Addis Ababa University

Addis Ababa Institute of Technology

Department of Electrical and Computer Engineering

A thesis submitted to the Addis Ababa Institute of Technology, School of Graduate Studies, Addis Ababa University

In partial fulfillment of the requirement for the degree of MASTER OF SCIENCE IN ELECTRICAL ENGINEERING (COMMUNICATION ENGINEERING)

By: Senay Semu

Advisor: Dr.Ing. Hailu Ayele

Declaration

I, the undersigned, declare that this thesis work is my original work, has not been presented for a degree in this or any other universities, and all sources of materials used for the thesis work have been fully acknowledged.

Senay Semu ____________________________

Name Signature

____

Date of submission

This thesis has been submitted for examination with my approval as a university advisor.

1. Dr.Ing. Hailu Ayele ____________________________
   Advisor Signature

ADDIS ABABA UNIVERSITY
SCHOOL OF GRADUATE STUDIES

PERFORMANCE ANALYSIS OF JOINT SOURCE CHANNEL CODING IN MIMO SYSTEMS

BY

SENAY SEMU TADESSE

ADDIS ABABA INSTITUTE OF TECHNOLOGY

APPROVAL BY BOARD OF EXAMINERS

Dr. Getahun Mukuria	______________________
Chairman, Dept of Graduate Committee	Signature
Dr. Hailu Ayele	_______________________
Advisor	Signature
_________________________	_______________________
Internal Examiner	Signature
_________________________	________________________
External Examiner	Signature

Acknowledgement

First and foremost thank you God for your help and patience throughout my life. I would like to express my deep and sincere gratitude to my great advisor Dr.Ing. Hailu Ayele, for the wholehearted guidance, continuous support, careful, detailed and constructive comments you gave on my thesis and for the significant contribution throughout this work. I have learned from you things that has not only improved the quality of this work, but also proved to be of great value for my future professional efforts.

I would also like to thank my parents, my brothers, my sisters and my dear friends for their everlasting love, patience and encouragement, without which I would not have succeeded.

Abbreviation

AC	Alternating Current
API	A-posteriori Information
AWGN	Additive White Gaussian Noise
BPSK	Binary phase shift Keying
CC	Centroid Condition
CELP	Codebook-excited Linear prediction
COVQ	Channel Optimized Vector Quantization
DC	Direct Current
DCT	Discrete Cosine Transform
DMC	Discrete Memoryless Channel
DRI	Decoder Reliability Information
FLC	Fixed Length Coding
JPEG	Joint Photographic Experts Group
JSCC	Joint Source Channel Coding
JSCD	Joint source Channel Decoding
LBG	Linde, Buzo and Gray
MIMO	Multiple Input Multiple Output
MPEG	Motion Picture Editors Guild
NNC	Nearest Neighbourhood Condition
OSTBC	Orthogonal Space Time Block Coding
SAI	Source A-priori Information
SDR	Signal to Distortion ratio
SSI	Source Significant information
VLC	Variable Length Coding
VQs	Vector Quantizers

List of Figures

List of Tables

Abstract

Joint source-channel coding/decoding techniques in multimedia communications have become a state-of-the-art and one of the challenging research subjects in the spatial communication area. They have great application prospective and deep impact in various manned space flights, satellite missions, mobile radio communications, and deep-space explorations. In the last few years, there have been influential achievements in Joint Source-channel coding/decoding studies. In this thesis one of the Joint Source Channel Coding (JSCC) technique, called Channel Optimized Vector Quantizer (COVQ) is used for encoding. The decoding is done by mapping the index of regions to the appropriate codebook vector. By doing the source coding and the channel coding optimized to each other some redundant information in the source will be left for the channel coding rather than removing it all.

The performance is evaluated over a multi-antenna communication system with orthogonal space-time block coding through Bit Error Rate (BER) and Signal-to-Distortion ratio (SDR). The results show that the Joint Source channel Coding scheme outperforms the Tandem System (Shannon's Separate coding theorem) in both performance measures (BER and SDR). Even if the main concern of the work is to apply the COVQ at the encoder side and see its performance on a Multiple Input Multiple Output (MIMO) channel, vector quantization source encoding and decoding concatenated with convolutional channel encoding and decoding methods are dealt for the purpose of comparison. The results of this comparison show that the JSCC scheme has better performance than what??. In addition, both Tandem system and JSCC scheme performance with respect to ~~the Bit Error Rate~~ (BER) and ~~Signal-to-Distortion Ratio~~ (SDR) are investigated on a Single Output Single Input channel. And it is observed that the JSCC scheme has a better performance than what??.

Keywords: Joint Source Channel Coding; Channel Optimized Vector Quantization

Contents

Chapter 1

1 Introduction

1.1 Background

The need for modern communication system which, is faster and capable of sending as much information as required, is becoming a day to day challenge for designers. System designers need to look for methods to decrease distortion and the delay of the system at an acceptable cost/complexity. One of the methods that are being studied today is the Joint Source Channel Coding, JSCC scheme, which makes use of the channel conditions to design the source encoder.

Shannon has established the basis for communication system which is based on the separate coding of the source which aims at removing the redundancy and channel coding to some redundancy to protect it from errors. Shannon's separation theorem is stated as follows: if the minimum achievable source coding rate of a given source is strictly below the capacity of the channel, then the source can be reliably transmitted through the channel by appropriate encoding and decoding operations.

According to Shannon's separation theorem arbitrary reliability can be achieved by performing the source coding independent of the channel followed by channel coding whenever the source entropy is less than the channel capacity for arbitrarily long block lengths [1]. But Shannon's separation theorem does not take into consideration constraints on system complexity and delay while real world communication systems are constrained by delay and complexity.

Separation theorem is known to hold true for sources and channels with memory, however, it fails under certain condition when the source and the channel are non stationary [2], as well as in multimedia networks. The third point is that there are channels for which the separation principle does not hold. An example of such channels is the multiuser channel whose capacity depends on the length of the source code [5].

Furthermore, in many wireless communication situations involving non-stationary sources/channels, the separation theorem may not hold. As a result of all these limitations on the tandem system, studying joint source-channel coding has attracted much interest.

In situations where such constraints occur Shannon's separate source channel coding theorem is observed to have inferior performance compared with systems which carry out source and channel coding jointly (Joint Source Channel Coding/Decodig). This is studied for example in [3] where joint source channel coding was shown to outperform Shannon's separate coding for systems having delay or complexity below a certain threshold.

There are several ways of jointly coding/decoding source and channel. In this thesis we have considered a method related to source quantization. This method of source quantization will be optimized depending on channel transition probabilities. The performance of this JSCC method is to be evaluated on a wireless Rayleigh fading MIMO channel.

1.2 Statement of the Problem

Unlike system in which the input is a bit-stream, in many applications the problem at hand is to transmit an analog source, such as the speech or image signals, over a digital communication system. In such cases, it is required to quantize the source; i.e., to approximate it with a source which has a finite alphabet. The performance of a quantizer is expressed by its signal-to-distortion ratio (SDR). Therefore, when an analog source is transmitted, it is reasonable to choose to minimize the end-to-end distortion of the system as the design goal, as opposed to approaches which opt to minimize the system bit, symbol, codeword, or frame error rate. We measure the inaccuracy by a distortion measure, such as the squared error between the original and the decoded sequences.

Let x and $\hat{x}$ be the original and reconstructed source vectors respectively, i.e, the system input and output. We are interested in this case to transmit analog sources over communication links with MIMO. The objective is to minimize the mean-square error under constraints on the average transmit power and channel bandwidth. The bandwidth constraint can simply be met by fixing the quantization rate and dimension, the modulation scheme, and the constellation size.

$$D = E\{||x - \hat{x}||^2\} \quad \ldots\ldots (1.1)$$

is the mean distortion.

A vector quantizer maps k-dimensional vectors in the vector space R^k (real numbers with dimension k) into a finite set of vectors $Y = \{y_i: i = 1, 2, ..., N\}$. Each vector y_i is called a code vector or a codeword. And the set of all the codewords is called a codebook. Associated with each codeword, y_i, is a nearest neighbor region called Voronoi region, and it is defined by:

$$V_i = \{x \in R^k: ||x - y_i|| \le \|x - y_j\|, for\ all\ i \ne j\} \quad \ldots\ldots (1.2)$$

$x, y_i, y_j \epsilon R^k$, which in our case stand for $\hat{x}$.

The set of Voronoi regions partition the entire space R^k such that:

$$\bigcup_{i=1}^{N} V_i = R^k \quad \text{...} (1.3)$$

$$\bigcap_{i=1}^{N} V_i = \emptyset \quad \text{...} (1.4)$$

Where V_i is a Voronoi region having dimension k.

The above vector quantization process does not take into account the channel transition probabilities. In the channel optimized vector quantization the channel state conditions will be used to adjust the vector quantization.

Distortion in Channel Optimized Vector Quantization is calculated as follows:

$$D_{COVQ} = \frac{1}{k}\sum_{i}\int_{s_i} p(x)\sum_{j} P_{Y|X}(j|i)d(x,C_j)dx$$

$$D_{COVQ} = \frac{1}{k}\sum_{i}\int_{s_i} p(x)\sum_{j} P_{Y|X}(j|i)\left|x - C_j\right|^2 dx$$

$$D_{COVQ} = \frac{1}{k}\sum_i \sum_j P_{Y|X}(j|i)\int_{s_i} p(x)\left|x - C_j\right|^2 dx \quad \text{..........................} (1.5)$$

Where C_j stands for a codeword in the codebook C and $d(x, C_j) = E\{|x - C_j|^2\}$

Where p(x) is the K dimensional density of the source and $P_{Y|X}(j|i)$is the probability of receiving *'j'* when *'i'* is sent.

Improving the VQ under the conditions of probabilistic index perturbation between the input and output of the channel is the subject of joint source-channel coding. The resulting VQ is called a channel optimized vector quantizer (COVQ). The basic idea of the COVQ is to design a VQ by incorporating the channel conditions into the design algorithm, trading off the quantization and channel noise in order to minimize the end-to-end distortion. From Equation 1.5 it can be seen that the mean of the distortion is obtained by averaging the distortion for all codewords $\sum_j P_{Y/X}(j|i)d(x,C_j)dx$ in the codebook and again averaging this value by the possible source vectors probability distribution, $\sum_i \int_{s_i} p(x)\sum_j P_{Y|X}(j|i)d(x,C_j)dx$.

This encoding technique is used with a MIMO channel as transmission medium. The MIMO system as it is already established by several researchers has the capability of increasing the capacity and the reliability of the wireless channel. [6]In particular, with channel knowledge at the receiver, a data rate increase proportional to the minimum of number transmit and receive antennas can be obtained by multiplexing data streams across the parallel channels associated with the channel gain matrix. Alternatively, multiple antennas enable transmit and/or receive diversity which decreases the probability of error, which in our case is given the prior attention.

In general, in this thesis a way of minimizing the distortion on analog signals will be designed by considering Joint Source Channel Coding for Channel Optimized Vector Quantization over a MIMO channel.

1.3 Objectives of the Study

The main objective of this thesis is to study the performance of Joint Source Channel Coding on MIMO communication Systems. This is done designing the system and simulating it on MATLAB software. Specifically,

- To understand the failures of Shannon's separate coding theorem.
- Exploring various aspects of modern communication systems (MIMO, OSTBC …) with respect to performance variation to be measured by distortion level and bit error rate.
- Studying the available methods of joint source channel coding methods.
- Detailed study and design of one of joint source channel coding method (Channel Optimized Vector Quantization).
- To study the performance of the COVQ JSCC method on various channels through simulation.
- Compare the performance difference of COVQ JSCC system with tandem system.

This objectives are met as it can be seen from the discussions found in the upcoming Chapters and the results put in the fourth Chapter.

1.4 Relevance of the Thesis Work

Based on separation theory, source and channel coders are usually implemented sequentially and independently. However, practical communication systems are constrained by complexity and latency. Oftentimes, hypothesis of the Shannon separation theory does not hold. For instance, image or video coder cannot usually achieve optimum compression performance. There is always residual redundancy in the output bitstream. In addition, the error (channel transmission error include random error, burst error and packet loss in radio channel) correction capabilities of block codes and convolutional codes are also limited; they cannot correct all errors in noisy channels. It has also been noted that the separation coding principle does not hold even theoretically in some practical communication systems [2]. Noting the limitation of the Shannon separation theory in practical applications, many scholars have focused their attention on the study of joint source-channel coding/decoding techniques. Through jointly optimizing source and channel parameters, they have achieved a great number of important results for the optimal transmission performance [8], [4]…etc.

In this thesis one of joint source channel coding schemes, Channel Optimized Vector Quantization (COVQ), will be used for encoding and decoding to minimize overall distortion on the system and to reduce the bit error rate. To optimize the vector quantizer with respect to the channel transition probability, index assignment optimization method will be implemented. Index assignment optimization determines the index of the code words in the codebook that minimizes the distortion. The indices so obtained are the ones that are to be transmitted.

1.5 Outline of the Thesis Work

The thesis is organized in five Chapters. The first Chapter is meant to give a brief introduction to the work on this thesis. In this Chapter the back ground on the basic communication system and the motivation behind the work in this paper is presented briefly. The problem is identified and the relevance of the solution proposed has also been discussed.

The second Chapter presents a review of related papers to the thesis work. In this Chapter details of basic communication system with its constraints will be discussed. Then based on the constraints, standard solutions like MIMO systems is to be presented in the next section. Then the available methods of joint source channel coding/decoding techniques like: - Joint Source Channel Coding, Joint Source Channel Decoding (JSCD) and Variable-length JSCC with variable-length JSCD is presented.

In Chapter three the system model is discussed in detail. The wireless channel model and its modification, the MIMO channel, analog source coding method (vector quantization) and its optimization with respect to the channel transition probabilities are the main topics in the Chapter. In addition orthogonal space time block coding combiner is also discussed. Then the transition probabilities of MIMO channel and the implementation algorithm for simulation in MATLAB will be presented.

In Chapter four the results of simulation are presented with their discussion and implications. Finally recommendations and conclusions are presented in Chapter five.

Chapter 2

2 Literature Review

This Chapter will review the basic communication system model, and the improvements made to it, like use of multiple receive and transmit antennas and modifications with respect to coding and decoding. The MIMO channel and various methods of Joint Source Channel Coding schemes are the main topics.

2.1 Basic Communication System Model

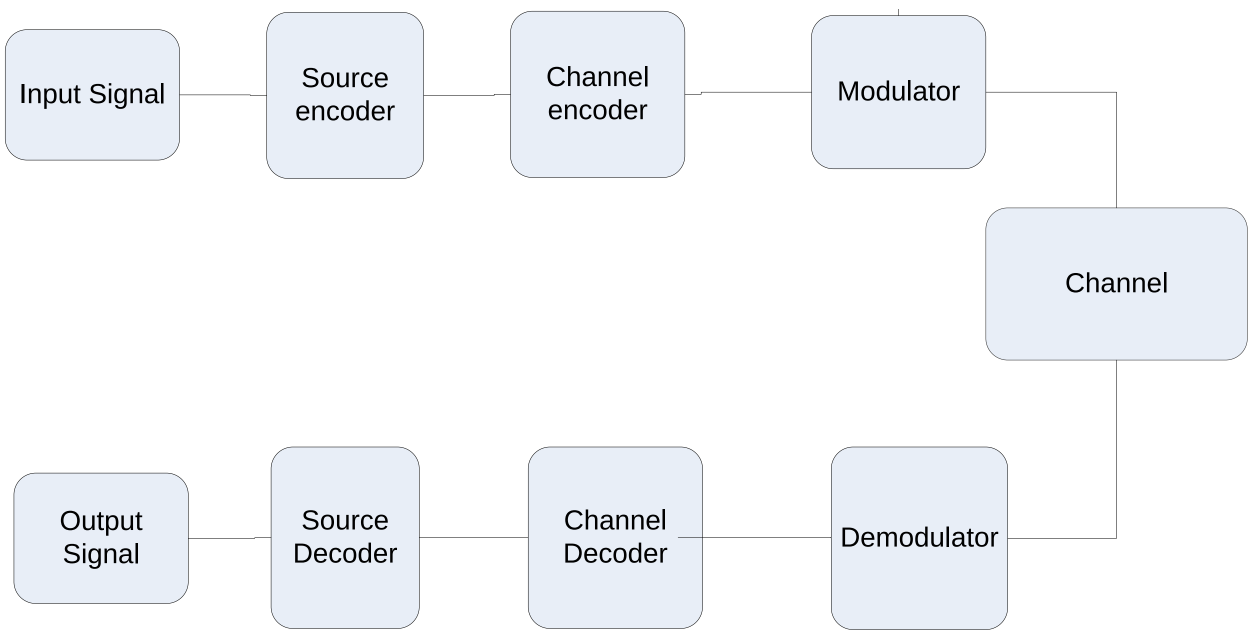

Figure 2.1 Communication System Model

Figure 2.1 illustrates the basic components of a digital communication system in their functionality order. The input can be either digital signal that is discrete in time and has finite number of output characters or an analog signal, such as audio, image or video signal [4]. In this basic communication system the source encoder converts the input signal, either analog or digital, into a sequence of binary digits efficiently (removing redundancy). Then the channel encoder adds some redundancy in a controlled way so as to reduce the effect of the errors encountered in the communication channel on the receiver side. So the added redundancy enhances the reliability of the system. Then the binary sequence from the channel encoder will be modulated by the modulator to change the bits in to electrical signal which is capable to be transmitted through any communication channel. The communication channel is the physical medium that is used to transmit the signal from the transmitter to the receiver antenna. Then at the receiving end the digital demodulator will reconstruct the binary bits which are sent from the channel encoder to the modulator from the waveform obtained by the receiving antenna. Then the channel decoder will try to restructure original information bits by using the redundancy added and the codebook used by the encoder. Finally the source decoder will change the binary bits into the original signal digital or analog. Distortion by the source encoder and source decoder and channel decoding errors might cause the received signal to be only an approximation to the original signal.

2.2 MIMO Channel

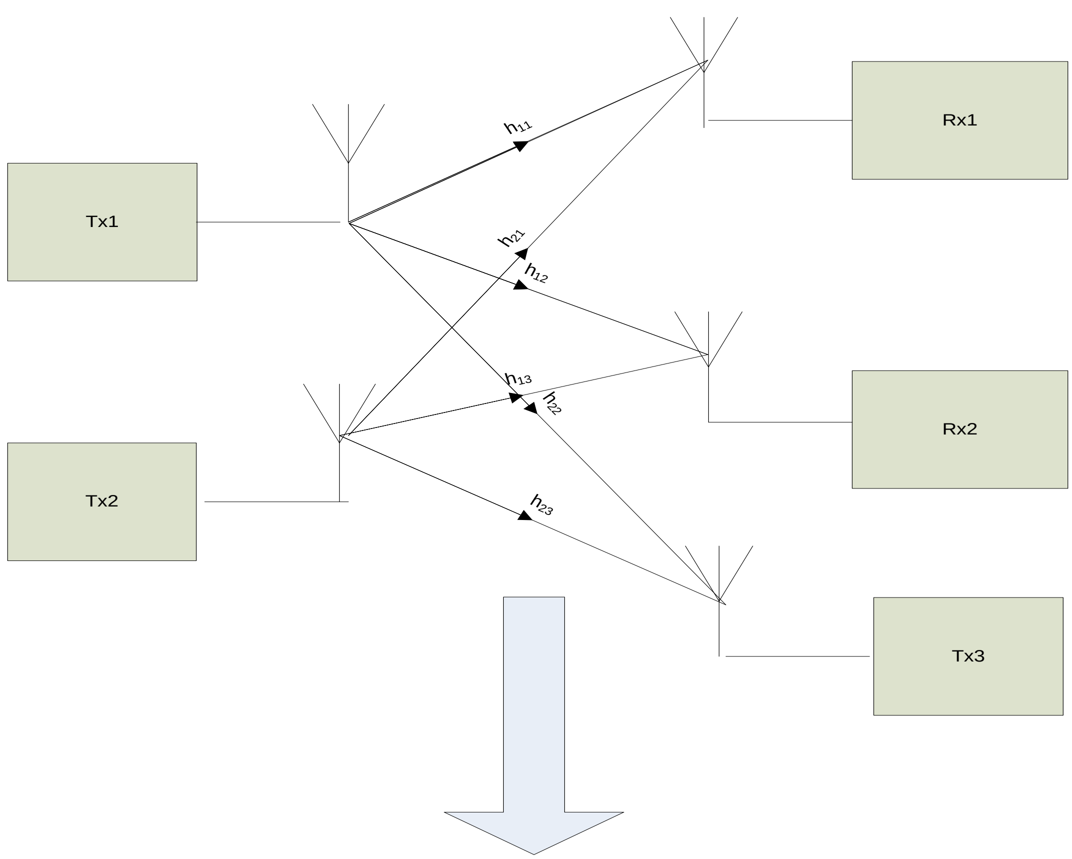

Figure 2.2 MIMO Channel

The channel in which the system model is to be applied on is a wireless Rayleigh fading MIMO channel. The number of antennas at the receiver and transmitter side is three and two respectively. For two transmitter and three receiver antennas, as shown in Figure 2.2, the channel gain matrix is given by (Compare equation (2.1) with Fig. 2.2)

$$H = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \\ h_{31} & h_{32} \end{bmatrix} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2.1)$$

Using MIMO channel instead of a SISO channel increases the capacity and the reliability of the channel. The point to point channels are constrained in bandwidth and reliability level. And since the propagation medium varies with time, the fading is time variant and owing to the Rayleigh distribution of the received amplitude, the channel gain can sometimes be so small that the channel becomes useless. So to increase the bandwidth and reliability level multiple transmitter antennas and receiver antennas are being considered. In this thesis this scheme is used to investigate the performance of Joint source-channel coding technique (specifically Channel Optimized Vector Quantization). Multiple antennas at the receiver or/and at the transmitter can be used for error combating (diversity), which is one way to mitigate this problem, and it amounts to transmitting the same information over multiple channels which fade independent of each other or receive the same information through several receiving antennas and combine them in appropriate manner. Depending on antenna number, the diversity might be obtained through transmit diversity where multiple transmitter antennas are used to transmit the same data and/or through receive diversity where multiple receiver. Time diversity and frequency diversity, where the same information transmitted at different frequency bands or different time instants and antenna diversity are also common diversity methods [7]. In this thesis a MIMO system with receive diversity is employed. MIMO system can also be used for increasing data rate (multiplexing). For example as it is studied in [6] for Gaussian channels the capacity of the channel, with r receiver and t transmitter antennas under power constraint P equals

$$C = \int_0^{\infty} \log(1 + P * \lambda/t) \sum_{k=0}^{m-1} \frac{k!}{(k+n-m)!} [L_k^{n-m}(\lambda)]^2 \lambda^{n-m} e^{-\lambda} \, d\lambda \ldots\ldots (2.2)$$

λ is eigen value of $W = HH'$ for $r < t$ and $W = H'H$ $r \geq t$

Where H is the channel gain matrix and H' its transpose, $L_k^{n-m}(\lambda)$ is Laguerre polynomial of order k.

And m = min{r, t} and n = max{r, t}. Where t is the number of transmitter antennas and r is the number of receiver antennas.

Consider t = 1. In this case m = 1 and n = r. Noting that $L_0^{n-m}(\lambda) = 1$ an application of Equation (2.1) yields the capacity as

$$\frac{1}{l(r)}\int_0^{\infty} \log(1 + p\lambda)\,\lambda^{r-1}e^{-\lambda}d\lambda, \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ (2.3)$$

Where l(r)=(r-1)! This shows the capacity increases with increasing number of receiver antennas due to the presence of the factor λ^{r-1} provided that λ is greater than one. In our case for two transmitter and three receiver antennas the above capacity function is expressed as:

$$C = \int_0^{\infty} \log\left(1 + P * \frac{\lambda}{2}\right) \sum_{k=0}^{1} \frac{k!}{(k+1)!}\left[L_k^1(\lambda)\right]^2 \lambda e^{-\lambda}\, d\lambda \ldots\ldots\ldots\ldots\ (2.4)$$

in which case the capacity also increases due to the existence of additional antennas.

In addition to using a MIMO channel if we use Joint Source Channel Coding (JSCC) for the encoder we can increase the capacity and the reliability more. This is true because we are considering the channel conditions prior to transmission and at the same time we can minimize the processing delay. The delay is decreased since the encoding is done at once in case of JSCC which is done twice (source coding and channel coding) in Shannon's separation theorem.

2.3 Joint Source Channel Coding/Decoding Schemes

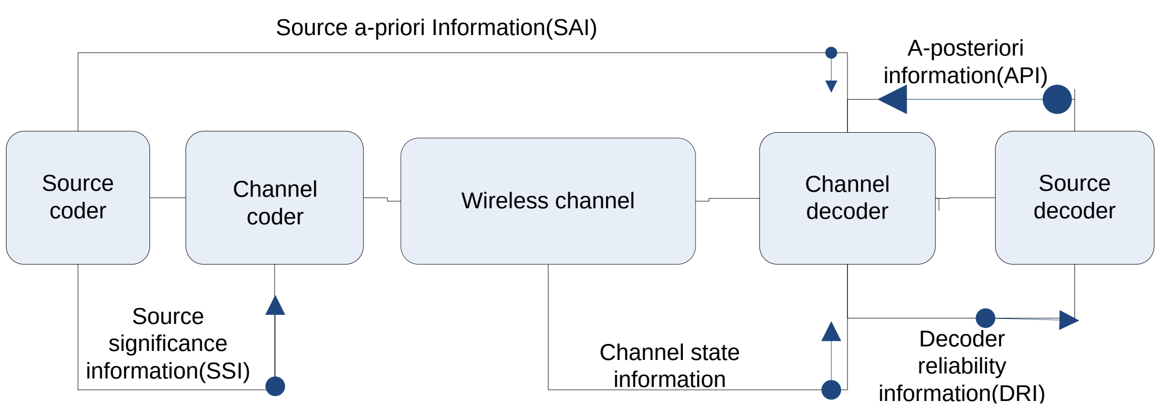

Figure 2.3 Communication System Model with JSCC

In the basic model for joint source channel coding given in the above Figure, channel coder uses the source significance information (SSI) from the source code. Therefore, the overall efficiency is enhanced by such joint source channel coding. Then the codeword sequence from the encoder after modulation is sent through the channel. The signal arrives at receiver side after being corrupted by fading and Gaussian noise. The decoder makes use of the source a priori information (SAI) from the encoder side and the channel state information from the channel estimator to carry out its task of decoding the original signal. Decoder Reliability Information (DRI) and the a-posteriori Information (API) between the channel decoder and the source decoder can be measured repetitively to achieve minimum joint decoding error.

Joint source channel coding theorem can be divided into the following categories based on the above principles [8].

2.3.1 Joint Source Channel Coding (JSCC):

This technique is based on optimizing the source and channel coding at the encoder side which is opposite to the source channel separation theorem. The following are joint source channel coding design schemes: JSCC for rate allocation, JSCC for unequal error protection, JSCC for optimized quantization, JSCC for resilient source coding, JSCC for rate distortion optimization coding and optimal rate control. In these methods the decoding is done separately as suggested by Shannon [1]. Following this we will see some details for the three most common methods of JSCC. Namely JSCC for unequal error protection, JSCC for rate allocation, JSCC for optimized quantization. These methods are common due to the reason that they result in better performance efficiency relative to the other JSCC methods.

2.3.1.1 JSCC for unequal error protection

The output bit stream of source coders has diverse impacts for quality of image and video reconstruction. Some data very sensitive to channel errors contribute more to reconstruction quality. This includes data such as head information, motion vector, Direct Current (DC) coefficients from the Discrete Cosine Transform (DCT), etc. Some other bits, such as Alternating Current (AC) coefficients, contribute less to reconstruction quality. However, if these bits are lost or mistaken, the performance of the reconstructed signal will not be reduced remarkably. So the channel coder gets this information from the source coder for optimized performance. Therefore to increase the overall performance protect those more vital data by better channel codes, and protect those less vital data by weaker channel codes. The error control concept of varying the protection level according to the significance is known as unequal error protection (UEP) [8].

Performing UEP strategy at the encoder requires partitioning the data into image and video components to divide bits into groups according to their level of significance.

2.3.1.2 JSCC for rate allocation

This technique is one of the JSCC methods that allocate optimum values of the rate (in bits per sample (bps)) for the source coder and the channel coder for a fixed value of an overall rate in order to minimize the end to end distortion. As the rate of source coding increases (which can be controlled by the rate of quantization) the distortion due to compression decreases. But the channel gets more robust error protection as the channel encoder rate decreases. In rate allocation JSCC method one has to find an optimal division of the overall rate. At higher SNR values more rates must be allocated for channel coding to increase the data transmission rate. But at lower values of SNR the source coder must be given more rates [13]. Common method of JSCC for rate allocation is to code the source and the channel separately but jointly optimize them. HOW??

The overall rate is the sum of the channel rate and the source rate. The same is true for the distortion.

$$R_{S+C} = R_C + R_S \quad (2.5)$$

$$D_{S+C} = D_S + D_C \quad (2.6)$$

Where R_{S+C} and D_{S+C} are the total rate and distortion respectively. R_C,R_S, D_S, D_C are the channel bit rate, source bit rate, source distortion and channel distortion respectively. The idea of joint source channel coding for rate allocation is to minimize the total distortion subject to the give channel bandwidth.

2.3.1.3 JSCC for optimized quantization

In this method of JSCC the source and the channel coding is done at the same time, no separate coding. The source coding (quantization for analog signals) will be optimized based on the channel transition probabilities. This method is the one used in this thesis and will be discussed later in detail.

2.3.2 Joint Source Channel Decoding (JSCD)

Joint Source Channel Decoding (JSCD) is mainly employed for fixed-length encoding (FLC), JSCD utilizing a priori information (SAI) for bit-level en/decoding such as codebook-excited linear prediction (CELP). The advantages of the FLC lie in its simple implementation and low complexity. However, the compression is not as efficient as the variable length encoding. The JSCD based on the FLC is mainly applied to speech coding systems. JSCD utilizing a priori information (SAI) for bit-level en/decoding can be implemented by exploring source a priori knowledge, helping to improve the system performance. One advantage of the algorithm is that it requires only modifications at the decoder side, without interfering with the source encoder, at the coder side, without using variable-length coding, the JSCD based on SAI for bit-level en/decoding is mainly applied to data coding systems. The other technique in Joint Source Channel Decoding (JSCD) is Variable-length coding (VLC). It is widely used in JPEG, JPEG2000, MPEG, H.264 image/video compression coding standards. The main advantages of the VLC lie in its high compression rates and easy error detections. However, it is very sensitive to channel noises and hard to synchronize. Even a single binary bit error could make the bit stream not decodable, until another synchronization mark is found. In order to improve the robustness of the VLC, JSCD for variable-length coding with bit-level decoding technology can be used to enhance the transmission performance [6]. There are also various specific techniques of JSCD. Some of them are presented here.

2.3.2.1 JSCD for Fixed-Length Encoding

Some early work focused on the source fixed-length coding (FLC), such as the JSCD methods based on differential pulse code modulation (DPCM) [22] or codebook excited linear prediction (CELP) [8]. The advantages of the FLC lie in its simple implementation and low complexity. However, the compression is not as efficient. The JSCD based on the FLC is mainly applied to speech coding systems.

2.3.2.2 JSCD Utilizing a Priori Information for Bit-level En/Decoding

Source statistical properties are also known as source a priori information (SAI). Source decoder and channel decoder can implement JSCD by utilizing SAI, helping to improve the system performance. In [4], Hagenauer proposed a basic framework for joint source-channel encoding/decoding. By extending the well-known soft output Viterbi algorithm (SOVA), he presented the APRI-SOVA algorithm, which incorporated the source a priori or a posteriori knowledge into the channel decoding. One advantage of the APRI-SOVA algorithm is that it requires only modifications at the decoder side, without interfering with the source encoder. As a result, a coding gain of 0.5 dB is achieved without increasing the bit rate [8]. Later, Boudreau developed the applications of the APRI-SOVA algorithm to speech coding standards. In 2002, Fingscheidt also presented a JSCD method, which improved the mobile communication system performance by exploring the source a priori knowledge [11].

2.3.3 Variable-length JSCC with Variable-length JSCD

Another approach to Joint Source Channel Coding/Decoding scheme is variable-length encoder and decoder. For variable-length JSCC with variable-length JSCD, Liu and Tu [7] constructed a new symbol-level joint trellis with compound states by merging a VLC trellis with a convolutional trellis. Based on this joint trellis, the symbol-level a posteriori probability (APP) decoding algorithm is also derived, which leads to a joint iterative decoding approach with symbol-level soft outputs.

The above categorization of joint source channel encoding and decoding schemes is based on the place where the technique is applied. In the following sections specific techniques of joint source channel coding encoding methods will be discussed in detail.

Chapter 3

3 System Model

3.1 Channel Model

The channel model in this thesis is a Discrete Memoryless Channel (DMC) which is defined by finite input and finite output values with a set of transition probabilities. This channel model is preferred since it is a simple model for real world channels. The channel transition probabilities are a function of the fading process in the wireless channel, in our case the Rayleigh fading channel, and the addition of Gaussian noise. In the following section some details of Gaussian and Rayleigh fading channels are presented.

3.1.1 Additive White Gaussian Channel

Gaussian channel is a discrete time continuous alphabet channel. For input signal x_t and output alphabet y_t the output at time t is given by the sum of the input signal x_t and n_t. The noise n_t is assumed to be flat in frequency (white RELALLY??) and it is independent and identically distributed,(i.i.d) taken from a Gaussian distribution

$$.y_t = x_t + n_t \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \text{(3.1)}$$

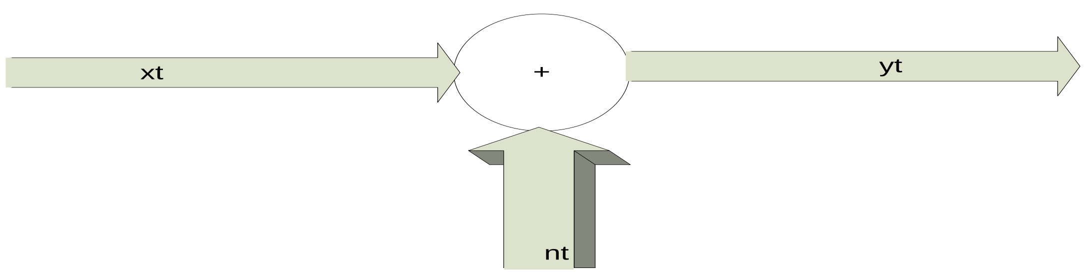

Figure 3.1 Gaussian Channel

3.1.2 Rayleigh Fading Channel Model

Rayleigh fading channel is also a discrete time continuous amplitude channel which passes the transmitted signal at different paths with different delay. Therefore with modeling perspective the difference between Gaussian channel and Rayleigh fading channel is that in Rayleigh fading channel the fading coefficients multiply the input signal to attenuate it before the Gaussian noise is added. The Rayleigh fading channel has multiple paths for the signal to arrive at the receiver with different arrival time. However the Gaussian channel has only a single path.

Therefore, in wireless communications, the output signal is a function of the input signal, the fading coefficients and the Gaussian noise.

$$y_t = h_t x_t + n_t \qquad (3.2)$$

Where the fading coefficient h_t is i.i.d Rayleigh distributed RV.

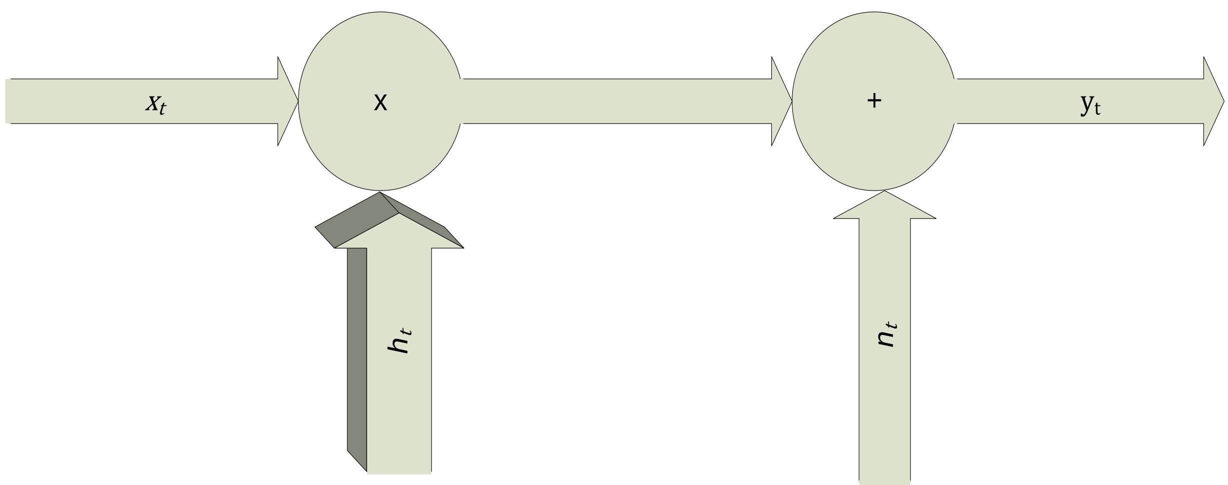

Figure 3.2 Rayleigh fading channel followed by AWGN channel

3.2 Vector Quantization as a Means of Source Coding

Generally there are two categories of source coding: lossless source coding and lossy source coding. In the first category the source is represented completely by the coded information; which means the source code can be decoded such that the decoded data and the original data are identical to each other. This only applies for digital data. On the other hand continuous amplitude sources can't be coded in a lossless manner. In such cases one has to apply lossy source coding through quantization.

3.2.1 Lossless Source Coding

Lossless source coding is data compression where decompression gives back an exact copy of the original. It is used in modems, for compressing text files,…, etc. Although images, sound, and video are more often compressed using a lossy compression technique to achieve better compression many compressors have a lossless mode. In addition, a lossless algorithm may be used as a building block in designing a lossy compressor.

Shannon established that there is a fundamental limit to lossless data compression. This limit, called the entropy rate, is denoted by H. The exact value of H depends on the information source, more specifically, the statistical nature of the source. It is possible to compress the source, in a lossless manner, with compression rate close to H. It is mathematically impossible to do better than H[1].

Some examples of lossless compression:

- Fixed-length to fixed-length (FF) coding: each symbol in a data is represented by equal length block codes. Suppose you have a program that uses files that contain only decimal digits, spaces, tabs, carriage returns, dots, and commas. On most computers these files would be stored in eight-bit ASCII. It is easy to write a special-purpose compressor that will compress these files by a factor of two just by encoding each ASCII character into a four-bit codeword.

- Fixed-length to variable-length (FV) coding: This method produces a variable length output bits from a block of input characters with fixed length. Even if all 256 possible characters occur in the file, it is still possible to compress it if some characters are more frequent than others. Suppose the files described in the previous example are not guaranteed to contain only the fifteen characters mentioned, but the remaining 241 characters occur very infrequently. The fifteen frequent characters could be encoded using codewords 0001 through 1111, while 0000 would be reserved as an escape code to indicate that the following eight bits should be interpreted as a single character.

- Variable-length to fixed-length (VF) coding: Suppose you have some files representing black and white images, with black pixels represented by 1-bits and white pixels by 0-bits. Often such files will have long runs of identical bits. A simple run-length compressor might encode runs of

1 through 128 0-bits using codewords 00000000 through 01111111 and encode runs of 1 through 128 1-bits using codewords 10000000 through 11111111. If there are many more long runs than short runs this would give good compression. Variable-length to variable-length (VV) coding: The compressor in the previous example might be improved by using short codewords for frequently-occurring run lengths at the expense of lengthening the codewords for less frequent run lengths.

Lempel–Ziv (LZ) and Huffman codes are examples of variable-length to fixed-length and fixed-length to variable-length lossless source coding methods, respectively.

3.2.2 Vector Quantization (Lossy Source Coding)

Usually it is not possible to exactly reconstruct the original data without distortion. For example for a continuous amplitude signal to be sent over a digital communication channel, it needs to be represented by a limited number of bits due to bandwidth constraint. In the processes of converting the continuous amplitude signal to a stream of bits, it is obvious that loss of some data will occur. In general this means that the signal has to be converted from its analog form to digital representation before transmission. The technique of conversion involves quantization followed by index assignment (digital values). The quantizer assigns the closest value in its output set to the analog information called an output level.

The oldest and common way of quantization is scalar quantization. Scalar quantizer (SQ) is specified by encoding function (E) and decoding function (D). The real line is mapped to a set of indices in the encoding and the reverse is done in the decoding function [9].

A scalar quantizer can be summarized as follows:

$$E : \mathrm{R} \rightarrow I = \{0, 1, \cdots, N-1\}, \quad D : I \rightarrow C = \{c_0, c_1, \cdots, c_{N-1}\} \ldots\ldots\ldots (3.3)$$

Where: N is the number of levels, I indices assigned for the codevectors, C is the codebook.

And the quantizer rate is:

$$r = log_2 N \text{ bits per symbol (bps)} \ldots\ldots\ldots (3.4)$$

Assuming that the input signal to the source encoder to be X, the scalar quantizer outputs $E(X) = \hat{X}$ in a quantization source coding system. The encoder finds the nearest quantization level from the set of quantization levels set on the real line. The encoder induces a partition $P = \{S_i\}_{i=0}^{N-1}$ of R and assigns indices to X based on the regions in which it is located.

The performance of the quantizer is measured by how accurately it represents X. And this can be measured by how small the distance between the original and the quantized signal is. In other words, distortion provides the measure of performance quantitatively as follows.

$d(X, \hat{X}) = |X - \hat{X}|^2$ is the distance between X *and* $\hat{X}$.

Based on the above equation, the distortion of the scalar quantizer is defined as

$$E = E\{d(X, \hat{X})\} = E\{|X - \hat{X}|^2\} \text{ .... (3.5)}$$

What is the difference between the three E's in (3.5)? Encoding function, expectation, …..

Generalizing the scalar input signal X into vector input signal **X** will make the quantizer vector quantizer which has multiple dimension according to the vector size. The vector quantizer again outputs an index from a finite set. Therefore a vector $X \epsilon R^K$ is quantized into a set of indices *I = {0, 1, 2,3,….,N-1}*. And the recovered signal $\hat{X}$ is to be chosen from a set of codebooks $C = \{c_0, c_1, c_2, \ldots, c_{N-1}\} \epsilon R^K$ which correspond to the indices.

The vector quantizer maps $X \epsilon R^K$ into $\hat{X} \epsilon R^K$, which is an element of C (codebook), $\hat{X}$ is an approximation of the continuous amplitude signal X in the quantization process.

Therefore, the vector quantizer can be defined as a function:

$$E: R^k \rightarrow C \text{ .... (3.6)}$$

The rate of the vector quantizer is determined by the number of bits per source symbol and is given by:

$$r = \frac{log_2 N}{K} bps \text{ .... (3.7)}$$

where K is the dimension of the vector quantizer.

The measure of distortion is also defined similarly as in the scalar quantizer case.

$$D = E\{d(X, \hat{X})\} = E\{|X - \hat{X}|^2\} \text{ .... (3.8)}$$

Where E{} is an expectation function.

In the above equation $X \; and \; \hat{X}$ are vectors. Similar to the scalar quantizer case the vector quantizer forms a partition region $P = \{S_i\}_{i=0}^{N-1}$, which has multidimensional cells. For analog source coding vector quantizer is usually used than the scalar quantizer since it has a better performance.

Vector quantizer outperforms scalar quntizer due to the following reasons [9].

- It gives larger degrees of freedom of choosing the shape of the quantization region, a feature that makes the VQs more important in terms of variety of the simple cell shapes.
- Vector quantizer has also factors that help to exploit the dependence of vector components and to make fractional bit rates per symbol possible due to the reason that they use multiple dimensions for quantization.
- The other benefit of a vector quantization is that it is possible to achieve the ultimate limit of the rate distortion function as the quantization dimension goes to infinity.

As it is stated above the optimal vector quantizer is one that minimizes the distortion between the source vectors X and its reproduction $\hat{\hat{X}}$. Therefore the goal of the vector quantizer is to find the encoding and decoding pair (E and D) with the least possible distortion subject to rate constraint and quantization level constraint.

The distortion of the vector quantizer is written as [9]:

$$D_{VQ} = \frac{1}{k} E\left[\left\|X - \hat{X}\right\|^2\right] \quad \text{……………..…………………….(3.9)}$$

D_{VQ} is mean of the distortions in the signal and k is quantization dimension.

$$D_{VQ} = \frac{1}{k} \sum_{j=0}^{N-1} E\left\{\left\|X - c_j\right\|^2 \middle| X \in S_j\right\} P(X \in S_j) \text{….….…. (3.10)}$$

$P(X \in S_j)$ is the probability of X being an element of region S_j and C_j is the j^{th} codeword that corresponds to S_j.

$$D_{VQ} = \frac{1}{k} \sum_{j=0}^{N-1} \int_{S_i} p(x) \left\|x - c_j\right\|^2 dx \quad \text{…………………..…. (3.11)}$$

Equation 3.11 finds the mean of the distortion in the vector quantizer. $p(x)$ is the source distribution probability.

The above problem has no global solution. However using the two necessary conditions for optimality we can find the solution for the above problem. This conditions are the nearest neighborhood condition (NNC) and centroid Condition (CC).

- *Nearest neighbour condition*: for a given set of cluster centroids, any data object can be optimally classified by assigning it to the cluster whose centroid is closest to the data object with respect to the distance function.

- *Centroid condition*: for a given partition, the optimal cluster representative minimizing the distortion is the *centroid* of the cluster members [10].

Using the Nearest Neighborhood and centroid conditions, the encoding regions are expressed as:

$$S_i = \{x \epsilon R^k : \|X - c_i\|^2 \leq \|X - c_j\|^2, for\ all\ j \neq i\} \ldots\ldots\ldots\ldots\ldots \quad (3.12)$$

The number of regions S_i (i=1,2,…,N) is the same as the quantization level (N). For each quantization level the above equation finds the corresponding training vectors nearest to the corresponding codewords.

And optimal codebooks are obtained from necessary condition for the optimality of a VQ called the centroid condition. It assigns the optimal codebook to a given encoding partition *P*. In other words, given encoding regions $\{S_i\}_{i=0}^{N-1}$, the VQ can only be optimal if the output code vectors are the centroids of the encoding regions.

$$c_j = E\{X|X \in S_i\} = \frac{\int_{X \in S_i} xp(x)dx}{\int_{X \in S_i} p(x)dx} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (3.13)$$

Equation 3.13 is used to calculate the codebooks using the centroid function. C_j is the codeword for j^{th} region. Where x is training vector in the data and *p(x)* is probability of occurrence of x. The equation finds the centroid of each region to determine the codebook.

The above two equations used iteratively will provide us an optimal codebook and optimal partition.

3.3 Channel Optimized Vector Quantization

All source coding methods described in the previous sections disregarded the channel. In pure source coding systems, the output of the encoder *E* is directly fed to the input of the decoder *D*. Vector quantization, the most important source coding scheme we reviewed, for instance, does not assume any statistical index perturbation between the encoder and decoder. However, noise makes an inevitable and notable effect in real-world communication systems.

When channel conditions are taken into account during source coding (quantization), the coding will be optimized to channel conditions. Such coding scheme is referred to as JSCC for Channel Optimized vector quantization (COVQ). In JSCC for COVQ the objective is to find optimal partitions and codebook, optimal with respect to the channel conditions (channel transition probability matrix).

For a given source, channel, quantization dimension k, and fixed rate r, we wish to find the optimal partition and codebook that minimizes the overall distortion.

The overall distortion of the COVQ is given as follows; the definition is similar to the one given for vector quantizer except it involves the transition probability which accounts for channel optimization.

$$D_{COVQ} = \frac{1}{k}\sum_i \int_{s_i} p(x)\sum \quad P_{Y|X}(j|i)d(x, C_j)dx$$

$$D_{COVQ} = \frac{1}{k}\sum_i \int_{s_i} p(x)\sum_j P_{Y|X}(j|i)\left|x - C_j\right|^2 dx$$

$$D_{COVQ} = \frac{1}{k}\sum_i \sum_j P_{Y|X}(j|i) \int_{s_i} p(x)\left|x - C_j\right|^2 dx \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots (3.14)$$

JSCC for COVQ is a result of improving the VQ under the conditions of probabilistic index perturbation between the input and output of the channel. Thus the basic idea of the COVQ is to design a VQ by incorporating the channel conditions into the design algorithm, trading off the quantization and channel noise in order to minimize the end-to-end distortion.

From Equation 1.5 the mean of the distortion is obtained by averaging the distortion for all codeword and a training vector x : $\sum_j P_{Y/X}(j|i)d(x, C_j)dx$ in the codebook and again averaging this value by the possible training vectors probability. $\sum_i \int_{s_i} p(x)\sum_j P_{Y|X}(j|i)d(x, C_j)dx$.

One proven advantage of the COVQ is that there is no need to add error-protection intended redundancy to the system and its performance is acceptable even without that sort of extra redundancy.

Similar to the VQ case, the global solution to this problem is unknown. However, there are generalized versions of nearest neighborhood condition (NNC) and central condition (CC) for the COVQ which enable us to find locally optimal solutions in the course of an iterative procedure similar to a VQ case. It starts from a suitably chosen initial codebook (using splitting algorithm and then modifying it by simulated annealing algorithm). These methods for initial code book selection will be discussed later.

For the overall distortion expressed by the following equation;

$$D_{COVQ} = \frac{1}{K}\sum_i \sum_j P_{Y|X}(j|i) \int_{S_i} p(x)\left|\left|x - c_j\right|\right|^2 dx, \quad \ldots\ldots\ldots\ldots\ldots\ldots (3.15)$$

The problem is to find the optimal partition P* and the optimal codebook c*. By using training sequence, replacing the integrals with summations (where?) and density functions with empirical weights for a training vector {x_1, x_2, … x_M} Equation 3.15 is modified as:

$$D_{COVQ} = \frac{1}{KM}\sum_{i=1}^{M}\sum_{j=0}^{N-1} P_{Y|X}\big(j\big|\varepsilon(x_i)\big)\, d(x_i, c_j)$$

$$D_{COVQ} = \frac{1}{KM}\sum_{i=1}^{M}\sum_{j=0}^{N-1} P_{Y|X}(j|\varepsilon(x_i))\,||x_i - c_j||^2 \ldots\ldots\ldots\ldots\ldots\ldots \text{(3.16)}$$

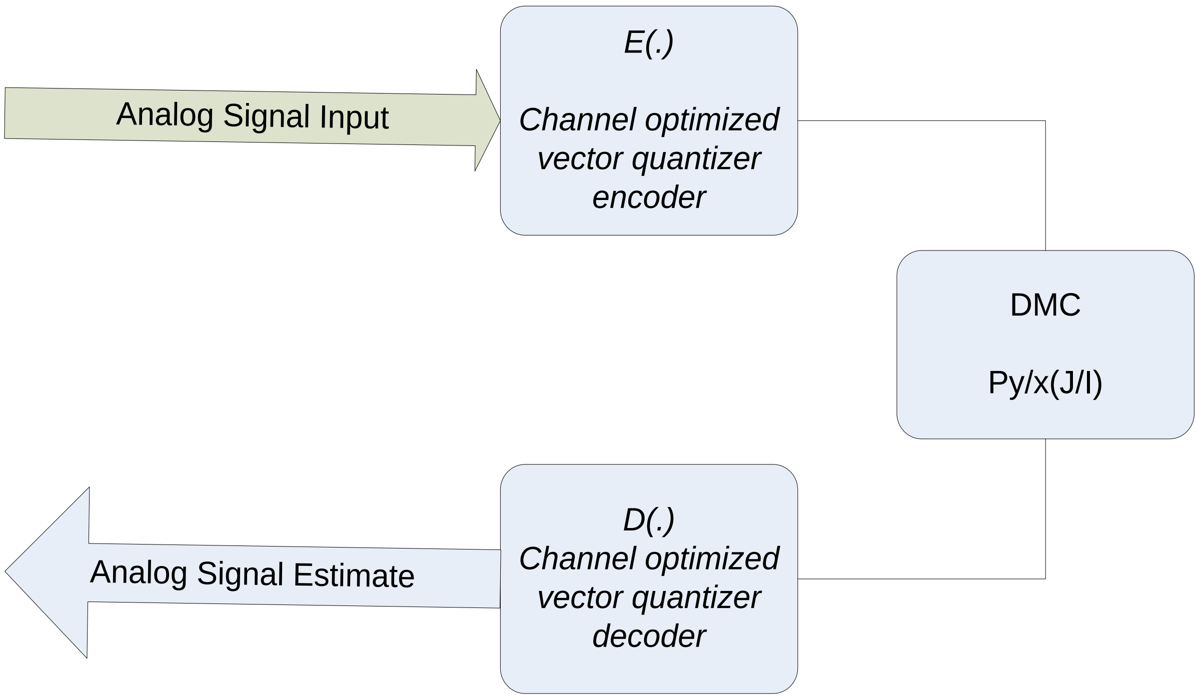

Figure 3.3 General block Diagram of communication system with channel optimized vector quantizer

Why is this figure require? Part of the discussion?

$\varepsilon(x_i)$ is the encoding function that finds the index of the training vector. M is the number of training vectors and K is the quantization dimension. Nearest neighborhood and centroid conditions are used to find the optimal codebook and the optimal partition sets.

Using generalized CC condition, the codevectors for the given partition set $p = \{S_0, S_1, S_2, , S_{N-1}\}$ are determined as:

$$c_j = \frac{\sum_{0}^{N-1} P_{Y|X}(j|i)\sum_{s_i} x}{\sum_{i=0}^{N-1} P_{Y|X}(j|i)|s_i|}, j = 0,1, \ldots N-1 \ldots\ldots\ldots\ldots\ldots \text{(3.17)}$$

Where $|s_i|$ is the number of training vectors in the quantization cell S_i.

Equation 3.17 finds the centroid of each partition averaged to its transition probability.

The encoder uses the codebook, C obtained from the above equation [23], to determine the optimal partition set. This partition set is given by:

$S_i^* = \left\{x: \sum_j P_{Y|X}(j|i)\left\|x - c_j\right\|^2 \le \sum_j P_{Y|X}(j|\hat{\imath})\left\|x - c_j\right\|^2, for\ all\ i \in I = \{0, 1, 2, \dots, \mathrm{N}-1\}., i \neq \hat{\imath}\right.$ for every i ϵ I={0, 1, 2,…, N−1}. ………………………………………………………… (3.18)

Using Equation 3.18 the training vectors nearest to each codeword are determined by averaging the Euclidean distance between each training vector and codewords by the transition probability of that codeword. Through this the training vectors are divided into the partition regions. Therefore, the encoding function ε is a function which finds the codebook for each training vector that minimizes the distortion at the receiver. This can be written mathematically as:

$$\varepsilon(x_i) = arg\ \min_{i \in I} \sum_{j=1}^{N-1} P_{Y|X}(j|i)\left\|x - c_j\right\|^2, x_i \epsilon \{\mathrm{x}_1, \mathrm{x}_2, \dots, \mathrm{x}_\mathrm{M}\} \quad (3.19)$$

The goal of the system is to transmit the random vector $\mathrm{X} \in \mathrm{R}^\mathrm{k}$ over the channel and form an estimate $\hat{\mathrm{X}}$ of X such that the distortion $\mathrm{E}\left\|\mathrm{X} - \hat{\mathrm{X}}\right\|^2$ is minimized. The channel optimized vector quantizer encodes $\{\mathrm{X}_\mathrm{n}\}$ input data at a rate of $\mathrm{r} = \frac{\log_2 \mathrm{N}}{\mathrm{k}}$ bits per symbol (bps). Therefore, the COVQ encoder is a mapping

$$\varepsilon: \mathrm{R}^\mathrm{k} \to \mathrm{I}_\mathrm{n} \triangleq \{0,1, \dots \mathrm{N}-1\} = \{0,1\}^{\mathrm{kr}} \quad (3.20)$$

Where N is the number of quantization regions and $\{0,1\}^{\mathrm{kr}}$ represents the binary representation of the indices. Then, the binary representation of $\varepsilon(\mathrm{X}_\mathrm{n}) = \mathrm{I}_\mathrm{n}$ is modulated and sent over the channel in kr(the number of bits in a symbol(index)) consecutive channel uses. The encoding is done using the decision regions $\{\mathrm{S}_\mathrm{i}\}_{\mathrm{i}=0}^{\mathrm{N}-1}\left(\mathrm{N} = 2^{\mathrm{kr}}\right)$ via the encoding rule:

$$\mathrm{X} \in \mathrm{S}_\mathrm{i} \leftrightarrow \mathrm{I} = \varepsilon(\mathrm{X}) = \mathrm{i} \quad (3.21)$$

The input index probability distribution is denoted by P_i for i = 0, 1, 2, …, N − 1. The indices will be changed into their binary representation and after modulation, will be transmitted with unit energy. The modulated signals are denoted by $\mathrm{W}_\mathrm{n}^1, \mathrm{W}_\mathrm{n}^2, \dots, \mathrm{W}_\mathrm{n}^{\mathrm{kr}}$ and form the vector $\mathrm{W}_\mathrm{n} \in \{-1, +1\}^{\mathrm{kr}}$. Each symbol is transmitted over the physical channel (MIMO channel) with full diversity. Since the corrupted signal at the receiver side is only a function of transmitted symbols at the same time unit, the received vector R_n consists of kr consecutive received values: $\mathrm{R}_\mathrm{n}^1, \mathrm{R}_\mathrm{n}^2, \dots, \mathrm{R}_\mathrm{n}^{\mathrm{kr}}$.

3.4 Orthogonal Space Time Block Coding Combiner (OSTBC)

As we are dealing with performance of JSCC on MIMO channels with OSTBC, it is good to discuss how we get the channel index transition probability of OSTBC based on soft decision decoding. Methods that exploit soft information can provide significant performance gains. In addition to using the soft information efficiently, the solution should allow the COVQ index transition probabilities to be determined in closed form, since this will make the COVQ design and encoding phases more practical. As we illustrate below, linear combining has both of the above properties. To get a better performance of the OSTBC combiner soft decision decoding is used. (SO WHAT? USED IN YOUR WORK?)

The objective is to design the chain of the blocks between the encoder and the decoder by a discrete memoryless channel and design an efficient vector quantizer system. Which means a transition probability function will replace the effects of those blocks between the transmitter antenna and the receiver antenna and the vector quantizer will be designed with less distortion as possible. For this the system is designed so as to incorporate the soft information of the orthogonal space time block coded channel. The space time decoding is designed by incorporating space time soft detector followed by linear combiner.

The output of the OSTBC soft decoder is given [12] by:

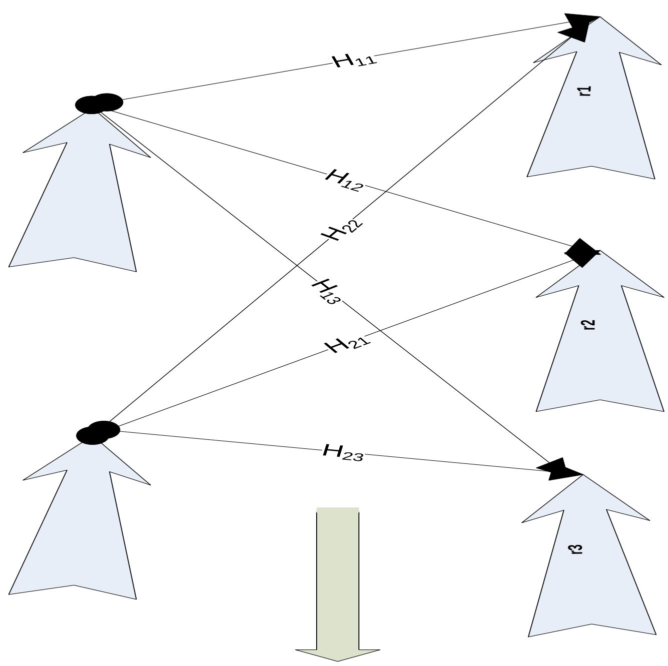

Figure 3.4 MIMO channel

$$r^j = g^{'} Y_j c + n^j \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (3.22)$$

Where: c is the codeword, r^j (see Figure 3.4) is the received signal, $g^{'} = g\sqrt{\gamma_s/t}$ for t transmit antennas, γ_s is the signal to noise ratio, g is the coding gain and n^j is the Gaussian noise

$$Y_j = \sum_i |H_{ji}|^2, j = 1, \ldots, L \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (3.23)$$

Where L is the number of receiver antennas.

The output of the OSTBC linear combiner is then fed to a scalar quantizer which acts as a soft decision decoder. Let $\{U_k\}_{k=0}^{N-1}$ be the decision regions of the scalar quantizer (soft information extractor) and $\{w_k\}_{k=0}^{N-1}$ be the code points, where $N = 2^q$ is the number of codewords (q is the quantization levels for the soft decision decoder). Since the output of the OSTBC linear combiner can have any real value the soft decision quantizer should have upper and lower unbounded decision regions.

$$U_k = \begin{cases} -\infty, if\ k = -1 \\ (k + 1 - N/2)\Delta,\ if\ k = 0, \ldots, N-2 \\ +\infty, if\ k = N-1 \end{cases} \ldots\ldots\ldots\ldots\ldots (3.24)$$

Where: Δ is quantization interval. And the quantization rule is simply:

$f(\delta_j) = k \quad if \quad \delta_j \leftarrow (U_{k-1}, U_k)$, k=1,…, N-1. Both Δ and $f(\delta_j)$ are scalar values.

3.5 Transition Probabilities of the Equivalent Discrete Memoryless Channel

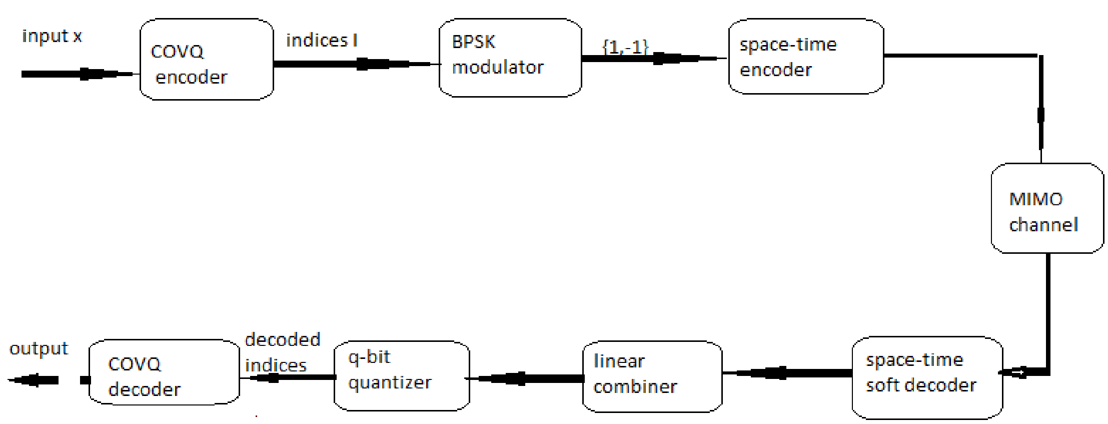

Figure 3.5 Communication system Model with JSCC

The transition probabilities that need to be determined are those corresponding to 2^{kr}, (from the definition of $rate\, r = \frac{log_2^N}{k}$) (I) input symbols and 2^{qkr}(J) output symbols. The discrete channel is the concatenation of the OSTBC encoder, the MIMO channel, the space time soft-decoder, the linear combiner with optimized weight coefficients, and the uniform quantizer. The discrete channel is equivalent to a binary-input 2^q output DMC used *kr* times (number of bits used to represent a symbol).

Therefore the required set of transition probabilities are $P(w_k/c_i)$, for all w_k which q tuple whre c_i for i=1 or 2 is the BPSK constellation point, and are assumed to acquire a value of 0 and 1 respectively.

$$P(\omega_k/c_i) = Q\big((U_{k-1} - c_i)\delta\big) - Q((U_k - c_i)\delta) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \text{(3.25)}$$

Where Q (.) is defined as:

$$Q(\delta) = p\big(c_i \to c_j\big) = \frac{1}{2}\left(1 - \frac{\delta_{ij}}{\sqrt{{\delta_{ij}}^2+2}} \sum_{k=0}^{KL-1} \binom{2k}{k} \frac{1}{\left(2({\delta_{ij}}^2+2)\right)^k}\right) \ldots\ldots\ldots\ldots \text{(3.26)}$$

as it is discussed in [12].

For a channel optimized vector quantizer with rate r and dimension k let's denote the binary representation of the index of decision regions S_i by $\{b_l\}_{l=1}^{kr}$ and that of the codevector ω_j by $\{B_l\}_{l=1}^{kr}$, where B_l is a binary q-tuple. As the DMC is memoryless, the COVQ index transition probabilities can easily be computed as:

$$P_{J|I}(j|i) = \prod_{l=0}^{kr} P(\omega_{B_l} | c_{(2-b_l)}) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \text{(3.27)}$$

This index transition probability distribution is used in the design of the channel optimized vector quantizer.

3.6 Simulated Annealing Algorithm (Related with the previous section?)

The code book to be generated from the COVQ which uses vector quantization optimized to channel transition probability matrix needs a suitably selected initial codebook. To select the initial codebook Simulated Annealing algorithm is used.

In the process of simulated annealing algorithm first a codebook will be generated by using vector quantization, which is only about the source coding. The codebook so obtained will be modified in such a way that the channel transition probabilities are taken into account such that the channel distortion is minimized. This is done through simulated annealing algorithm. The procedures in the algorithm are presented below.

The distortion introduced by the channel denoted by D_c is given by:

$$D_c = \frac{1}{k}\sum_{i=0}^{N-1}\sum_{j=0}^{N-1} P_{Y/X}(j/i)P_i d(c_i, c_j) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \text{(3.28)}$$

The above equation gives a good clue (HOW?) to select the initial codebook since index assignment to the codewords determines the channel distortion. The distortion in the above equation can be minimized over the order of the index assignment, {0, 1, 2, …, N-1}, since Dc (channel distortion) is indirectly a function of the indices via the codebook and the index transition matrix.

Steps of Simulated Annealing Algorithm

1. *Randomly pick an initial order of the indices, I = {0, 1, 2, …, N-1}, ,for the code book obtained through vector quantization and an initial temperature T0.*
2. *Choose the next order of indices I' again randomly and calculate the change in the distortion Dc using the present and previous randomly chosen indices. If* $\Delta Dc \leq 0$ *replace I with I' and go to step 3. Otherwise, replace I with I' with probability* $e^{-\Delta Dc/T}$ *and go to 3.*
3. *If there is no distortion change after a specific number of perturbations Ncut , proceed to step 4. If there is a change in the distortion go back to step 2.*
4. *Multiply the temperature by some factor α which is between 0 and 1. If the temperature is less than prescribed minimum temperature Tf or the system appears to be stable (not much distortion change) stop with I as the final order of the code book. Otherwise go back to the second step.*

3.7 Channel Optimized Vector Quantizer Algorithm

For the design of the channel optimized vector quantizer the Linde, Buzo and Gray (LBG) [9] algorithm is used.

Let $N = 2^n$ the number of codewords in their respective regions, k quantization dimension M the number of training vectors. Choose a fixed $\in > 0$ as the target stopping threshold and δ as the perturbation constant for the splitting algorithm, the algorithm selected for initial codebook selection. The procedure of the LBG algorithm is stated as follows:

1. ***Start:*** *N*=1 the counter for the number of codevectors and*

$$c_0^* = \frac{1}{M}\sum_{m=1}^{M} x_m$$

as the only codevector for the one level quantizer. The globally optimal one level codebook of a training sequence is the centroid of the entire sequence.

Calculate

$$D^{(1)} = D^* = \frac{1}{kM}\sum_{m=1}^{M} \|x_m - c_0^*\|^2$$

as the initial and optimal average distortion of the one-level quantizer. Thus the first initial codebook found for the one-level quantizer (which is also optimal) is $c_1^{(1)} = c_1^* = \{c_0^*\}$.

2. ***Splitting****(obtains a code book by first taking the average of the training vectors as initial codebook and then continues by adding and subtracting a constant vector until the size of the codebook equals to the number of codevectors): to find the initial codebook of N*(the counter for the number of codevectors) level quantizer from the N*/2 level quantizer ,set*

$$c_i^{(1)} = c_i^* + \delta,$$
$$c_{i+N^*}^{(1)} = c_i^* - \delta,$$

for i=0,1,.....N-1, where* δ *is a constant perturbation vector* $\boldsymbol{\delta} = \delta * \mathbf{1}$*, where **1** is k dimensional vector with all elements equal to 1. Superscripts indicate the iteration number.*

3. Iteration:

In this step of the algorithm the codebook obtained from the splitting algorithm in the previous step will be used to find the first partition set. Then the two equations (Equation 3.17 and Equation 3.18) obtained by using the NNC and CC conditions are used iteratively to reach at the optimum codebook and partition set that minimizes the total distortion. The iteration continues until the difference between the distortions of two consecutive iterations is below $\in > 0$ *(stopping criteria)*

3.1. Set the initial distortion $D^{(1)} = D^*$ *and the iteration counter j=1.*

3.2. Given j^{th} codebook $C^{(j)} = \{c_0^j, c_1^j, \ldots, c_{N^*-1}^j\}$, assign each training vector x_m to its corresponding encoding region to determine the j^{th} partition cell $P^{(j)} = \{S_i^{(j)}\}_{i=0}^{N^*-1}$ according to the rule given in Equation 3.19

$$\varepsilon^j(x_m) = arg\, min_{i \in I_{N^*}} \sum_{i=1}^{N-1} P_{Y|X}(j|i)\left\|x - c_j\right\|^2,\ m = 1,2,\ldots,M.$$

$where\ I_{N^*} = \{0,1,\ldots\ldots\ldots,N^* - 1\}.$

3.3. Compute the new codebook using Equation 3.17

$$c_{j+1} = \frac{\sum_{i=0}^{N-1} P_{Y|X(j|i)} \sum_{s_i} x}{\sum_{i=0}^{N-1} P_{Y|X}(j|i)|s_i|}$$

3.4. Compute the updated distortion from Equation 3.16:

$$D_{COVQ}(j) = \frac{1}{KM}\sum_{i=1}^{M}\sum_{j=0}^{N-1} P_{Y|X}(j|\varepsilon(x_i))\ ||x_i - c_j||^2$$

$if\ \frac{D^{(j-1)} - D^{(j)}}{D^{(j-1)}} > \in, go\ to\ step\ 3.2$

3.5. Set the final codebook for this iteration as:

$$c_i^* = c_i^j,\quad i = 1, 2, \ldots\ldots\ldots N^*$$

$$and\ set\ D^* = D^{(j)}.$$

Go to step 2 and repeat the splitting and iteration procedures until $N^* \geq N$.

3.8 Decoding of the Received Signal

Orthogonal space-time block combiner (DECODING? See the title.) is used to combine the signals received at the three antennas. Then the bit streams obtained from the combiner is used to find the index of the region it represented. Then this index of region is used to find the code the transmitted signal corresponds to from the codebook. Linear Combiner: The combiner used is OSTBC linear combiner which is a variation of the classical maximum ratio combining (MRC) [4] in which the signals to be combined (the space-time soft-decoded signals) have different noise variances. If we let $\delta_i^j = \frac{r_i^j}{g'Y_j}$, the linear combiner output will be

$$\delta_i = \sum_{j=1}^{L} \alpha_j \frac{r_i^j}{g'Y_j} = \sum_{j=1}^{L} \alpha_j (c_i + n_i^j) \ldots\ldots\ldots \quad (3.29)$$

Where c_i is the transmitted symbol, α_j's are the weighting coefficients at each receiver antenna

The distribution of n_i^j, the noise component in the above equation, is Gaussian with zero mean and variance $\frac{g}{g'^2 Y_j}$ [12]. In linear combining, the objective is to choose the weights $\{\alpha_j\}_{j=1}^{L}$ so that the signal to noise ratio is maximized by enforcing $\sum_{j=1}^{L} \alpha_j = 1$.

Chapter 4

4 Results and Conclusion

As it is presented in the previous c~~C~~hapters one of the JSCC techniques has been chosen and an algorithm has been developed. In this Chapter the performance of the system model will be presented. The specific technique chosen for implementation is the one which optimizes the source coding based on the channel transition probability. This technique is known as Joint Source Channel Coding for Channel Optimized Vector Quantization (JSCC for COVQ).

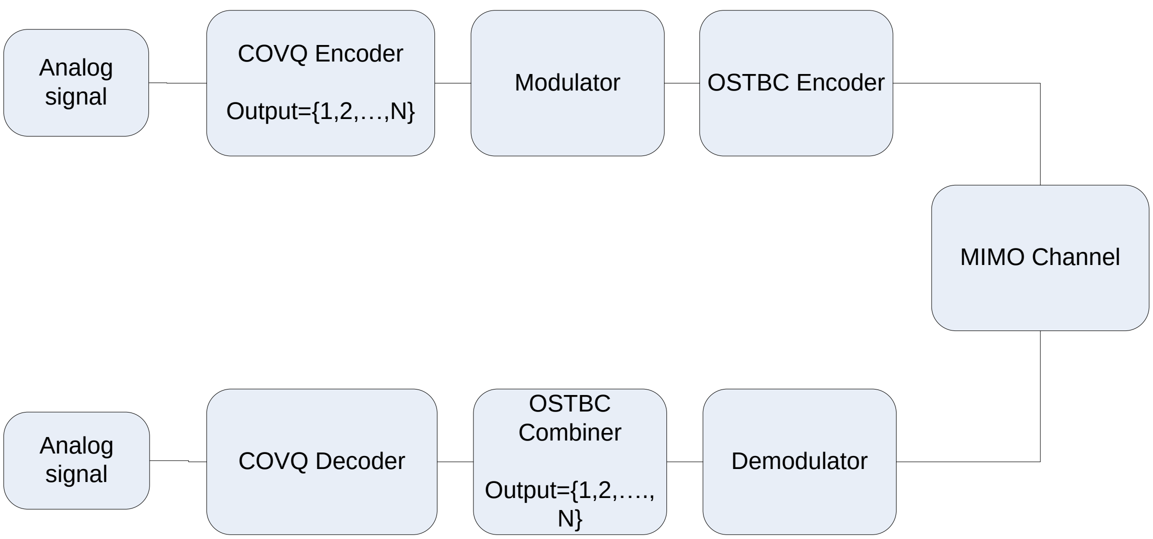

Figure 4.1 Communication system model with which the system is developed

Using the above system model various results are obtained that show the performance of the system as a function of system parameters. And results that compare the system with other communication systems are also presented.

Both quantitative and qualitative methods are used to study the performance of the communication system. The quantitative measures are for SDR vs SNR, and BER vs SNR results. The qualitative results are for the comparison of the images. For the comparison of the images different channel models and different encoding and decoding methods are considered.

The performance measures are ~~signal to distortion ratio~~ SDR and bit error rate versus signal to noise ratio. The simulation is done on a software tool called MATLAB R2010B.

The simulation parameters are summarized in the following two tables.

Rayleigh fading MIMO Channel with COVQ				
Quantization Level	64	Sampling Time Symbols/second???	10^{-4}	Code Rate for the OSTBC Fraction of Data bits
	128		10^{-5}	0.5
	256		10^{-6}	0.75

Table 4.1 Simulation parameters for Rayleigh flat fading MIMO channel with COVQ

The above table is simulation parameters used for comparison of SDR and BER within the model of Rayleigh Fading MIMO channel with COVQ.

Comparison of JSCC for COVQ On MIMO with other		
1	Comparison with Rayleigh fading channel	The comparison is done with respect to both SDR and BER.
2	Comparison with Tandem system (vector quantizer as source coding and convolutional codes as channel coding are used in this case)	

Table 4.2 Parameters for comparison the system model with other systems

Transmit correlation matrix $T_x = \begin{bmatrix} 1 & 0.17 \\ 0.17 & 1 \end{bmatrix}$

Recieve correlation matrix $R_x = \begin{bmatrix} 1 & 0.17 & 0.11 \\ 0.17 & 1 & 0.14 \\ 0.11 & 0.14 & 1 \end{bmatrix}$

Where are these values obtained from? A reference? Where $T_x(i,j)$ for i, $j=1, 2$ represents the degree of dependence of the signals transmitted from the i^{th} and the j^{th} transmit antenna. And $R_x(i.j)$ for $i, j = 1, 2, 3$ represents the degree of dependence of the signals received at the i^{th} and the j^{th} receive antenna.

The above correlation matrices are assumed based on from a study on "MIMO antenna performance for handsets and data terminals" made by Ethertronics (antenna developer and manufacturer company) [24].

For all of the simulations a Doppler frequency shift of $f_D = 40\text{Hz}$ is assumed. Why? Type of Environment??

Results for both signal to distortion ratio and bit error rate are simulated at SNR values ranging from 0dB usually up to 20dB and sometimes lesser.

Through the results that are presented below it is demonstrated that using Joint Source Channel Coding (JSCC) scheme called Channel Optimized Vector Quantization (COVQ) over perform the Shannon separate coding scheme. The performances are demonstrated at different levels of quantization and different rates with an objective to look for the optimal values.

In the system model the channel has the following characteristics

- Number of paths of the multipath Rayleigh fading channel is 2. The delay of the second path from the first is 10^-6.
- Modulation type is baseband BPSK modulation/demodulation.
- MIMO: for the MIMO channel 2 transmitter and 3 receiver antennas are used. The antennas at both the transmitter and receiver side are assumed to be not fully separated in space.
- And for the MIMO channel, Orthogonal Space Time Block Code encoder and combiner is used.
- Two code rates (fraction of data bits in a symbol), 1/2 and ¾ MIMO are also compared.

The type of channel coding used?

System performance (signal to distortion ratio) comparison at different quantization levels

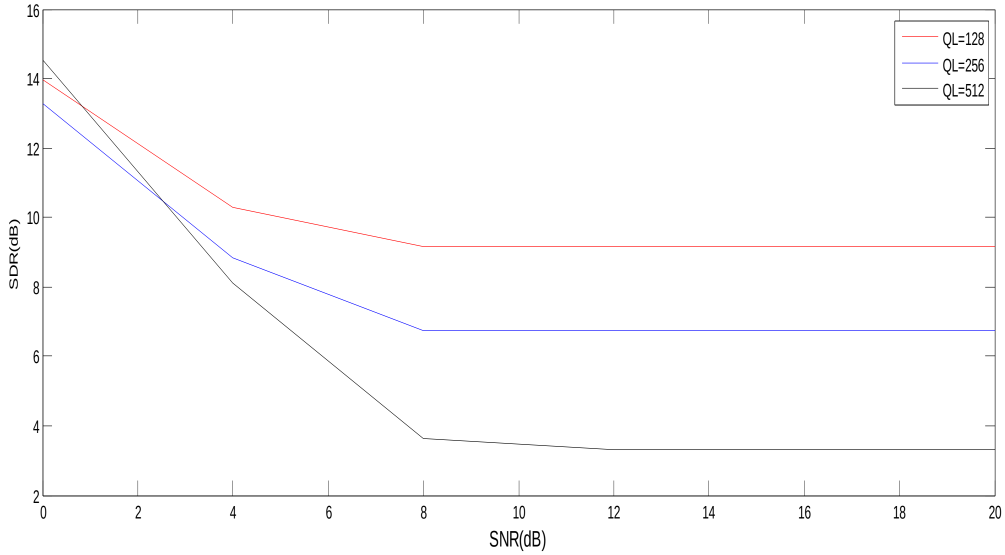

Figure 4.2 Distortion of the signal with increasing value of the SNR at different quantization level.

As the above result shows the distortion increases with decreasing quantization?????? level specially at higher SNR values and has a significant decrease of the distortion level with increasing the power of the transmitted signal. At higher values of the signal to noise ratio increasing the power has no use.

Performance of the channel optimized vector quantizer at different quantization levels on the bit error rate.

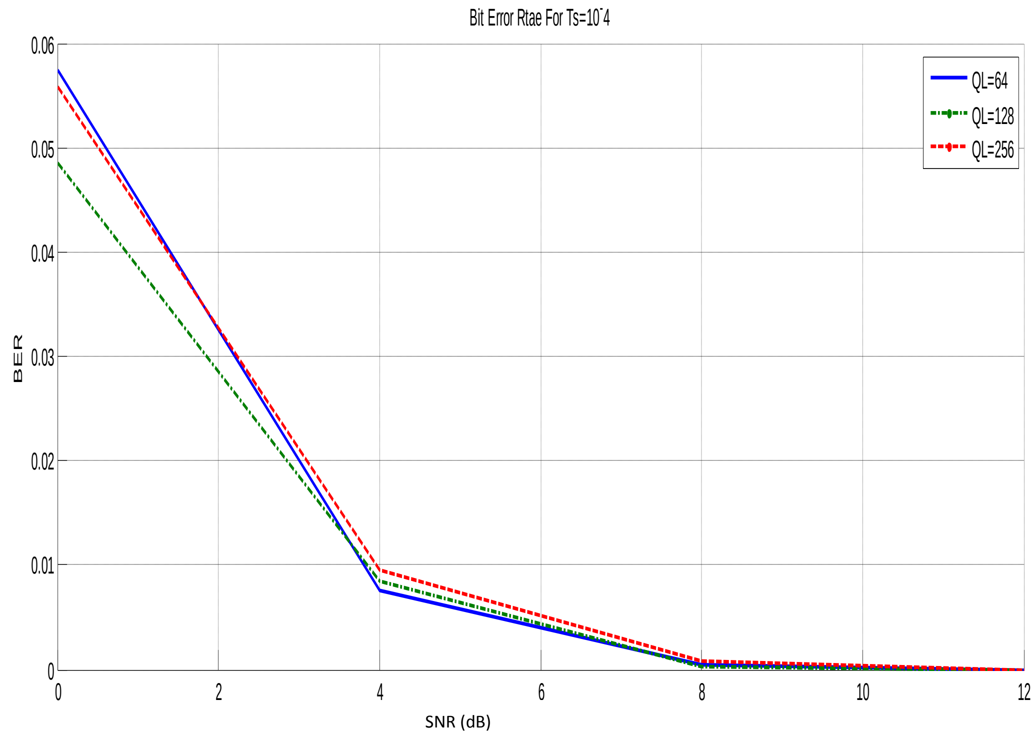

Figure 4.3 BER of COVQ at different quantization levels with sampling time=10^-4

Again with increasing the signal power, the performance of the system (Channel Optimized Vector Quantizer encoder with MIMO channel) measured in a bit error rate increases. At lower SNR values higher quantization level gives a better quality signal in relative to lower quantization level. However, this improvement is not that much significant as compared to the complexity increment resulted from increasing the quantization level. This result is obtained at a sampling rate of 10^-4.

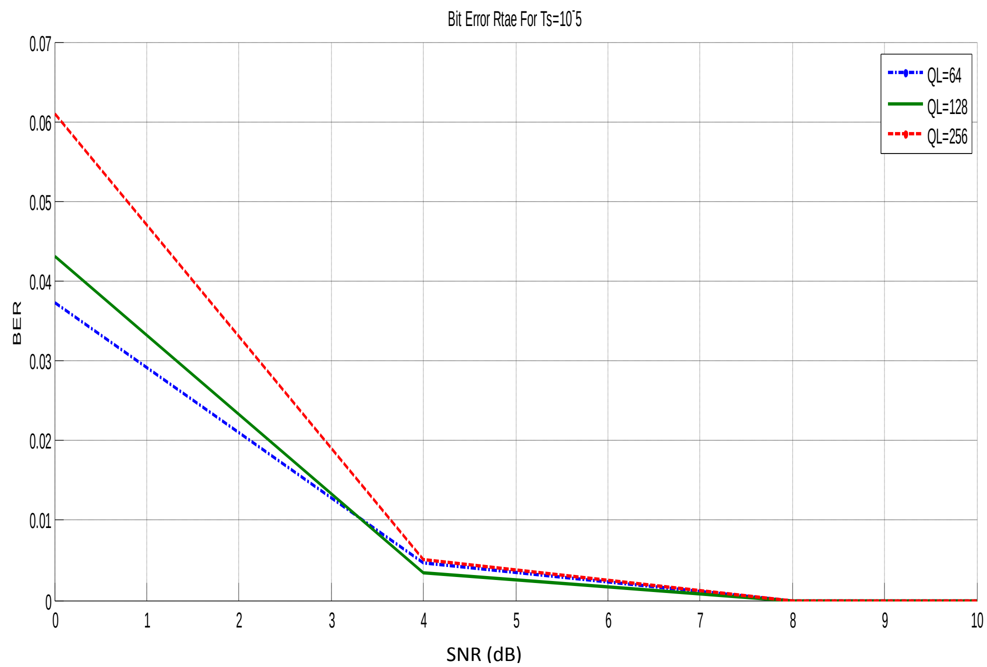

Figure 4.4 BER of COVQ at different quantization levels with sampling time=10^-5

From Figure 4.4, it can be observed that the same explanation given for Figure 4.3 holds. But the performance gap for different quantization levels at lower signal to noise ratio is more visible. That is due to the reason that in a noisier channel (quantization distortion) increasing the rate has an adverse effect. However, at a higher signal to noise ratio values the performance is almost the same at both sampling rates. The sampling rate for the result given in Figure 4.4 is 10^-5.

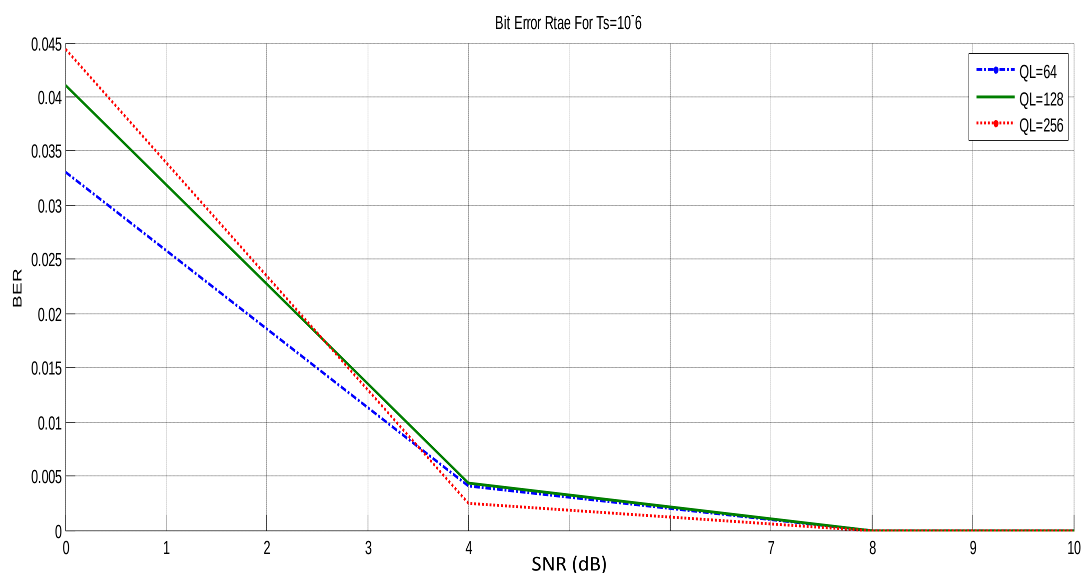

Figure 4.5 BER of COVQ at different quantization levels with sampling time, 10^-6

The same simulation run as for the above two results but at a sampling rate of 10^-6 is displayed in Figure 4.5. Not different from the above result it shows that we must decrease the data rate for lower values of quantization for a given SNR value.

From the BER versus SNR values displayed in Figure 4.3, Figure 4.4 and Figure 4.5 increasing the quantization level does not pay back the complexity introduced by increasing the quantization level. For instance in Figure 4.5 at 2dB SNR value the reduction in BER obtained by increasing the quantization level from 64 to 256 is only 0.5%. The difference is even lesser for smaller sampling rates as it can be observed from Figures 4.3 and 4.4. This situation is more pronounced at higher SNR values. The error rate depends on the channel conditions and the channel capacity. So the error rate increases as the data rate is increased.

If only the results related to BER are considered increasing the quantization level has no significant return. However it can be seen from the results of SDR versus SNR that increasing the quantization level has significant differences.

In general from the above Figures we can deduce that the system goes to higher performance with respect to its BER values at any of the quantization levels and rates for higher signal to noise ratio levels. But at lower values of the SNR for higher values of the quantization levels it is good to increase the rate for increased performance as the result in the above Figures depicts.

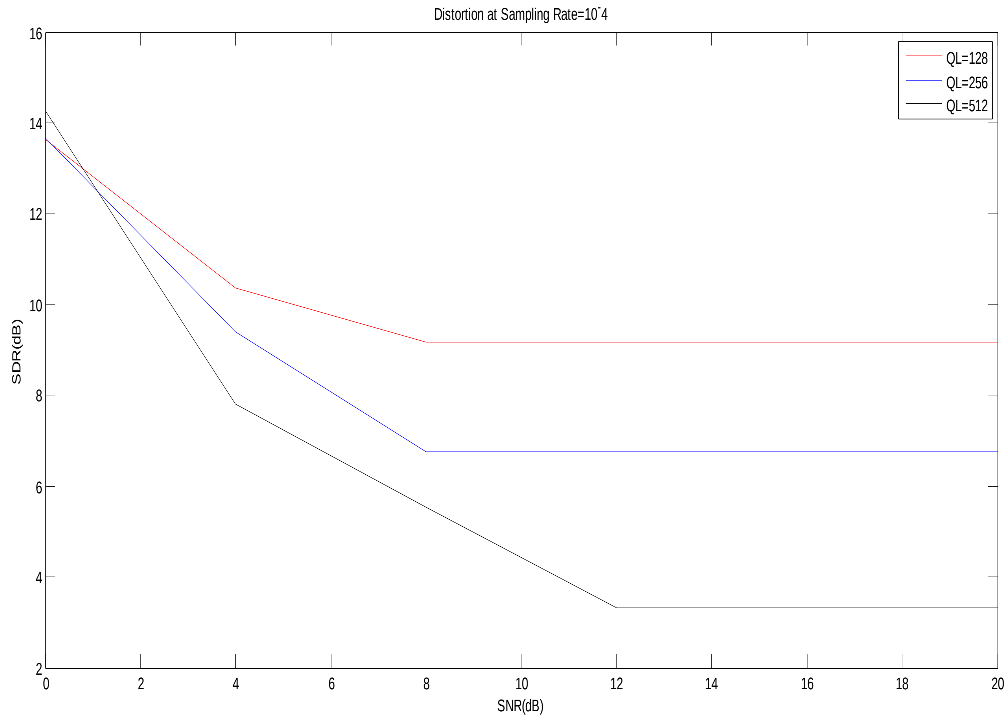

Figure 4.6 Distortion ratio of COVQ at different quantization levels with sampling time=10^-4

Similar to the BER performance case the distortion of the system (COVQ encoding and MIMO channel) has lesser distortion at higher SNR values for any level of quantization. But it performs well for higher quantization levels in general at higher signal to noise ratio and viseversa. The distortion at any level seems fixed for SNR values greatr than 10, however it does not converge to zero as in the BER case since it inherent behavior of a quantizer. The sampling rate for the above Figure is 10^-4.

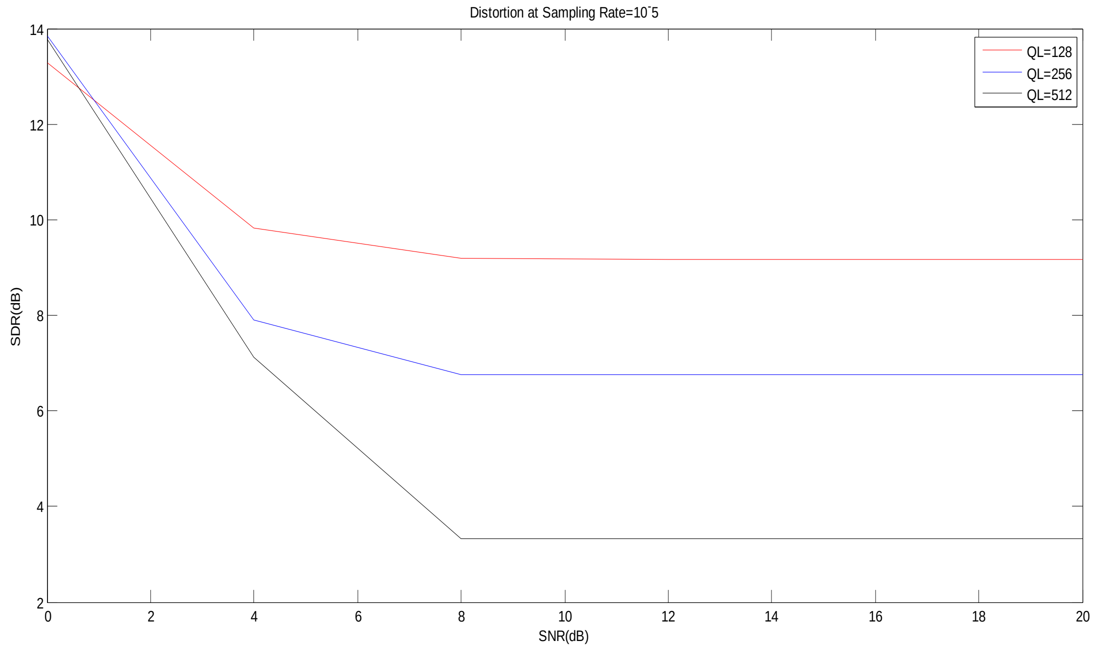

Figure 4.7 Distortion ratio of COVQ at different quantization levels with sampling time=10^-5

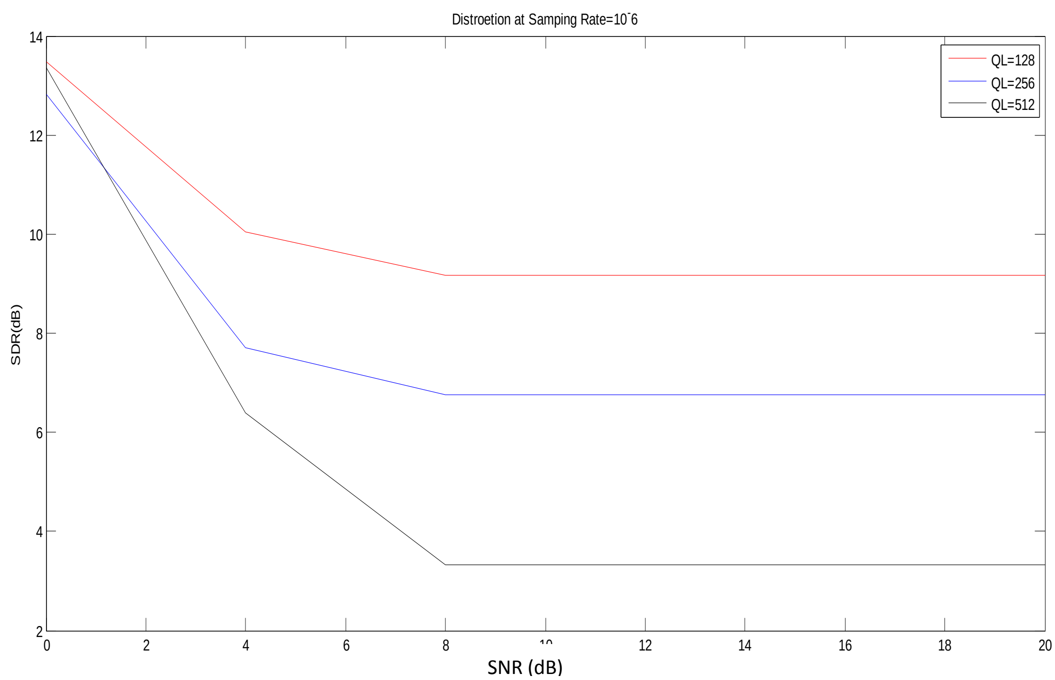

Figure 4.8 Distortion ratio of COVQ at different quantization levels with sampling time=10^-6

The results in Figure 4.7 and Figure 4.8 are simulated in similar manner with the result in Figure 4.6, with a different sampling rate (10^-5 and 10^-6 respectively for Figure and Figure 4.8). In the signal to distortion ratio case, for larger quantization level it is good to use smaller sampling rate for better quality. This is only true when the channel is not in a good condition. For a channel which is less noisy the performance can be increased by increasing both the quantization level and the sampling rate, i.e. when power is not a constraint. Comparably the gap of the distortion curves at different quantization levels increases with increasing the sampling (data rate). Similar to the result in Figure4.6, the results in Figure 4.7 and Figure 4.8 have almost constant values at an SNR values greater than the one at Figure 4.4. Opposite to the bit error rate performance, the signal to distortion ratio at lower rates has good performance values for lager quantization levels.

The results displayed above for BER and SDR performance of JSCC for COVQ have consistent results for most of the SNR values. However, these results are seen to be inconsistent in some points of all the plots especially at low SNR values. Through simulation of these results obtained for different images I have seen that this point of inconsistency gets closer to zero as the size of the transmitted data increases. Therefore I suppose that this inconsistency will be omitted as the size of the data is increased.

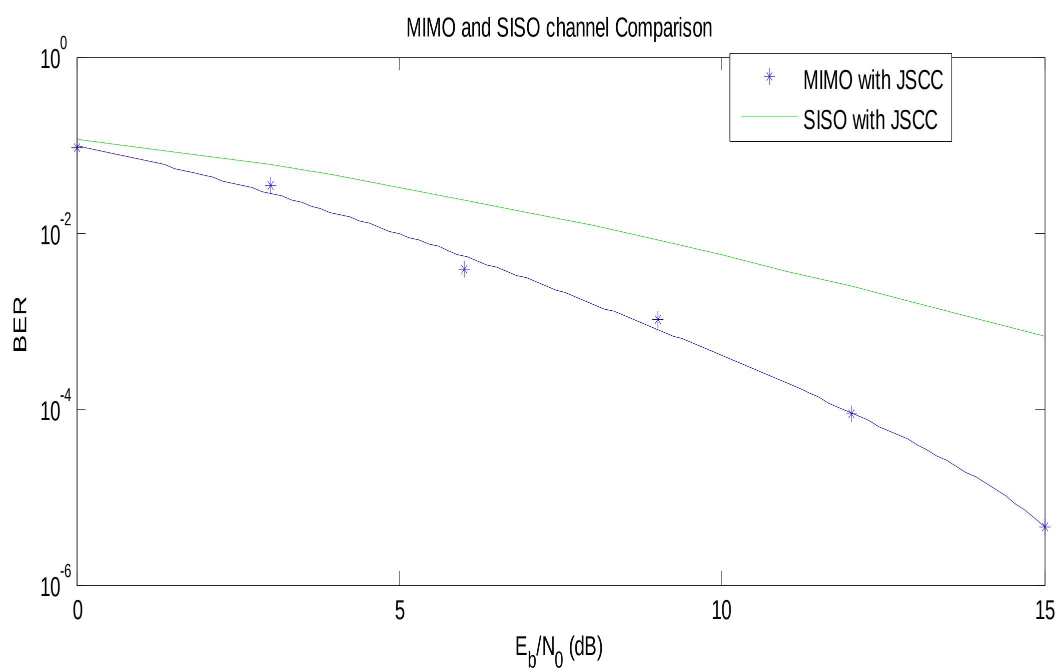

Figure 4.9 Performance Comparison of JSCC applied on MIMO and SISO.

'In the previous result for the above figure I made the sampling time for the mimo and siso different the result put here for the same sampling time.'

The above Figure depicts the obvious superior quality (by how much?) of the multiple input multiple output channel over the Rayleigh channel which is measured by the data bit error rate values. Increasing the number of transmit or receive antennas results in a SNR gain due to space diversity.

BTW: Figure 4.9 is the only MIMO related result in the entire thesis though MIMO stands bold in the title

As it is expected the performance of the two code rates are almost the same which is shown in Figure 4.10. This is due to the fact that the channel coding is inbuilt together with the source coding.

Therefore there is no need for to allocate bits for redundancy and decrease the rate. This implies we can use the largest rate possible in the MIMO system for joint source channel coding without affecting the performance.

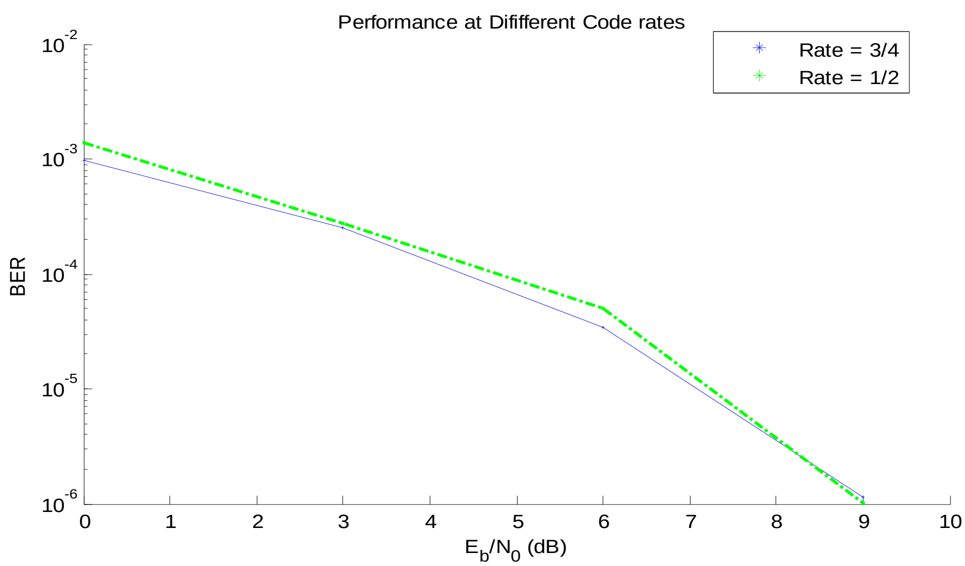

Figure 4.10 JSCC at different code rates

The following result shows the performance superiority of the JSCC scheme over the separate coding. The distortion on image is larger in case of tandem system than the JSCC scheme especially at higher lower SNR values.

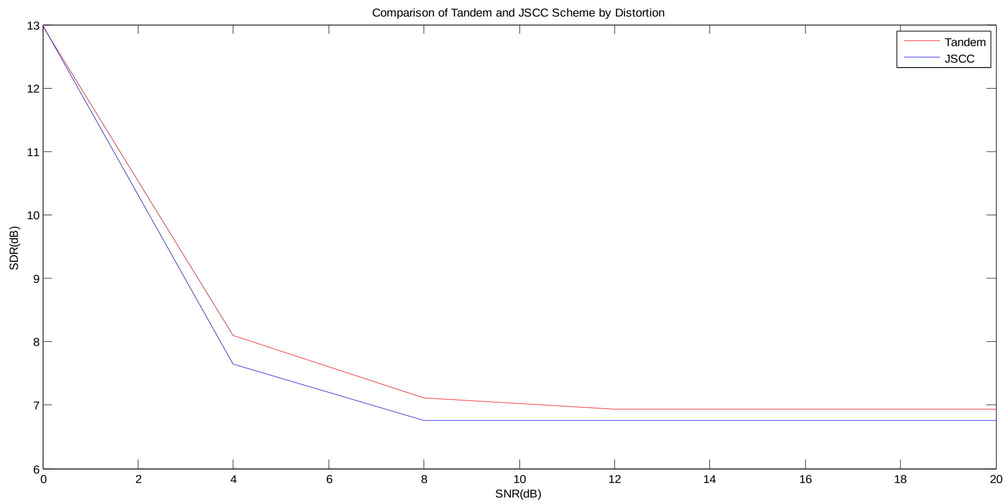

Figure 4.11 Distortion Comparison of Tandem System with JSCC System

As it can be seen from the result in Figure 4.11 joint source channel coding scheme outperforms the Shannon separate source coding and channel coding scheme which is measured by the signal too distortion ratio at different SNR values. However the distortion gap decreases with increasing channel quality. Even finally they both converge to a constant distortion level for stable channel condition. This is due the best quality of the MIMO channel.

The following results are made part of the paper to show the performance differences of the joint source channel coding from that of the separate coding qualitatively and also the performance differences of the MIMO channel and the single Rayleigh channel used with JSCC encoding.

The images shown in Figure 4.12 are the transmitted image (used in the simulation) and its copy as received through the Rayleigh(SISO) and the MIMO channel. These Figures are displayed to see the difference between the Rayleigh and the MIMO channel quality. As it is expected the MIMO channel has better quality.

Figure 4.13 and Figure 4.14 are the copy of the transmitted image as received through the MIMO channel. The difference is in their coding scheme: the result in Figure 4.13 is encoded using Shannon's separate coding theory and the result in Figure 4.14 is encoded using JSCC for channel optimized vector quantization encoding.

So these Figures are used for the comparison of the quality of the Figures in the two encoding cases. The image transmitted using the JSCC for COVQ scheme is better than the tandem scheme. For example at SNR value of 10, there is a distortion difference of 0.5 per pixel on average.

(a) Transmitted image

(b) Received image through MIMO channel

(c) Received image through Rayleigh channel

Figure 4.12 Image: a. transmitted, b. received through MIMO channel, c. received through Rayleigh channel

Figure 4.13 Image received from a MIMO channel which is sent using COVQ encoding

Figure 4.14 Image received from a MIMO channel which is sent using separate encoding

Chapter 5

5 Conclusions and Recommendations

5.1 Conclusions

The goal of this thesis is to design????? COVQ and implement it on a MIMO channel so that the performance of the communication system increases in terms of the signal-to-distortion-ratio and bit error rate, while reducing or keeping the complexity as low as possible and increasing data rate.

First of all, motivated by the data high compression rate of a vector quantization source coding, we use an algorithm that uses the channel transition probabilities to optimize the source coding based on the channel conditions before transmission. The algorithm involves designing a VQ initially using the splitting method and then use simulated annealing algorithm to optimize the index assignment for choosing an initial code.

Due to the fact that the encoder has done source coding optimized to the channel conditions there is no need to use the channel coder separately and in effect this decreases the delay.

As it has been seen from the results given in Chapter four the performance of the channel optimized vector quantizer is acceptable without the use of any other source coder. Since the channel coding and the source coding are done at the same time (no need for extra redundant bits for channel coding) it was possible to use the MIMO at its highest possible rate.

All of the achieved performance gains increase notably with the increase of quantization rate.

5.2 Recommendations

This paper dealt with reviewing different techniques of joint source channel coding/decoding, which are found to increase the performance of communication system by jointly optimizing the source and the channel coder/decoder. Later on a specific JSCC scheme called Channel Optimized Vector Quantization (COVQ) is chosen for detail analysis and it is designed for

MIMO channel. Then the performance improvement is observed through Signal to Distortion ratio (SDR) and Bit Error Rate (BER).

Ideas for future work:

- Other joint source channel coding schemes on MIMO.
- Use Channel state information at the receiver for power management.
- Make use of the soft information to calculate the probability rather than using the empirical probability.

6 References

[1] C. E. Shannon, "A mathematical theory of communication (Part 1)," Bell Syst. Tech. J.,1948.

[2] Sridhar Vembu Sergio, Verdu, Fellow, IEEE, and Yossef Steinberg, January, "The Source Channel Separation Theorem Revisited", IEEE Transactions on Information Theory, 1995

[3] Jongtae Lim and David L. Neuhoff, Joint And Tandem Source-Channel Coding with Complexity And delay Constraints, June 2001.

[4] John Proakis, "Digital communication", Mc Graw Hill, 4th edition.

[5] Thomas Cover, Elements Of Information theory, John Wiley & Sons, Inc, 1991.

[6] Emre Telatar,Capacity of Multi-antenna Gaussian Channel,1995.

[7] Erik G.Larsson and Petre Stoica, Space-Time Block Coding For Wireless Communication, Cambridge University Press, 2003.

[8] TU Glu Faung, LIU JianJun,et al, Studies and advances on joint source-channel encoding/decoding techniques in flow media Communications, SCIENCE CHINA Information Sciences Vol. 53 No. 1: 1–17, January 2010.

[9] Hamidreza Ebrahimzadeh Saffar, Channel Optimized Vector Quantization:Iterative Design Algorithms, Queen's University Kingston, Ontario, Canada Augest 2008.

[10] P. Fra¨nti1 and J. Kivijarvi, Randomized Local Search Algorithm For The Clustering Algorithm, Springer-Verlag London Limited, Chinese Journal of Electronics Vol.20, No.1, 2000.

[11] TU Guofang and LIU JianJum…et al, Joint Source-Channel En/Decoding Based on a New Symbol-Level Joint Trellis, January 2011.

[12] Firouz Behnamfar, Single And Multiple Antenna Communication Systems: Performance Analysis And Joint Source-Channel Coding, Queen's University Kingston, Ontario, Canada, September 2004.

[13] Wei Xiang, Joint Source-Channel Coding for Image Transmission and Related Topics, University of South Australia, December 2003.

[14] Dimitris Bertsimas and John Tsitsiklis,Simulated Annealing, Statistical Science Vol.8,No.1,10-15, 1993.

[15] X. Zhu and R. D. Murch, "Performance analysis of maximum likelihood detection in a MIMO antenna system," IEEE Trans. Commun., vol. 50, pp. 187-191, Feb. 2002.

[16] M. Ruf and J. Modestino, \Rate-distortion performance for joint source and channel coding of Images," in Proc. IEEE Int. Conf. Image Proc., pp. 77-80, 1995.

[17] Igor V. Kozintsev, "Signal processing for joint source channel coding of digital images", 2004.

[18] Xuqi Zhu, Yu Liu and Lin Zhang "Distribute Joint Source-Channel Coding in Wireless Sensor Networks" , 2009.

[19] Hamid Jafarkhani, Paschalis Ligdas, and Nariman Farvardin, "ADAPTIVE RATE ALLOCATION IN A JOINT SOURCE/CHANNEL CODING FRAMEWORK FOR WIRELESS CHANNELS", 1996.

[20] KTH LCN Technical Report, Resource Allocation for Wireless Video Streaming: Bit-Rate Adaptation and Packet Scheduling May 2006.

[21] Antonio Pascual Iserte, Channel State Information and Joint Transmitter Receiver Design in Multi Antenna Systems, December 2004.

[22] Y. Linde, A. Buzo and R. M. Gray "An algorithm for vector quantizder design,"IEEE Trans. Inform. Theory, vol. 39, pp. 868–876, Jan. 1980.

[23] H. Kumazawa, M. Kasahara and T. Namekawa, "A construction of vector quantizers for noisy channels," Electronics Eng. in Japan, vol. 67-B, January 1984.

[24] MIMO Antenna Performance for Handsets and Data Terminals, Ethertronics, November 2008.

7 Appendix

Matlab Code for the system:

```
%this function initlbg finds the index for each input value.
%its input arguments are the discrete signal and the level
%and the outputs are the index of the regions and the codebook for the
%training signalthe function uses centroid and voronoi functions
%iteratevely to reach at an optimum point

function [c0 indr]= initlbg(x,n)
delta=.01;
[m,t]=size(x);
c0=x*(ones(t,1))/t;%initial codebook for the first iteration
d0=(x-c0(:,1)*ones(1,t));%initial distortion
d1=norm1(d0)/t;
indr=ones(1,t);
jj=1;
vv=zeros(m,n);
while(jj<n)
   for kk=1:jj
       vv(:,2*kk-1)=c0(:,kk)-delta;
       vv(:,2*kk)=c0(:,kk)+delta;
   end
   vv=vv(:,1:2*jj);
   indr= voronoi2(x,vv);
   jj=jj*2;
   [c0 d2]=centroid(x,jj,indr);
   if((d1-d2)/d1>delta)
       while ((d1-d2)/d1>delta)
        % vv=vv(:,1:2*jj);
        d1=d2;
        indr= voronoi2(x,vv);
        %jj=jj*2;
        [c0 d2]=centroid(x,jj,indr);

       end
   end
   d1=d2;
end
return
>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
>>>>>>>>>>



%this function finds the regions of data vectors and it returns the index
%of the corresponding region. the input arguments are the training vector
%and code book

function indr2=voronoi2(x,c)

[m,t]=size(x);
[m,n]=size(c);
```

```
indr2=zeros(1,t);
for i=1:t
   x1=norm1(x(:,i)-c(:,1));
   indr2(i)=1;
   for j=2:n
      d=norm1(x(:,i)-c(:,j));
      if(d<=x1)
         x1=d;
         indr2(i)=j;
      end
   end
end
return

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
>>>>>>>>>>% this function finds the average value of traing vectors in aregion and
% the total distortion

function [c,e]=centroid(x,n,indr)
[m,t]=size(x);
c=zeros(m,n);
e=0;
for jj=1:n
   indrep=find(indr==jj);
   xjj=x(:,indrep);
   [m lxjj]=size(xjj);
   if lxjj~=0
      c(:,jj)=(xjj*ones(lxjj,1))/lxjj;
      d1=(c(:,jj)*ones(1,lxjj)-xjj);
      e=e+norm1(d1);
   end
end
e=e/t;
return

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

% this function finds the average value of traing vectors in aregion and
% the total distortion

function [c,e]=centroid(x,n,indr)
[m,t]=size(x);
c=zeros(m,n);
e=0;
for jj=1:n
   indrep=find(indr==jj);
   xjj=x(:,indrep);
   [m lxjj]=size(xjj);
   if lxjj~=0
      c(:,jj)=(xjj*ones(lxjj,1))/lxjj;
      d1=(c(:,jj)*ones(1,lxjj)-xjj);
      e=e+norm1(d1);
   end
end
e=e/t;
```

```
return

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

%this image data is used in trainig the quantizer
%the function returns a number in double format of the image

function a=image1()
rgb_img = imread('ngc6543a.jpg'); % Load the image
image(rgb_img) % Display the RGB image
I = .2989*rgb_img(:,:,1)...
+.5870*rgb_img(:,:,2)...
+.1140*rgb_img(:,:,3);
%a=im2col(I,[2 2],'distinct');
a=double(I);
return

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
>>>>>>>>>>>

% this function evaluates the distortion on the image due to the channel
% transition probabilities and it is used to choose the random index
% assignment in simmulated annealing algorithm

function [sum]=chan_dist(c0,pi,prob)
[k n]=size(c0);
sum=0;
for i=1:n
    for j=1:n
        sum=sum+prob(j,i)*pi(i)*norm1(c0(i)-c0(j));
    end
end
sum=sum/k;
return


>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
>>>>>>>>>>>

% the following code is written to choose the best index assignment for the
% initial code to minimize the distortion in the channel
% it takes as an input the intial code obtained from the simple vector
% quantizzer and its level of quantization

function [c0] = sim_anneal(c0,n)
distn=0.3;
del_d=.5;
T0=10;Tf=.00025;alpha=.87;Ncut=200;

while (T0>=Tf&&del_d==0)
    y=randi(n,[1,n]);
    for i=1:n
        c0(y(i))=c0(i);
    end
```

```
    d0=chan_dist(c0,pi,pro);
    while (Ncut>=1)
        y1=randi(n,[1,n]);
          for i=1:n
              c0(y(i))=c0(i);
              ends
              d1=chan_dist(c0,pi,prob);
              del_d=d1-d0;
          if del_d<=0
              y=y1;
          end
          if del_d>0;
              p=exp(-del_d/T0);
              x=randsrc(1,1,[[-1 1]; [p 1-p]]);
              if x==-1
                 y=y1;
              end
          end
    Ncut=Ncut-1;
    end
    T0=alpha*T0;
    end
end
return
```

>>

```
%this is the 3x2 MIMO channel
% Define system parameters
 function  [receivedData col2 errors lendata ]= mimochan2(numTx,numRx,SNR,data,col1,Ts)
    %numTx = 2; numRx = 3;
    Ts = 1e-6;
    maxDopp = 30;
    [numBits col]= size(data);
    col2=col1;
    %SNR = 10;

    % Create modulator and encoder system objects
    hMod = comm.QPSKModulator('BitInput',true,...
                'SymbolMapping','Gray');
    hDemod = comm.QPSKDemodulator('SymbolMapping','Gray',...
                'BitOutput', true);
    hOSTBCEnc = comm.OSTBCEncoder('NumTransmitAntennas', numTx);
    hOSTBCComb = comm.OSTBCCombiner('NumTransmitAntennas', numTx,...
                'NumReceiveAntennas', numRx);

    % Create MIMO channel object
    chan = mimochan(numTx, numRx, Ts, maxDopp);
    chan.NormalizePathGains = 1;
    chan.StorePathGains = 1;
    chan.ResetBeforeFiltering = 1;

    % Generate data
```

```
  % data = randi([0 1],numBits,1);

   % Modulate data
   modData = step(hMod, data);

   % Encode modulated data using OSTBC and transmit through Rayleigh and AWGN channel
   encData = step(hOSTBCEnc, modData);

   % Transmit signal
   rxMIMO = filter(chan, encData);
   receivedSignal = awgn(rxMIMO,SNR,'measured');

   % Decode and demodulate received signal
   decData = step(hOSTBCComb, receivedSignal,...
                reshape(chan.PathGains, numBits/2, 2,3));
   receivedData = step(hDemod, decData);

   % Compute number of bit errors in received data
   errors = biterr(data, receivedData);
           lendata= length(data);
 return
```

>>

```
%the following function find x the index transition probabilities of the 2x3
%MIMO channel empirically

function [prob error]=condition_prob(snr,n)
prob=zeros(n,n);
indr=(1:n);
error=0;

   for m=1:100;
        [mod col]=modu(indr);
        [mout col2 errors lendata]=mimochan2(2,3,snr,mod,col);
        error=errors+error;
        dout=demog(mout,col2);
        dout=round(dout);
        dout=dout';
        size(dout);
        dout(1,find(dout>n))=n;
        dout(1,find(dout<1))=1;
        for k=1:n
          for j=1:n
             if(dout(1,k)==indr(1,j))
                prob(k,j)=prob(k,j)+1;
             end
          end
        end

   end
```

```
  %prob(:,:)=prob(:,:)';
 error=error/(100*lendata);
 prob(:,:)=prob(:,:)/100;
 %errorA((i/3)+1,1)=error;
 %SNR=(0:3:24);
 %plot(SNR,errorA);
return
```

>>

```
%this function rearranges the image data into a binary sequence to be able
%to transmitted over a channel

function [mod col]=modu(indr)
x=de2bi(indr);
[row  col]=size(x);

for k=1:col

       mod(((row*(k-1))+1):row*k,1)=x(:,k);
end

return
```

>>

```
% this function undo the rearrangement of the image data by 'modu' function
% into its original form

function img =demog(mimout,col2)
[row col]=size(mimout);
row=row/col2;
img=zeros(row,col2);
for k=1:col2
   x(1:row,k)=mimout(((row*(k-1))+1):row*k,1);
end
img=bi2de(x);
return
```

>>

```
% in this function the decoding algorithm is implemented. it simply maps
% the incoming index of region to its appropriate codevector from the code
% book

function img =decode_(indr,out1,n)
 indr=indr';
[x y]=size(indr);
%load code_book_;
```

```
[u v]=size(out1);
img = zeros(u ,y);
for i=1:y
    index=round(indr(i));
    if(index<=0)
        index=1;
    end
    if (index>256)
            index=256;
    end

    img(:,i)=out1(:,index);
end

return
```

>>>

```
%rayl is a function written to find the distortion and the bit error rate
%for a rayleigh channel
%it take as input sampling time, doppler frequency the difference between
%the two conscutive SNRs and the final SNR, and quantization level

function [disn err]=rayl(ts,fd,jump,final,n)
%function [disn err]=rayl(ts,fd,k,n)
disn=zeros(1,final/jump+1);

tau1=0;  tau2=10^-6;
err=zeros(1,6);
for l=1:100
    i=1;
    for k=0:jump:final
        oinput=image1();
        [row col]=size(oinput);
        [out1 out2]=initlbg(oinput,n);
        [mod col]=modu(out2);
        mod1=pskmod(mod,2);
        chan1 = rayleighchan(ts,fd,[tau1  tau2],[15 10]);
        rout(:,1) = filter(chan1,mod1(:,1));
        nrout=awgn(rout,k,'measured');
        dmod1=pskdemod(nrout,2);
        dout=demog(dmod1,col);
        oimg=decode_(dout,out1,n);
        err1(i)=r_err(mod1,dmod1);
        size(err1);
        disn(i)=(norm1(oimg-oinput))/(row*col);
        %imshow(int8(oimg));
        i=i+1;
    end
        err(1,:)=err(1,:)+err1(1,:);
end
plot(0:jump:final,err(1,:)/100);
return
```

>>

```
%this function plots the distortion and bit error rate vs SNR at different
%quantzation levels for the JSCC scheme
function Ber_(jump,final,qun_ini,qun_final,Ts)
n=qun_ini;
i=0;
while qun_final >= n*2
   if i==0
       n=qun_ini;
   else
       n=n*2;
   end
[u v]=img_disn1(jump,final,n,Ts);
[row col]=size(u);

sizeu=size(u)

DIST(i+1,1:col)=u;
ERR(i+1,1:col)=v;
N(i+1)=n;
i=i+1;
end

 plot(0:jump:final,ERR(1,:),0:jump:final,ERR(2,:),0:jump:final,ERR(3,:));
 legend('N(1)','N(2)','N(3)','N(4)');

 figure

 plot(0:jump:final,DIST(1,:),0:jump:final,DIST(2,:),0:jump:final,DIST(3,:));
 legend('N(1)','N(2)','N(3)','N(4)');

return
```

>>

```
% this function plots the image distortion for both tandem system and jscc
% scheme over 2x3 mimo channel

function imge_dst(xn)

for snr=0:5

    snr1=4*snr;
[oinput oimg]=test(n,snr1);
oimg1=test1(n,snr1);
```

```
imshow(int8(oinput));
[row col]=size(oinput);

 disn1(snr+1)=sum(sum(abs(oimg-oinput)))/(row*col);
 disn2(snr+1)=sum(sum(abs(oimg1-oinput)))/(row*col);

end

plot(0:4:20,disn1,'r');
hold on
plot(0:4:20,disn2,'b');

return
```

>>>

```
% this is a test function for the tandem system using convolutional channel
% coding

function [oinput oimg]=test(n,snr)
oinput=image1();
[out1 out2]=initlbg(oinput,n);
sizeofout1=size(out1);
[mod col]=modu(out2);
[mout col2]=mimochan2(2,3,snr,mod,col);
dout=demog(mout,col2);
oimg=decode_(dout,out1);
return
```

>>>

```
% this is a test function for Channel Optimized Vector Quantization, JSCC
% scheme

function [oimg1]= test1(n,snr)
oinput=image1();

[prob err] =condition_prob(20,n);
probab=size(prob);
[out1 out2]=initlbg(oinput,n);
[a,b]=size(out2);
pi=zeros(1,n);
for i=1:n
        [b a]=size(find(out2==i));
        pi(i)=a;
end
pi(:,:)=pi(:,:)./n;
out1 = sim_anneal(out1,n);
out2 = voronoi2(oinput,out1);
%sizeofout1=size(out1)
[mod col]=modu(out2);
```

```
%mout=mimochan1(mod,.001,40,0,10^-6);
%dout=demog(mod,col);
%%oimg=decode_(dout,out1);
%figure
%imshow(int8(oimg));
[mout col2]=mimochan2(2,3,snr,mod,col);
dout=demog(mout,col2);
oimg1=decode_(dout,out1);
return
```

>>>