.3 normalized cross-correlation factors for a first order Gauss-Markov process with different timing jitter variances is provided.

	$\rho_{zx}(0)$	$\rho_{zx}(1)$	$\rho_{zx}(2)$	$\rho_{zx}(3)$	$\rho_{zx}(4)$	$\rho_{zx}(5)$
$\sigma^2 = 0.001$	-0.0091	0.0055	-0.0014	0.0006	-0.0003	0.0002
$\sigma^2 = 0.01$	-0.0900	0.0545	-0.0133	0.0059	-0.0033	0.0021
$\sigma^2 = 0.02$	-0.1775	0.1069	-0.0255	0.0113	-0.0063	0.0040
$\sigma^2 = 0.05$	-0.4250	0.2526	-0.0561	0.0245	-0.0137	0.0087

Table 2.3: Normalized jitter noise cross-correlation coefficients with input for different values of timing jitter variance

Chapter 3

Spectral Shaping

In this chapter a ”Spectral Shaping” scheme is proposed to tackle the problem of jitter noise in high memory regime.As we have already seen in high memory regime for timing jitter,jitter noise exhibits a high frequency spectrum;The heart of scheme presented in this chapter is this property which in other words is the large negative correlation factor between two consecutive jitter samples. The channel model of interest is an additive white Gaussian noise channel with timing jitter.The relationship between the input and output of such a channel can be written as:

$$y_n = x_n + z_n + w_n \tag{3.1}$$

where z_n is the additive jitter noise as before and w_n is and additive white Gaussian process independent of input x_n.Values x_n are chosen i.i.d. from the finite size constellation points.Throughout this thesis all the simulations are based on either *4PAM* or *BPSK* input constellations.Also without loss of generality lets assume input power is scaled to unity.

An empirical distribution of jitter noise for first order Gauss-Markov timing jitter with $\sigma^2 = 0.01$ and $r = 0.99$ is depicted in figure 3.1 .The distribution is obtained by a long run of channel (3.1) for *4PAM* input in absence of Gaussian noise.The empirical distribution (solid curve) is far from a Gaussian distribution of same variance (dashed curve).This imply

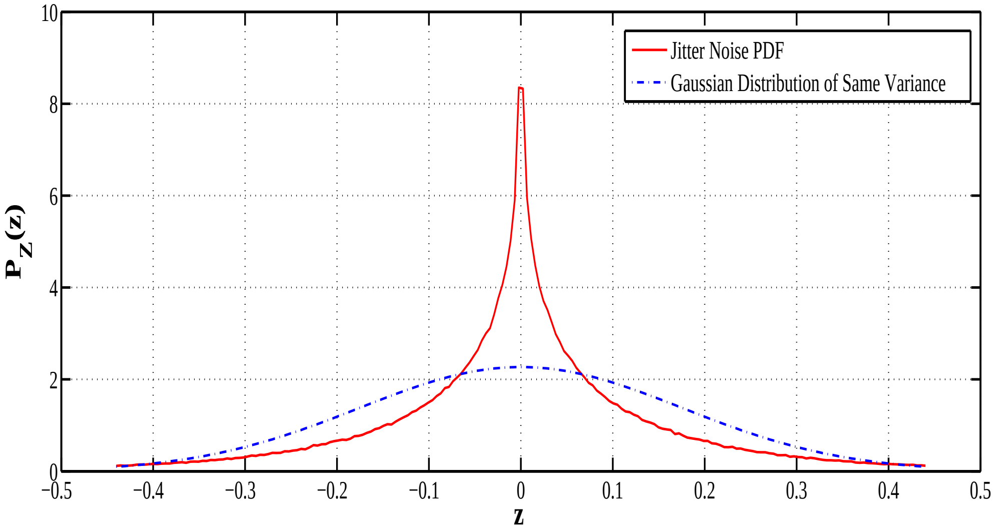


Figure 3.1: Jitter noise distribution vs. Gaussian distribution of the same variance

that for the channel of (3.1) a maximum likelihood decoder is no more a minimum distance decoder.

As we have seen in previous chapter for correlation factors of jitter noise due to strongly correlated timing jitter,are much bigger comparing to cross-correlations with input.This gives us a hint that in order to improve the performance of a jitter channel it can be much more useful to focus on spectral of jitter noise rather than its cross-correlation with input.

What we will do is to expand the constellation points over an orthonormal basis where the power spectrum of the basis cover different portions of the spectrum.The high spectrum basis will encounter stronger jitter noise.Then,the solution is to allocate less rate to worse dimension or adjust the power allocation between the good and bad dimension.An example of a good candidate for the orthonormal basis is Hadamard basis.

3.1 Hadamard transform

The Hadamard transform H_m is a $2^m \times 2^m$ matrix.The Hadamard matrix can be defined recursively as follows:

$$\begin{cases} H_0 = 1 \\ H_m = \frac{1}{\sqrt{2}} \begin{pmatrix} H_{m-1} & H_{m-1} \\ H_{m-1} & -H_{m-1} \end{pmatrix} & , m > 0 \end{cases} \tag{3.2}$$

Where $\frac{1}{\sqrt{2}}$ is a normalization factor.Hadamard matrices are only composed of zeros and ones.For example, for H_1 and H_2,

$$H_1 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \tag{3.3}$$

$$H_2 = \frac{1}{2} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix} \tag{3.4}$$

- Hadamard transform is an orthonormal transform, which means $HH^t = I$, where I is identity matrix.
- $H = H^t$,Hadamard matrix is symmetric .

3.2 Spectral shaping using H_1

As shown in figure 3.2 , modulation points are expanded over the orthonormal bases using 2 dimensional Hadamard transform. x' process, the output of block H is generated as follows:

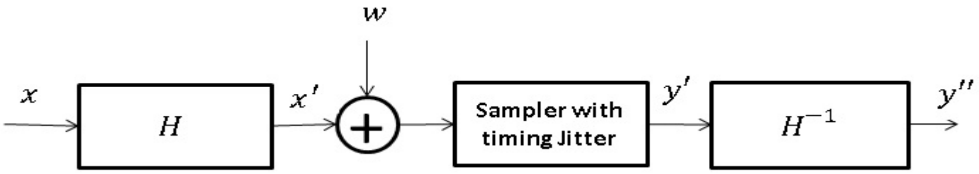


Figure 3.2: Expansion over Hadamard basis

$$\begin{pmatrix} x'_1 & x'_3 & x'_5 & \cdots \\ x'_2 & x'_4 & x'_6 & \cdots \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \times \begin{pmatrix} x_1 & x_3 & x_5 & \cdots \\ x_2 & x_4 & x_6 & \cdots \end{pmatrix} \tag{3.5}$$

Therefore,

$$y'_n = x'_n + z'_n + w'_n \tag{3.6}$$

Input process x is a stationary white process,and H transform is orthonormal preserving average power of input process.The Process x'_n is also white, since,

$$R_{x'x'}(1) = E[x'_1 x'_2] = E[(x_1 + x_2)(x_1 - x_2)] = 0 \tag{3.7}$$

This implies spectrum $P_{z'}(f) = P_z(f)$ as if there was no transformation in input.however,

$$y''_n = x_n + z''_n + w''_n \tag{3.8}$$

where,

$$\begin{pmatrix} z''_1 & z''_3 & z''_5 & \cdots \\ z''_2 & z''_4 & z''_6 & \cdots \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \times \begin{pmatrix} z'_1 & z'_3 & z'_5 & \cdots \\ z'_2 & z'_4 & z'_6 & \cdots \end{pmatrix} \tag{3.9}$$

The process z'' is no more stationary as its average power over 2 dimensions (dimension 1: odd transmissions, dimension 2: even transmissions) is different.To interpret what is happening on either dimensions consider filters g_l (low-pass) and g_h (high-pass) with impulse responses as following :

$$\textbf{low pass filter } \mathbf{g_l} \textbf{ :} \begin{cases} \begin{pmatrix} g_l(0) \\ g_l(-1) \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} & \\ g_l(i) = 0 & , i \neq -1, 0 \end{cases} \tag{3.10}$$

$$\textbf{high pass filter } \mathbf{g_h} \textbf{ :} \begin{cases} \begin{pmatrix} g_h(0) \\ g_h(-1) \end{pmatrix} = \begin{pmatrix} 1 \\ -1 \end{pmatrix} & \\ g_h(i) = 0 & , i \neq -1, 0 \end{cases} \tag{3.11}$$

The process z'' on its first dimension can be taught of being obtained through filtering process z' by g_l and decimating even transmission times.Since z' is concentrated around high frequencies,and g_l is low-pass, consequently,jitter noise power on first dimension would decrease.On the other hand, for second dimension g_h is a high-pass filter resulting stronger noise on second dimension.Rewriting equation (3.9) :

$$\begin{aligned} \begin{pmatrix} z''_1 & z''_3 & z''_5 & \cdots \\ z''_2 & z''_4 & z''_6 & \cdots \end{pmatrix} &= \frac{1}{\sqrt{2}} \begin{pmatrix} z'_1 + z'_2 & z'_3 + z'_4 & z'_5 + z'_6 & \cdots \\ z'_1 - z'_2 & z'_3 - z'_4 & z'_5 - z'_6 & \cdots \end{pmatrix} \\ &= \frac{1}{\sqrt{2}} \begin{pmatrix} (g_l * z')_{(2k-1)} \\ (g_h * z')_{(2k-1)} \end{pmatrix} \quad k = 1, 2, ... \end{aligned} \tag{3.12}$$

For jitter noise power on each dimension we have,

$$\mathbf{P_{odd}} = E[z''_{2k-1} z''_{2k-1}] = \tfrac{1}{2} E[(z'_{2k-1} + z'_{2k})^2] = (1 + \rho) R_{z'z'}(0) \quad k = 1, 2, ... \tag{3.13}$$

$$\mathbf{P_{even}} = E[z''_{2k} z''_{2k}] = \tfrac{1}{2} E[(z'_{2k-1} - z'_{2k})^2] = (1 - \rho) R_{z'z'}(0) \quad k = 1, 2, ... \tag{3.14}$$

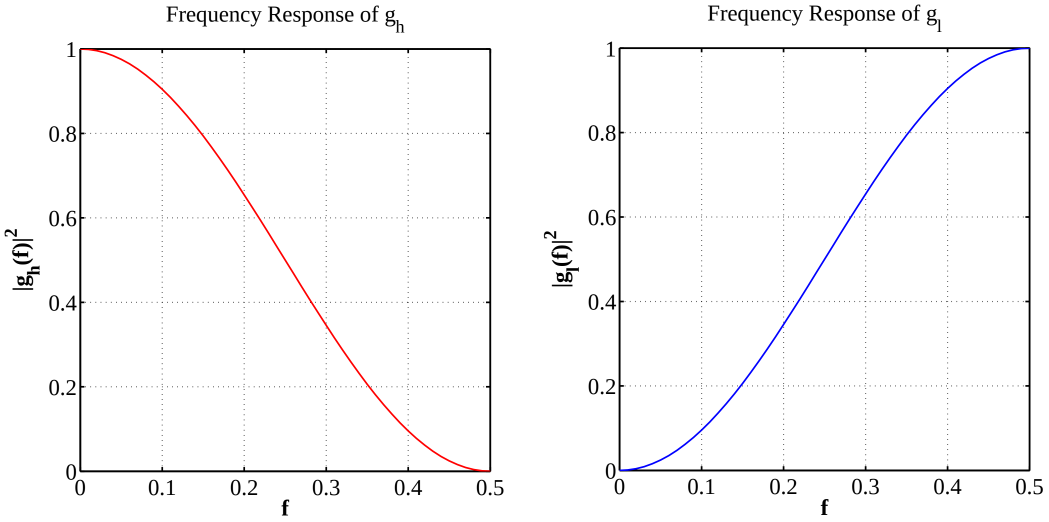


Figure 3.3: Frequency responses of low pass g_l (left) and high pass g_h (right) filters

Where $\mathbf{p_{odd}}$ is the jitter noise power on odd signalling transmission time slots, $\mathbf{p_{even}}$ is the jitter noise power on even signalling transmission time slots and $\rho = \frac{R_{z'z'}(1)}{R_{z'z'}(0)}$.Indeed, the power of noise is split along two transmission proportional to eigenvalues of covariance matrix,

$$\mathbf{R} = \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}$$

Since,the column of Hadamard matrix is actually the eigenvectors for covariance matrix $\mathbf{R}$.In our case of interest where ρ has a significant negative value there will be a strong jitter on even transmission times.furthermore,

$$E[z''_{2k-1}z''_{2k}] = \frac{1}{2}E[(z'_{2k-1} + z'_{2k})(z'_{2k-1} - z'_{2k})] = 0 \tag{3.15}$$

Which indicate the two parallel dimensions are independent in terms of jitter noise.We can now exploit these properties (uneven independent dimensions) to improve system in terms of *BER*.For instance, allocating less rate to the worse dimension or adjusting energy allocations.Some simulation results and *BER* curves are presented in section (3.4) .

3.3 Spectral shaping using higher order Hadamard transforms

Same approach as section (3.2) can be applied to expand modulation points over higher orders Hadamard bases.For higher order transformations there will be more dimensions with unalike strength of jitter noise on each, which gives higher "degrees of freedom" for power and rate allocation. For H_2 expansion :

$$\begin{aligned}\begin{pmatrix} z''_1 & z''_5 & z''_9 & \cdots \\ z''_2 & z''_6 & z''_{10} & \cdots \\ z''_3 & z''_7 & z''_{11} & \cdots \\ z''_4 & z''_8 & z''_{12} & \cdots \end{pmatrix} &= \frac{1}{2}\begin{pmatrix} z'_1+z'_2+z'_3+z'_4 & z'_5+z'_6+z'_7+z'_8 & z'_8+z'_9+z'_{10}+z'_{11} & \cdots \\ z'_1-z'_2+z'_3-z'_4 & z'_5-z'_6+z'_7-z'_8 & z'_8-z'_9+z'_{10}-z'_{11} & \cdots \\ z'_1+z'_2-z'_3-z'_4 & z'_5+z'_6-z'_7-z'_8 & z'_8+z'_9-z'_{10}-z'_{11} & \cdots \\ z'_1-z'_2-z'_3+z'_4 & z'_5-z'_6-z'_7+z'_8 & z'_8-z'_9-z'_{10}+z'_{11} & \cdots \end{pmatrix} \\ &= \frac{1}{2}\begin{pmatrix} (g_a * z')_{(4k-3)} \\ (g_b * z')_{(4k-3)} \\ (g_c * z')_{(4k-3)} \\ (g_d * z')_{(4k-3)} \end{pmatrix} \quad k = 1, 2, ... \end{aligned} \tag{3.16}$$

Hence,this breaks down the channel into 4 unalike dimension in terms of jitter noise power.The frequency response corresponding to filters on each dimensions are :

$$g_a(f) = 1 + e^{j2\pi f} + e^{j4\pi f} + e^{j6\pi f}$$

$$g_b(f) = 1 - e^{j2\pi f} + e^{j4\pi f} - e^{j6\pi f}$$

$$g_c(f) = 1 + e^{j2\pi f} - e^{j4\pi f} - e^{j6\pi f}$$

$$g_d(f) = 1 - e^{j2\pi f} - e^{j4\pi f} + e^{j6\pi f}$$

Hence, with regarding to jitter noise power the order of dimensions is $a \rightarrow c \rightarrow d \rightarrow b$, "a" being the best and "b" being the worst.

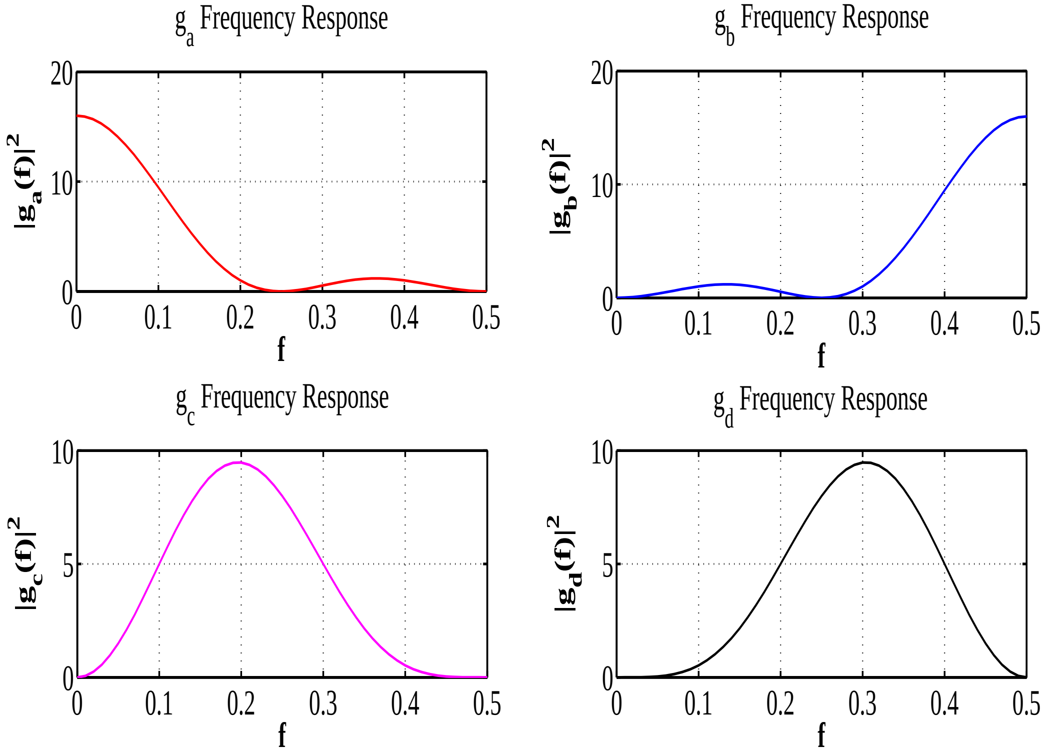


Figure 3.4: Frequency responses of g_a (top left), g_b (top right),g_c (bottom left) and g_d (bottom right) filters

3.4 Simulation results

In this section simulation results of *BER* curves for the proposed schemes is presented.Gray labelling is used to label the constellation points.The baseline is a minimum distance decoder while transmitting 2 bits on each dimension using *4PAM* modulation points.Please recall that we are considering strongly correlated jitter noise which is concentrated around high frequencies.Furthermore, only rate allocations with integer rates is considered.

As it can be seen the *BER* has decreased only in high *SNR* regime.The reason is in low *SNR* the Gaussian noise is the dominant noise which is identical on both transmissions.Thus, any asymmetry in terms of rate and power allocation on dimensions may degrade system performance in terms of *BER*.

3.4.1 Projecting constellation points on H_1, *BER* curve

Figure 3.5 compares *BER* for different *SNR*s in presence of jitter noise.*SNR* is the signal to AWGN power ratio not including jitter noise power.Timing jitter is AR(1) process with variance σ^2=0.01 and $r = 0.99$.The marked dashed curve is the baseline which is basically *4PAM* constellations on each dimension with equally distributed power.The solid line shows the *BER* behaviour after expanding modulation points on H_1 bases, using *8PAM* constellation on better and *BPSK* on worse dimension.

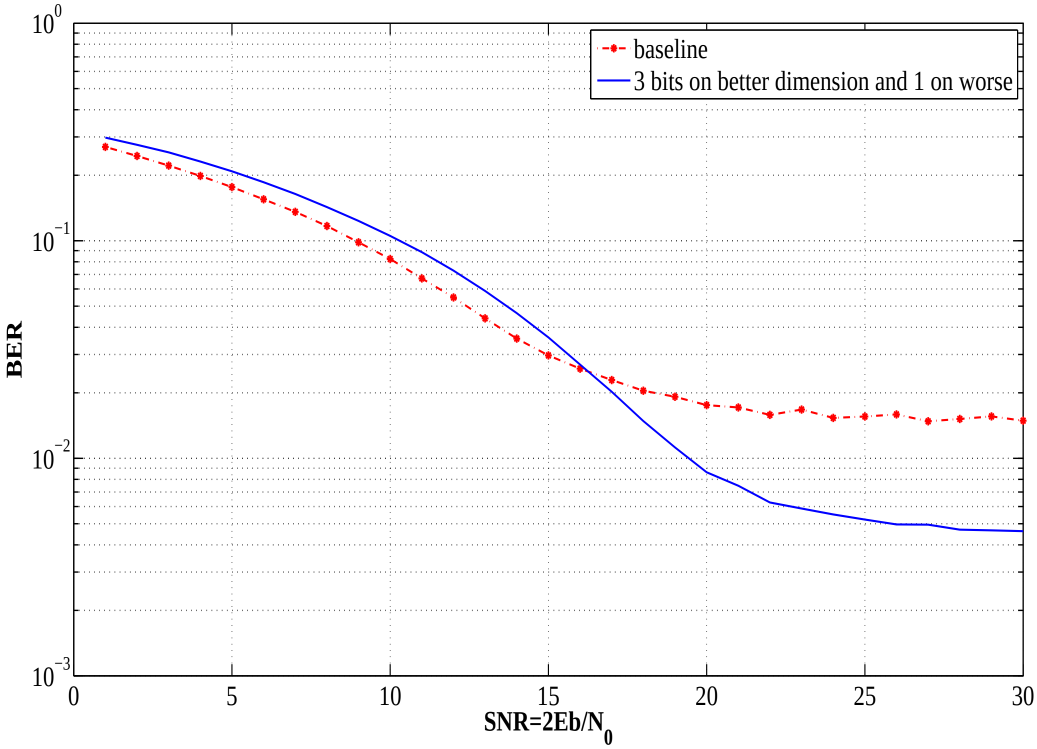


Figure 3.5: *BER* curves for spectral shaping over H_1 ,*8PAM* for better and *BPSK* for worse transmission

Some illustrations on the figure :

- Along with the fact that different number of bits is allocated on different dimensions (first and second transmission which face different noise powers), the power of input signal should also be optimized so as to achieve the lowest *BER*
- For the curve plotted above power is apportioned among dimensions with ratio $\frac{\mathbf{P}_{8PAM}}{\mathbf{P}_{BPSK}} = \frac{1.75}{0.25}$, this has been found through exhaustive search to optimize for high *SNR* regime.
- The cross point of the two curves can be moved to the *SNR* region of interest by adjusting the power between 2 dimensions
- Although the simulation has been done for a fair amount of input data bits still the curves might not seem smooth enough.Since there is a high memory in jitter, if the jitter noise falls in a bad state it is going to stuck in that state for a long time, resulting an un-smooth graph
- To better optimize for *BER* one can consider allocating non integer information bits on 2 dimensions

3.4.2 4 dimensional spectral shaping

For the same timing jitter discussed above the *BER* curve in case of expanding constellation points on H_2 is depicted below.The best dimension carries 3 bits of information *8PAM*, the worse 1 bit *BPSK* and 2 bits *4PAM* for the other 2.Powers are adjusted exhaustively to better the performance in high *SNR* regime where jitter noise is dominant.

Illustrations about the curve :

- Same as before the input signal powers for each dimension is adjusted so as to get the lowest *BER* when jitter noise is stronger comparing to AWGN noise

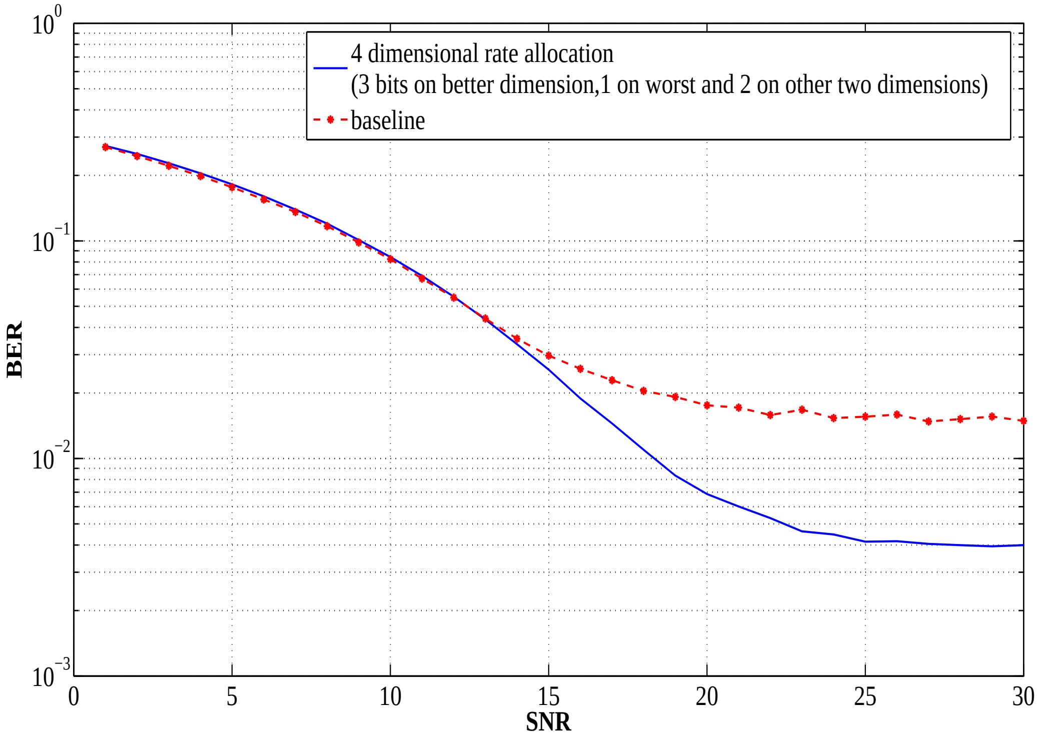


Figure 3.6: BER curves for spectral shaping over H_2 ,

- The signal power on dimension 1 to 4 is apportioned as follows : $(\mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3, \mathbf{P}_4) = (1.75, 0.25, 1.2, 0.8)$ .Which is, better channels have bigger share of the total power.

- As it can be seen, the cross point of two curves is now moved a bit toward a lower SNR, however,this cross point can still be manipulated by adjusting input signal powers among dimensions

- Comparing to expansion over H_1 as expected expansion over H_2 gives us a bit of improvement in terms of BER

- For even more improvement non integer rate allocation can be considered among dimensions

3.4.3 Expansion over H_1 versus expansion over H_2

As just seen the expansion over Hadamard bases improved the BER performance specially in high SNRs.Now the question can be how much gain is obtained by shaping with H_2 instead of H_1.Figure 3.7 answer this question.All the settings for the curves are just as described in preceding subsections.

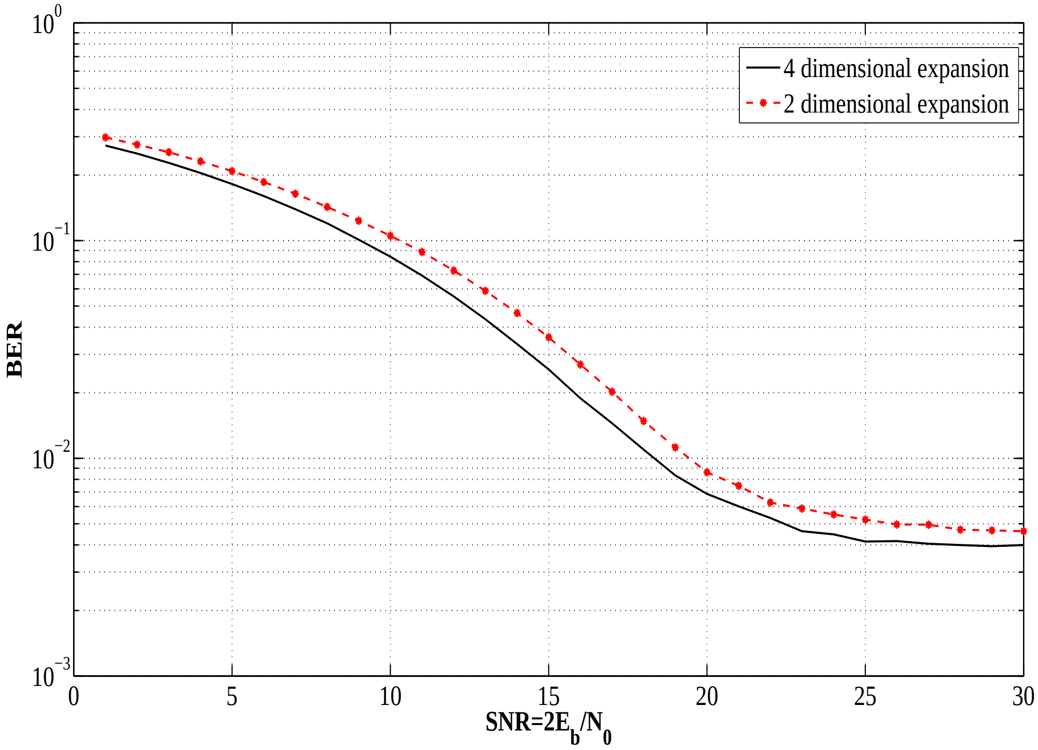


Figure 3.7: Expansion over H_1 vs expansion over H_2,

Chapter 4

Viterbi Decoder

4.1 Theory

In general case the communication channel with timing jitter can be modelled as *ISI* channel with random *ISI* coefficients. For a causal system where only previous symbols in time can contribute in *ISI* terms of current symbol,

$$y_n = \sum_{k=0}^{\infty} a_k(\zeta_{\mathbf{n}}) x_{n-k} \tag{4.1}$$

for FIR (Finite Impulse Response)communication system 4.1 becomes

$$y_n = \sum_{k=0}^{l} a_k(\zeta_{\mathbf{n}}) x_{n-k} \tag{4.2}$$

where $l+1$ is the length of the impulse response.Coefficients a_k are deterministic functions of jitter times $\zeta_{\mathbf{n}}$.If in addition to jitter the data is also corrupted by AWGN noise,

$$y_n = \sum_{k=0}^{l} a_k(\zeta_{\mathbf{n}}) x_{n-k} + w_n \tag{4.3}$$

Our observation of the channel is trajectories of process y_n and we are interested to best estimate the signal path x_n.In order to find best estimate of x_n we first have to clarify what we mean by "best estimation".In the subsequent parts by best estimate of x_n we mean :

$$\text{Choose } (\hat{x}_k, \hat{\zeta}_k)_{k \leq n} \text{ such that } \mathbb{P}(x_k = \hat{x}_k, \zeta_{\mathbf{k}} = \hat{\zeta}_k \ \forall k \leq n) \text{ is maximaized}$$

Next step is how to compute the most probable path estimate. This can be done using a recursive or better to say a dynamic programming algorithm called *Viterbi Algorithm*.VA was first a mean for decoding convolutional codes with an optimum no-sequential algorithm [18].Then it was observed that [14] VA is indeed a solution to dynamic programming.Lets see how we can use VA in our special case of interest.

In order to compute maximum probability pass we have to choose mapping functions $g_1, g_2, ..g_n$ such that

$$(g_1(y_1, ..., y_n), g_2(y_1, ..., y_n), ..., g_n(y_1, ..., y_n)) =$$

$$\underset{(\hat{x}_1,...\hat{x}_n,\hat{\zeta}_1...,\hat{\zeta}_n)}{\operatorname{argmax}} \int \prod_{k=1}^{n} I_0(x_x - \hat{x}_k) f_{x_1,...,x_n,\zeta_1,..,\zeta_n|y_1,...,y_n}(y_1, ...y_n; dx_1, ...dx_n, d\hat{\zeta}_1, ..., d\hat{\zeta}_n) \tag{4.4}$$

where f is conditional probability density function and

$$I_0(a) = \begin{cases} 1 & , a = 0 \\ 0 & , a \neq 0 \end{cases} \tag{4.5}$$

If the input sequence x_n is chosen from a finite set

$$(g_1(y_1, ..., y_n), g_2(y_1, ..., y_n), ..., g_n(y_1, ..., y_n)) =$$

$$\underset{(\hat{x}_1,...\hat{x}_n,\hat{\zeta}_1...,\hat{\zeta}_n)}{\operatorname{argmax}} \sum \prod_{k=1}^{n} I_0(x_x - \hat{x}_k) \mathbb{P}_{x_1,...,x_n,\zeta_1,..,\zeta_n|y_1,...,y_n}(y_1, ...y_n; x_1, ...x_n, \hat{\zeta}_1, ..., \hat{\zeta}_n) \tag{4.6}$$

Lets now examine equation 4.6 in our case.For the sake of simplicity we will first assume that the impulse response length of the channel is only 2.

4.1.1 *Viterbi decoder* for channel with only one *ISI* term

The output of channel in this case is

$$y_n = a_0(\zeta_{\mathbf{n}})x_n + a_1(\zeta_{\mathbf{n}})x_{n-1} + w_n \tag{4.7}$$

Our focus is a first order Gauss-Markov model (AR(1)) for timing jitter process $\zeta_{\mathbf{n}}$.However, these can be extended to higher order Markov processes.Thus, the following random vectors form a Markov chain :

$$(x_n, x_{n-1}, \zeta_{\mathbf{n}}) \longleftrightarrow (x_{n-1}, x_{n-2}, \zeta_{\mathbf{n-1}}) \longleftrightarrow (x_{n-2}, x_{n-3}, \zeta_{\mathbf{n-2}})$$

Consequently y_n would be a Hidden Markov Model.For a reason we will explain later for now assume timing jitter $\zeta_{\mathbf{n}}$ is a dicrete valued process. Applying equation 4.6 and Bayes formula :

$$\begin{gathered}(g_1(y_1, ..., y_n), g_2(y_1, ..., y_n), ..., g_n(y_1, ..., y_n)) = \\ \underset{(\hat{x}_1,...\hat{x}_n,\hat{\zeta}_1...,\hat{\zeta}_n)}{\operatorname{argmax}} \; \mathbb{P}_{x_1,...,x_n,\zeta_1,..,\zeta_n,y_1,...,y_n}(y_1, ...y_n; \hat{x}_1, ..., \hat{x}_n, \hat{\zeta}_1, ..., \hat{\zeta}_n) =\end{gathered} \tag{4.8}$$

$$\underset{(\hat{x}_1,...\hat{x}_n,\hat{\zeta}_1...,\hat{\zeta}_n)}{\operatorname{argmax}} \; \mathbb{P}_{x_1}(\hat{x}_1)...\mathbb{P}_{x_n}(\hat{x}_n)\mathbb{P}_{\zeta_1}(\hat{\zeta}_1).\mathbb{P}_{\zeta_2|\zeta_1}(\hat{\zeta}_2 \mid \hat{\zeta}_1).\mathbb{P}_{\zeta_n|\zeta_{n-1}}(\hat{\zeta}_n \mid \hat{\zeta}_{n-1})f(y_1, ..., y_n \mid \hat{x}_1, \hat{\zeta}_1, ..., \hat{x}_n, \hat{\zeta}_n) \tag{4.9}$$

Where to derive 4.9 we have use the facts that all $x_n s$ are generated i.i.d and independent of all timing jitter $\zeta_{\mathbf{n}}$ and that $\zeta_{\mathbf{n}}$ form a first order Markov process.Furthermore,using independence of AGWN process w_n we know given $(\zeta_{\mathbf{k}}, x_k, x_{k-1})$, all y_k are independent of each other we can simplify 4.9 as

$$\underset{(\hat{x}_1,...\hat{x}_n,\hat{\zeta}_1...,\hat{\zeta}_n)}{\operatorname{argmax}} \; \mathbb{P}_{\zeta_1}(\hat{\zeta}_1).\mathbb{P}_{\zeta_2|\zeta_1}(\hat{\zeta}_2 \mid \hat{\zeta}_1).\mathbb{P}_{\zeta_n|\zeta_{n-1}}(\hat{\zeta}_n \mid \hat{\zeta}_{n-1}).f(y_1 \mid \hat{x}_1, \hat{\zeta}_1).f(y_2 \mid \hat{x}_1, \hat{x}_2, \hat{\zeta}_2)...f(y_n \mid \hat{x}_n, \hat{x}_{n-1}, \hat{\zeta}_n) \tag{4.10}$$

Since terms $\mathbb{P}_{x_k}(\hat{x}_k)$ are all equal and like a constant in maximizing the likelihood, they have been removed from 4.10 . $\log x$ is an increasing function which leads to $argmax_x f(x) = argmax_x \log f(x)$, consequently,

$$(g_1(y_1, ..., y_n), g_2(y_1, ..., y_n), ..., g_n(y_1, ..., y_n)) =$$

$$\underset{(\hat{x}_1,...\hat{x}_n,\hat{\zeta}_1...,\hat{\zeta}_n)}{\operatorname{argmax}} (log\mathbb{P}(\hat{\zeta}_1) + logf(y_1 \mid \hat{x}_1, \hat{\zeta}_1) + \sum_{k=1}^{n} \log \mathbb{P}(\hat{\zeta}_k \mid \hat{\zeta}_{k-1}) + \log f(y_k \mid \hat{x}_k, \hat{x}_{k-1}, \hat{\zeta}_k)) \tag{4.11}$$

term $\log f(y_k \mid \hat{x}_k, \hat{x}_{k-1}, \hat{\zeta}_k)$ in 4.11 is indeed,

$$\log f(y_k \mid \hat{x}_k, \hat{x}_{k-1}, \hat{\zeta}_k) = -\frac{(y_k - a_0(\hat{\zeta}_k)\hat{x}_k - a_1(\hat{\zeta}_k)\hat{x}_{k-1})^2}{2\sigma_w^2} \tag{4.12}$$

Since the conditional density function f is the Gaussian distribution of corresponding AWGN process w_n.Therefore,

$$(g_1(y_1, ..., y_n), g_2(y_1, ..., y_n), ..., g_n(y_1, ..., y_n)) =$$

$$\underset{(\hat{x}_1,...\hat{x}_n,\hat{\zeta}_1...,\hat{\zeta}_n)}{\operatorname{argmax}} (log\mathbb{P}(\hat{\zeta}_1) - \frac{(y_1 - a_0(\hat{\zeta}_1)\hat{x}_1)^2}{2\sigma_w^2} + \sum_{k=1}^{n} \log \mathbb{P}(\hat{\zeta}_k \mid \hat{\zeta}_{k-1}) - \frac{(y_k - a_0(\hat{\zeta}_k)\hat{x}_k - a_1(\hat{\zeta}_k)\hat{x}_{k-1})^2}{2\sigma_w^2}) \tag{4.13}$$

Now lets go back to *Viterbi Algorithm*.VA is basically a dynamic programming algorithm solving optimization problem where our general problem can be broken to smaller sub-problems with same structure.The idea behind *Viterbi* is to define likelihood (or cost) time indexed functions

$$v_l(\hat{x}_1^l, \hat{\zeta}_1^l) = log\mathbb{P}(\hat{\zeta}_1) - \frac{(y_1 - a_0(\hat{\zeta}_1)\hat{x}_1)^2}{2\sigma_w^2} + \sum_{k=1}^{l} \log \mathbb{P}(\hat{\zeta}_k \mid \hat{\zeta}_{k-1}) - \frac{(y_k - a_0(\hat{\zeta}_k)\hat{x}_k - a_1(\hat{\zeta}_k)\hat{x}_{k-1})^2}{2\sigma_w^2} \tag{4.14}$$

Eventually the goal is to maximize the likelihood v_n and finding the corresponding path $(\hat{x}_1^n, \hat{\zeta}_1^n)$.

Viterbi **Recursions**

$$v_l(\hat{x}_1^l, \hat{\zeta}_1^l) = v_{l-1}(\hat{x}_1^{l-1}, \hat{\zeta}_1^{l-1}) + \log \mathbb{P}(\hat{\zeta}_l \mid \hat{\zeta}_{l-1}) - \frac{(y_l - a_0(\hat{\zeta}_l)\hat{x}_l - a_1(\hat{\zeta}_l)\hat{x}_{l-1})^2}{2\sigma_w^2} \tag{4.15}$$

Equation 4.15 is the heart of our *Viterbi* algorithm.Starting from time stamp 1 ,at each time stamp l using the already calculated likelihood $v_{l-1}(\hat{x}_1^{l-1}, \hat{\zeta}_1^{l-1})$ for all possible $\hat{\zeta}_l$ we will find the triples $(\hat{x}_{l-1}, \hat{x}_l, \hat{\zeta}_{l-1})$ that maximizes $v_l(\hat{x}_1^l, \hat{\zeta}_1^l)$.We have to keep track of both the likelihood term $v_l(\hat{x}_1^l, \hat{\zeta}_1^l)$ and the triple $(\hat{x}_{l-1}, \hat{x}_l, \hat{\zeta}_{l-1})$ which lead to that maximum for all possible $\hat{\zeta}_l$ (This is the reason why we considered a quantized version of timing jitter so as to be able to compute likelihood for all possible $\hat{\zeta}_l$ which is only possible if for all time stamp l, $\hat{\zeta}_l$ has a finite cardinality.In fact the running time of the algorithm is scaled with $\Theta(\mid \hat{\zeta} \mid^2)$).

This is done until we get to time n. The procedure continues until the likelihood value v at time n is obtained.Then using a backtrack part of the algorithm (the winner triple saved at each time stamp) its then possible to decode $\hat{x}_n$ corresponding to the most likely path.Algorithm 1 is the pseudo code for this *Viterbi Decoder.Viterbi Algorithm* execution time is linear with n.For faster execution values $a_0(\zeta), a_1(\zeta)$ for all quantized version of ζ have to be computed and saved in a lookup table.Running time of the algorithm also depends on quantization levels for ζ which as mentioned before is $\Theta(\mid \hat{\zeta} \mid^2)$).As the *pdf* for random variable ζ is Gaussian, quantized values of ζ is better to be obtained non-uniformly based on its pdf.In next section we will extend these steps for the case that impulse response of system has longer length.

```
// first time stamp
for $\hat{\zeta}_1 \in$ quantized versions of $\zeta$ do
    $v_1(\hat{\zeta}_1) \leftarrow \max_{\hat{x}_1} (\log \mathbb{P}(\hat{\zeta}_1) - \frac{(y_1 - a_0(\hat{\zeta}_1)\hat{x}_1)^2}{2\sigma_w^2})$;
    $d_1(\hat{\zeta}_1) \leftarrow \operatorname{argmax}_{\hat{x}_1} (\log \mathbb{P}(\hat{\zeta}_1) - \frac{(y_1 - a_0(\hat{\zeta}_1)\hat{x}_1)^2}{2\sigma_w^2})$;
end
// finding likelihoods and corresponding data at each time stamp
for $k = 2, ..., n$ do
    for $\hat{\zeta}_k \in$ quantized versions of $\zeta$ do
        $v_k(\hat{\zeta}_k) \leftarrow \max_{\hat{x}_k, \hat{\zeta}_{k-1}} (v_{k-1}(\hat{\zeta}_{k-1}) + \log \mathbb{P}(\hat{\zeta}_k \mid \hat{\zeta}_{k-1}) - \frac{(y_k - a_0(\hat{\zeta}_k)\hat{x}_k - a_1(\hat{\zeta}_k)d_{k-1}(\hat{\zeta}_{k-1}))^2}{2\sigma_w^2})$ ;
        $(d_k(\hat{\zeta}_k), z_k(\hat{\zeta}_k)) \leftarrow$
        $\operatorname{argmax}_{\hat{x}_k, \hat{\zeta}_{k-1}} (v_{k-1}(\hat{\zeta}_{k-1}) + \log \mathbb{P}(\hat{\zeta}_k \mid \hat{\zeta}_{k-1}) - \frac{(y_k - a_0(\hat{\zeta}_k)\hat{x}_k - a_1(\hat{\zeta}_k)d_{k-1}(\hat{\zeta}_{k-1}))^2}{2\sigma_w^2})$;
    end
end
// back track part to find data $(g_1^n)$ corresponding to most likely path
$f_n \leftarrow \operatorname{argmax}_{\hat{\zeta}_n \in \text{quantized version of } \zeta} v_n(\hat{\zeta}_n)$
for $k$=1,...,$n$ do
    $f_{n-k} \leftarrow z_{n-k+1}(g_{n-k+1})$;
    $g_{n-k} \leftarrow d_{n-k+1}(g_{n-k+1})$;
end
```

Algorithm 1: *Viterbi Algorithm for jitter channel with one ISI term*

4.1.2 Decoder extension to higher order *ISI* terms

For the more general case of finite impulse response channel of 4.3 it is still possible to find the most likely path with same algorithm but with more complicated likelihood terms.For this case we have to modify equation 4.10 as follows :

$$\begin{aligned}\underset{(\hat{x}_1,...\hat{x}_n,\hat{\zeta}_1...,\hat{\zeta}_n)}{\text{argmax}} \; &\{\mathbb{P}_{\zeta_1}(\hat{\zeta}_1).\mathbb{P}_{\zeta_2|\zeta_1}(\hat{\zeta}_2 \mid \hat{\zeta}_1).\mathbb{P}_{\zeta_n|\zeta_{n-1}}(\hat{\zeta}_n \mid \hat{\zeta}_{n-1}).f(y_1 \mid \hat{x}_1, \hat{\zeta}_1) \\ &.f(y_2 \mid \hat{x}_1, \hat{x}_2, \hat{\zeta}_2)...f(y_k \mid \hat{x}_k, ..., \hat{x}_{k-l}, \hat{\zeta}_n)...f(y_n \mid \hat{x}_n, ..., \hat{x}_{n-l}, \hat{\zeta}_n)\}\end{aligned} \tag{4.16}$$

Since each channel output y_k depends on l previous channels input which is x_{k-l}^{k}.For the conditional probability distribution f ,

$$\log f(y_k \mid \hat{x}_{k-l}^{k}, \hat{\zeta}_k) = -\frac{(y_k - \sum_{i=0}^{l} a_i(\hat{\zeta}_k)\hat{x}_{k-i})^2}{2\sigma_w^2} \tag{4.17}$$

Therefore,

$$\begin{gathered}(g_1(y_1, ..., y_n), g_2(y_1, ..., y_n), ..., g_n(y_1, ..., y_n)) = \\ \underset{(\hat{x}_1,...\hat{x}_n,\hat{\zeta}_1...,\hat{\zeta}_n)}{\text{argmax}} \; (log\mathbb{P}(\hat{\zeta}_1) - \frac{(y_1 - a_0(\hat{\zeta}_1)\hat{x}_1)^2}{2\sigma_w^2} + \sum_{k=1}^{n} \log \mathbb{P}(\hat{\zeta}_k \mid \hat{\zeta}_{k-1}) - \frac{(y_k - \sum_{i=0}^{k-1+(l-k+1)^+} a_i(\hat{\zeta}_k)\hat{x}_{k-i})^2}{2\sigma_w^2})\end{gathered} \tag{4.18}$$

where

$$x^+ = \begin{cases} x & x > 0 \\ 0 & \text{otherwise} \end{cases}$$

Updating these likelihoods, same *Viterbi Algorithm* can be used to find the most likely data path.The updated algorithm is presented at Algorithm2 .

```
// first time stamp
for $\hat{\zeta}_1 \in$ quantized versions of $\zeta$ do
    $v_1(\hat{\zeta}_1) \leftarrow \max_{\hat{x}_1} (\log \mathbb{P}(\hat{\zeta}_1) - \frac{(y_1 - a_0(\hat{\zeta}_1)\hat{x}_1)^2}{2\sigma_w^2});$
    $d_1(\hat{\zeta}_1) \leftarrow \operatorname{argmax}_{\hat{x}_1} (\log \mathbb{P}(\hat{\zeta}_1) - \frac{(y_1 - a_0(\hat{\zeta}_1)\hat{x}_1)^2}{2\sigma_w^2});$
end
// finding likelihoods and corresponding data at each time stamp
for $k = 2, ..., n$ do
    for $\hat{\zeta}_k \in$ quantized versions of $\zeta$ do
        $v_k(\hat{\zeta}_k) \leftarrow \max_{\hat{x}_k, \hat{\zeta}_{k-1}} (v_{k-1}(\hat{\zeta}_{k-1}) + \log \mathbb{P}(\hat{\zeta}_k \mid \hat{\zeta}_{k-1}) - \frac{(y_k - \sum_{i=0}^{k-1+(l-k+1)^+} a_i(\hat{\zeta}_k)\hat{x}_{k-i})^2}{2\sigma_w^2})$ ;
        $(d_k(\hat{\zeta}_k), z_k(\hat{\zeta}_k)) \leftarrow$
        $\operatorname{argmax}_{\hat{x}_k, \hat{\zeta}_{k-1}} (v_{k-1}(\hat{\zeta}_{k-1}) + \log \mathbb{P}(\hat{\zeta}_k \mid \hat{\zeta}_{k-1}) - \frac{(y_k - \sum_{i=0}^{k-1+(l-k+1)^+} a_i(\hat{\zeta}_k)\hat{x}_{k-i})^2}{2\sigma_w^2});$
    end
end
// back track part to find data $(g_1^n)$ corresponding to most likely path
$f_n \leftarrow \operatorname{argmax}_{\hat{\zeta}_n \in \text{quantized version of } \zeta} v_n(\hat{\zeta}_n)$
for $k$=1,...,$n$ do
    $f_{n-k} \leftarrow z_{n-k+1}(g_{n-k+1});$
    $g_{n-k} \leftarrow d_{n-k+1}(g_{n-k+1});$
end
```

Algorithm 2: *Viterbi Algorithm for jitter channel (Extended version)*

4.2 Simulation Results

Simulation results evaluating *Viterbi Decoder* performance for the AWGN channel corrupted with sampling time jitter is presented in this section.Just like before a Gauss-Markov model with high memory is considered for timing jitter.Recall that for the channel of interest we have

$$y_n = \sum_{k=0}^{l} a_k(\zeta_{\mathbf{n}}) x_{n-k}$$

The *ISI* terms due to jitter in fact depends on coefficient functions $a_k(.)$ which indeed are the pulses used to transmit data over the channel.As before these pulses assumed to be *sinc* functions.However,one can also consider other pulses as *raised cosine*.In terms of *ISI* , *sinc* functions are the extreme case since its decay in time is slower than any *raised cosine* function of same energy.For *sinc* functions

$$a_k(t) = sinc(k+t) = \frac{sin(\pi(k+t))}{\pi(k+t)} \tag{4.19}$$

The *Viterbi Decoder* design depends on the levels of quantized versions of time jitter ζ.Increasing the quantization levels might improve the decoder approximation of *ISI* terms and hence improve in terms of *BER*, however, since the running time of the algorithm is scaled with the square of the number of jitter states, the complexity of the decoder is increased with number of quantization levels.

4.2.1 Considering 4 states for timing jitter

First of all,using MATLAB software the timing jitter which indeed is a first order Gauss-Markov process is quantized into 4 states.The quantization can be done several ways including uniform and non-uniform(Lloyd quantization).Non-uniform quantization is in fact a clustering algorithm to minimize the expected value of square of quantization error.Then,empirical stationary and transition probabilities of states is obtained to be used in likelihood terms of each branch. The *BER* curve which compares *Viterbi Decoder* and a minimum likelihood decoder is depicted below.

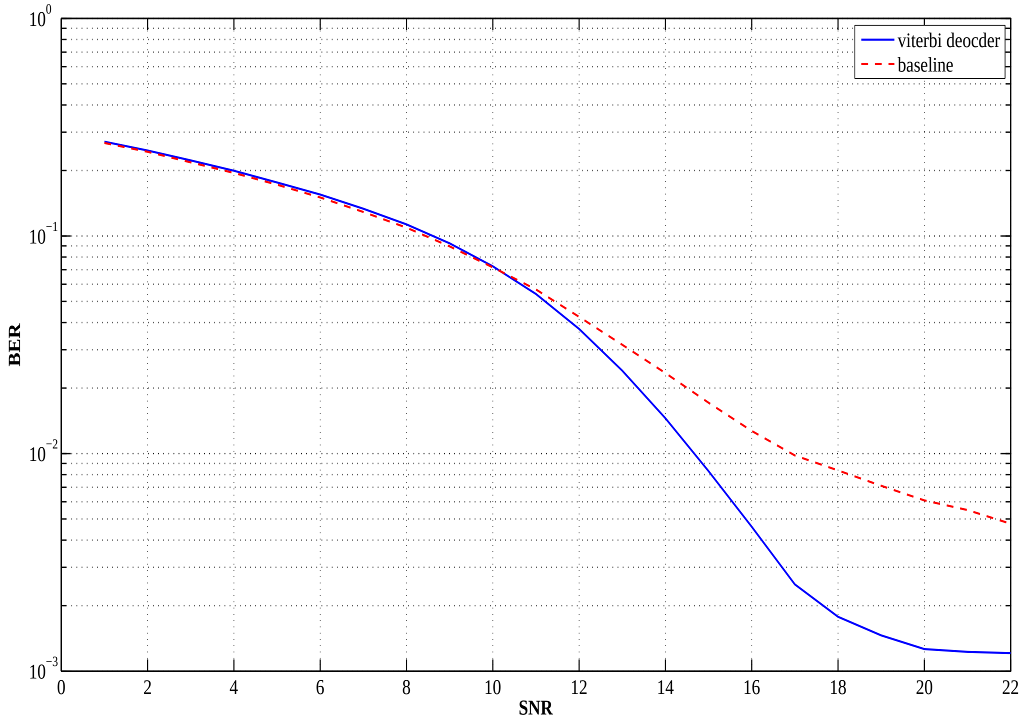


Figure 4.1: Viterbi decoder with 4 jitter states versus minimum distance decoder

Illustrations on figure 4.1

- **SNR** is the signal to Additive Gaussian noise power ratio not including jitter noise
- The time jitter process is a Gauss-Markov process with variance $\sigma^2 = 0.01$ and correlation factor $r = 0.99$
- Channel impulse response is considered to be 6 which means only 5 previous symbols would participate in *ISI* terms.

4.2.2 Considering 8 states for timing jitter

In this section we have simulated the *Viterbi Decoder* with 8 states for jitter channel to see how much doubling the states effect on *BER* curve.

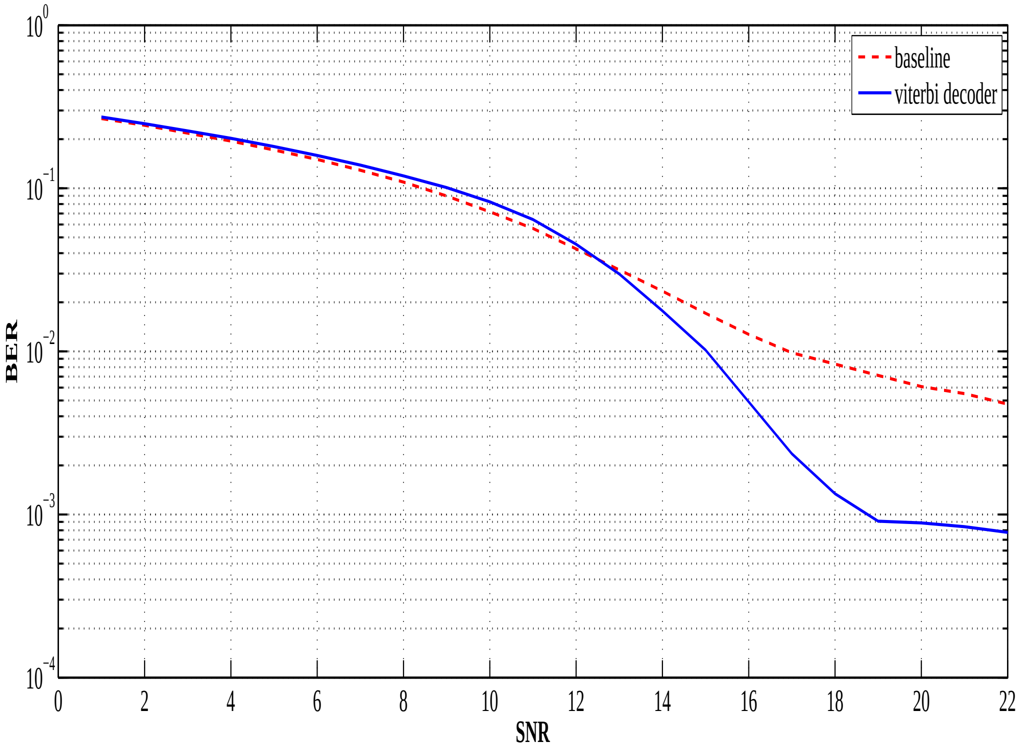


Figure 4.2: Viterbi decoder with 8 jitter states versus minimum distance decoder

Compared to decoder with 4 states for timing jitter this decoder result in a bit better performance in terms of *BER*.Although,since decoding time is doubled,this might not be a better choice.

Chapter 5

Conclusion

In communication systems where sampling time jitter becomes a bottleneck for system performance it is of a great importance to be able to characterize the effects of this jitter on system performance;Either in terms of *BER* or amplitude error or any other performance measure of interest.This was the main goal of this thesis.In chapter 2 the amplitude error due to sampling time jitter was characterized.Then In next chapters the communication system is designed such that the performance of communication system in terms of *BER* measure is increased.

Lots of improvements is achieved using suggested schemes and decoders.That being said,there is still more room for further improvements.The spectral shaping method can be further improved using better transformations as Hadamard transformation is not necessarily optimal.Furthermore, other non-integer rate allocations can better the system performance.

For the *Viterbi* decoder exploiting the cross-correlation factors with input signal in likelihood terms might result in better performance.Moreover,similar algorithms like *BCJR* algorithm can be examined.*BCJR* algorithm would find a-posteriori probability of states at each time rather than maximum likelihood path(*Viterbi*).Another interesting fact about the *BCJR* algorithm is that it will find these a-posteriori based on all the observed output up to last time instant.In other words it will exploit the future memory for decoding as well.

Furthermore,the *Viterbi* algorithm can be designed to speed up the calculations for finding the likelihood path.For example, parallel processing based *Viterbi* architecture.

References

[1] R. Ahlswede and J. Wolfowitz. Channels without synchronization. *Adv. Appl. Probabil*, (3):383–403, 1971.

[2] Venkat Chandar Aslan Tchamkerten and Gregory Wornell. Communication under strong asynchronism. *IEEE TRANSACTION ON INFORMATION THEORY*, 55(10), OCTOBER 2009.

[3] Constant Paul Marie Jozef Baggen. *An information theoritic approach to timing jitter.* PhD thesis, UNIVERSITY OF CALIFORNIA,SAN DIEGO, 1993.

[4] A.V. Balakrishnan. On the problem of time jitter in sampling. *IRE*, April 1962.

[5] Thomas M. Cover and Joy A.Thomas. *Elements of Information Theory.* Wiley, 1991.

[6] M. C. Davey and D. J. C. MacKay. Reliable communication over channels with insertions, deletions and substitutions. *IEEE TRANSACTIONS ON INFORMATION THEORY*, 47(2):687–698, February 2001.

[7] S.N. Diggavi and M. Grossglauser. Bounds on the capacity of deletion channels. *in Proc. IEEE Int. Symp. Inf. Theory*, page 421, 2002.

[8] R.L. Dobrushin. Shannon's theorems for channels with synchronization errors. *Problemy Peredachi Informatsii*, 3(4), 1967.

[9] D. Fertonani and T. M. Duman. Novel bounds on the capacity of binary channels with deletions and substitutions. *in Proc. IEEE Int. Symp. Inf. Theory*, page 25522556, 2009.

[10] D. Fertonani and T. M. Duman. Upper bounding the deletion channel capacity by auxiliary memoryless channels. *in Proc. IEEE Int. Conf. Commun*, pages 1–5, 2009.

[11] Vahid Tarokh Hugues Mercier and Fabrice Labeau. Bounds on the capacity of discrete memoryless channels corrupted by synchronization and substitution errors. *IEEE TRANSACTIONS ON INFORMATION THEORY*, 58(7), 2012.

[12] Tektronix Inc. Understanding and characterizing timing jitter. http://www.to-way.com/tf/tekjit.pdf.

[13] H. Nyquist. Certain factors affecting telegraph speed. *Bell System Technical Journal*, 1924.

[14] J. K. Omura. On the viterbi decoding algorithm. *IEEE Transactions on Information Theory*, 15(1):177–179, January 1969.

[15] Claude E. Shannon. Communication in the presence of noise. *Proceedings of the IRE*, 1949.

[16] Aslan Tchamkerten Venkat Chandar and David Tse. Asynchronous capacity per unit cost. *IEEE TRANSACTION ON INFORMATION THEORY*, 55(10), JUNE 2010.

[17] Aslan Tchamkerten Venkat Chandar and Gregory Wornell. Optimal sequential frame synchronization. *IEEE TRANSACTION ON INFORMATION THEORY*, 54(8), AUGUST 2008.

[18] A. J. Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, 13(2):260–269, April 1967.

[19] R. Motwani W. Zeng, J. Tokas and A. Kavcic. Bounds on mutual information rates of noisy channels with timing errors. *in Proc. IEEE Int. Symp. Inf. Theory*, pages 709–713, 2005.

[20] Wikipedia. Jitter. `http://en.wikipedia.org/wiki/Jitter`.