Development of Rate-Compatible Structured LDPC CODEC Algorithms and Hardware IP

Project Final Report

School of Electrical and Computer Engineering
Georgia Institute of Technology
December 2006

Jaehong Kim
Demijan Klinc
Woonhaing Hur
Dr. Aditya Ramamoorthy
Dr. Sunghwan Kim
Dr. Steven W. McLaughlin

Table of Con

List of Abbreviati

ACK	Acknowledg
APP	A Posteriori
ARQ	Automatic F
AWGN	Additive Wl
BEC	Binary Eras
BER	Bit Error Ra
BP	Belief Propa
BPSK	Binary Phas
BSC	Binary Sym
CRC	Cyclic Redu
CSI	Channel Sta
E^2RC	Efficiently-I
eIRA	extended IR
FEC	Forward Err
FER	Frame Error
HARQ	Hybrid Auto
IC	Integrated C
IR	Incremental
IRA	Irregular Re
LDPC	Low-Densit
LLR	Log Likelih
MAP	Maximum A
MMSE	Minimum M
NACK	Negative Ac
PEG	Progressive
QC	Quasi-Cycli
QPSK	Quadrature
RCPC	Rate-Compa
V-BLAST	Vertical Bel
VLSI	Very Large
WER	Word Error

CHAPTER I

INTRODUCTION

Low-density parity-check (LDPC) codes by Gallager [1] had been forgotten for several decades in spite of their excellent properties, since the implementation of these codes seemed to be impossible at that time. These codes were rediscovered in the middle of the 1990s [2] and were shown to achieve Shannon limit within 0.0045dB [3]. LDPC codes are now considered good candidates for the next-generation forward error correction (FEC) technique in high throughput wireless and recording applications. Their excellent performance and iterative decoder make them appropriate for technologies such as DVB-S2, IEEE 802.16e [4], and IEEE 802.11n [5], [6].

While semiconductor technology has progressed to an extent where the implementation of LDPC codes has become possible, many practical issues still remain. First and foremost, there is a need to reduce complexity without sacrificing performance. Second, for applications such as wireless LANs, the system throughput depends upon the channel conditions and hence the code needs to have the ability to operate at different rates. Third, while the LDPC decoder can operate in linear time, it may be hard to perform low-complexity encoding of these codes. In particular, the class of irregular LDPC codes introduced by Richardson *et al.* [7] may have high memory and processing requirements, especially at short block lengths. While the encoding time can be reduced substantially using the techniques presented in [8] at long block lengths, their techniques may be hard to apply at short block lengths. The other option is to resort quasi-cyclic

(QC) LDPC or algebraic constructions that can be e
Irregular repeat-accumulate (IRA) codes were i
codes have a linear-time encoder and their perfor
LDPC codes. This class of codes was extended, c
Yang *et al.* [11], where they demonstrated high-rate

A popular technique for achieving rate adaptati
rate-compatible puncturing. A rate-compatible pu
applying to incremental redundancy (IR) hybrid
systems, since the parity bit set of a higher rate coc
lower rate code [12]. The RCPC scheme has anot
encoder and decoder while operating at different rat
transmitter sends depends on the rate requirement.
not transmitted are treated as erasures. Thus, pu
solution to the rate-adaptation problem.

Motivated by these observations, this report first
LDPC codes with short block lengths. Based on the
codes is proposed that can be efficiently encoded a
compatible fashion. The proposed LDPC codes
encoder and have good performance under punctur
we verify that the proposed codes show good thr
applied to IR-HARQ systems over time-varying cha

CHAPTER II

BACKGROUND RESEARCH

Channel coding is an essential technique to cope with errors occurring in channels of communication systems and storage systems. Channel coding has flourished in two branches. Channel errors can be corrected with forward error correction (FEC) codes. On the other hand, a receiver may request retransmission of the previous data if it fails to recover them, which is called automatic repeat request (ARQ). FEC codes can be classified into block codes, such as cyclic codes and LDPC codes, and tree codes, such as convolutional codes and Turbo codes. In this chapter, we briefly explain the block codes where LDPC codes are specified.

Let us consider linear block codes over the binary field $F_2 \triangleq (\{0,1\},+,\times)$. Let F_2^N be the N-dimensional vector space over F_2. Then, an (N, K) linear block code C is defined as K-dimensional subspace of F_2^N, where K is a data word length and N is a codeword length. Since C is a subspace of dimension K, there are K linearly independent vectors $\boldsymbol{g}_0, \boldsymbol{g}_1, \cdots, \boldsymbol{g}_{K-1}$ which span C. Let $\boldsymbol{m} = [m_0, m_1, \cdots, m_{K-1}]$ be the data word and $\boldsymbol{c} = [c_0, c_1, \cdots, c_{N-1}]$ be the corresponding codeword in the code C. The mapping $\boldsymbol{m} \rightarrow \boldsymbol{c}$ is thus naturally written as $\boldsymbol{c} = m_0\boldsymbol{g}_0 + m_1\boldsymbol{g}_1 + \cdots + m_{K-1}\boldsymbol{g}_{K-1}$. This relationship can be represented in the matrix form $\boldsymbol{c} = \boldsymbol{m}G$, where G is a $K \times N$ matrix;

$$G = \begin{bmatrix} \mathbf{g}_0 \\ \mathbf{g}_1 \\ \vdots \\ \mathbf{g}_{K-1} \end{bmatrix}$$

We call the matrix G the generator matrix for C

The encoding process can be viewed as an injectiv

K-dimensional vector space into vectors from th

ratio

$$R = \frac{K}{N}$$

is called *code rate*.

On the other hand, the null space $C^{\perp}$ of C has

$N-K$ linearly independent vectors $\mathbf{h}_0, \mathbf{h}_1, \cdots, \mathbf{h}_{N-}$

have for any $\mathbf{c} \in C$ that

$$\mathbf{h}_i \cdot \mathbf{c}^T = 0, \quad \forall$$

This relationship can be represented in the matrix

H is the so-called *parity-check matrix* defined as

$$H = \begin{bmatrix} \mathbf{h}_0 \\ \mathbf{h}_1 \\ \vdots \\ \mathbf{h}_{N-K-1} \end{bmatrix}$$

A low-density parity-check code is so called bec

low density of 1s. We address the details of LDPC

2.1 Low-Density Parity-Check Codes

Every LDPC code is uniquely specified by its parity-check matrix H or, equivalently, by means of the *Tanner graph* [13], as illustrated in Figure 2.1. The Tanner graph consists of two types of nodes: *variable nodes* and *check nodes*, which are connected by edges. Since there can be no direct connection between any two nodes of the same type, the Tanner graph is said to be *bipartite*. Consider an LDPC code defined by its corresponding Tanner graph. Each variable node, depicted by a circle, represents one bit of a codeword, and every check node, depicted by a square, represents one parity-check equation.

$$H = \begin{array}{c} \begin{matrix} a & b & c & d & e & f & g \end{matrix} \\ \begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix} \end{array} \begin{array}{l} \\ \leftarrow A \\ \leftarrow B \\ \leftarrow C \\ \leftarrow D \end{array}$$

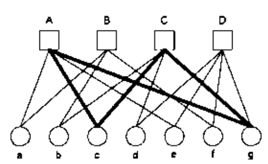

Figure 2.1 A parity-check matrix and its Tanner graph; Thick lines in the graph implies cycle 4.

Since we are considering N codeword length and K data word length, the Tanner graph contains N variable nodes and M check nodes, where $M = N - K$. Let us denote the parity-check matrix $H = (h_{ij})_{1 \le i \le M, 1 \le j \le N}$. Then, the i-th check node is connected to the j-th variable node if and only if $h_{ij} = 1$. For example, 1 in column f and row D in the parity-check matrix in Figure 2.1 corresponds to an edge connection between variable

node f and check node D in the Tanner graph. If the

variable or check node, we say that node has degree

degree 2 and check node D has degree 4. Tan

visualization tool for a variety of issues concerning

Definition 2.1: A *cycle* of length l in a Tanner

that begins and ends at the same node, whereby eve

The *length* of a cycle is the number of edges in th

many cycles of different lengths in their Tanner gra

Definition 2.2: The *girth* in a Tanner graph is the

The girth has a great importance for the code's p

bipartite, the smallest girth has length 4, as sho

However, it is desirable to avoid short cycles in des

can cause poor performance.

An *ensemble* of LDPC codes is defined by two

distributions, called a *degree distribution pair*, for t

$$\lambda(x) = \sum_{i=2}^{d_v} \lambda_i x$$

$$\rho(x) = \sum_{i=2}^{d_c} \rho_i x$$

where λ_i is the fraction of edges emanating from

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | **sales@docketalarm.com** | **1-866-77-FASTCASE**