TRIBHUVAN UNIVERSITY

INSTITUTE OF ENGINEERING

Pulchowk Campus

Thesis No: 066/MSI/620

DESIGN OF LOW-DENSITY PARITY-CHECK CODES FOR AUTOMATIC REPEAT-REQUEST BLOCK-FADING CHANNEL

by

Utsab Pokharel

620/MSI/2066

Thesis

Submitted to

Masters of Science in Information and Communication Engineering

Department of Electronics and Computer Engineering

Nov-2013

DESIGN OF LOW-DENSITY PARITY-CHECK CODES FOR AUTOMATIC REPEAT-REQUEST BLOCK FADING CHANNEL

by

Utsab Pokharel

Thesis Supervisor

Sharad Kumar Ghimire

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Information and Communication Engineering

Department of Electronics and Computer Engineering

Institute of Engineering Pulchwok Campus

Tribhuvan University

Lalitpur, Nepal

Nov-2013

COPYRIGHT ©

Head
Department of Electronics and Computer Engineering
Institute Of Engineering
Pulchowk Campus
Lalitpur, Nepal

Recommendation

The undersigned certify that they have read and recommended to the Department of Electronics and Computer Engineering for acceptance, a thesis entitled **“Design of Low-Density Parity-Check Codes for Automatic Repeat-Request Block Fading Channel ”**, submitted by **Utsab Pokharel** in partial fulfillment of the requirement for the award of the degree of **“Master of Science in Information and Communication Engineering”**.

....

Supervisor: Sharad Kumar Ghimire

Program Coordinator

MSICE, IOE

…………………………………..

External: Dilli Ram Adhikari

Deputy Manager

Nepal Telecom

Departmental Acceptance

The thesis entitled **"Design of Low-Density Parity-Check Codes for Automatic Repeat-Request Block Fading Channel "**, submitted by **Utsab Pokharel** in partial fulfillment of the requirement for the award of the degree of **"Master of Science in Information and Communication Engineering"** has been accepted as a bonafide record of work independently carried out by him in the department.

……………………………………..

Dr. Arun Kumar Timalsina
Head
Department of Electronics and Computer Engineering,
Institute of Engineering,
Tribhuvan University,
Pulchowk, Nepal.

ACKNOWLEDGEMENT

Firstly, I would like to thank my supervisor, Sharad Kumar Ghimire for providing continual valuable encouragement, support and direction throughout thesis work. His advice, guidance and support are gratefully acknowledged. I would also like to thank him for reviewing the versions of this thesis.

I would like to show my greatest appreciation to Dr. Arun Kumar Timalsina, and Department of Electronics and Computer Engineering for providing the platform to me for showing venture in this thesis work.

I would also like to acknowledge the assistance of Prof. Dr. Shasidhar Ram Joshi, Dr. Subarna Shakya, Dr. Sanjeeb Prasad Panday, Dr. Surendra Shrestha, Dr. Nanda Bikram Adhikari, Jitendra Manandhar, Daya Sagar Baral, for supporting me.

Many thanks to Manohar Rajbhandari, Rajeev Prajapati, Ranjan Timsina, Prabhat Adhikari, Kailash Khadka, Prem Chandra Roy, Manish Mallick for their help and for many insightful and useful discussions at different point of time during this period.

This project is the result of dedication and encouragement of many individuals. My sincere and heartfelt appreciation goes to all of them.

And finally, I would like to express my gratitude for the support of my parents and family members during these studies.

ABSTRACT

Many practical communication systems such GSM, EDGE or system using orthogonal frequency division multiplexing (OFDM) can be modeled as the block fading channel. Communication over the block fading channel is particularly challenging because the transmitted codeword undergoes only a finite number of fading blocks making the channel non-ergodic. Hence there is a non-zero probability (called the outage probability) with which the communication is not possible. One practical way of reducing the outage probability is the use of adaptive technique such as ARQ systems. Subject to no delay constraints, outage can be avoided. In practice however, most system are subject to delay constraint imposed. Design of coding systems for ARQ system with delay constraint is considered in this thesis.

The Low-Density Parity-Check (LDPC) code is used as a mother code. The LDPC codes are specifically designed to achieve the maximum diversity in a fixed rate transmission system. A novel algorithm to design Root-Check LDPC codes based on Progressive Edge Growth (PEG) techniques for block-fading channels is proposed, which outperform codes constructed by the existing techniques .The concept of the check splitting to the LDPC code is extended to provide an incremental redundancy coding scheme specifically for the block fading channel in this thesis.

By construction, the proposed coding scheme achieves the outage probability, high level of the diversity and effectively adapts the transmission rate to the instantaneous channel conditions.

Keywords: Low-Density Parity-Check (LDPC), Automatic Repeat-Request (ARQ), Progressive Edge Growth (PEG), Root-Check, and Check Splitting.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF SYMBOLS

$\overline{N_{s_j}^{l}}.$	Neighborhood of variable node s_j at depth l
$\hat{\mathbf{m}}$	Estimate of the transmitted message
$N_c^{(l)}$	Number of check nodes at ARQ round l
$N_s^{(l)}$	Number of variable nodes at ARQ round l
C	Codebook
c_i	i^{th} check node
D_s	Variable node degree sequence
d_{si}	Degree of variable node s_i
Eb	Symbol Energy
E_{si}	Set containing all edges incident to variable node s_i
F	Number of Fading Blocks per ARQ round
$\boldsymbol{G}$	Generator matrix
$\boldsymbol{G_l}$	Generator matrix for l th ARQ round
$\boldsymbol{H}$	Parity check matrix
$\boldsymbol{H_l}$	Parity check matrix for l th ARQ round
$\boldsymbol{h}_{l,b}$	Fading gains in l^{th} ARQ round b^{th} block
I	Mutual Information
$\boldsymbol{I_K}$	Identity matrix of size $K \times K$
K	Number of information bits on N length codeword
L	maximum number of allowed ARQ rounds
$\boldsymbol{m}$	Information sequence on transmitter
M	Number of Check Equations on N length Codeword /Number of Check Nodes
N	Number of variable nodes/ Code Length
$N0$	Noise power
$P(x)$	Check nodes degree distribution
P_i	Number of check nodes of degree i
P_{out}	Outage probability

R	Code rate
R_l	Rate of the l th ARQ round
s_i	ith variable node
T_c	Coherent time
$\boldsymbol{U}$	Sets containing check nodes
$\boldsymbol{V}$	Sets containing variable nodes
$\boldsymbol{w}_{l,b}$	Elements of noise vector in l^{th} ARQ round b^{th} block
$\boldsymbol{x}$	Codeword vector
x_i	i-th Symbol on a codeword
$\boldsymbol{x}_l$	Vector of transmitted symbols in l^{th} ARQ round
$\boldsymbol{x}_{l,b}$	Transmitted symbol in l^{th} ARQ round b^{th} block
$\boldsymbol{y}_l$	Vector of received symbols in l^{th} ARQ round
$\boldsymbol{y}_{l,b}$	Received symbol in l^{th} ARQ round b^{th} block
δ	Diversity
$\Lambda(x)$	Variable node degree distribution
Λ_{i}	Number of variable nodes of degree i
ρ	Average signal-to-noise ratio
ϕ	Decoding function
Φ	Encoding function

ABBREVIATIONS

ARQ	Automatic Repeat Request
AWGN	Additive White Gaussian Noise
BPSK	Binary Phase Shift keying
CN	Check Node
EDGE	Enhanced Data GSM environment
EXIT	Extrinsic Information Transfer
FEC	Forward Error Correction
FER	Frame Error Rate
GSM	Global System of Mobile
ISI	Inter Symbol Interference
LDPC	Low Density Parity Check
MDS	Maximum Distance Separable
OFDM	Orthogonal frequency division multiplexing
PEG	Progressive Edge Growth
RA	Repeat Accumulate
SNR	Signal-to-Noise Ratio
VN	Variable Node

CHAPTER 1: INTRODUCTION

Communication over wireless system is challenging due to mobility of the devices and multipath propagation of the transmitted signal. The main challenge is to mitigate the multi path losses termed as fading. The fading can be in time domain, termed as time-selective fading and in frequency domain, frequency –selective fading.

The block fading channel was first introduced in [1]. It is a convenient model that describes the behavior of slow fading channels such as frequency hopping and *orthogonal frequency division multiplexing* (OFDM) systems. Coding over block fading channels is much different from coding over AWGN channels. There is a non zero probability that reliable communication is not possible. The main reason for this is that a codeword is transmitted only on a finite number of fading blocks.

In a flat fading channel, the fading coefficients remain constant for the duration of the block and vary between the blocks according to some specific distribution. The fundamental limit of the block fading channel is its *outage probability*, the probability that the channel cannot support its actual data-rate that is the word error rate is lower bounded by the outage probability [2].

1.1 Background

The noisy channel coding theorem, first presented by Shannon in 1948 [3], states that for every communications channel, however noisy it may be, there exists a maximum, in most cases positive, rate, denoted R, at which information can be transmitted with vanishingly small probability of error. This is in contrast to what was believed until then, namely that the information rate must tend to zero in order for the probability of error to also tend to zero. The maximum rate, at which reliable communication is achievable, is called the *channel capacity*.

There are various notions of capacity for fading channels, including *ergodic capacity* [4], *outage capacity* [2] [1] and *delay limited capacity* [5]. The ergodic capacity for fading channels is the maximum achievable rate without any delay constraints. For a fast fading environment, a long code can undergo ergodicity in the fading process and hence can achieve arbitrarily close to the limit. In the case of slow fading subject to a constant delay constraints, the notion of outage capacity more often used. An information outage event is defined as the event where a channel is unable to support the actual rate of transmission. More specifically, outage occurs when the instantaneous mutual information between the channel inputs and outputs are smaller than the transmission rate. The probability associated with this outage event is called the outage probability [2]. The ε-outage capacity is defined as the maximum achievable rate that can be supported with an outage probability of less than ε [1]. Similar to the outage capacity, the delay limited capacity is defined as the zero-outage capacity [5]. A detailed survey of these capacity concepts can be found in [2].

Unfortunately, Shannon's proof is not constructive, as he did not provide ways to construct such codes but merely proved their existence. Since 1948, a whole scientific region (coding theory) has been dedicated to the design and implementation of codes for efficient communication. In the past few years, some practical code classes which can achieve transmission at rates very close to the channel capacity have been introduced. Two major representatives of this class of codes are LDPC and Turbo codes.

Error control / correction is needed in order to mitigate errors during transmission. The two main error control techniques are ARQ transmission schemes [6] and forward error correction (FEC) techniques. ARQ coding techniques are used for applications where feedback from the receiver is possible. There are three basic types of ARQ protocols *stop-and-wait* ARQ, *go-back-N* ARQ and *selective-repeat* ARQ. Hybrid ARQ schemes combine FEC and ARQ coding schemes. A comprehensive study of ARQ techniques is available in [6].

One of the first use of ARQ (both pure and hybrid) in block fading channels is proposed in [7] in which the outage capacity of a block fading channel is to be minimized by using ARQ techniques. The improved reliability in an ARQ system is achieved at the expense of transmission delay, leading to the tradeoff between delay and reliability. This tradeoff was studied in [8] , in which reliability called the SNR exponent or the optimal diversity order is defined in terms of the asymptotic slope of the outage probability curve.

Without delay constraints, the number of ARQ rounds to transmit a codeword can be arbitrarily large, thereby eliminating outages. In most practical communication systems, delay constraints do exist. The delay constraints can either be a short term delay constraint or a long term delay constraint. A short-term delay constraint is over a single codeword limiting the maximum number of ARQ rounds that a codeword can use [9]. Long-term delay constraint on the other hand is imposed only in long run, a limit on average number of ARQ rounds a codeword can use.

1.2 Scope of the Study

OFDM is often used to overcome the frequency–selective nature of the channel to provide high transmission data rate over a wireless channel. In addition to OFDM, adaptive coding and modulation, multi user detection and automatic repeat request (ARQ) schemes increase the reliability over wireless communication system. In OFDM, the available bandwidth is divided into sub orthogonal channels, thereby eliminating inter symbol interference (ISI). Each channel then undergoes a flat fading. Hence, OFDM can reduce a channel into several independent flat fading sub channels. In other words, the channel can be conveniently modeled as the block (flat) fading channel [2], [1]. Therefore deeper understanding of communication over block fading channels can give us insight into designing practical systems in a wireless environment.

1.3 Application

The block fading channel [2] [1] is commonly used to model a slowly fading channel. Examples of practical systems modeled by flat fading channels include slow time- and/or frequency hopping schemes such as global system for mobile (GSM) communication and enhanced data GSM environment (EDGE). By the use of modulation schemes such as OFDM, a channel can also be modeled as a block fading channel. And OFDM has been incorporated in many wireless standards such as IEEE 802.11 (WiFi) and IEEE 802.16 (WiMax).

1.4 Objective

Adaptive transmission based on ARQ schemes can help to minimize the word error rate in a block fading channel [7]. For non-real-time applications, there is no delay constraint before which a message frame needs to be decoded. Hence the number of ARQ rounds per codeword can be arbitrarily large to eliminate outages. In reality, most applications are subject to some delay constraints.

The objective of this thesis is to design practical codes that can achieve arbitrarily close to the outage limits over the ARQ systems. Namely to achieve

1. Maximum Diversity
2. Coding gain

Diversity is defined as slope of the error-rate curve as a function of the signal-to-noise ratio on a double logarithmic scale and upper-bounded by the intrinsic diversity of the channel given by the slope of the outage limit. Coding gain determines how close the error rate curve is to the outage curve. When full diversity is attained, the coding gain yields a measure of signal-to-noise ratio distance to the outage limit.

1.5 Thesis Organization

In Chapter 2, a brief description of some coding techniques that are useful for design of LDPC code is given. In Section 2.1 a graphical representation (tanner graph) of the capacity approaching LDPC codes is reviewed. A brief description of degree distribution is given in Section 2.2. A design algorithm for LDPC code is reviewed in Section 2.3. Different techniques for code extension to generate incremental redundancy are described in Section 2.4 and the performance of check splitting codes over AWGN channels is discussed.

In Chapter 3, research methodology and specific system model used for the ARQ code construction is described. The outage probabilities for the model are given.

Good codes that can perform close to capacity for AWGN channels need are not necessarily good on block fading channels. In Chapter 4, Coding schemes that are used to achieve close to outage limit and maximal diversity ARQ block fading channels are described. In Section 4.1 Root-check LDPC codes that achieves the maximal diversity for a fixed rate code over a block fading channel is described. In Section 4.2, a novel algorithm to design Root-Check LDPC codes based on Progressive Edge Growth (PEG) techniques for block-fading channels is proposed. This is followed by the check split algorithm that achieves close to these limits. Simulation results are given and discussed in Section 4.4.

Chapter 5 gives concluding remarks followed by areas for future research in Section 5.2.

CHAPTER 2: LITERATURE REVIEW

In this Section, we will review some the coding techniques that are useful for design of LDPC code. As the name implies, Low-Density Parity-Check (LDPC) codes are linear block codes which have a parity-check matrix with a low density of non-zero entries, i.e. a sparse matrix. The definition of “low density” is a bit loose, but generally matrices with less than 10% non-zero entries are considered to be sparse. The number of non-zero entries in a sparse matrix increases linearly with respect to matrix dimensions, and not quadratically as in regular (dense) matrices. This parity-check matrix can be stored very efficiently using sparse matrix techniques.

The first LDPC codes, introduced by Gallager [10], were regular codes. This means that a constant number of symbols participate in each check equation, and each symbol participates in a constant number of check equations. In terms of the parity-check matrix **H**, this means that the column and row weights are constant. As shown in [11], much better performance can be achieved by allowing irregular variable and check node degrees.

The LDPC code in systematic form is specified by its sparse parity check matrix **H**:

$$\boldsymbol{H} = [\,\boldsymbol{I}_{N-K}\ \ \boldsymbol{P}\,], \qquad \text{equation (2.1)}$$

where $\mathbf{I}_{N\text{-}K}$ is the identity matrix of size (N-K) and **P** is an (N-K)-by-K matrix. Then the generator matrix for the code is:

$$\boldsymbol{G} = [\boldsymbol{P}'\,\boldsymbol{I}_K] \qquad \text{equation (2.2)}$$

2.1 Graph Representation of LDPC Codes

We can construct a graphical representation of an LDPC code using *Tanner graphs* [12]. Tanner graphs are bipartite graphs, a bipartite graph (or bigraph) is a graph whose vertices can be divided into two disjoint sets U and V, such that every edge connects a vertex in U to one in V

In our case we have two sets containing *variable nodes* and *check nodes*. Variable nodes correspond to codeword bits and check nodes correspond to check equations.

Fig. 2.1: Symbols used for representing variable nodes and check nodes.

We can construct a Tanner graph from an $M \times N$ parity-check matrix $\boldsymbol{H}$ as follows:

1. Add N numbered variable nodes to the variable node set and N numbered check nodes to the check node set.

2. Add an edge between check node c_i and variable node s_j if and only if the corresponding entry h_{ij} in $\boldsymbol{H}$ is equal to 1.

Step 2 actually illustrates the fact that the codeword bit s_j participates in check equation c_i. More generally, a '1' at position i in row j of the parity-check matrix, indicates a connection between codeword bit i and parity-check equation j. Consequently, each row indicates which bits participate in the corresponding check equation, and each column indicates which check equations the corresponding bit participates in. Some useful properties/definitions regarding of bipartite graphs are:

1. The degree of a node is defined as the number of edges connected to that node.
2. A cycle is a path on the graph such that the start vertex and the end vertex are the same.
3. Bipartite graphs cannot contain odd-lengthed cycles.
4. The minimum cycle length of a graph is called the girth of the graph.
5. The minimum girth of a Tanner graph is 4.

Graph representation of LDPC codes plays a vital role in decoding, since the most widely used decoding algorithm is in fact an algorithm for representation. For the decoding algorithm to work well, it is essential for the graph to have as large a girth as possible. Equivalent representations of the same code can differ significantly in terms of girth, so a particularly interesting and important problem is finding the best possible representation marginalizing functions, which can be implemented using a certain graph

2.2 Degree Distributions

Assume that an LDPC code has length N and that the number of variable nodes of degree i is Λi, so $\sum_i \Lambda_i = N$. In the same fashion, the number of check nodes of degree i is P_i, so $\sum_i \mathrm{P_i}$ = N (1 -R) = M, where R is the *design rate* of the code. The following compact notation is more convenient

$$\Lambda(x) = \sum_{\mathrm{i}=1}^{l_{max}} \Lambda_{\mathrm{i}} x^i \qquad \mathrm{P}(x) = \sum_{\mathrm{i}=1}^{r_{max}} \mathrm{P_i} x^i \qquad \text{equation (2.3)}$$

From the above definitions, we get the following useful relationships:

$$\Lambda(1) = N \qquad P(1) = N(1 - R) \qquad \text{equation (2.4)}$$

$$R(\Lambda, \mathrm{P}) = 1 - \frac{\Lambda(1)}{\mathrm{P}(1)} \qquad \Lambda'(1) = P'(1) \qquad \text{equation (2.5)}$$

The first four relationships should be obvious. If we rewrite the left part of the last one as

$$\Lambda'(1) = \sum_{\mathrm{i}=1}^{l_{max}} i\Lambda_{\mathrm{i}} \qquad \text{equation (2.6)}$$

The *normalized variable node degree distribution* from a node perspective is defined as

$$L(x) = \frac{\Lambda(\mathrm{x})}{\Lambda(1)} = \sum_{\mathrm{i}=1}^{l_{max}} L_{\mathrm{i}} x^{i} \qquad \text{equation (2.7)}$$

The *normalized check node degree distribution* from a node perspective is defined as

$$R(x) = \frac{P(\mathrm{x})}{P(1)} = \sum_{\mathrm{i}=1}^{l_{max}} P_{\mathrm{i}} x^{i} \qquad \text{equation (2.8)}$$

2.3 Progressive Edge Growth Algorithm

Our goal is to construct a graph having a girth as large as possible, given the code length and degree distribution, which is a rather hard combinatorial problem. However, a suboptimal yet simple and well performing algorithm was proposed in [13] .

The basic idea is pretty straightforward: the graph is constructed in an edge-by-edge tanner with each edge placed so that it has the minimum possible impact on the overall graph girth. This means that, fundamentally, an edge is placed between the variable node in question and the most distant check node in the graph. In the optimal case where the distance is infinite, i.e. no path exists between the variable node and the check node, the new edge creates no additional cycles.

Given the number of variable nodes, the number of check nodes and the variable node degree distribution, this algorithm assigns degrees to each variable node according to the degree distribution and calculates the degree distribution of the check nodes, making it as uniform as possible.

2.4 Code Extension and Breaking Cycles in Tanner Graphs

The following techniques can be used for expanding the graph of an existing LDPC code in order to create a new code. A pleasant side-effect of this expansion is that many cycles of the corresponding Tanner graph are broken in the process, which is beneficial to the performance of the decoding process

2.4.1 Column Splitting

Let C be an LDPC code of length N. We can extend this code by splitting each column of its parity-check matrix into $q >= 2$ columns, where q is the extension factor. The resulting matrix will be even sparser than the original, so the resulting code will definitely be an LDPC code. If columns are split properly, the resulting codes can be very good. Column splitting increases the code's length and decreases its rate. Many codes of different rates and lengths can be constructed from a single mother code.

By column splitting, the degrees of the variable nodes in the extended code are reduced. Thus, the resulting graph has fewer cycles than the original graph. An example of a length-4 cycle being broken by column splitting can be seen below:

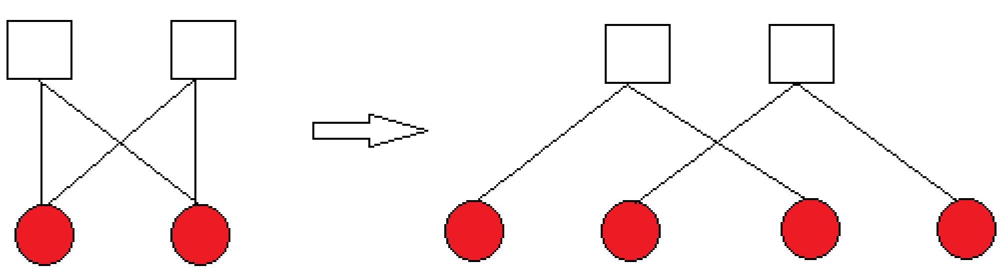

Fig. 2.2: Breaking a length-4 cycle by column splitting

The 2 original variable nodes have been split into 4 nodes, with each being connected to the 2 common check nodes by only one edge.

2.4.2 Row Splitting

Let C be an LDPC code with an M×N parity-check matrix H. We can extend this code by splitting each row of H into q >=2 rows, where q is the extension factor. The resulting matrix will be even sparser than the original, so the resulting code will definitely be an LDPC code. Row splitting increases the code's number of parity check equations, so the rate is decreased.

Accordingly, by row splitting, the degrees of the check nodes in the extended code are reduced. Thus, the resulting graph has fewer cycles than the original graph. An example of a length-4 cycle being broken by row splitting can be seen below.

The 2 original check nodes have been split into 4 nodes, with each being connected to the 2 common variable nodes by only one edge.

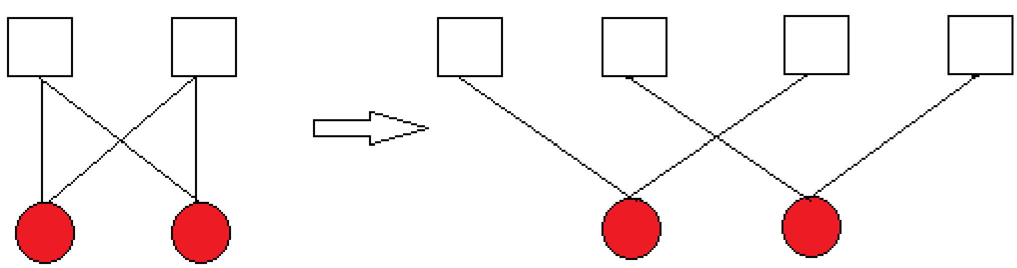

Fig. 2.3: Breaking a length-4 cycle by row splitting

2.4.3 Check Splitting Technique

Check splitting [14]is an effective way of generating incremental redundancy. Before the proposal of check splitting, puncturing was used on LDPC and RA codes to generate incremental redundancy. Puncturing of some of the variable nodes disables some of the checks in the code thereby decreasing the performance of the code and a very low rate code is generated. Then a high rate code is obtained by 'puncturing' some of the codeword symbol. The punctured codeword will then be transmitted. If the channel is too noisy for successful decoding, transmitter can send some of the previously punctured/deleted symbols until decoding is successful.

As the puncturing variable nodes disables some of the check nodes and extending a code can lead to new cycles, the code performance will be jeopardized. Motivated from these limitations of puncturing, an effective way to generate incremental redundancy by splitting checks was proposed.

The idea of check splitting is as follows. Consider a relatively high rate (K, N) linear code defined by a $(N - K) \times N$ parity check matrix H . Hence, the codeword are all $\boldsymbol{x} = [x_1, \dots, x_N]^{\boldsymbol{T}}$ such $that\ H\,\boldsymbol{x} = 0$. We can extend this code to slightly lower rate by increasing the code length to $N + 1$ and existing parity check equation with two of them (hence maintaining the cardinality of the code as2^K). To split a check, its neighbors are identified first. These neighbors are then divided into two almost equal sets.

Suppose the check equation can be written as

$$\sum_{i \in V_1} x_i = \sum_{j \in V_2} x_j \qquad \text{equation (2.9)}$$

Where $V_1, V_2 \subseteq \{1, \dots, N\}$. Then we replace the equation with the following ones:

$$\sum_{i \in V_1} x_i = x_{N+1} \qquad \text{equation (2.10)}$$

$$\sum_{j \in V_2} x_j = x_{N+1} \qquad \text{equation (2.11)}$$

The newly formed check has the variable nodes in these sets as neighbors and a new variable node is attached in order to satisfy the check equations. This new variable node is included in the new parity check matrix. Hence we have extended the code to a $(K\ , N + 1\ )$ code.

This algorithm is repeated until we obtain the required code rate. The check splitting algorithm normally divides the code of degree d into two checks of degree $[d/2 + 1]$ and$[d/2 + 1]$. Dividing the codes in this way ensures that, the check splitting does not create any new cycles in the graph, thus avoiding any error floors at high SNR. Any check with degree d can be effectively split to reduce the rate only d - 1 times. Any further splits result in the code being repeated. Since all the newly formed variable nodes are of degree two, the check splitting does not increase the complexity of the code compared to extension algorithms.

For codes obtained from puncturing, message passing based decoding is always conducted via low rate code’s factor graph. Hence, decoding complexity is more a less the same no matter how good the channel is. On the other hand, for code generated by check splitting, the factor graph defining the code involves only the variables that are actually transmitted. Hence if the channel has high SNR, then decoding is conducted via a small graph and hence has less decoding complexity.

CHAPTER 3: RESEARCH METHODOLOGY

The research methodologies applied throughout this thesis were based on a combination of mathematical analysis and computer simulations. Some of the software code that was written in order to support the simulations was based on the MATLAB programming language. In most of the cases where speed improvements were required to reduce simulation time, various functions were custom coded using the C programming language environment. Simulation code based on system model described in Section 3.1, developed for obtaining the results throughout this thesis ran on an 2.53GHz Intel Core i3 Processor in a Windows 7 operating system.

In Section 3.1 the system model for ARQ system will be presented followed by the simulation result of outage probability for block fading channels in Section 3.2.

3.1 System Model

Consider an ARQ system where the underlying channel is modeled by block-fading channel with a coherence time of T_c channel uses and each ARQ round consists of F independent block-fading periods. Thus each ARQ round comprises $F\ Tc$ channel uses. The delay constraint is specified by L, maximum number of allowed ARQ rounds that a codeword can be transmitted. In this thesis, a general ARQ system that uses stop and wait protocol is considered. Decoding is attempted after receiving each ARQ round. If decoding is successful, a positive acknowledgement to the transmitter is sent over an error free feedback link. Otherwise, a request for transmission of another ARQ round is made. Note that each ARQ round has F fading blocks with coherence time of Tc.

Let $\boldsymbol{x}_{l,f}$ be a vector of length Tc denoting the input of the block-fading channel in block f of ARQ round l, where $f = 1, \dots\dots, F$ and $l = 1, \dots\dots, L$. Furthermore, let $\boldsymbol{x}_l = (\boldsymbol{x}_{l,1}, \dots\ , \boldsymbol{x}_{l,\mathrm{F}})$ represent the concatenated input in ARQ round l .The corresponding output of the block-fading channel is represented by $\boldsymbol{y}_{l,f}$ and $\boldsymbol{y}_l$, respectively. The received signal at the f th block and l th ARQ round is given as

$$\boldsymbol{y}_{l,f} = \sqrt{\rho}\, \boldsymbol{h}_{lf}\, \boldsymbol{x}_{l,\mathrm{f}} + \boldsymbol{w}_{l,f} \qquad \text{equation (3.1)}$$

where the fading gains $\boldsymbol{h}_{l,f}$ are Rayleigh distributed and the elements of the noise vector $\boldsymbol{w}_{l,b}$ are independently Gaussian distributed with variance σ^2. The average signal to noise ratio (SNR) is given by ρ . Let $\boldsymbol{h}_l = (\boldsymbol{h}_{l,1},\ldots ,\boldsymbol{h}_{l,\mathrm{F}})$ denote the vector of fading gains in ARQ round l. We assume binary phase shift keying (BPSK) modulation with alphabet {1, -1} and perfect channel state information at the receiver.

In the first ARQ round, the encoder (Φ) uses a code of rate R_1 to encode the information sequence $\mathbf{m}$ to $\mathbf{x_1}$.

$$\boldsymbol{x_1} = \boldsymbol{m} * \boldsymbol{G_1} \qquad \text{equation (3.2)}$$

where $\mathbf{G_1}$ is generator matrix for first ARQ round obtained by linear transformation of $\mathbf{H_1}$ as described in Section 2. and $\mathbf{H_1}$ is parity check matrix constructed using Root-Check-PEG algorithm as described in Section 4.2.

Standard sum-product algorithm [15] is employed at the decoder with a maximum of 20 iterations. Decoding begins following the reception of an ARQ round. If the received codeword can be decoded, the receiver sends back a one-bit acknowledgement signal to the transmitter via a zero-delay error-free feedback link. The transmission of the current codeword ends immediately following the acknowledgement signal and the transmission of the next codeword in the queue starts. If an error is detected in the received codeword before the L^{th} ARQ round, the receiver requests another ARQ round by issuing a one-bit negative acknowledgement.

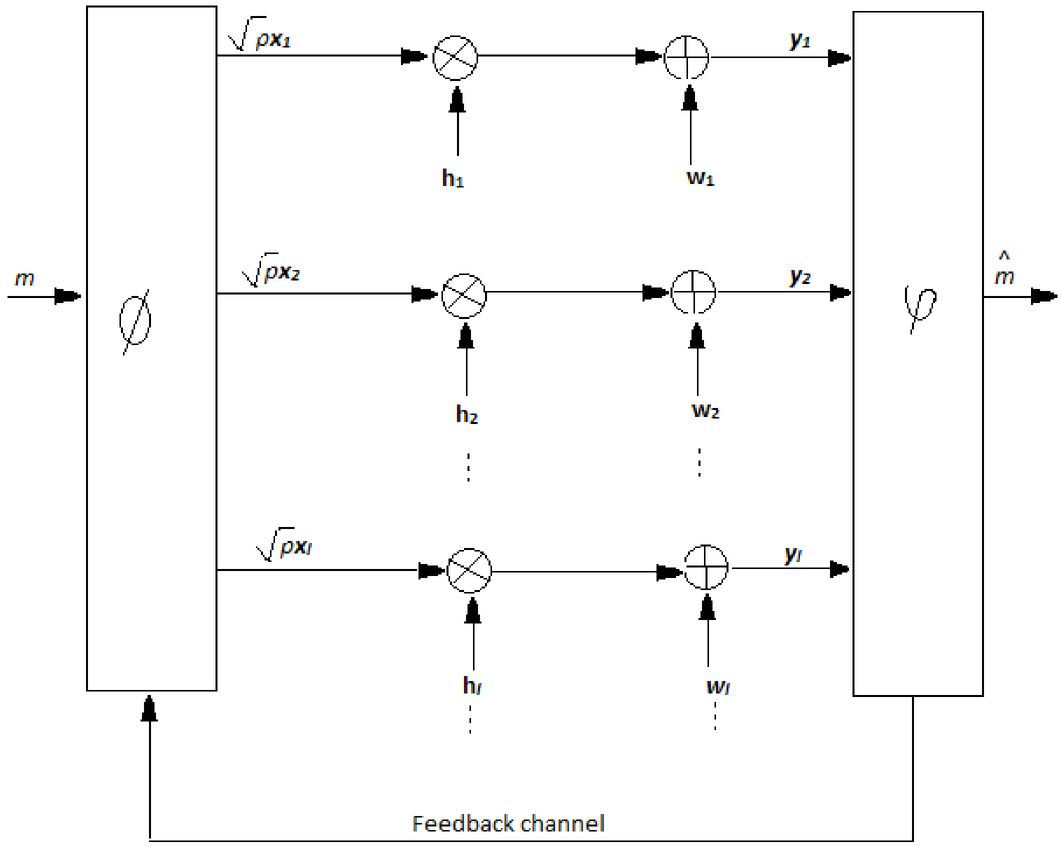

Fig. 3.1: System model

In the following ARQ rounds, the encoder transmits symbols $\mathbf{x}_l$, with decreasing the effective rate in ARQ round l to $R_l = R_1 / l$.

$$\boldsymbol{x_l = m{*}G_l} \qquad \text{equation (3.3)}$$

where $\mathbf{G}_l$ is generator matrix for lth ARQ round obtained by linear transformation of $\boldsymbol{H_l}$ as described in Section 2. And $\boldsymbol{H_l}$ is parity check matrix constructed using Check-Splitting technique as described in Section 4.3.1.

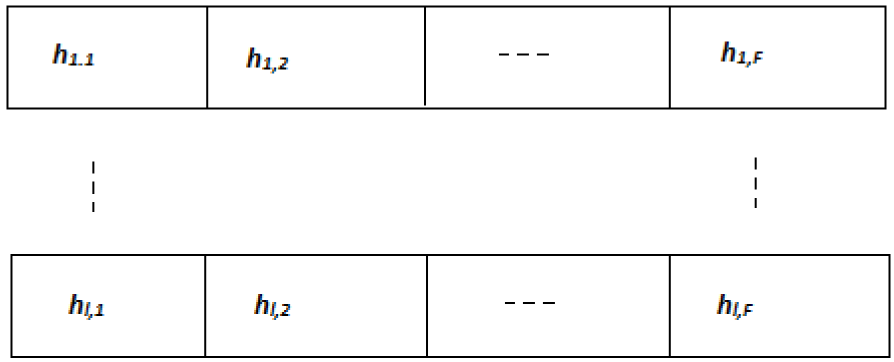

Fig. 3.2: Different notations

3.2 Outage Probability and Diversity

Coding over block fading channels is quite different from coding over AWGN channels, because the performance of a code over a block fading channel depends on the random fading gain which varies from block to block. The performance of codes in block fading channels is mainly characterized by diversity and coding gain. The diversity is the slope of the error-rate curve. The maximum diversity a code can achieve is upper bounded by the slope of outage curve because probability of error is lower bounded by the outage probability. Coding gain on the other determines how close the error rate curve is to the outage curve.

The system model described can be modified to accommodate fixed rate and ARQ block fading systems. For fixed rate systems, the maximum number of allowed ARQ rounds per codeword is restricted to one; while for ARQ systems the maximum number of allowed ARQ rounds is given by L. The fading gains for the block fading scenario are drawn from a Rayleigh distribution with each of the blocks being independent and identically distributed.

The word error of any block fading channel is lower bounded by the outage probability. The outage probability for a fixed rate system is given by

$$P_{out}(\rho, R) = Pr\{ I(\boldsymbol{x}; \boldsymbol{y} \mid \boldsymbol{h}) < R_1 \} \qquad \text{equation(3.4)}$$

where $I(\mathbf{x}; \mathbf{y} \mid \mathbf{h})$ is the normalized conditional mutual information between the input and output given the fading gain $\boldsymbol{h}$ and R_1 is the code rate and is given by [16]

$$I(\mathbf{x}; \mathbf{y} \mid \mathbf{h}) = \frac{1}{F}\sum_{f=1}^{F}\frac{1}{2}\log_2\left(1 + 2R\frac{E_b}{N_0}h_{1f}^2\right) \qquad \text{equation(3.5)}$$

Furthermore, ρ is the average SNR with which the codeword is transmitted. In words, if the channel does not support the rate of transmission, there is an outage event. The closed form formula for the outage probability is not available in general and it is usually obtained by using Monte-carlo simulations.

An ARQ system is said to be in outage if the receiver fails to decode the transmitted information after the reception of the L^{th} ARQ round. The outage probability is thus defined by

$$P_{out}(\rho, R_1) = Pr\{ I(\mathbf{x}_l; \mathbf{y}_l \mid \mathbf{h}_l) < R_1 \} \qquad \text{equation(3.6)}$$

where $I(\mathbf{x}_l; \mathbf{y}_l \mid \mathbf{h}_l)$ denotes the normalized mutual information between the received vector $\mathbf{y}_l$ and the transmitted codeword $\mathbf{x}_l$ given the fading gains $\mathbf{h}_l$ for lth ARQ round, ρ is the average SNR and R_1 is the rate of transmission of the first ARQ round.

Then for 'L' th ARQ round mutual information is given by

$$I(\mathbf{x}_l; \mathbf{y}_l \mid \mathbf{h}_l) = \frac{1}{L*F}\sum_{l=1}^{L}\sum_{f=1}^{F}\frac{1}{2}\log_2\left(1 + 2R\frac{E_b}{N_0}h_{lf}^2\right) \qquad \text{equation (3.7)}$$

The diversity of the system is defined as the slope of the outage probability in a log-log scale. The maximum diversity of a fixed rate system at rate R over a block-fading channel with B blocks and a maximum number of L ARQ rounds is given by the Singleton bound

$\delta_{LF} \triangleq 1+[LF (1 - R_L]$

$$= 1 + \left[LF\left(1 - \frac{R_1}{L}\right)\right] \qquad \text{equation (3.8)}$$

The rate of the first ARQ round is $R_1 = \frac{k}{FT_c}$. From the Singleton bound the maximum rate R_1 that allows full diversity is R_1 = 1/ F, which will be used in the rest of this thesis. It follows that k = T_c.

The SNR-exponent for the case of fixed rate is also given by the Singleton bound [17]and for the case of the ARQ coding scheme [9], the SNR-exponent is upper bounded by

$$\delta_F \triangleq 1 + \left[FL\left(1 - \frac{R}{\log_2 M}\right)\right] \qquad \text{equation (3.9)}$$

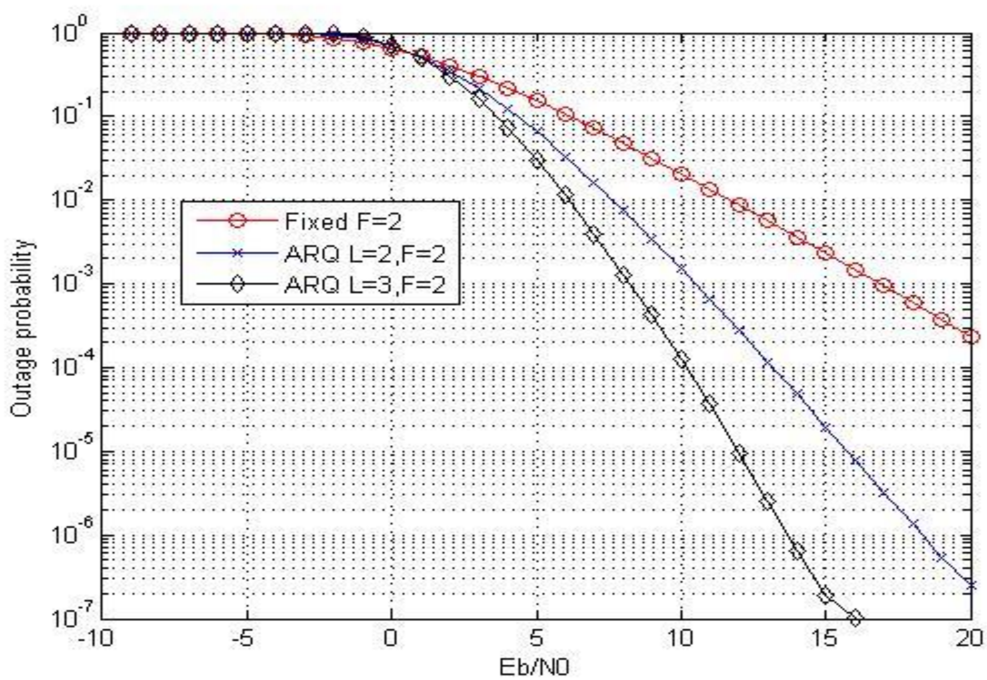

Fig. 3.3: Outage probability of fixed and ARQ block fading channel

CHAPTER 4: CODING OVER BLOCK FADING CHANNEL

Good codes that can perform close to capacity for AWGN channels need are not necessarily good on block fading channels. It was shown that codes that are maximum distance separable (MDS) are sufficient to achieve maximal diversity order. However, MDS codes are not sufficient to achieve outage probability [18] because coding is done over a finite number of fading blocks and hence the word error probability does not simply depend on the Hamming distance of the code. Instead, the performance of the code is dependent on a slightly different parameter called the block-wise Hamming distance [17]. Any code that has a large minimum block-wise Hamming distance need not necessarily have a large Hamming distance.

The design of outage-achieving codes is complex as the channels are non-ergodic. Since the channels are non-ergodic, EXIT functions [19] designed for the ergodic channels cannot be used.

In this Section, code structures based on PEG algorithm and check splitting techniques are incorporated in order to simultaneously achieve coding gain, incremental redundancy scheme and maximal diversity. Finally the simulation results of proposed coding technique over the block fading channels will be presented at the end of this chapter.

4.1 Root-Check LDPC Code

Root checks proposed in [17] are defined by a graphical representation. Before we describe root check LDPC codes, we will introduce various graphical notations which follows the notations used in Section 2.1 and depicted in Fig. 4.1.

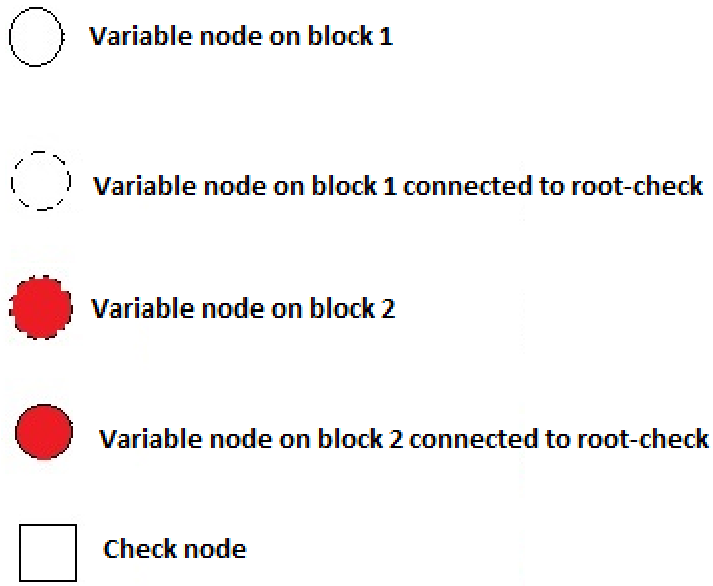

Fig. 4.1: Notations used in graph for root-check LDPC code **[20]**

Consider a LDPC code for a block fading channel over two blocks. A root check for the first block is defined as a check node that has only one neighbour in the 1st block while all the remaining neighbours belong to the second block. The root check for second block can be defined similarly. A graphical representation of a (2, 4) regular root check LDPC code for $N = 8$ is shown in Fig. 4.2. Each node is labeled according to the following convention. The notations i and p in Fig. 4.2 denotes the systematic part of the code and parity part of the code respectively. Similarly, the number represents the block it belongs to. For example the label $1i$ implies that the variable node corresponding to a systematic bit transmitted in block 1. In order to guarantee full diversity, all the systematic bits are required to connect to at least one root checks.

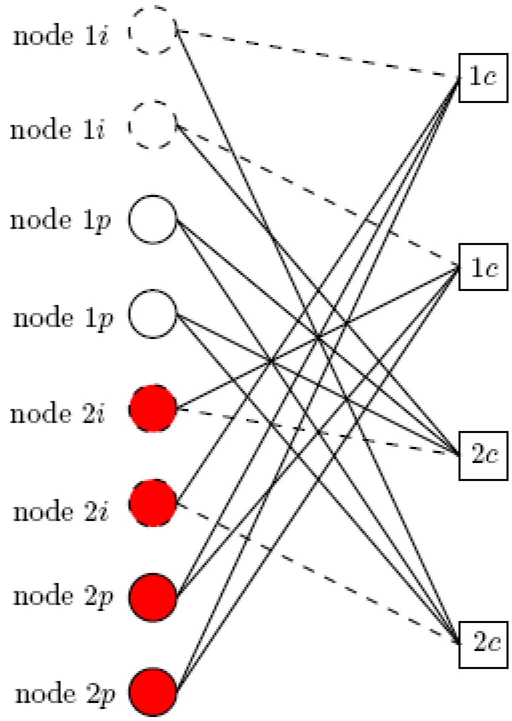

Fig. 4.2: Regular root check LDPC code **[16]**

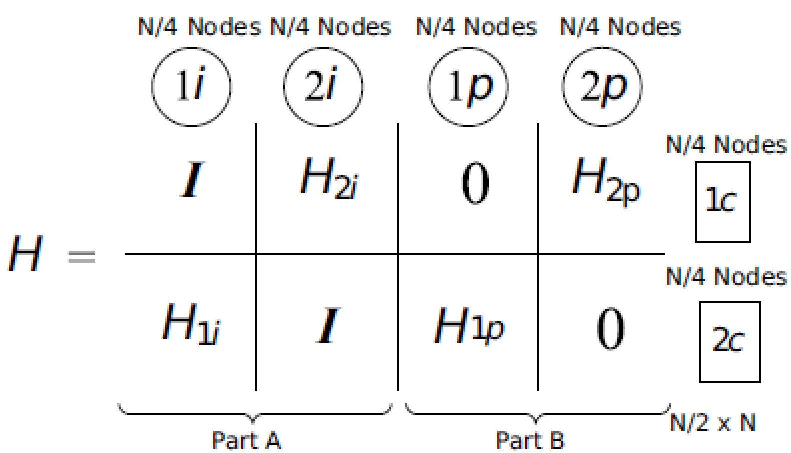

Fig. 4.3: Parity check matrix **[16]**

The coding structure shown in Fig. 4.2 was shown to achieve the outage probability for the case of block-erasure channels [18]. On general block-fading channel however, the codes were shown only to achieve the maximum diversity order. In [18], it was shown that achieving the diversity order does not depend on the degree distribution of the code but only on the coding structure. The degree distribution of the code however is important in maximizing the coding gain of the code.

Proposition 7 in [17] shows that root check LDPC codes have full diversity when decoded using belief propagation algorithm. It was shown that the diversity is achieved in the first decoding iteration and in the subsequent iteration diversity is preserved.

4.2 Construction of Root-Check LDPC Codes

A specific code construction is required in order to maximize the block-wise Hamming distance. LDPC codes and turbo codes that have a good block-wise Hamming distance were designed in [11] respectively where these codes achieve the maximum diversity. A Comprehensive study of different techniques for construction of LDPC codes is available in [21].

In this Section we present the proposed PEG Root-Check algorithm for the construction of good Root-Check LDPC codes for fix rate system. First of all, we will introduce some definitions and notations. Then, we will present the pseudo code of our proposed algorithm. In this thesis we consider the case of a block-fading channel with$F = 2$.

The variable node degree sequence D_S is defined [16], to be the set of column weights of $\mathbf{H}$ as designed, and is prescribed by the variable node degree distribution $\Lambda(x)$. Moreover, D_S is arranged in non-decreasing order. The original PEG algorithm upon which the proposed algorithm, called PEG Root-Check, is based constructs $\mathbf{H}$ by operating progressively on variable nodes to place the edges required by D_S. The Variable Node (VN) of interest is labeled s_j and the candidate check nodes are individually referred to as c_i. The PEG algorithm chooses a check node c_i to connect to the variable node of interest s_j by expanding a sub-graph from s_j up to maximum depth l. The set of check nodes found in this sub-graph is denoted $N_{s_j}^{l}$ while the set of check nodes of interest, those not currently found in the sub-graph, are denoted $\overline{N_{s_j}^{l}}$. For the PEG algorithm, a check node is chosen at random from the minimum weight check nodes of this set.

4.2.1 Pseudo-code for the PEG Root-Check Algorithm

Initialization: We define the indicator vectors $\mathbf{w}_1$ and $\mathbf{w}_2$, uni-dimensional vectors given by:

$$\mathrm{w}_1 = \left[0 \times \frac{M}{2}\ ,\ 1 \times \frac{M}{2}\right] \qquad \text{equation (4.1)}$$

$$\mathrm{w}_2 = \left[1 \times \frac{M}{2}\ ,\ 0 \times \frac{M}{2}\right] \qquad \text{equation (4.2)}$$

which are modeled on that of [13], the original PEG paper for tracking $\overline{N_{s_j}^{l}}$. as the sub-tree expands. Imposing $\mathbf{w}_1$ and $\mathbf{w}_2$ for appropriate sets of VNs forces the parity-check matrix to have the form of Fig. 4.3. The pseudo code for proposed PEG Root-Check algorithm is as follows:

1. **for** $j = 1 : n$ **do**
2. **for** $k = 0 : D_S(j) - 1$ **do**
3. **if** $(j \leq \frac{N}{4}) \parallel (\frac{N}{2} < j \leq \frac{3N}{4})$ **then**
4. $\boldsymbol{w}_{PEG} = \boldsymbol{w}_1$
5. **else**
6. $\boldsymbol{w}_{PEG} = \boldsymbol{w}_2$
7. **end if**

8. **if** $k == 0$ &$j < \frac{N}{2}$ **then**

Place the edge (cj , s_j) to ensure Root-check structure.

9. **if** $k == 0$ & $j > \frac{N}{2}$ **then**

10. Choose candidate at random from minimum weight CNs of the CN set indicated by $\boldsymbol{w}_{PEG}$.

11. **else**

12. Expand tree from the VN of interest under current setting. As the tree expands, for any CNs newly added to the tree the corresponding entry of $\boldsymbol{w}_{PEG}$ is set to 0.

13. Expand the tree to depth l s.t. the weight of $\boldsymbol{w}_{PEG}$ stops decreasing but is greater than 0 **or** the weight of $\boldsymbol{w}_{PEG} \neq 0$ but the weight of $\boldsymbol{w}_{PEG}$at the next level $l+1, = 0$.

14. Place the edge (c_i, s_j) randomly among the minimum weight check nodes of the set indicated by $\boldsymbol{w}_{PEG}$.

15. **end if**

16. **end for**

17. **end for**

This algorithm will be used to construct the Mother Code for the ARQ system.

4.3 Adaptive Transmission over Block Fading Chanel

In an ARQ system, specific construction of codes is required to incorporate the ARQ nature in code design. ARQ codes on a block fading channel were considered in [7]. The hybrid ARQ codes designed in [7] considered rate 1/2 block or convolutional codes. The design was later extended to incorporate LDPC codes over block fading channels in [22]. All the code designs described above used puncturing to obtain ARQ coding schemes. Check splitting technique [14] is recently proposed to generate incremental redundancy under the setup of ARQ in AWGN channels. A novel algorithm of check splitting of root check LDPC codes that preserve the root check properties was proposed in [20].

It has been shown in [14] that the choice of mother code has an effect on the performance of the check splitting codes. In order to achieve the maximum diversity over the block fading channels, the mother code used for check splitting should be designed to achieve the diversity over the first ARQ round. Root check LDPC codes are maximal diversity achieving codes over fixed rate systems. Hence, root check LDPC code is purposed as mother code to achieve the maximum diversity over ARQ block fading channels in this thesis.

4.3.1 Check Splitting of Root-Check LDPC Codes

We consider check splitting [20] within a root-check LDPC code operating at the maximum rate $R_1 = 1/F$ for full diversity. In this case, all check nodes are root-checks. To maintain full diversity, the proposed check split algorithm is based on retaining the root-checks after check splitting. Let $N_s^{(l)} = FTl$ and $N_c^{(l)}$ denote the number of variable nodes and check nodes at ARQ round *l*, respectively. Recall that one additional variable node, representing a new parity-check bit, is generated for each check split. We therefore need to perform a sufficient number of check splits per ARQ round to keep the rate as $R_1 = \frac{R_1}{l}$ .It follows that

$$N_c^{(l)} = N_v^{(l)} \left(1 - \frac{1}{Fl}\right) = FTl - T \qquad \text{equation (4.3)}$$

In each ARQ round each root-check is split at least one time, and a fraction of the $\frac{1}{F-1}$ retained root-checks will be split twice. For $F = 2$, all retained root-checks are split two times. The root-check splitting is performed according to the split rules in Eqs. (2.9), (2.10) and (2.11) in Section 2.4.3, and demonstrated for a simple case of two blocks per ARQ round and a maximum of two ARQ rounds (i.e. $F = 2$ and $L = 2$) in Fig. 4.4.

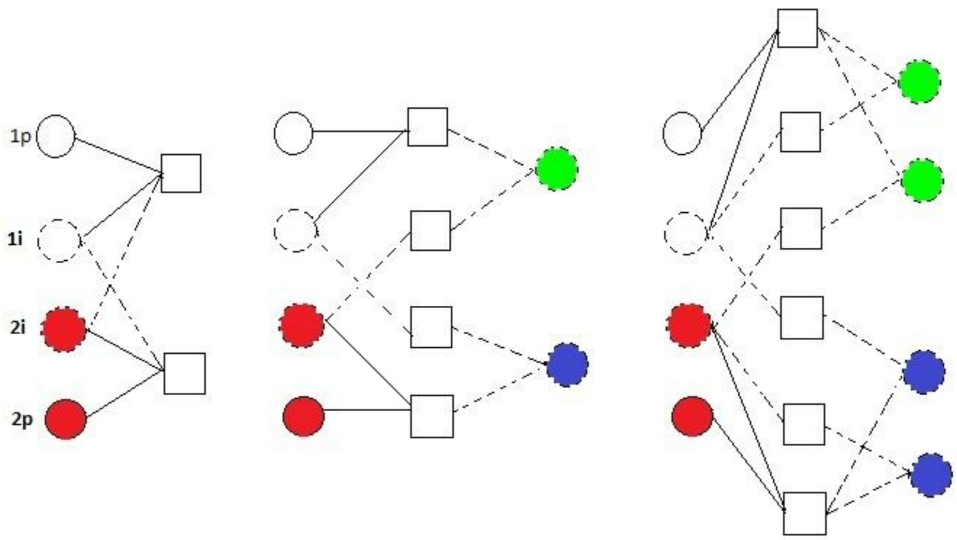

Fig. 4.4: Check splitting algorithm demonstrated for $L=2$, $F=2$ and $R_1 = \frac{1}{2}$

We follow the notation in Sections 2.1 and 4.1 where circles in the figure represent groups of variable nodes and squares represent groups of check nodes. Dashed lines correspond to edges of degree one while solid lines correspond to connections of higher degree. As discussed above, the retained root-checks need to be split again in this case. The initial root-check LDPC code is shown on the left. In the first stage of the splitting algorithm, each root-check is split. For the lower check split V_1 contains the index of the single node from 1*i* and V_2 contains the indices of nodes from 2i and 2*p*. The same procedure is applied to the upper check split. In both cases, the root-checks are retained, and the two new checks represent repetitions. The procedure is then repeated for the two retained root-checks in the second stage, leading to the graph to the far right. For the lower retained root-check, V_1 contains the index of one systematic node from 2*i*, while V_2 contains the indices of remaining nodes from 2*i* and 2*p*. The check splitting is mainly adding repetition checks, which ensures diversity and adds coding gain.

4.4 Performance of Proposed Coding Technique

In this Section, we present the performance of the proposed coding technique for system model described in Section 3.1 Rayleigh block fading channel is used with $F = 2$ independent fading blocks. All LDPC codes simulated here are (3,6) regular LDPC codes with rate $R = 1/2$ .The outage limit is drawn in dashed line in each figure for reference. The message symbol (***m***) length *K* is 500 bits.

4.4.1 Performance of the PEG Algorithm for Fix Rate System

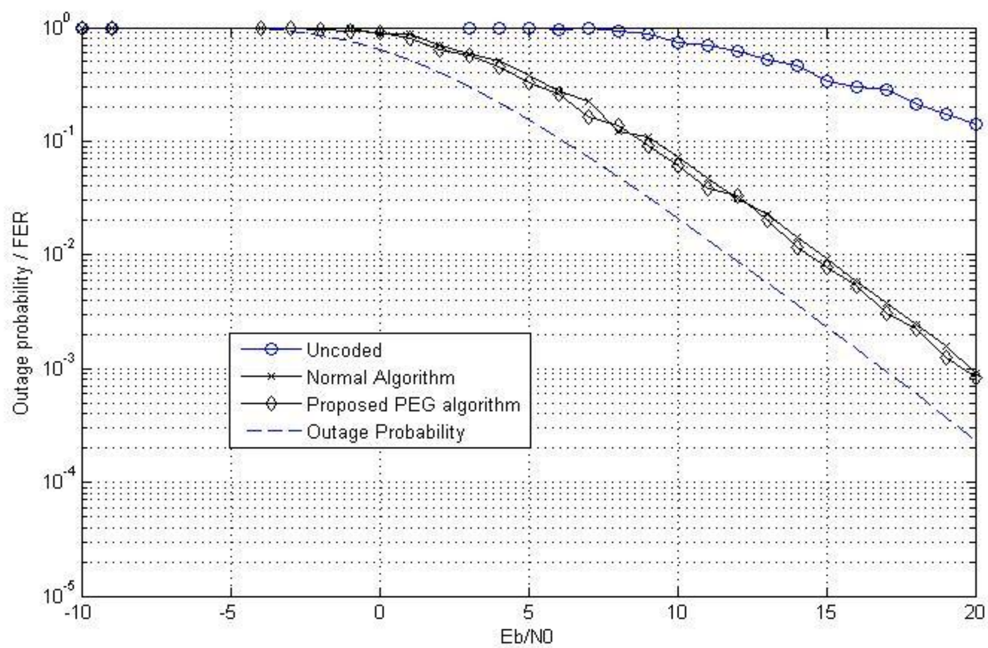

Fig. 4.5: FER performance comparison for root-check LDPC codes over a block-fading channel with F = 2 an N = 1000

In this Section, we present the performance of the proposed PEG-Root-Check LDPC codes for fix rate system as described in Section 3.2. It is stated in [16] that a maximum of 20 iterations are enough to obtain a fair analysis in terms of FER performance for block-fading channels with code word of length N 1200 bits.

The performance of a PEG based LDPC code [16] is also shown in Fig 4.5. From the results, our proposed PEG-Root-Check LDPC code outperforms the other Root-Check based LDPC codes and can save in average up to 0.5dB for the same FER performance. The better result presented by our proposed PEG-Root-Check LDPC codes enforces that a Root-Check LDPC code generated with the proposed PEG-based algorithm produces a better performance in terms of FER.

4.4.2 Performance of Proposed Coding Technique on ARQ System

In this Section, the check split algorithm defined in Sections 2.4.3 and 4.3.1 is extended to block fading channels. First, the coding technique used in [20] is analyzed followed by proposed coding technique in this thesis.

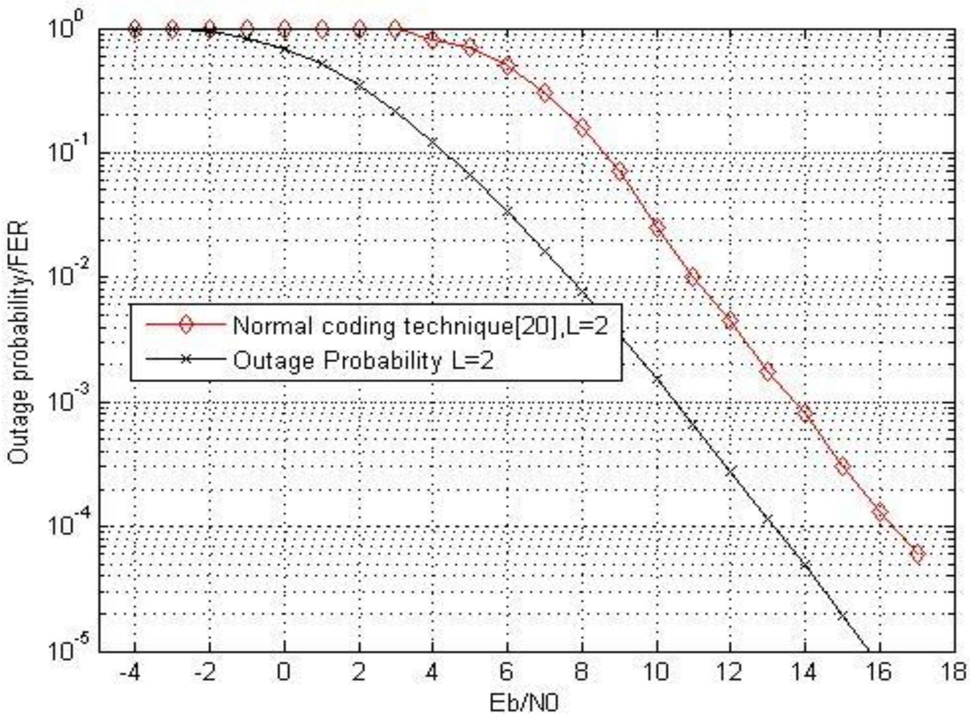

Fig. 4.6: Performance of normal coding technique with $L = 2, F = 2$ and $R1 = ½$.

As described in [17], root check codes are necessary but not sufficient to attain the outage probability. In this Section, we used root check codes that achieve the SNR-exponent on block fading channels and check splitting of root check LDPC. The code construction used in this thesis is used to analyze the performance over block fading channels. Simulation results in Fig. 4.6 show that normal coding technique can achieve full diversity at high-SNR. The solid line represents the outage probability of ARQ system with $L = 2, F = 2$ and $R1 = ½$. The solid line marked with '<>' is the code performance of normal coding technique used in [20].

In the following, the coding technique for the system model described in Section 3.2 is analyzed. The code structure of proposed technique is similar to using a root check LDPC code in first ARQ round as described in Section 4.1, and repeating the code over subsequent rounds.

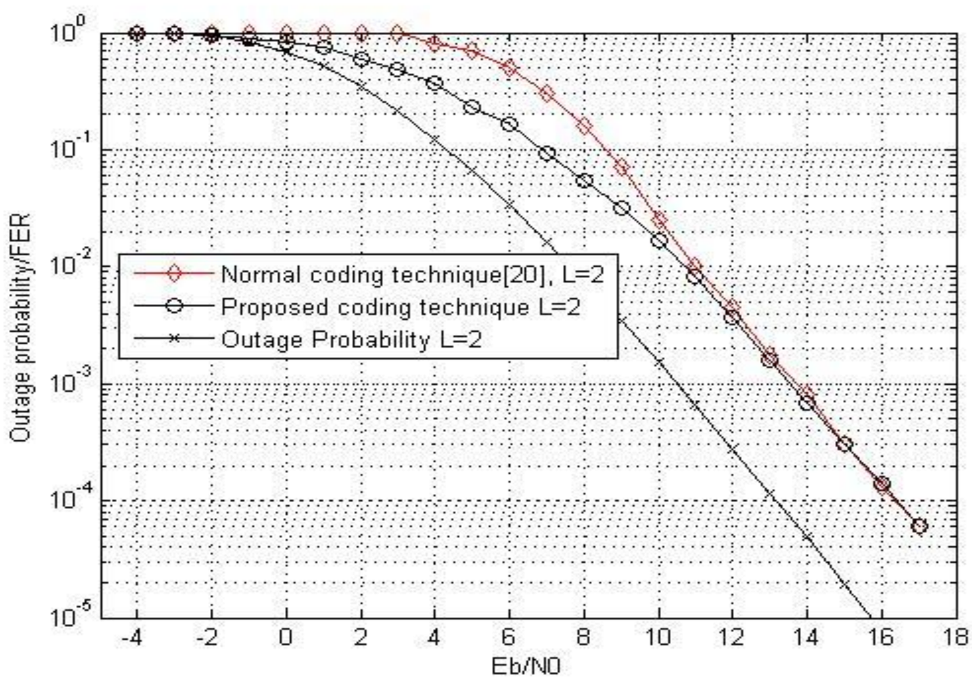

Fig. 4.7: Performance of proposed coding technique with $L = 2, F = 2$ and $R1 = ½$.

The simulation results for the proposed coding technique has been shown to perform better compared to coding technique used in [20]. In Fig. 4.7.The solid line represents the outage probability of ARQ system with L=2, F=2 and R_1= ½. The solid line marked with '<>'is the code performance of normal coding technique used in [20]. The solid line marked with '**o**' is the code performance of proposed coding technique. The proposed codes are shown to achieve the SNR-exponent. The coding gain of the proposed algorithm showed improvement over the normal coding technique for low SNR region and an average of 2 dB but the performance of the code is a further 3 dB away from the outage probability for the maximum allowed ARQ round L=2.

4.4.3 Performance of Proposed Code on Subsequent ARQ Rounds

In this section the performance of the proposed coding technique on subsequent ARQ rounds is analyzed. The same mother code is used and extended for subsequent ARQ round using check splitting algorithm.

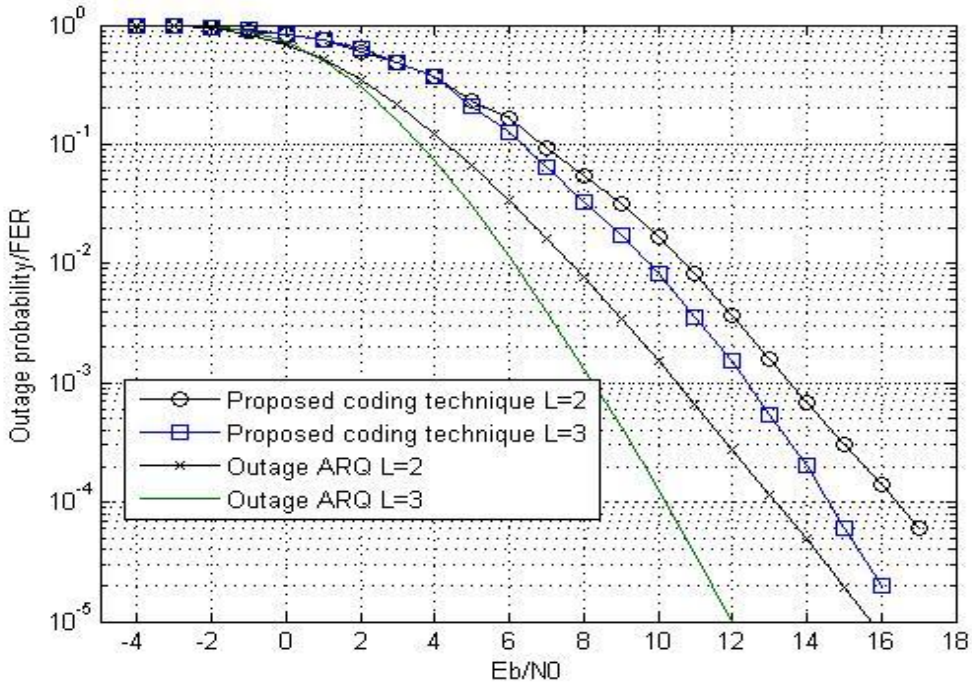

Fig. 4.8: Performance of proposed LDPC code for ARQ rounds $L = 2$ and $L = 3$.

The simulation results in Fig. 4.8 showed that the code for ARQ round L=3 achieves full diversity but it is still away from the outage probability for maximum allowed ARQ round L=3.

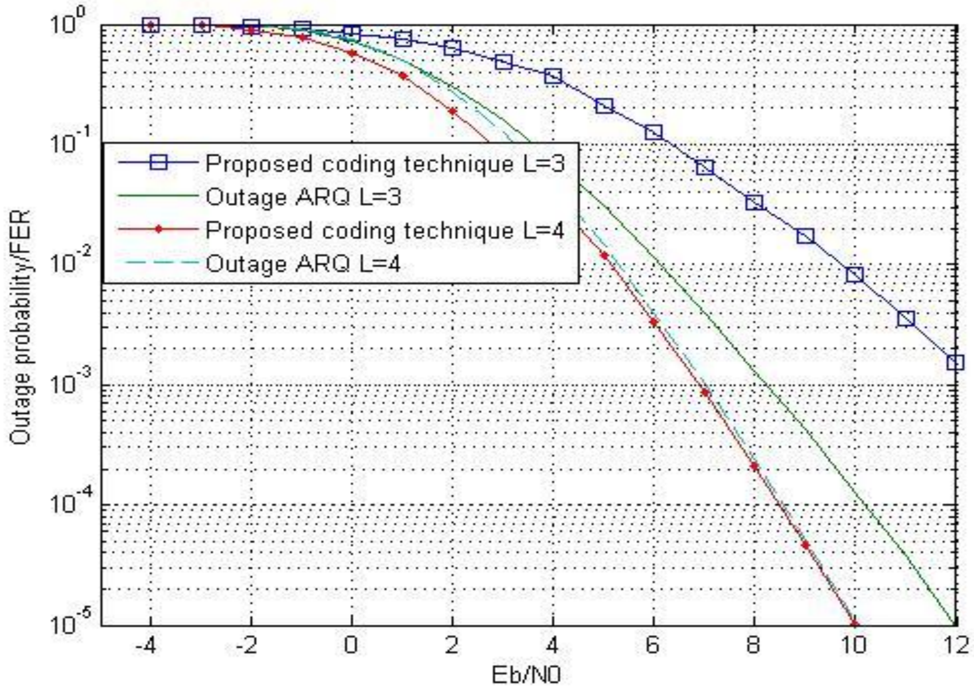

Fig. 4.9: Performance of proposed LDPC code for ARQ rounds $L = 3$ and $L = 4$.

The simulation results in Fig 4.9 showed that the performance of (3, 6) LDPC code with maximum allowed ARQ round $L = 4$ is close to the theoretical limit and attains the outage limit.

4.5 Summary

In this chapter, an algorithm to construct root check LDPC codes over the block fading channels was introduced. The simulation results have shown that the codes obtained by the proposed algorithm have shown to perform better in terms of the coding gain over the block fading channels. And this code is used as a mother code over ARQ block fading channel. Check Splitting is used to extend the code for adaptive transmission which provides incremental redundancy. And simulation results showed that proposed coding technique achieve diversity and coding gain over the ARQ block fading channels and is close to theoretical limit. And the (3, 6) LDPC code achieves the outage limit for maximum allowed ARQ round $L = 4$.

CHAPTER 5: DISCUSSIONS AND CONCLUSIONS

In this chapter, the main contributions of the thesis are discussed in Section 5.1. A few areas of further research are discussed in Section 5.2.

5.1 Discussions

In Chapter 3, Firstly, the Methodologies used for the thesis is discussed. The outage probability for ARQ block fading channels is derived. This derivation was based on the derivation of outage probability for fixed rate codes. The monte-carlo simulations of outage probability over ARQ block fading channels for various maximum allowed ARQ rounds and various fading blocks per ARQ round was shown to have different diversities as predicted by the singleton bound.

In Chapter 4, Firstly, root check LDPC codes were introduced. On general block-fading channel, the codes in [17] was shown that root check LDPC codes have full diversity when decoded using belief propagation algorithm. A novel algorithm to design Root-Check LDPC codes based on Progressive Edge Growth (PEG) techniques for block-fading channels was proposed. Simulation results showed that proposed algorithm outperform codes constructed by the existing techniques [16].

Coding over block fading channels was extended for ARQ system based on the new technique of check splitting of LDPC codes. In order to achieve the maximal diversity, the root check LDPC codes were chosen as mother code for subsequent check splitting. Check splitting of root check LDPC codes that preserve the root check properties is similar to the repetition of the systematic symbols. The performance gain of the codes obtained from the application of the new algorithm over regular root check LDPC code were illustrated for block fading channels. Simulation results showed that the new codes were close to theoretical limit. Also, the codes were shown to achieve the maximal diversity over the block fading channels.

The simulation results of the coding technique used in [20] were compared to that of the codes obtained from the proposed algorithm. The simulation results showed that the coding gain using the proposed algorithm is higher compared to using traditional algorithm. Simulation results showed that the (3, 6) LDPC code achieves the outage limit for maximum allowed ARQ round $L = 4$.

5.2 Future Enhancements

1. The degree distribution of the code is important in maximizing the coding gain of the code. So choice of degree distribution in design of mother code needs to be considered.
2. Power allocation for ARQ channels and coding over these power allocated channels also needs to be considered. Besides using ARQ techniques over the channels, power allocation to each of the fading blocks can also improve the performance.
3. Coding techniques on power allocated channels also need to be considered.

REFERENCES

[1] L. Ozarow, S. Shamai and A. Wyner, "Information theoretic considerations for cellular mobile radio," *IEEE Tran. Veh. Technol,* vol. 43, pp. 359-378, May 1994.

[2] E. Biglieri, J. Proakis and S. Shamai, "Fading channels: Information theoretic and communication aspects," *IEEE Trans. Inf. Theory,* vol. 44, pp. 2619-2692, Oct 1996.

[3] C. E. Shannon, "A mathematical theory of communication," *Bell Systems Technical Journal,* vol. 27, pp. 379-423, 1948.

[4] A. Goldsmith and P. Varaiya, "Capacity of fading channels with channel side information," *IEEE Trans. Inf. Theory,* vol. 43 no. 11, pp. 1986-1992, Nov 1997.

[5] S. Hanly and D. Tse, "Multi-access fading channel-Part II: Delay-limited capacities," *IEEE Trans. Inf. Theory,* vol. 44 no. 11, pp. 2816-2831, Nov 1998.

[6] S. Lin and D. Costello, Error control coding: Fundamentals and applications, Upper Saddle River: N.J: Pearson-Prentice Hall, 2004.

[7] E. Malkamaki and L. H., "Performance of Truncated Type-II Hybrid ARQ Schemes with Noisy Feedback over Block Fading Channels," *IEEE transactions on communications,* vol. 48 no. 9, pp. 1477-1487, Sept 2000.

[8] L. Zheng and D. Tse, "Diversity and multiplexing: A fundamental tradeoff in multiple-antenna channels," *IEEE Trans. Inf. Theory,* vol. 45 no. 5, pp. 1073-1096, May 2003.

[9] A. Chuang, A. G. i. Fabregas, L. Rasmussen and I. Collings, "Optimal throughput-diversity-delay tradeoff in ARQ block fading channels," *IEEE Inf. Theory work-shop*, 2006.

[10] R. Gallager, "Low-density parity-check codes"," *IRE Transactions on Information Theory*, pp. 21-28, Jan. 1962.

[11] M. Luby, M. Mitzenmacher, A. Shokrollahi and D. Spielman, "Analysis of low density codes and improved designs using irregular graphs," *Proc. 30th Annu. ACM Symp. Theory of Computing*, pp. 249-258, 1998.

[12] M. R. Tanner, "A recursive approach to low complexity codes," *IEEE Trans. Inf. Theory*, vol. 27, pp. 533-547, Sept. 1981.

[13] X. Y. Hu, E. Eleftheriou and D. M. Arnold, "Regular and irregular progressive edge growth Tanner graphs," *IEEE Trans. Inf. Theory*, vol. 51 no. 1, pp. 386-398, Jan 2005.

[14] M. Good and F. Kschischang, "Incremental redundancy via check splitting," *23rd Biennial Symp on communications*, pp. 55-58, May-Jun 2006.

[15] P. H. Siegel, J. Hou and L. B. Milstein, "Performance analysis and code optimization of low density parity-check codes on rayleigh fading channels," *IEEE journal on selected area in communications*, vol. 19 No. 5, pp. 924-924, May 2001.

[16] A. G. Uchoa, C. Healy, R. C. de Lamare and R. D. Souza, "LDPC codes based on Progressive Edge Growth techniques for block fading channels," *8th International Symposium on Wireless Communication Systems (ISWCS)*, pp. 392-396, Nov 2011.

[17] J. Boutros, A. G. i. Fabregas, E. Biglieri and G. Zemor, "Design and analysis of low-density parity-check codes for block fading channels," in *Inf. Theory Appl. Workshop*, San Diego, USA, Jan 2007.

[18] A. G. i. Fabregas and G. Caire, "Coded modulation in block fading channel: coding theorems and code construction," *IEEE Trans. Inf. Theory,* vol. 52, pp. 91-114, Jan 2006.

[19] S. t. Brink, "Convergence behaviour of iterative decoded parallel concatenated codes," *IEEE Trans. Communications*, vol. 49 no. 10, pp. 1727-1737, Oct 2001.

[20] S. K. Kambhampati, G. Lechner, T. Chan and L. K. Rasmussen, "Check splitting of root-check LDPC codes over ARQ block-fading channels," in *Communications Theory Workshop (AusCTW)*, Feb 2010.

[21] T. Richardson and R. Urbanke, Modern coding theory, Cambridge University Press, 2008.

[22] N. Varnica, E. Soljanin and P. Whitting, "Incremental Redundancy Hybrid ARQ with LDPC AND Raptor Codes," in *IEEE Int. Symp. Inf. Theory , pp. 995-999*, Adelaide, SA, Sep 2005.