Efficient Decoding and Application of Rateless Codes

by

Ali AbdulHussein

B.A.Sc., Simon Fraser University, Burnaby, B.C., 2006

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF
THE REQUIREMENTS FOR THE DEGREE OF

Master of Applied Science

in

The Faculty of Graduate Studies

(Electrical and Computer Engineering)

The University Of British Columbia

(Vancouver)

September, 2008

Abstract

Fountain codes have recently gained wide attention in the communications research community due to their capacity-approaching performance and rateless properties that allow them to seamlessly adapt to unknown channel statistics. This thesis offers two key contributions. For the first, we consider the problem of low complexity decoding of Luby Transform (LT) and Raptor codes, which are classes of Fountain codes. We introduce a decoding method which has a significantly reduced computational load compared to the commonly used alternative of message-reset decoding with a flooding schedule. This method combines the recently proposed technique of informed dynamic scheduling combined with incremental decoding. Simulation results for the example of the binary symmetric channel show complexity reductions (in terms of the total required number of decoding iterations) by 87% compared to conventional message-passing decoding and 54% compared to a recently proposed incremental decoding scheme for Raptor codes.

Having proposed our novel decoding method, we then focus on applying rateless codes to free-space optical (FSO) transmission systems. FSO systems enable high-speed communication with relatively small deployment costs. However, FSO systems suffer a critical disadvantage, namely susceptibility to fog, smoke, and similar conditions. A possible solution to this dilemma is the use of hybrid systems employing FSO and radio frequency (RF) transmission. As for the second contribution of this

thesis, we propose the application of rateless coding for such hybrid FSO/RF systems. The advantages of our approach are (i) the full utilization of available FSO and RF channel resources at any time and (ii) very little feedback from the receiver. In order to substantiate these claims, we establish the pertinent capacity limits for hybrid FSO/RF transmission and present simulation results for transmission with off-the-shelf Raptor codes, which achieve realized rates close to these limits under a wide range of channel conditions.

Table of Contents

List of Tables

List of Figures

Acknowledgements

I wish to express my sincere thanks to my supervisor, Prof. Lutz Lampe. This thesis and work would have not been accomplished without his expert advice and effective communication. I am particularly grateful to him for giving me the opportunity to explore and excel in the field of communications.

I would like to express a special word of thanks to Anand Oka, who tirelessly assisted me in every step of this work. I also thank my friends who offered encouragement when it was most needed.

A special thanks to my mother and father who supported me throughout my stay at the University of British Columbia.

Dedication

This thesis is dedicated to my parents

Samar Ibrahim & AbdulSalam Hamze

for their perseverance over the years,
as they have given me the opportunity
to study at this reputable university
in this great region of the world.

Chapter 1

Introduction

Fixed-rate codes such as low-density parity-check (LDPC) codes, are typically optimal for channels with known statistics [19]. Practically, however, the channel statistics are unknown to the transmitter in most cases, and fixed-rate codes may fail to perform as well. Fountain (rateless) codes have been introduced in [13] to address this problem. Luby Transform (LT) and Raptor codes are two classes of these codes. They have received considerable attention in the recent past due to their inherent ability to adapt to channel conditions and their capacity-approaching performance. It has been shown that these codes offer good performance not only for erasure channels, but also for other channels such as the binary symmetric channel (BSC), the additive white Gaussian noise channel (AWGNC) and fading channels [4, 17, 20].

In this thesis work, we focus on two main issues concerning the application of rateless codes. First, we investigate the problem of decoding rateless codes more efficiently. Second, we study the application of rateless codes to FSO transmission systems.

To address the former issue, we first introduce the existing methods of decoding rateless codes, with a greater focus on scheduling techniques and stopping criteria. We then note that in order to decode rateless codes more efficiently, we have to

accomplish the following:

1. Fast convergence of decoding, i.e., less computation required before decoding is completed successfully.

2. Early termination of decoding, i.e., smarter method to detect the end of a decoding process, and hence stop earlier without degrading error-performance.

The decoding of rateless codes proceeds in multiple attempts, and each attempt is composed of several iterations. If in one attempt, the decoder fails to decode successfully (error-free), another attempt is executed upon the arrival of more information from the channel. As such, to realize fast convergence, we utilize a method of retaining information from previous decoding attempts as presented in [8]. We combine this method with a newly presented scheduling technique, which was initially introduced for fixed-rate codes [2]. In this scheduling technique, only less reliable sections of the decoding graph are active in the decoding process. This results in improved (faster) convergence compared to the commonly used techniques. The combination of the aforementioned techniques results in a novel decoding method which we term Incremental Decoding with Informed Dynamic Scheduling (IDIDS).

Secondly, to achieve early termination, we present a novel hybrid stopping criterion which allows for termination of decoding much earlier. This criterion tests for successful completion at a very fine interval (less then one iteration) and is hence capable of stopping the decoding attempt at a much earlier stage than traditional criteria. We apply this criterion to IDIDS, and find through simulations, that decoding complexity can be reduced greatly.

In the later sections of this thesis, we introduce FSO transmission systems, their applications, and the constraints limiting their performance. We then discuss the existing techniques utilized to combat such constraints such as the use of error control codes, interleaving and spatial diversity. We discover that media diversity is the only way to perform well as proposed as in [10], where a radio frequency (RF) channel is used in conjunction with the FSO channel. However, this approach is deemed inefficient because of duplication of information on RF and FSO channels. Realizing this fact, we propose our alterative which can achieve a close-to-optimal performance under a wide spectrum of channel conditions, yet using minimal feedback and avoiding duplication. Our method utilizes a practical implementation of Raptor codes. Simulations show that these codes can approach the associated capacity limit of the hybrid FSO/RF channel under various channel conditions.

1.1 Thesis Outline

The remainder of this thesis is organized as follows:

In Chapter 2, we introduce and compare rateless codes to fixed-rate codes. We then briefly present the procedure for encoding LT and Raptor codes. Subsequently, we discuss the existing techniques for decoding rateless codes with several scheduling recipes. We also talk about the recently presented method for incremental decoding. Finally, we review the stoping criterion for decoding rateless codes.

Chapter 3 presents the first contribution of this thesis. A new method for incremental decoding is presented and discussed. Then, this method is combined with

the freshly introduced sequential scheduling technique to result in a novel method of decoding rateless codes. A new stoping criterion is catered for this combination to achieve maximal decoding cost reduction. This chapter concludes with simulation results showing the reduction achieved for both LT and Raptor codes.

In Chapter 4, we shift our focus to the problem of combating constraints limiting performance of FSO systems. We introduce the existing techniques and then propose our method which is based on the application of rateless codes to hybrid FSO/RF systems. The transmission model and the proposed coding scheme are discussed in details. Simulation results confirm the superiority of our method.

Chapter 5 summarizes the thesis conclusions and outlines few suggestions for future work.

1.2 Contributions

The overall work of this thesis has lead to two publications as well as a recent submission as follows:

- Ali AbdulHussein, Anand Oka and Lutz Lampe. Decoding With Early Termination for Rateless (Luby Transform) Codes, In *Proc. of the IEEE Wireless Communications and Networking Conference (WCNC)*, Las Vegas, USA, March 2008.

- Ali AbdulHussein, Anand Oka and Lutz Lampe. Decoding with Early Termination for Raptor Codes, *IEEE Communications Letters*, 12:444-446, June 2008.

- Ali AbdulHussein, Lutz Lampe and Anand Oka. Rateless Coding for Hybrid FSO/RF Communication System, submitted to the *IEEE Communications Letters*, Aug 2008.

Chapter 2

Rateless Codes

In this chapter, we introduce rateless codes and discuss their differences with fixed-rate codes. Next, we present methods for encoding and decoding rateless codes with more details on techniques used to schedule rateless decoding. The degree distributions used to encode LT codes are also reviewed briefly in this Chapter. We finally introduce the stopping criterion utilized to terminate rateless decoding.

2.1 Background

While fixed-rate codes such as LDPC codes have been shown to approach capacity for channels with known statistics [19], in many practical situations the channel statistics are unknown at the transmitter or, as in e.g. broadcast transmission, different transmitter-receiver links experience different statistics, and hence fixed-rate codes may perform poorly. To address this issue, the concept of rateless codes, and in particular Fountain codes, is appealing. LT and Raptor codes, invented by Luby [13], are classes of Fountain codes which are capacity-achieving for erasure channels.

For a Fountain code, transmitted bits could be either the input bit itself (degree one) or the exclusive-or of multiple input bits. The size of this multiplicity is referred to as degree and hence a Fountain code is characterized by its degree distribution.

The success of the LT codes depends critically on using the (robust) Soliton Degree Distribution and long code lengths [13]. However, to maintain low complexity and latency in practical applications, [20] has proposed an optimized degree distribution where the maximum degree is limited to 67. Since this results in a significant error floor, a fixed high-rate outer code like LDPC needs to be used to give good asymptotic performance. Such a concatenation (high-rate outer code and LT code) has been termed a Raptor code [4, 20]. It has been shown that these codes offer good performance not only for erasure channels, but also for other channels such as the binary symmetric channel (BSC), the additive white Gaussian noise channel (AWGNC) and fading channels [4, 17, 20].

2.2 Encoding Rateless Codes

This section first reviews the method of encoding LT codes based on a pre-defined degree distribution. The following section then discusses various degree distributions given in literature. The encoding of Raptor codes is also presented.

2.2.1 LT Encoding

The factor graph shown in Figure 2.1 represents the graph for a typical LT code. The LT code encoder, in general, linearly transforms a k-bit input vector (v_1, v_2, v_3, ..., v_k), labeled as white circles in the graph, into an infinite stream of check symbols (c_1, c_2, c_3, ..., c_n), labeled as squares, which passes through a given channel having an arbitrary capacity. The decoder collects channel output symbols progressively (y_1,

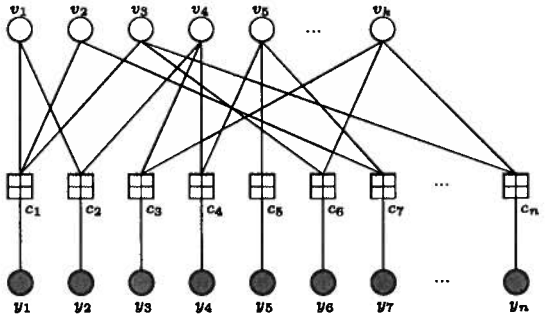

Figure 2.1: LT Code Factor Graph

y_2, y_3, ..., y_n), labeled as grey circles, till it has sufficient information to infer the bits (v_1, v_2, v_3, ..., v_k). The LT code rate is then given by

$$R_{\text{LT}} = \frac{k}{n}, \tag{2.1}$$

where the random variable n is the number of check symbols required for successful (error-free) decoding. The rate R_{LT} is therefore a random variable as well that depends on the channel noise realization.

The process of generating a check symbol is described as follows:

- The value of the encoded symbol is the exclusive-or of d randomly chosen input symbols from the k-bit input vector, where d is called the symbol degree ($1 \leq d \leq k$).
- The value for d is chosen from a degree distribution. More about the design and analysis of degree distributions can be found in the next section.

The resulting code can be thought of as an irregular (non-uniform d) low-density code. The connections between the input and check symbols can be represented by

a generator matrix. Let G be the matrix associated with the factor graph shown in Figure 2.1. This G, therefore, can be written as

$$G = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & \dots & 0 \\ 1 & 0 & 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 0 & 1 & 0 & \dots & 1 \\ 0 & 0 & 0 & 1 & 1 & \dots & 0 \\ 0 & 0 & 0 & 0 & 1 & \dots & 0 \\ 0 & 0 & 1 & 0 & 0 & \dots & 1 \\ 0 & 1 & 0 & 0 & 1 & \dots & 0 \\ . & . & . & . & . & \dots & . \\ . & . & . & . & . & \dots & . \\ . & . & . & . & . & \dots & . \\ 0 & 0 & 1 & 0 & 1 & \dots & 1 \end{bmatrix}.$$

Each row in G is associated with a single check symbol of the code, and the 1's represent the indices of the input bits connected to this check symbol. For instance, the first row, [1 1 1 0 0 ... 0], represents the first check symbol's, c_1, connection to v_1, v_2 and v_3. Therefore, c_1 is given by

$$c_1 = v_1 \oplus v_2 \oplus v_3. \tag{2.2}$$

2.2.2 LT Degree Distributions

The degree distribution of an LT code gives the probabilities, $\rho(d)$, of having a check symbol with degree d. The design of such a distribution has to guarantee two elements:

1. Some encoded symbols should have high degrees to ensure that all input bits are connected to some check symbols. For successful decoding, every input bit must have at least one connection with a check symbol.

2. There must be few check symbols with low degrees, such as one and two, so that the decoding process can get started.

In achieving this, one distribution, named the Ideal Soliton Distribution, has been suggested in [13] and is given by

$$\begin{aligned} \rho(1) &= 1/n, \\ \rho(d) &= \frac{1}{d(d-1)}. \end{aligned} \tag{2.3}$$

where n is the size of the input vector. However, this distribution is impractical because it may lead the decoding process to a dead-end when there are not enough degree-one check symbols to keep the decoding process going. Notice that this distribution guarantees only one degree-one check symbol in average. To mitigate this issue, the Robust Soliton Distribution has been suggested alternatively. This distribution has two extra parameters, c and δ, which are specifically designed to ensure

that the expected number of degree-one check symbols is about

$$S = c \log\left(\frac{n}{\delta}\right)\sqrt{n}. \tag{2.4}$$

instead of 1 which is what the Ideal Soliton Distribution guarantees. c is a constant, and δ is the bound on the probability that the decoding process fails after $n + \epsilon$ received check symbols, where $\epsilon > 0$ is an overhead factor which accounts for the suboptimality of the (finite-length) code with respect to capacity. The Robust Soliton Distribution [13] is defined as

$$v(d) = \frac{\rho(d) + \tau(d)}{Z}, \tag{2.5}$$

where $\rho(d)$ is given by equation (2.3) and $\tau(d)$ is given by

$$\tau(d) = \begin{cases} \frac{S}{nd} & \text{for } d = 1, 2, ..., (n/S) - 1 \\ \frac{S}{n}\log(\frac{S}{\delta}) & \text{for } d = n/S \\ 0 & \text{for } d > n/S \end{cases} \tag{2.6}$$

and the normalization factor Z is defined as

$$Z = \sum_{d}(\rho(d) + \tau(d)). \tag{2.7}$$

2.2.3 Raptor Encoding

The Raptor encoding process is composed of an outer code encoder (which is typically an LDPC encoder) and an LT code encoder which was described in Section 2.2.1.

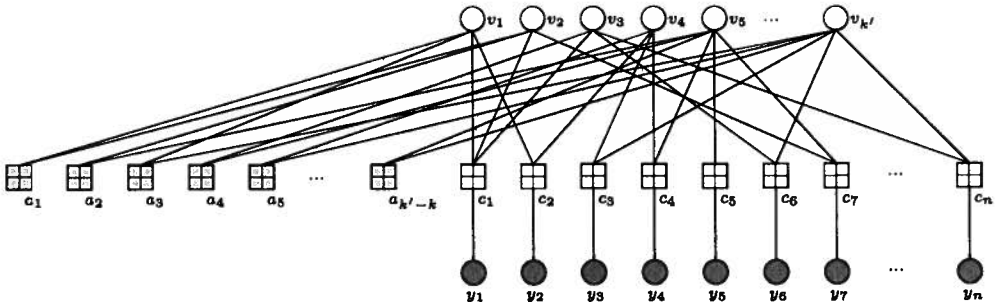

Figure 2.2: Factor graph of a Raptor code after transmission of n check bits.

The factor graph of a typical Raptor code is shown in Figure 2.2. The outer encoder encodes the k-bit input vector into the k'-bit codeword, labeled as white circles in the factor graph. The check symbols for this outer code is given by the vector (a_1, a_2, a_3, ..., $a_{k'-k}$), labeled as grey squares. The k'-bit codeword is then fed into the LT code encoder to produce an infinite stream of check symbols (c_1, c_2, c_3, ..., c_n), labeled as white squares. The channel output symbols are (y_1, y_2, y_3, ..., y_n), labeled as grey circles. The rate of the Raptor code, R_{RAPTOR}, is given by

$$R_{\text{RAPTOR}} = \frac{k}{n}, \tag{2.8}$$

where the random variable n is the number of check symbols required for successful decoding. The rate R_{RAPTOR} is also a random variable that depends on the channel noise realization. The LT portion of the Raptor code is characterized by a generator matrix, G, though the outer code is characterized differently (a parity check matrix if LDPC is used).

2.3 Decoding Rateless Codes

Similar to fixed-rate codes, e.g. LDPC codes, rateless LT and Raptor codes (codes defined on factor graphs) can be decoded with message-passing algorithms such as belief propagation (BP). BP decoding can proceed with different scheduling techniques. A scheduling technique is a recipe of how to communicate reliability information among input and check symbols during the BP process. These techniques will be discussed in the following section.

Different from fixed-rate codes, however, the decoding graph of rateless codes grows incrementally with time as new check symbols become available. Normally the decoder requires more symbols when it fails to decode with the available ones. Therefore, a fresh decoding attempt is made after the arrival of each new batch of check symbols on a newly defined factor graph. However, to maintain low decoding complexity (which is typically measured by the total number of iterations until successful decoding), it is desirable to terminate each decoding attempt as early as possible and hence reduced the overall cost. To do so, a smart stopping criterion is required. The next Chapter of this thesis presents our novel method to terminate decoding early though keeping error performance intact.

In the general case of decoding rateless codes, the decoder collects channel output symbols y_i progressively until it believes to have sufficient information to infer the k input bits. In other words, a receiver-side estimate of the channel statistics (capacity C) is made and the first decoding attempt is executed when n_{min} channel symbols are collected. n_{min} is given by

$$n_{min} \approx k/C. \tag{2.9}$$

If the decoded information is erroneous, which is typically detected by making use of a cyclic redundancy check (CRC) with only a few bits of additional redundancy, another decoding attempt is made after a new batch of n_{inc} samples has been received.

If the number of the current decoding attempt is denoted by κ, then the total number of received channel symbols is given by

$$n = n_{min} + (\kappa - 1)\, n_{inc}, \tag{2.10}$$

and the instantaneous code rate is the one given in 2.1 for LT codes or in 2.8 for Raptor codes. Figure 2.2 illustrates the factor graph for a Raptor code, where we labeled the n parity-check nodes involved in the decoding by c_i, $i = 1, 2, \ldots, n$. Note that the decoding of LT and Raptor codes can be executed similarly with the exception that the check nodes of the outer code (which typically belong to a high-rate LDPC code) do not exist in the factor graph of an LT code. The scheduling techniques discussed in the next section apply equally to the decoding of both codes.

2.4 Scheduling Techniques for Decoding Rateless Codes

Numerous scheduling techniques for BP decoding on factor graphs have been discussed in literature [11, 18]. This section reviews the two main scheduling techniques that

can be used for decoding rataless as well fixed-rate codes, for example in LT, Raptor and LDPC codes. First, we discuss the simpler case, which is called "flooding" or "parallel" message-passing. We then describe IDS, which is a version of "sequential" message-passing. Decoding with a flooding schedule, being the most widely used method, will be the bench mark to which we compare our proposed methods in simulations.

2.4.1 Flooding (FL) Schedule for BP Decoding

Generically, BP involves the exchange of a-posteriori likelihood messages between nodes (i.e. input and output symbols) of the factor graph [11]. The flooding schedule consists of several "iterations" of message-passing. Before describing this procedure, let us define the following:

- $\mathcal{N}(v_j)$, called the neighborhood of v_j, is the set of all check nodes connected to variable node v_j,
- $\mathcal{N}(c_i)$, called the neighborhood of c_i, is the set of all variable nodes connected to check node c_i,
- $M_{v_j,c_i}^{(\ell)}$ denotes the message sent from the variable node v_j to the check node $c_i \in \mathcal{N}(v_j)$, in the ℓ^{th} iteration,
- $m_{c_i,v_j}^{(\ell)}$ denotes the message sent from the check node c_i to the variable node $v_j \in \mathcal{N}(c_i)$, in the ℓ^{th} iteration, and
- Z_{c_i} is the log-likelihood ratio (LLR) corresponding to the channel information

on the check node c_i. Note that for different channel models, different equations for calculating this LLR are utilized.

In every iteration, the following message update rules are applied in parallel to all variable (input) and check (output) nodes in the factor graph:

$$\tanh\left(\frac{m^{(\ell)}_{c_i,v_j}}{2}\right) = \tanh\left(\frac{Z_{c_i}}{2}\right) \prod_{\substack{v_x \in \mathcal{N}(c_i) \\ v_x \neq v_j}} \tanh\left(\frac{M^{(\ell)}_{v_x,c_i}}{2}\right), \tag{2.11}$$

$$M^{(\ell+1)}_{v_j,c_i} = \sum_{\substack{c_y \in \mathcal{N}(v_j) \\ c_y \neq c_i}} m^{(\ell)}_{c_y,v_j}\,. \tag{2.12}$$

Then the LLR of each variable node v_j is calculated as

$$\Theta^{(\ell)}_j = \sum_{c_i \in \mathcal{N}(v_j)} m^{(\ell)}_{c_i,v_j}\,, \quad \forall j \in \{1,...,k\}. \tag{2.13}$$

After a certain number of iterations a hard-decision is made on $\Theta^{(\ell)}_j$ for $j = 1, 2, \ldots, k$ to obtain the estimated data vector $(\widehat{v}_1, \widehat{v}_2, \widehat{v}_3,...,\widehat{v}_k)$. The way this hard-decision is made also depends on the channel model being used.

2.4.2 Informed Dynamic Schedule (IDS) for BP Decoding

Recently, various sequential schedules have been discussed in literature. Different from the parallel updates in the FL schedule, a sequential schedule prescribes a predefined sequence of variable-node [9] or check-node [24] updates. (Note that an "update" is a much smaller unit of computation than an "iteration".) Simulations as well as theoretical analysis demonstrated that sequential scheduling converges faster

than flooding when applied to decoding codes defined on factor graphs [18]. This increase in convergence speed does not increase the decoding complexity per iteration and leaves the error performance intact. Therefore, it can result in reduced overall decoding cost.

One version of sequential scheduling, called IDS, was recently presented in [2]. Unlike traditional sequential scheduling techniques which typically follow a pre-defined update schedule, IDS uses the latest available reliability information to re-schedule future updates (follows dynamic update schedule). IDS, therefore, has the ability to adaptively shift the BP *update focus* towards less reliable sectors of the factor graph, thus avoiding wastage of efforts in exchanging messages among nodes that have already reached high reliability. This leads to faster movement of the high reliability information within the decoding graph, which therefore results in faster convergence.

In IDS each message is associated with a residual, which is simply the absolute difference between its value before and after an update. For the use of IDS with rateless codes we consider the residuals $|m^{new}_{c_i,v_j} - m^{old}_{c_i,v_j}|$ for messages sent from a check node c_i to variable nodes $v_j \in \mathcal{N}(c_i)$, where $m^{old}_{c_i,v_j}$ and $m^{new}_{c_i,v_j}$ are the messages before and after an update, respectively. Each check node, c_i, is then assigned the ordering metric of the largest residual among all those of $v_j \in \mathcal{N}(c_i)$ as follows

$$r_{c_i} = \max_{v_j \in \mathcal{N}(c_i)} \left| m^{new}_{c_i,v_j} - m^{old}_{c_i,v_j} \right|. \tag{2.14}$$

The check node which maximizes this metric is deemed to provide the largest innovation and is hence selected for the next update. In IDS, the ordering metrics for all

check-nodes are calculated according to equation (2.14) and stored in a sorted queue Q. Then, an *update* consists of the following:

- The node c_i with the highest ordering metric, r_{c_i}, value is selected and messages to all $v_j \in \mathcal{N}(c_i)$ are calculated and propagated according to equation (2.11),
- this same metric, r_{c_i}, is then set to zero and Q is re-ordered. The only residuals that change after the updates corresponding to c_i are those of $v_j \in \mathcal{N}(c_i)$,
- equation (2.12) is then used to calculate all messages from these variable nodes to the check nodes in $\mathcal{N}(v_j)$ (except c_i) for each $v_j \in \mathcal{N}(c_i)$,
- the ordering metrics of these check nodes are re-calculated and re-ordered in Q, and the one with the highest value is selected for the next update.

The above steps are repeated n times, where n is the number of check symbols in the factor graph. These n updates count as a full iteration. Iterations are executed until a maximal pre-defined number of iterations is reached or a stopping criterion is employed.

Note that unlike the case of a flooding schedule, with IDS, it is legitimate to speak of a *fraction* of an iteration. A *fraction* of an iteration can be just few updates (less than n). This concept will be significant when we introduce our novel stopping criterion, which can stop an iteration before its end, in the next Chapter.

2.5 Incremental Decoding (ID) of Rateless Codes

Having discussed two techniques used in scheduling BP decoding, we now discuss a method of retaining information from one decoding attempt to another which was presented in [8] and termed Incremental Decoding (ID).

Due to the dynamic nature of rateless LT and Raptor codes' factor graph, instead of starting from scratch upon the arrival of every new batch of check symbols as in message-reset decoding (MRD), ID *continues* decoding taking into account the results of the previous decoding attempt. For the time being assume that a FL schedule is used (cf, Section 2.4.1) and hence we call these two techniques MRDFL and IDFL.

Generally, both MRD and ID techniques proceed in several decoding attempts. The first attempt, which is identical for both methods, starts when $n_{min} \approx k/C$ check symbols are available (cf. Section 2.2). The later decoding attempts, which are executed differently by ID and MRD, commence at the arrival of each new batch of n_{inc} check symbols. A decoding attempt consists of several iterations. In each iteration, all variable and check-nodes are updated according to equations (2.11) and (2.12). Instead of initializing messages to zero at the beginning of each attempt as done in MRD, ID utilizes the information produced in the previous attempt. One way of realizing this [8] is to initialize the value of a message in a decoding attempt to its corresponding value from the last iteration of the previous attempt as follows:

$$m_{c_i,v_j}^{(1,\kappa)} = m_{c_i,v_j}^{(\ell_{last},\kappa-1)}, \tag{2.15}$$

where $m_{c_i,v_j}^{(\ell,\kappa)}$ is the message passed from the check-node c_i to the variable-node v_j in

the ℓ^{th} iteration of the κ^{th} decoding attempt. This utilization of existing reliability information results in reduced decoding cost.

We will refer to this method simply as ID in the following. In the next chapter, we also introduce an alternative method of retaining information from one attempt to another.

2.6 Stoping Criterion for Decoding Rateless Codes

In the case of fixed-rate codes such LDPC codes, the check-sum satisfaction test is used to terminate the decoding process and a pre-defined number of iterations is applied in each attempt. The test uses the following metric

$$\mu = \frac{C_S}{N_c}, \tag{2.16}$$

where C_S is the number of satisfied check node equations (a check-node equation is satisfied when the exclusive-or sum of its variable node edges is equal to zero) and N_c is the number of check nodes in the factor graph. The test is satisfied when $\mu \geq \Gamma$ where Γ is a user-defined threshold. Γ is usually set to 1 (i.e. stop decoding when all check-node equations are satisfied). Typically, a maximal number of iterations is applied in each attempt and the entire decoding process is halted when the aforementioned test is satisfied as in decoding LDPC codes.

To achieve minimal decoding cost, a designated criterion can be used to stop

the decoding attempt before reaching the maximal number of iterations. For this, we cannot simply use the test given in (2.16) because full satisfaction of check-sum equations cannot be achieved all the time in the early decoding attempts. Therefore, a different Γ value is required for each attempt, which is rather tedious to optimize for each different code and attempt specially when applied to rateless codes, where N_c is dynamic. To address this, we present an alternative stopping criterion in the next Chapter which can stop a decoding attempt at a fraction of an iteration when applied with IDS.

It is important to note that in practice, CRC bits are utilized to indicate the correctness of the decoded word and hence to stop the decoding process entirely.

Chapter 3

Decoding with Early Termination for Rateless Codes

3.1 Introduction

In this chapter, we consider the problem of decoding rateless codes with early termination, and thus low complexity. In particular, we concentrate on the example of LT and Raptor codes. To achieve early termination, fast convergence of BP decoding and a criterion for termination, commonly referred to as a stopping criterion, are needed. The former problem can be tackled using two, not necessarily mutually exclusive, approaches:

- Firstly, and as was presented in Section 2.5, it has been suggested that instead of starting the decoding process from scratch during every decoding attempt, some information should be retained and utilized from previous decoding attempts [8]. This technique, called incremental decoding (ID), typically results in a significantly reduced number of iterations in the later decoding attempts.

- Secondly, as is well known for the case of fixed-rate codes, convergence can be improved by clever design of the schedule of message passing. For LDPC

codes a method called informed dynamic scheduling (IDS) has recently been proposed in [2] and was discussed in Section 2.4.2. In this procedure, only unreliable subsets of the factor graph are required to update their messages, while much of the graph remains quiescent. This results in a greatly reduced number of updates compared to the often used flooding schedule, and thus faster convergence.

The principal contributions of this chapter, therefore, are:

1. The *combined* application of the concepts of incremental decoding with IDS to the decoding problem of rateless codes.

2. The formulation of a novel stopping criterion for early stopping to minimize decoding complexity.

As for the first contribution, we present an alternative way of retaining information from one decoding attempt to the next, different from ID, which was discussed in Section 2.5. We term this method Biased Incremental Decoding (BID). Then we apply this method with IDS for decoding rateless codes. We also combine ID with IDS. This results in two novel techniques for efficient decoding of rateless codes: Incremental Decoding with Informed Dynamic Scheduling (IDIDS) and Biased Incremental Decoding with Informed Dynamic Scheduling (BIDIDS).

As for the second contribution, we note that, to the best of our knowledge, the existing literature on rateless codes does not explicitly address the problem of when to stop a decoding attempt. We therefore propose a new hybrid stopping criterion that is sensitive enough to terminate decoding within a fraction of an iteration, thus

extracting the maximum possible efficiency of BID/ID with IDS (the notion of a fraction of an iteration is explained in Section 2.4.2). The simulation results show that the devised decoding schemes with early termination achieve considerable reductions in decoding complexity compared to the conventional decoder, where complexity is measured in terms of total number of decoding iterations over all decoding attempts.

3.2 Biased Incremental Decoding (BID)

Having introduced ID, we propose a different way of retaining information from one decoding attempt to the next, BID. Instead of initializing messages, the estimates (or belief) of the variable nodes v_j from the previous decoding attempt are used as a-priori information, or bias, in the next decoding round. However, to avoid stalling of message passing due to positive coupling, a forgetting factor α needs to be applied. Therefore, all variable messages are initialized with the scaled bias at the start of each decoding attempt as follows

$$M_{v_j,c_i}^{(1,\kappa)} = \alpha\Theta_j^{(\ell_{last},\kappa-1)}, \quad for\ all\ c_i \in \mathcal{N}(v_j), \tag{3.1}$$

where $\Theta_j^{(\ell_{last},\kappa-1)}$ is the LLR of variable-node v_j calculated with the equation (2.13) in the last iteration of the $(\kappa-1)^{th}$ decoding attempt and $0 \leq \alpha < 1$, is used in the κ^{th} decoding attempt. We will call this technique BIDFL when a flooding schedule is used.

3.3 Combination of ID/BID with IDS

We now combine ID/BID with IDS to synergize their computational advantages. Together, this accomplishes BP decoding with early termination and thus reduces complexity compared to the conventional BP decoding approach, i.e., message-reset decoding with flooding schedule (MRDFL). The combination of ID and BID with IDS results in two novel techniques for decoding rateless codes: IDIDS and BIDIDS.

In IDIDS the decoder follows the ID procedure for propagating information from one decoding attempt to the next, as discussed in Section 2.5, and in each decoding attempt it uses the schedule defined by IDS to update the check nodes. Moreover, IDIDS also retains the ordering metric queue Q from the previous decoding attempt.

BIDIDS applies the BID procedure for retaining information across successive decoding attempts, and employs IDS to update check nodes. We note that when applying IDS to the problem of decoding rateless codes, which are defined through their generator matrix rather than through their parity-check matrix as LDPC codes, an initial flooding iteration is required to initialize the node ordering metrics r_{c_i} defined in (2.14).

The pseudo code for the IDIDS and BIDIDS decoding algorithms is given in Table 3.1. Both methods employ the new hybrid stopping criterion described in the following section.

Table 3.1: Pseudo code for IDIDS/BIDIDS algorithms for decoding Rateless codes

1:	Collect n_{min} received samples
2:	Perform one flooding iteration
3:	Compute all r_{c_i} (2.14) and generate a sorted Q
4:	*// do one decoding attempt*
5:	Initialize messages according to (2.15) for IDIDS or (3.1) for BIDIDS
6:	**for** $\ell = 1 \ldots \ell_{max}$ **do** *// one iteration*
7:	**for** $k = 1 \ldots n/w$ *// one round of w updates*
8:	Perform w IDS updates
9:	Calculate $\mu_{\mathcal{L}}$ (3.3)
10:	**if** stopping criterion (3.4) is satisfied **then** go to line 15
11:	**end for**
12:	Calculate $\mu_{\mathcal{H}}$ (3.2)
13:	**if** stopping criterion (3.4) is satisfied **then go to** line 15
14:	**end for**
15:	**if** CRC is satisfied **then go to** line 18
16:	Collect n_{inc} new received samples
17:	**Go to** line 4
18:	Terminate decoding

3.4 A Hybrid Stopping Criterion for IDIDS and BIDIDS

Instead of using the metric μ in (2.16), we present a novel hybrid stopping criterion for terminating a decoding attempt in IDIDS and BIDIDS. This is necessary for the following reasons:

1. Though the metric μ is useful for fixed-rate codes, it cannot be used for rateless codes because the value of Γ needs to be adjusted when the code length and rate are varied.

2. In IDS an iteration is a set of several successive updates. If a satisfactory reliability is reached for a fraction of an iteration, it would be efficient to stop the decoder at that point before completing a full iteration. Therefore, an update-sensitive criterion is necessary.

The hybrid stopping criterion suggested in this section is composed of two independent detectors: $\mathcal{H}$ and $\mathcal{L}$, each of which functions optimally in different regions of the instantaneous code rate R. Detector $\mathcal{H}$ measures the check-sum satisfaction *difference* over an interval of an iteration with the metric $\mu_{\mathcal{H}}$ which we define as

$$\mu_{\mathcal{H}} = \left| \left(\frac{C_S}{n}\right)_{\ell} - \left(\frac{C_S}{n}\right)_{\ell-1} \right|. \tag{3.2}$$

Detector $\mathcal{L}$, in contrast, employs the metric $\mu_{\mathcal{L}}$ which measures the repetitiveness of variable nodes being updated between two consecutive intervals of w updates of check nodes in one iteration. Let V_i be the set of all variable-nodes updated during

interval number i. Similarly, let V_{i+1} be the set of variable nodes updated during the interval number $i+1$. We define $\mu_{\mathcal{L}}$ as

$$\mu_{\mathcal{L}} = \frac{|V_i \bigcap V_{i+1}|}{|V_{i+1}|}, \tag{3.3}$$

where $|V_{i+1}|$ is the cardinality of V_{i+1}. For instance, in Figure 2.2, let $w = 2$ and suppose at interval i, the check nodes c_1 and c_2 are selected and hence the variable nodes (v_1, v_2, v_3, v_4) in their neighborhood are updated. During the next interval, $i+1$, the check nodes c_5 and c_7 are selected and the variable nodes (v_2, v_5) are updated. Therefore, $\mu_{\mathcal{L}} = 1/2$ since only v_2 is repetitively updated (note that during w check-node updates, it is possible that some check nodes are selected more than once). Then we formally define our hybrid stopping criterion as

$$[\mu_{\mathcal{H}} \leq \gamma_{\mathcal{H}}] \text{ or } [\mu_{\mathcal{L}} \geq \Gamma_{\mathcal{L}}], \tag{3.4}$$

where $\gamma_{\mathcal{H}}$ and $\Gamma_{\mathcal{L}}$ are user-defined thresholds *independent* of the code rate and length. Note that $\mu_{\mathcal{H}}$ is sampled at every iteration whereas $\mu_{\mathcal{L}}$ is sampled at every interval of w updates within one iteration.

The hybrid stopping criterion essentially bridges the two very different convergence behaviors exhibited by the IDIDS/BIDIDS decoder in two distinct regions:

- High-rate region: information is scarce and typically several iterations are executed before the messages converge. Hence there is little incentive for stopping within a fraction of an iteration. Instead, a good indication of convergence is given by the fact that the status of check-sum satisfaction is not changing

appreciably, which is detected by the metric $\mu_{\mathcal{H}}$.

- Low-rate region: convergence is rapid, often within a single iteration, and hence it is worthwhile to try and stop in a fraction of an iteration. This cannot be accomplished by $\mu_{\mathcal{H}}$. Instead, we need a more fine-grained detector. From simulations we observed that in the low-rate region, convergence implies repetitive updates confined to a small set of variable nodes, often in the form of a limit-cycle. This phenomenon is sensed by the metric $\mu_{\mathcal{L}}$. However, since the detector $\mathcal{L}$ is not efficient in the high-rate region, where the order of message updates looks practically pseudo random, we need to use a hybrid combination of detector $\mathcal{H}$ and $\mathcal{L}$ as in equation (3.4) to achieve early termination with maximal efficiency. For simplicity, we may set $\gamma_{\mathcal{H}} = 0$ and $\Gamma_{\mathcal{L}} = 1$. This implies that $\mu_{\mathcal{H}}$ checks for strict invariance in check-sum satisfaction whereas $\mu_{\mathcal{L}}$ checks for an update limit-cycle.

3.5 Computational Complexity

The ultimate goal of the work presented in this chapter is to reduce the overall cost of decoding rateless codes by reducing the number of BP iterations required. For fair comparison between IDIDS/BIDIDS and other existing techniques, we try to keep the cost per iteration equal. In both MRDFL and BIDFL/IDFL (FL refers to the use of a flooding schedule), message-update operations are the major contributor to the overall cost. However, in sequential techniques such as IDIDS/BIDIDS, on top of the message-update operations, the following two additional operations contribute to

the overall cost:

1. Calculating the residual, r_{c_i}, associated with each check node c_i.

2. Ordering these metrics in a queue Q.

As for the problem of computing residuals, we employ the min-BP approximation

$$m'_{c_i,v_j} = \prod_{\substack{v_x \in \mathcal{N}(c_i) \\ v_x \neq v_j}} \text{sign}\,(M_{v_x,c_i}) \min_{\substack{v_x \in \mathcal{N}(c_i) \\ v_x \neq v_j}} (|M_{v_x,c_i}|), \tag{3.5}$$

to compute approximate residuals $|m^{new}_{c_i,v_j} - m^{old}_{c_i,v_j}|$ and approximate ordering metrics r_{c_i} in (2.14), cf. [2]. The approximate calculation of ordering metrics reduces the cost substantially without degrading the error-performance which was validated by our simulations.

The problem of ordering can also be handled with modest cost since we only need to sort the full queue Q once at the start of the decoding process. New metrics can then be inserted into Q progressively during the update process.

3.6 Simulation Results

To evaluate the effectiveness of our suggested methods, we obtain results through Monte Carlo simulations. We discuss the results for simulation conducted for the LT and Raptor codes separately. The following two sections present the examined methods, the simulation settings and the results.

3.6.1 Simulations for LT codes

In this section, we present simulation results to quantify the complexity savings achieved with the new decoding methods compared to the conventional MRDFL decoding applied to LT codes. As motivated above, we adopt the number of iterations until decoding is terminated as the measure of complexity.

In order to separate the effects of informed scheduling and incremental decoding, we compare the complexities of the following decoding schemes:

- MRDFL, IDFL: the conventional methods with flooding schedule. MRDFL will be the bench line, to which we compare our suggested methods.
- BIDFL(α): the incremental decoding technique with our suggested method of retaining information from one attempt to another, biased incremental decoding (α is the scaling factor)
- IDIDS(w): our method which combines incremental decoding with IDS (w is the update interval length)
- BIDIDS(w,α): our method which combines our suggested biased incremental decoding with IDS (scaling factor α and update interval length w)

As interesting examples, we assume the BSC with a capacity of $C = 0.50$ and $C = 0.33$ bit/(channel use), respectively, and the LT information-word length is chosen $k = 2500$. We employ the following degree distribution from [20, Table I]:

Table 3.2: Values for parameters w and α used in simulations.

	$C = 0.50$		$C = 0.33$	
	w	α	w	α
BID	N/A	0.2	N/A	0.01
IDIDS	500	N/A	750	N/A
BIDIDS	500	0.05	750	0.01

$$\begin{aligned}\Omega(x) &= 0.007969x + 0.493570x^2 + 0.166220x^3 \\ &\quad +0.072646x^4 + 0.082558x^5 + 0.056058x^8 \\ &\quad +0.037229x^9 + 0.055590x^{19} + 0.025023x^{65} \\ &\quad +0.003135x^{66} .\end{aligned} \tag{3.6}$$

The values for the parameters w and α have been empirically optimized for this LT code to minimize the total decoding cost and are shown in Table 3.2 for $C = 0.50$ and $C = 0.33$ bit/(channel use). In all cases, the first decoding attempt commences when k/C received samples are available, and a maximum number of 50 iterations is applied.

First, we consider the bit-error rate (BER) performances for the different decoding methods, since a reduction in decoding complexity should not come at the cost of increased error rates. Figures 3.1 and 3.2 show the BERs for the different decoders as a function of the inverse instantaneous rate R^{-1}. It can be seen that there are hardly any differences between the BER performances, which makes a fair comparison based on decoding complexity fairly straightforward. We note that the relatively large distance to the respective capacity limits, e.g., an overhead of about 10 % is required

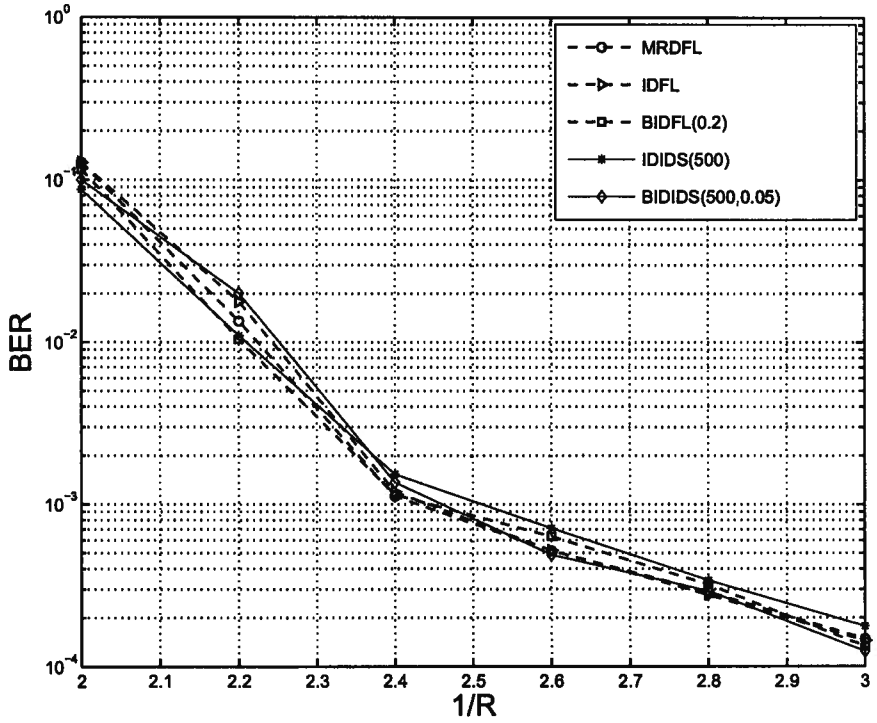

Figure 3.1: BER as a function of R^{-1} for different decoding methods. BSC with $C = 0.50$ bit/(channel use).

at a BER of 10^{-2}, can be reduced by increasing the word length k, while the error-floor behavior is alleviated by using an outer code [4, 17, 20] (this will be evident with the Raptor code simulations in the next section).

We now turn to the complexity comparison. Figures 3.3 and 3.4 plot the average total number of iterations as a function of R^{-1} for the BSC with $C = 0.50$ and $C = 0.33$ bit/(channel use), respectively. Considering the curves for MRDFL and IDFL we observe that ID can lower the number of iterations until termination by up to a factor of 6 compared to MRD. Similar complexity reductions are achieved with BID for the case of $C = 0.50$ bit/(channel use), while notably smaller gains are seen for the case of $C = 0.30$ bit/(channel use).

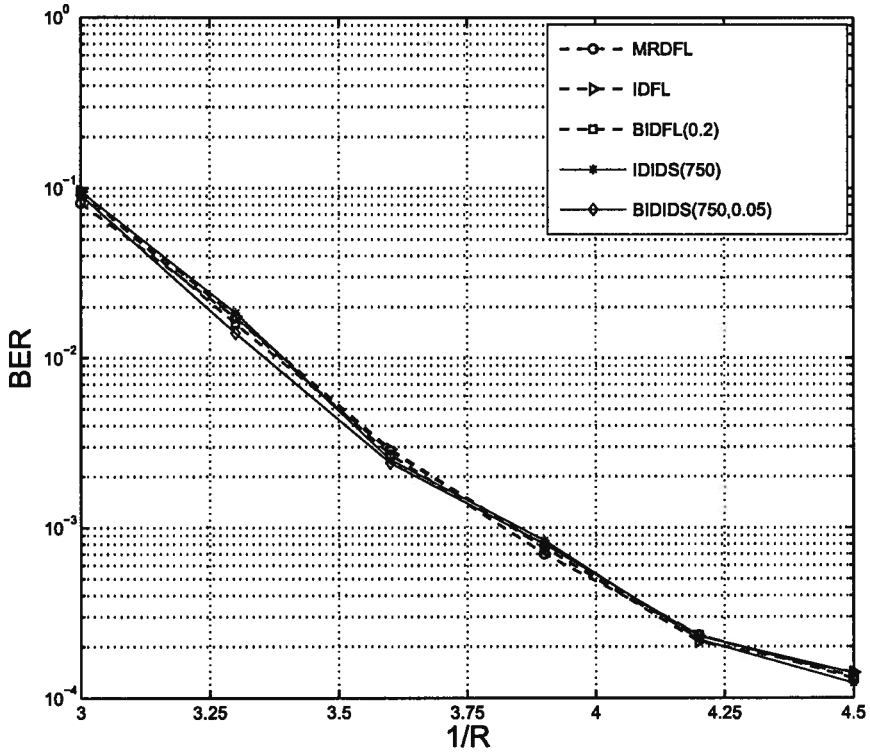

Figure 3.2: BER as a function of R^{-1} for different decoding methods. BSC with $C = 0.33$ bit/(channel use).

Table 3.3: Improvement factors in terms of number of iterations for the different algorithms compared to MRDFL. BSC with $C = 0.33$ bit/(channel use).

R^{-1}	IDFL	BID(0.01)	IDIDS(750)	BIDIDS(750, 0.01)
3.3	1.1	1.0	4.7	4.0
3.9	4.6	1.6	47.7	7.3
4.5	6.1	1.6	56.0	7.0

Table 3.4: Improvement factors in terms of number of iterations for the different algorithms compared to MRDFL. BSC with $C = 0.50$ bit/(channel use).

R^{-1}	IDFL	BID(0.2)	IDIDS(500)	BIDIDS(500, 0.05)
2.2	1.3	1.3	4.8	3.4
2.6	5.8	4.7	57.1	10.4
3.0	6.3	4.8	59.8	10.1

Complexity is further reduced if ID and BID are combined with IDS. In particular the new IDIDS method yields another improvement factor of 10, while the complexity gain with the BIDIDS method saturates earlier. We attribute this to the fact that freezing messages according to (2.15) allows a more structured retrieval of information than biasing in (3.1). We note that the new stopping criterion (3.4) and its ability to stop at a fraction of an iteration when employed with IDS are crucial for accomplishing early termination.

Tables 3.4 and 3.3 list the complexity reduction factors (in iterations) for the different algorithms when compared with MRDFL. Improvement factors in the range of $4-60$ can be achieved depending, through the instantaneous rate, on the error-rate performance (see Figures 3.1 and 3.2).

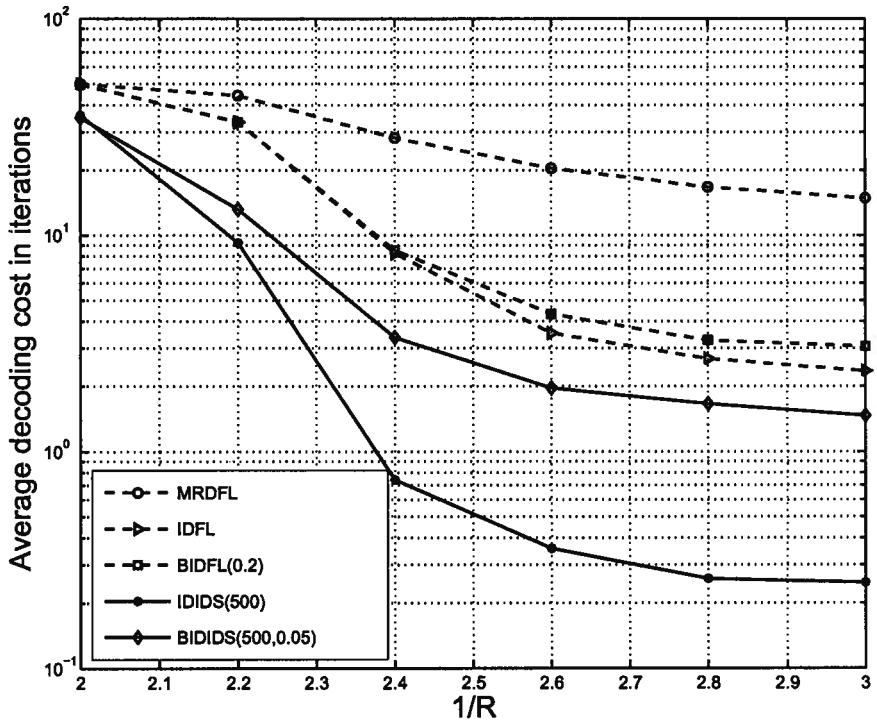

Figure 3.3: Complexity in terms of number of iterations as a function of R^{-1}. BSC with $C = 0.50$ bit/(channel use).

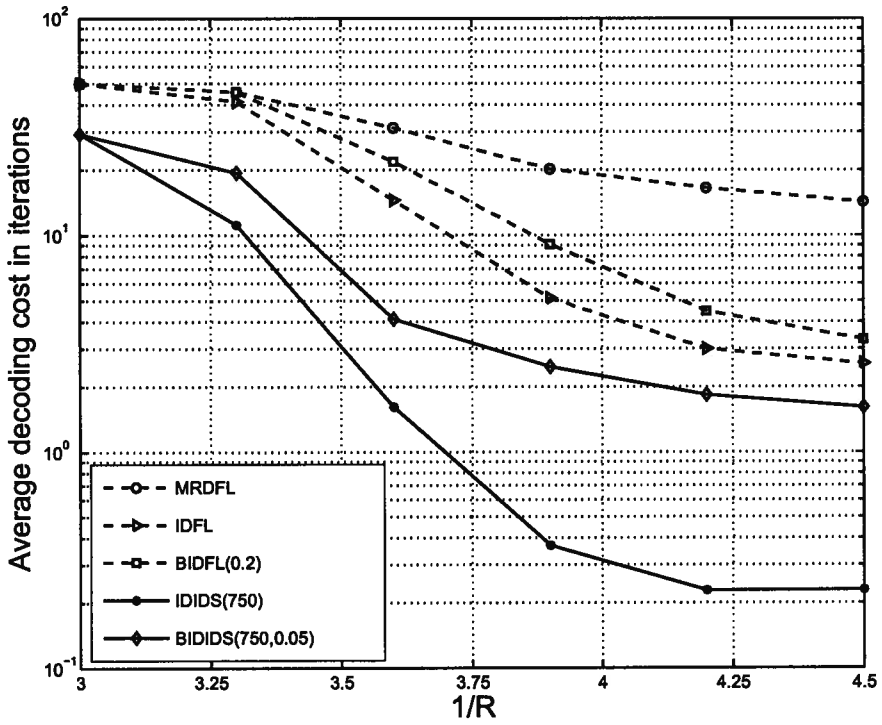

Figure 3.4: Complexity in terms of number of iterations as a function of R^{-1}. BSC with $C = 0.33$ bit/(channel use).

3.6.2 Simulations for Raptor Codes

In this section, we show quantitative results for the complexity reduction achieved by the proposed IDIDS method with the stopping criterion (3.4) when applied to the full Raptor code. Our measure of complexity is the total number of iterations until decoding is successful, i.e., cyclic-redundancy check is positive. Again we try to separate the effects of informed dynamic scheduling and incremental decoding, so we compare the complexities of the following decoding schemes:

- MRDFL($L_{\max}$, T)
- IDFL($L_{\max}$, T)
- IDIDS($L_{\max}$, T)

where T is the interval in number of samples between successive decoding attempts and $L_{\max}$ is the maximal number of iterations per attempt. We note that $L = L_{\max}$ was applied in previous works for Raptor codes (e.g. [8]), where L is the actual number of iterations in a decoding attempt.

For a fair comparison with IDIDS, we apply the check-sum satisfaction detector $\mathcal{H}$ to stop decoding attempts in MRDFL and IDFL. In IDIDS, the interval length $w = 100$ is chosen. As benchmark scheme, we consider MRDFL(100,100) without any termination, i.e., $L = L_{\max} = 100$, cf. [8]. As an interesting transmission example, we assume the BSC with capacity $C = 0.50$ bit/(channel use). The Raptor code consists of a rate-0.95 regular LDPC code and an LT code generated using the degree distribution in (3.6). The input-word length is chosen $k = 9500$ bits, and the first decoding attempt commences when k/C received samples are available.

First, we make sure that the different BP schedules and early termination do not affect the rate performance of Raptor codes. To this end, Figure 3.5 shows a scatter plot for the achieved rate for 50 transmitted words (i.e., the rate $R = k/n$ for which the transmitted word was decoded correctly). That is, for each point the x-value is the rate achieved with the benchmark scheme, MRDFL(100,100) without early termination, and the y-value is the rate achieved with one of the other decoding schemes for the same transmitted data and received samples. We consider both $T = 100$ and $T = 1$ for the ID schemes, and set $L_{\max} = T$ to fix the ratio $T/L_{\max}$ as in [8]. It can be seen that all measured points lie very close to the 45-degree line, which clearly manifests the equivalent performance achieved with the different decoders. In case of IDFL, this is consistent with the findings in [8], according to which only the ratio T/L determines the rate performance (assuming $L = L_{\max}$ in [8]).

Next, we compare decoding complexities. Figure 3.6 shows the scatter plot for the required number of iterations for the same 50 transmissions and decoding schemes that were considered in Figure 3.5. In addition, lines with different slopes are included to emphasize the approximate multiplicative gains in terms of complexity reduction. A number of observations can be made.

Firstly, the complexity of MRDFL can be reduced by a factor of two due to check-sum-satisfaction stopping. This is interesting in its own right, since only MRDFL without stopping was used in [8] for a comparison with IDFL. Secondly, we observe that IDFL accomplishes further notable complexity savings. IDFL(1,1) is somewhat advantageous over IDFL(100,100), which again is consistent with [8], where the use of $L = 1$ was advocated for incremental decoding. Thirdly, it can be seen that the

proposed IDIDS is by far the most efficient decoding scheme. The total number of iterations is only 13%, 27%, and 46% of those required for MRDFL without stopping (not shown in Figure 3.6), MRDFL with stopping, and IDFL(1,1), respectively. We note that the new stopping criterion (3.4) and its ability to stop at a fraction of an iteration when employed with IDS are crucial for accomplishing early termination and hence yielding these considerable reductions in complexity. Furthermore, since the number of message updates in IDIDS is adapted to the amount of new information available during each decoding attempt, the choice of the interval T between successive decoding attempts is not critical.

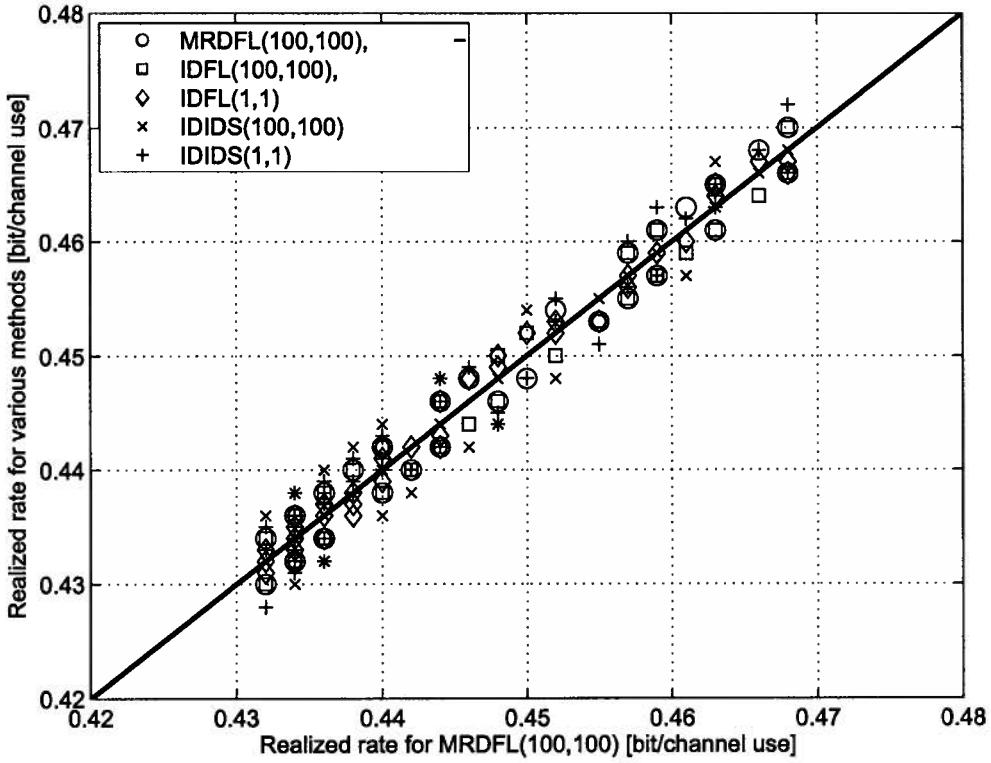

Figure 3.5: Scatter plot for the realized rate for various methods. Arguments $(L_{\max}, T)$ mean that a decoding attempt is made every T newly received samples with $L \leq L_{\max}$ iterations. Check-sum satisfaction stopping for MRDFL/IDFL. IDIDS uses early termination with (3.4). x-axis: MRDFL(100,100) with $L = L_{\max}$.

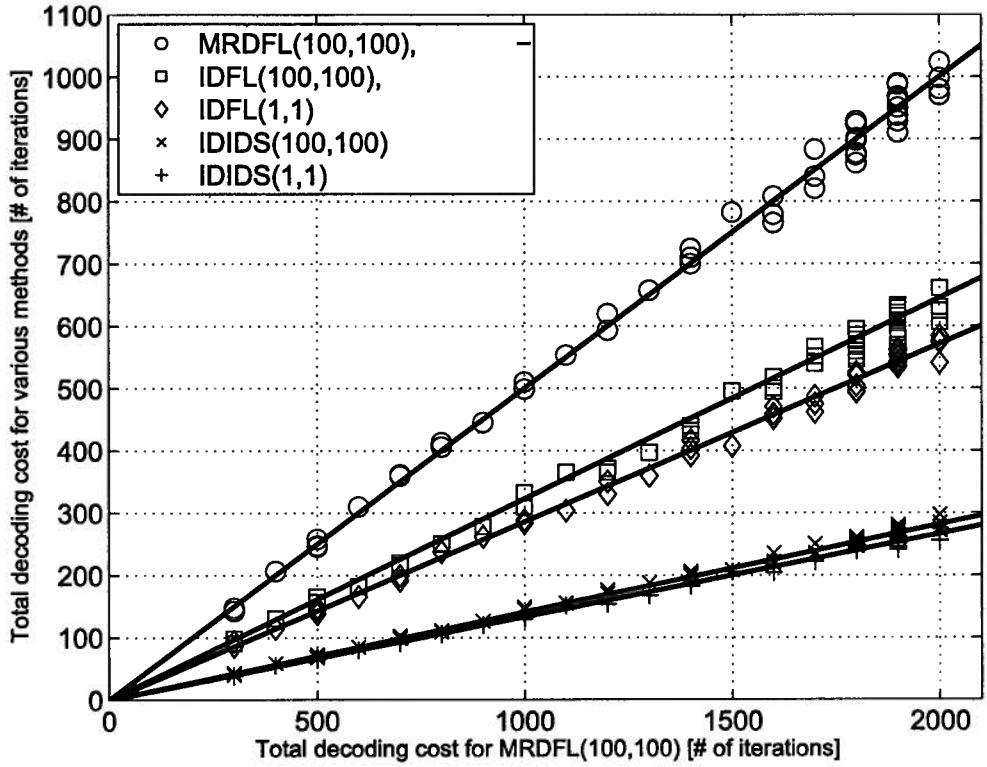

Figure 3.6: Scatter plot for the total cost in number of iterations until successful decoding. Arguments $(L_{\max}, T)$ mean that a decoding attempt is made every T newly received samples with $L \leq L_{\max}$ iterations. Check-sum satisfaction stopping for MRDFL/IDFL. IDIDS uses early termination with (3.4). x-axis: MRDFL(100,100) with $L = L_{\max}$.

Chapter 4

Rateless Coding for Hybrid Free-Space Optical and Radio-Frequency Communication

4.1 Introduction

Free Space Optics (FSO) refers to the transmission of modulated visible or infrared (IR) light beams through the atmosphere for optical communication. In this sense, it is similar to fiber, but instead of enclosing the data stream in a glass fiber, FSO uses light beams to transmit data through air. FSO communication systems have received renewed interest due to their low deployment costs and potential use in high-throughput applications like last mile access [1]. FSO transmits eye-safe light beams from one point to another using low-power IR lasers in the teraHertz spectrum. FSO light beams are directed to highly sensitive photodetector receivers. These receivers are telescopic lenses which can collect the photon stream carrying digital data used in applications such as video transmission, radio signals and internet connectivity. Transmission is directional, making it more secure than RF technologies though the

two points, transmitter and receiver, have to be within line-of-sight of each other.

FSO communication systems technology has been in development since the 1960s and was initially used in aerospace and military applications. The recent demand for higher bandwidth resulted in more interest in FSO systems and their applications. However FSO systems suffer from many factors, the most prominent of which are:

- Atmospheric turbulence, also known as scintillations: this results in light variations at the receiver side due to incoherence in the temperature and pressure of atmosphere [14].

- Scattering: this phenomenon is caused by fog. This is the most destructive factor impeding the development of FSO systems [14].

The above factors are typically modeled as a probabilistic process with carefully chosen distributions [15, 25].

To combat the deterioration of signal quality due to adverse atmospheric conditions, the use of error control codes and interleaving has been proposed in [3]. However, since temporal diversity is hardly available for high-speed terrestrial FSO links unless very large interleavers are employed, the utilization of spatial diversity at both the transmitter and receiver has been studied in [12, 23] as a means to mitigate the fluctuation of signal strength due to atmospheric turbulence. In this context, the use of space-time coding has been advocated in [6] and [26] for optical communication. However, none of the above diversity schemes can tolerate prolonged extreme weather conditions, in particular turbulence and scattering, which limits the availability of FSO links.

The only way to perform well when the optical channel suffers prolonged impairments is to use media diversity, as was proposed in e.g. [10] and [1], where an RF channel is used in conjunction with the FSO channel. The practical heuristic for such hybrid FSO/RF systems is that fog and rain, which affect FSO and RF, do rarely occur simultaneously [10]. Commercially available hybrid solutions use the RF link as a hot-standby backup for the FSO link, to be used only when the FSO channel is inoperative [1]. Needless to say, this approach to hybrid communication is inefficient because of duplication of the messages on the RF and FSO channels, since such repetition codes perform far from the overall system capacity. This inefficiency can be mitigated only via a non-trivial channel code for the overall hybrid channel.

There are two ways to approach this coding problem:

1. In the first approach, one assumes that the transmitter knows the exact channel conditions on both the RF and FSO links, and hence selects the best possible multiplexing ratio (which defines the portion of data to be transmitted on each link) and code rates to exploit the capacity of the hybrid channel under those specific channel conditions. Fixed-rate codes, such as LDPC codes, can be used in this case. This approach, however, relies on significant feedback from the receiver to the transmitter and reconfigurable coding and/or modulation schemes (adaptation of signal bandwidth is typically not a practical option for a hybrid FSO/RF system). Recently, [22] (cf. also [21]) has presented such a coding scheme based on non-uniform punctured LDPC codes, where the code rates for FSO and RF transmission are adjusted according to the instantaneous channel conditions.

2. The second approach, which we propose in this chapter, aims to achieve a close-to-optimal throughput under a broad spectrum of channel conditions, while using very little feedback. Thus we do not adapt the transmitter to change the relative utilization of the sub-channels (fixed multiplexing ratio). However our receiver does adapt to the channel variations and, by sending infrequent low-rate feedback, achieves a throughput close to capacity under any channel condition with the given multiplexing ratio. We achieve this robustness by using a practical implementation of rateless Fountain codes [13]. In particular, we provide simulative evidence that a single moderate-length Raptor code design [20] enables us to closely approach the associated capacity limit of the hybrid FSO/RF channel under varying channel conditions. Our results are in line with the findings in [4, 17] that Raptor codes designed for the binary erasure channel perform remarkably well for BSC and AWGN channels too, even though they are provably not universal for these classes of channels. Finally, we remark that the proposed rateless coding scheme still leaves the door open for the possible adaptation of the transmitter multiplexing ratio. For example, the constellation size for RF transmission could be modified for coarse adaptation to the channel condition, while rateless coding accomplishes adaptation on a finer (bit-wise) grain.

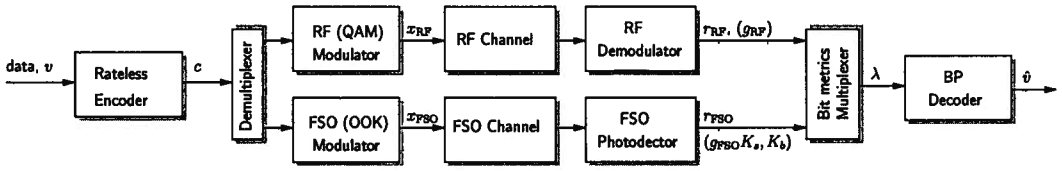

Figure 4.1: Block diagram of the proposed coded hybrid FSO/RF transmission system.

4.2 Transmission Model and Proposed Coding Scheme

In this section, we first introduce the channel model and signal schemes used for hybrid FSO/RF transmission and then describe the proposed coding scheme. The block diagram of the coded hybrid transmission system is shown in Figure 4.1.

4.2.1 Transmission Model

The transmission model consists of a transmitter, channel and a receiver and is described in details in the following sections.

4.2.1.1 Transmitter

The considered hybrid system consists of two transmitter-receiver pairs, one for FSO transmission and another for RF transmission.[1] Both transmitters receive a bit stream from a binary encoder, which is specified in detail in Section 4.2.2. The two transmitters can be described as follows:

- The RF transmitter maps $m_{\mathrm{RF}} = \log_2(M)$ binary symbols to an M-ary quadra-

[1]Extensions to multiple FSO or RF transceivers are straightforward.

ture amplitude modulation (M-QAM) signal point x_{RF} using Gray mapping. The QAM constellation is typically non-binary, i.e., $M > 2$, for bandwidth-efficient transmission. The signal elements x_{RF} are sent with a baud rate of $1/T_{\mathrm{RF}}$ symbols per second using conventional pulse shapes such as a root-raised cosine.

- The FSO transmitter employs intensity modulation with on-off keying (OOK), which is the prevalent modulation format for FSO communication. The signal elements are $x_{\mathrm{FSO}} \in \{0, 1\}$ and the optical signal intensity is P_{FSO} during the on-period and zero during the off-period, each of which is of length T_{FSO}. Obviously, one data bit is represented by one OOK symbol, i.e., $m_{\mathrm{FSO}} = 1$.

4.2.1.2 Channel and Receiver

The FSO receiver applies direct detection. To formulate the transmission model, we adopt the often used photon-count model for direct detection with an ideal photodetector where the detector output r_{FSO} has a Poisson count distribution with mean parameter μ_{FSO} which is given by

$$\mu_{\mathrm{FSO}} = \begin{cases} g_{\mathrm{FSO}} K_s + K_b & \text{for} \quad x_{\mathrm{FSO}} = 1 \\ K_b & \text{for} \quad x_{\mathrm{FSO}} = 0 \end{cases} \tag{4.1}$$

The constants K_s and K_b are given by

$$K_s = \eta P_{\mathrm{FSO}} T_{\mathrm{FSO}} / (hf), \tag{4.2}$$

$$K_b = \eta P_b T/(hf), \tag{4.3}$$

where η is the efficiency of the photodetector, f denotes the center frequency of the transmission, h is Planck's constant, and P_b represents the power from background radiation incident on the photodetector [5], [23]. The variable g_{FSO} accounts for signal attenuation and random signal intensity fluctuation due to scintillation. While a number of statistical scintillation models exist in the literature [23, 25] and also attenuation is time varying due to changing weather conditions, e.g., conditions of fog and smoke, in the context of this work we are content to notice that the coherence time for g_{FSO} is much larger than the time horizon for one codeword.

The RF system uses a line-of-sight link in the microwave or millimeter wave band and it is therefore well modeled by the classical AWGN channel with a gain factor $\tilde{g}_{\text{RF}}$. Again, the time variation of $\tilde{g}_{\text{RF}}$ due to e.g. occurrence of rain is extremely slow compared to the transmission speed in units of codewords. The RF receiver consists of a classical matched-filter front-end followed by baud-rate sampling. That is, the sampled receiver output in the equivalent complex baseband is written as

$$r_{\text{RF}} = g_{\text{RF}} x_{\text{RF}} + n_{\text{RF}} \,, \tag{4.4}$$

where $g_{\text{RF}} = \tilde{g}_{\text{RF}} e^{j\phi}$ denotes the complex channel gain, n_{RF} is complex AWGN with variance $P_n = \mathcal{N}_0/T_{\text{RF}}$ and $\mathcal{N}_0$ is the one-sided noise power spectral density.

The FSO/RF receiver passes the received samples r_{FSO} and r_{RF} along with the parameters $(g_{\text{FSO}} K_s, K_b)$ for the FSO channel, and (g_{RF}, P_n) for the RF channel to

the bit-metric calculator (see Figure 4.1).

4.2.2 Proposed Coding Scheme

4.2.2.1 Encoder and Multiplexing

The hybrid FSO/RF system employs a single encoder of a binary rateless code. More specifically, we apply the Raptor codes which were introduced in Chapter 2, whose encoder was discussed in Section 2.2.3. The code bits are then demultiplexed into two bit-streams, one entering the RF QAM modulator and the other one being input to the FSO OOK modulator. To be more specific, let us define the positive, mutually prime integers n_{FSO} and n_{RF} as

$$\frac{n_{\mathrm{FSO}}}{n_{\mathrm{RF}}} = \frac{m_{\mathrm{FSO}}/T_{\mathrm{FSO}}}{m_{\mathrm{RF}}/T_{\mathrm{RF}}} \,. \tag{4.5}$$

Then, out of a block of $n_{\mathrm{tot}} = n_{\mathrm{FSO}} + n_{\mathrm{RF}}$ bits $[c_i, c_{i+1}, \ldots, c_{i+n_{\mathrm{tot}}-1}]$, the first n_{FSO} bits are passed to the FSO modulator and the remaining n_{RF} bits are RF modulated. The formulation in (4.5) makes the mild assumption that $m_{\mathrm{FSO}}T_{\mathrm{RF}}/(m_{\mathrm{RF}}T_{\mathrm{FSO}})$ can be expressed as a rational number.

Encoding of the message $[v_1, v_2, \ldots, v_k]$ is terminated when a positive acknowledgment of successful decoding has been received.

4.2.2.2 Metric Computation and Demultiplexing

Based on the received samples and the channel parameters, decoding metrics in the form of log-likelihood ratios are generated.

For FSO transmission with the channel and detector model described in Section 4.2.1, the log-likelihood ratio (LLR) associated with x_{FSO} can be written as

$$\lambda_{\mathrm{FSO}} = r_{\mathrm{FSO}} \log\left(1 + \frac{K_s}{K_b}\right) - K_s \,. \tag{4.6}$$

For RF transmission, the LLR for the ℓth bit mapped to x_{RF} is given by

$$\lambda_{\mathrm{RF},\ell} = \max_{x_{\mathrm{RF}} \in \mathcal{X}_{\ell,0}} \left(|r_{\mathrm{RF}} - g_{\mathrm{RF}} x_{\mathrm{RF}}|^2\right) - \max_{x_{\mathrm{RF}} \in \mathcal{X}_{\ell,1}} \left(|r_{\mathrm{RF}} - g_{\mathrm{RF}} x_{\mathrm{RF}}|^2\right) , \tag{4.7}$$

$0 \leq \ell \leq m_{\mathrm{RF}} - 1$, where $\mathcal{X}_{\ell,b}$ denotes the subset of the QAM signal constellation for which the ℓth bit of the label is equal to $b \in \{0, 1\}$.

According to the demultiplexing of code bits described above, groups of λ_{FSO} and $\lambda_{\mathrm{RF},\ell}$ are multiplexed into one stream of LLRs $[\lambda_1, \lambda_2, \ldots, \lambda_n]$, which then is forwarded to the decoder (see Figure 4.1).

4.2.2.3 Decoder

The decoder for the rateless code collects LLRs λ_i progressively and a first decoding attempt is made when it is believed that sufficient information for successful decoding has been collected. A good estimate for the number n of collected LLRs is given by

$$n = \frac{k(1+\epsilon)}{C_{\mathrm{H}}} \,, \tag{4.8}$$

where C_{H} denotes the constrained capacity of the hybrid FSO/RF channel as defined in Section 4.3, and $\epsilon > 0$ is an overhead factor which accounts for the suboptimality

of the (finite-length) Raptor code with respect to capacity.

Decoding is performed using the well-known BP on the factor graph as discussed in Section 2.3. The variable nodes c_i are initialized with the LLRs λ_i, $1 \leq i \leq n$, and extrinsic likelihood messages are exchanged between the variable and check nodes accordingly to a pre-defined schedule. If the decoded message $[\hat{v}_1, \hat{v}_2, \ldots, \hat{v}_k]$ after BP decoding is deemed unreliable, which could be determined by either making use of a cyclic redundancy check (CRC) code or estimating the bit-error rate (BER) based on the final LLRs [7], a new decoding attempt is scheduled after a new batch of n_{inc} samples has been received. The value of n_{inc} will depend on the update schedule applied for BP decoding. Notice that we only use MRDFL for decoding in this Chapter.

Once the decoded message $[\hat{v}_1, \hat{v}_2, \ldots, \hat{v}_k]$ is deemed correct, decoding is terminated and a 1-bit acknowledgment signal is transmitted to the encoder.

4.3 Constrained Capacity of the Hybrid FSO/RF Channel

In order to benchmark the proposed coding scheme, we now specify the information-theoretic limit for the hybrid FSO/RF channel.

To this end, we first note that the FSO and RF modulation formats are not optimized for maximal mutual information and assumed constant regardless of the channel condition. While the former assumption is rooted in the practical constraints for FSO and RF transmission technology, the latter could be relaxed assuming that

a more powerful feedback channel is available in order to perform adaptation of the signal constellation for the RF channel. However, as long as the size M of the QAM constellation is chosen such that m_{RF} is at least 1 bit above the Shannon capacity of the RF AWGN channel, hardly any gains are achievable with a larger alphabet size [16]. Hence, choosing a reasonably large constellation size, say 64QAM, is sufficiently optimal.

Secondly, we note that we are not interested in the capacity in the sense that coding with a rate strictly lower than capacity achieves an arbitrarily low error rate for every FSO and RF channel realization, i.e, every $(g_{\mathrm{FSO}}K_s, K_b)$ and (g_{RF}, P_n). Instead of this compound channel capacity we are interested in the capacity where the transmitter is allowed to adapt its codebook according to the current channel state.

Let us define the states s_{FSO}, s_{RF}, and s_{H} of the FSO, RF, and hybrid FSO/RF channel as follows:

$$s_{\mathrm{FSO}} = [g_{\mathrm{FSO}}K_s, K_b] \tag{4.9}$$

$$s_{\mathrm{RF}} = |g_{\mathrm{RF}}|^2/P_n \tag{4.10}$$

$$s_{\mathrm{H}} = [s_{\mathrm{FSO}}\, s_{\mathrm{RF}}] \tag{4.11}$$

Furthermore, let the mutually prime integers η_{FSO} and η_{RF} be defined by

$$\frac{\eta_{\mathrm{FSO}}}{\eta_{\mathrm{RF}}} = \frac{T_{\mathrm{RF}}}{T_{\mathrm{FSO}}}\,, \tag{4.12}$$

i.e., $\eta_{\text{FSO}}/\eta_{\text{RF}}$ is the relative frequency of FSO and RF channel usages. Then, since the FSO and RF channels are used in parallel, the channel capacity for given s_{H} can be written as

$$C_{\text{H}}(s_{\text{H}}) = \frac{1}{\eta_{\text{FSO}}} \left[\eta_{\text{FSO}} C_{\text{FSO}}(s_{\text{FSO}}) + \eta_{\text{RF}} C_{\text{RF}}(s_{\text{RF}})\right] , \tag{4.13}$$

where normalization with η_{FSO} renders the capacity unit "bit per FSO channel use", and $C_{\text{FSO}}(s_{\text{FSO}})$ and $C_{\text{RF}}(s_{\text{RF}})$ respectively denote the constellation-constrained capacities of the FSO and RF channel in bit per FSO and RF channel use for a given channel state. These are given by

$$C_{\text{FSO}}(s_{\text{FSO}}) = 1 - \mathbb{E}\left\{1 + \exp\left((-1)^{x_{\text{FSO}}} \lambda_{\text{FSO}}\right)\right\} , \tag{4.14}$$

$$C_{\text{RF}}(s_{\text{RF}}) = m_{\text{RF}} - \sum_{\ell=1}^{m_{\text{RF}}} \mathbb{E}\left\{1 + \exp\left((-1)^{b_\ell} \lambda_{\text{RF},\ell}\right)\right\} , \tag{4.15}$$

where b_ℓ denotes the ℓth bit in the label of x_{RF}, and the expectation $\mathbb{E}\{\cdot\}$ is with respect to the joint probabilities $p(r_{\text{FSO}}, x_{\text{FSO}})$ and $p(r_{\text{RF}}, b_\ell)$, respectively. We note that neither $C_{\text{FSO}}(s_{\text{FSO}})$ nor $C_{\text{RF}}(s_{\text{RF}})$ lend themselves for closed-form expressions, and, hence, can be evaluated using Monte Carlo integration.

4.4 Simulation Results

In this section, we present numerical results and simulative evidence that the proposed coded hybrid FSO/RF scheme performs close to the information-theoretic limits. The considered hybrid system employs 64QAM modulation for RF transmission, which is

practically optimal as long as $C_{\mathrm{RF}}(s_{\mathrm{RF}}) \leq 5$ bit/(RF channel use) [16]. As specified in Section 4.2.1.1, OOK modulation is used for FSO signalling. Furthermore, we assume a wavelength of 1550 nm, a photodetector efficiency of $\eta = 0.5$, and a normalized background radiation of $P_b T = -170$ dBJ [23], which yields the average background photon count $K_b = 39$. The FSO channel quality is adjusted through variation of K_s, and the RF channel condition is determined by the signal-to-noise ratio (SNR) given by

$$\mathrm{SNR}_{\mathrm{RF}} = \frac{|g_{\mathrm{RF}}|^2 \mathbb{E}\{|x_{\mathrm{RF}}|^2\}}{P_n}. \tag{4.16}$$

For simulations, we apply a Raptor code consisting of a rate-0.95 regular LDPC code and an LT code generated using the degree distribution given in equation 3.6. The information word length is chosen as $k = 9500$, which renders the codeword length n moderate in all but very unfavorable channel conditions (simultaneously for both FSO and RF channel). We emphasize that the application of an off-the-shelf Raptor code is motivated by the results in [4, Section IV] and [17] for binary output symmetric channels and the lack of a universally optimal Raptor code. Decoding is continued until the correct codeword has been found, which emulates the use of a CRC outer code. The number n of check bits required for successful decoding is recorded and the realized rate R is determined according to

$$R = \frac{\eta_{\mathrm{FSO}} + \eta_{\mathrm{RF}}}{\eta_{\mathrm{FSO}}} \frac{k}{n}, \tag{4.17}$$

whose unit is bit per FSO channel use.

We are mainly interested in showing the ability of the coded hybrid FSO/RF system to perform consistently close to the capacity limit for

1. different FSO and RF channel conditions (s_{FSO}, s_{RF}) and thus different capacities $C_{\mathrm{FSO}}(s_{\mathrm{FSO}})$ and $C_{\mathrm{RF}}(s_{\mathrm{RF}})$, and

2. for various relative frequencies of channel usage (η_{FSO}, η_{RF}) and thus different multiplexing ratios.

For this purpose, we first consider different combinations of capacities $C_{\mathrm{FSO}}(s_{\mathrm{FSO}})$ and $C_{\mathrm{RF}}(s_{\mathrm{RF}})$, and adapt the relative frequencies of usage η_{FSO} and η_{RF} such that the total hybrid FSO/RF capacity $C_{\mathrm{H}}(s_{\mathrm{H}})$ from (4.13) is always 1 bit/(FSO channel use). For example, for $C_{\mathrm{FSO}}(s_{\mathrm{FSO}}) = 0.9$ bit/(FSO channel use) and $C_{\mathrm{RF}}(s_{\mathrm{RF}}) = 2.0$ bit/(RF channel use), we use $\eta_{\mathrm{FSO}} = 20$ and $\eta_{\mathrm{RF}} = 1$. Table 4.1 summarizes the parameters for the 54 considered setups. Figure 4.2 shows the realized rate R (in the form of a surface) from (4.17), averaged over 200 transmitted information words for each parameter set from Table 4.1, as a function of the capacities $C_{\mathrm{FSO}}(s_{\mathrm{FSO}})$ and $C_{\mathrm{RF}}(s_{\mathrm{RF}})$ of the component channels. We observe that the performance surface is nearly flat and that the capacity limit is closely approached by realized rates of about 90 % of capacity. That is, the same rateless code design performs consistently well for very different mixtures and qualities of FSO and RF channels. From this we conclude that the proposed coded system makes effective use of available mutual information regardless of whether it is provided through a Poisson or a Gaussian channel.

In a second experiment, we keep the relative frequency of FSO and RF channel usages constant, and vary the capacities $C_{\mathrm{FSO}}(s_{\mathrm{FSO}})$ and $C_{\mathrm{RF}}(s_{\mathrm{RF}})$ such that a rel-

Table 4.1: Considered channel capacities C_{FSO}, C_{RF} and channel usage ratios $\eta_{\text{FSO}}/\eta_{\text{RF}}$ defined in (4.12) for hybrid FSO/RF transmission such that the total capacity $C_{\text{H}} = 1$ bit/(FSO channel use). The corresponding realized rates are shown in Figure 4.2. (C_{RF} in [bit/RF channel use] and C_{FSO} in [bit/FSO channel use])

C_{FSO} \ $\frac{\eta_{\text{FSO}}}{\eta_{\text{RF}}}$	C_{RF} = 1	2	3	4	5	6
0.1	9/8	9/4	10/3	9/2	11/2	20/3
0.2	5/4	5/2	15/4	5/1	25/4	15/2
0.3	7/5	20/7	13/3	23/4	14/2	17/2
0.4	5/3	10/3	5/1	20/3	25/3	10/1
0.5	2/1	4/1	6/1	8/1	10/1	12/1
0.6	5/2	5/1	15/2	10/1	25/2	15/1
0.7	10/3	20/3	10/1	27/2	33/2	20/1
0.8	5/1	10/1	15/1	20/1	25/1	30/1
0.9	10/1	20/1	30/1	40/1	50/1	60/1

atively wide range of overall capacities $C_{\text{H}}(\boldsymbol{s}_{\text{H}})$ is scanned. In this way we examine the code's ability to self-tune to time-varying channel conditions which lead to different hybrid channel capacities. Twelve specific parameters combinations are given in Tables 4.2 and 4.3. For each combination we calculate the hybrid channel capacity $C_{\text{H}}(\boldsymbol{s}_{\text{H}})$ in (4.13) and simulate the realized rate R in (4.17). Figures 4.3 and 4.4 show the scatter plot of R versus $C_{\text{H}}(\boldsymbol{s}_{\text{H}})$. Also included is the 45-degree line, on which the measured points would come to lie if the realized rate exactly equaled the respective capacity value. It can be seen that all measured points are fairly close to the 45-degree line. While the variations among the rate points for a given capacity $C_{\text{H}}(\boldsymbol{s}_{\text{H}})$ are alike to the variation of the per-codeword error rate for fixed-rate codes, we note that an increase in capacity results in a very similar increase in realized rate. Together with Figure 4.2 these results show the robustness of the rateless-coded hybrid FSO/RF systems in performing under different channel conditions and approaching

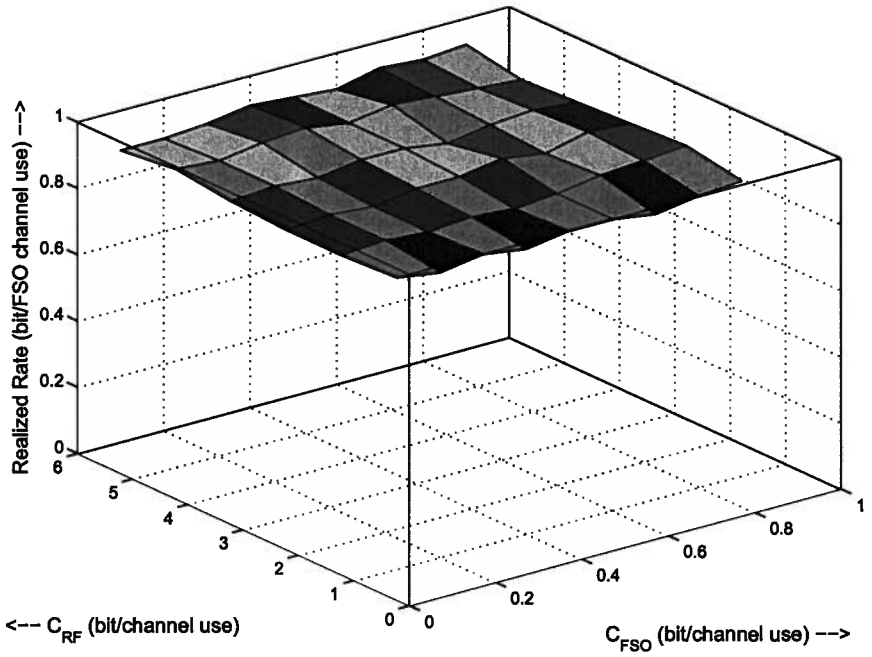

Figure 4.2: Surface plot for the realized rates when $C_{\mathrm{H}} = 1$ bit/(FSO channel use) and the parameters given in Table 4.1.

the respective capacity limit.

The scenarios for the parameters in Tables 4.1, 4.2 and 4.3 were chosen to point out the suitability of rateless coding for hybrid FSO/RF transmission, and not so much to show the benefit of joint transmission compared to switching between FSO and RF transmission. The latter follows immediately from a comparison of $C_{\mathrm{H}}(s_{\mathrm{H}})$ and $\max\{C_{\mathrm{FSO}}(s_{\mathrm{FSO}}), C_{\mathrm{RF}}(s_{\mathrm{RF}})\}$, and it could be formulated in terms of outage probability, if a certain minimal rate needs to be guaranteed, or in terms of average capacity difference, if throughput is of interest. The results presented in this section have provided evidence that such a capacity-based analysis is justified since $C_{\mathrm{H}}(s_{\mathrm{H}})$ is indeed the appropriate performance parameter for the proposed coded system.

Table 4.2: Different parameter combinations simulated for the scatter plot in Figure 4.3. (C_{RF} in [bit/RF channel use], C_{FSO} and C_{H} in [bit/FSO channel use])

Combination	C_{RF}	C_{FSO}	η_{RF}	η_{FSO}	C_{H}
1	2.00	0.30	1	6	0.63
2	3.00	0.50	1	6	1.00
3	4.00	0.70	1	6	1.37
4	4.20	0.80	1	6	1.50
5	5.00	0.90	1	6	1.73
6	6.00	0.90	1	6	1.90

Table 4.3: Different parameter combinations simulated for the scatter plot shown in Figure 4.4. (C_{RF} in [bit/RF channel use], C_{FSO} and C_{H} in [bit/FSO channel use])

Combination	C_{RF}	C_{FSO}	η_{RF}	η_{FSO}	C_{H}
1	2.00	0.30	2	6	0.97
2	3.00	0.50	2	6	1.50
3	4.00	0.70	2	6	2.03
4	4.20	0.80	2	6	2.20
5	5.00	0.90	2	6	2.57
6	6.00	0.90	2	6	2.90

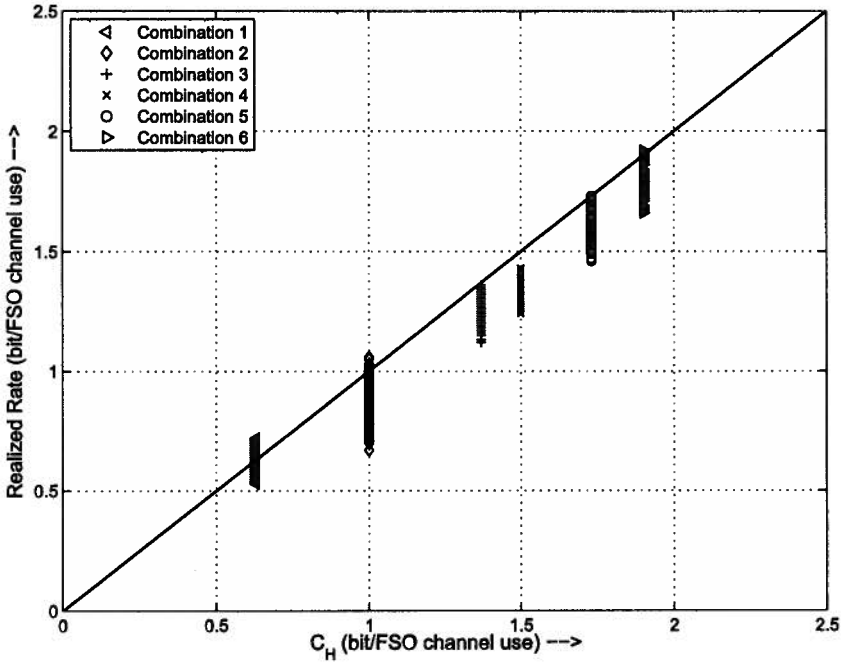

Figure 4.3: Scatter plot of realized rate versus capacity C_{H} for the parameter combinations given in Table 4.2. The 45°-line is included as a reference.

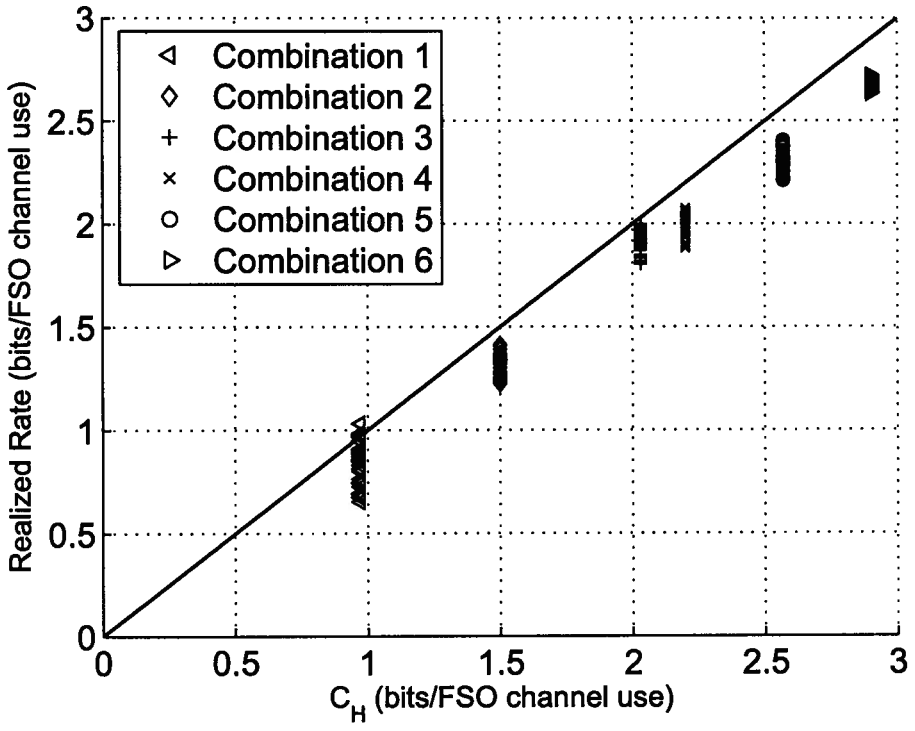

Figure 4.4: Scatter plot of realized rate versus capacity C_{H} for the parameter combinations given in Table 4.3. The 45°-line is included as a reference.

Chapter 5

Conclusions and Future Work

In this thesis, we developed a method for decoding rateless LT and Raptor codes. We further applied these rateless codes to hybrid free-space optical and radio-frequency communication.

First we reviewed rateless codes, namely LT and Raptor codes, then discussed the existing decoding algorithms in the second Chapter. Then, we investigated the problem of efficient decoding of rateless codes. It was noted that faster convergence and early termination of decoding attempts are the key to reducing decoding complexity. To this end, we proposed two new decoding methods, namely BIDIDS and IDIDS, which build on the ideas of incremental decoding and informed dynamic scheduling, that improve convergence of the BP process. Furthermore, in order to execute early termination, we formulated a new hybrid stopping criterion, which is particularly suited for rateless codes when decoded with the informed dynamic schedule.

As for LT codes, simulations for different channel capacities showed that complexity improvement factors of up to 60 are possible with IDIDS/BIDIDS, without any degradation in error-rate performance. For Raptor codes, on the other hand, simulation results for the example of the BSC with a capacity of 0.5 bit/(channel use) have shown that the proposed IDIDS affords significant savings in decoding complexity

without sacrificing performance in terms of realized rate.

In Chapter 4, we proposed the application of rateless codes for hybrid FSO/RF transmission systems. Hybrid FSO/RF systems combine the best of two worlds, i.e., robustness of FSO to rain and RF to fog and atmospheric turbulence, and thus render the system performance robust to short and long-term variations of the FSO and RF transmission channel. This is critical for the adoption of FSO as a reliable high-speed access technology. The distinct feature of the proposed scheme is that it enables to realize (most of) the potential advantages due to parallel FSO and RF channels without the need for redesign or reconfiguration of the transmitter-side coding or modulation. We have established the pertinent information-theoretic limits, and have provided simulative evidence that these limits are well approached with practical off-the-shelf Raptor code designs over wide ranges of channel conditions. We thus believe that the proposed scheme is suitable for practical implementation of efficient hybrid FSO/RF transmission systems.

For future work, the rateless coding scheme proposed in this thesis for FSO/RF systems can be investigated at more depth. For instance, the developed IDIDS/BIDIDS methods can be applied to decoding Raptor codes used in the proposed hybrid FSO/RF scheme. Also, adaption of the transmitter multiplexing ratio to the channel can be explored. In addition, the constellation size for the RF transmission can be modified for coarse adaptation to the channel condition, while rateless coding accomplishes adaptation on a finder (bit-wise) grain.

Bibliography

[1] S. Bloom and W. Hartley. The last-mile solution: Hybrid FSO radio. In *Whitepaper, AirFiber Inc.*, May 2002.

[2] A.I. Vila Casado, M. Griot, and R. Wesel. Informed dynamic scheduling for belief-propagation decoding of LDPC codes. In *Proc. of IEEE Int. Conf. Commun. (ICC)*, Glasgow, UK, June 2007.

[3] V.W.S. Chan. Coding for the turbulent atmospheric optical channel. *IEEE Trans. on Communications*, 30:269–275, January 1982.

[4] O. Etesami and A. Shokrollahi. Raptor codes on binary memoryless symmetric channels. *IEEE Trans. on Information Theory*, 52(5):2033 – 2051, May 2006.

[5] R.M. Gagliardi and S. Karp. *Optical Communications.* John Wiley & Sons, New York, 1976.

[6] S.M. Haas, J.H. Shapiro, and V. Tarokh. Space-time codes for wireless optical channels. In *Proc. IEEE International Symposium on Information Theory (ISIT)*, page 244, Washington, DC, USA, June 2001.

[7] P. Hoeher. Adaptive modulation and channel coding using reliability information. In *Proc. of 5th International OFDM-Workshop*, pages 14.1–14.4, Hamburg, Germany, September 2000.

[8] K. Hu, J. Castura, and Y. Mao. Reduced-complexity decoding of Raptor codes over fading cannels. In *Proc. of IEEE Global Commun. Conf. (GLOBCOM)*, San Francisco, CA, USA, Nov.-Dec. 2006.

[9] H. Kfir and I. Kanter. Parallel versus sequential updating for belief propagation decoding. *Physica A*, 300:259–270, 2003.

[10] I. I. Kim and E. Korevaar. Availability of free space optics (FSO) and hybrid FSO/RF systems. In *Proc. SPIE, Optical Wireless Communications IV*, volume 4530, pages 84–95, Denver, CO, USA, August 2001.

[11] F.R. Kschischang, B.J. Frey, and H.A. Loeliger. Factor graphs and the sum-product algorithm. *IEEE Trans. on Information Theory*, 47:498 – 519, February 2001.

[12] E. Lee and V. Chan. Optical communication over the clear turbulent atmospheric channel using diversity. *IEEE Journal on Selected Areas in Communications*, 22:1896–1906, November 2004.

[13] M. Luby. LT codes. In *Proc. 43rd Annual IEEE Symp. on Foundations of Comp. Sc. (FOCS)*, pages 271–280, Vancouver, BC, Canada, November 2002.

[14] A.K. Majumdar and J.C. Ricklin. Effects of the atmospheric channel on free-space laser communications. In *Proc. SPIE, Free-Space Laser Communications V*, volume 5892, pages 58920K1–58920K16, San Diego, CA, USA, July 2005.

[15] S.S. Muhammad, P. Kohldorfer, and E. Leitgeb. Channel modeling for terrestrial free space optical links. In *Proc. 7th International Conference on Transparent Optical Networks (ICTON)*, volume 1, pages 407– 410, Barcelona, Spain, July 2005.

[16] L.H. Ozarow and A.D. Wyner. On the capacity of the Gaussian channel with a finite number of input levels. *IEEE Trans. on Information Theory*, 36(11):1426–1428, November 1990.

[17] R. Palanki and J.S. Yedidia. Rateless codes on noisy channels. In *Proc. IEEE International Symposium on Information Theory (ISIT)*, page 37, Chicago, IL, USA, June-July 2004. (see also www.merl.com/papers/TR2003-124/).

[18] P. Radosavljevic, A. de Baynast, and J.R. Cavallaro. Optimized message passing schedules for ldpc decoding. In *Proc. 39th Asilomar Conf. Signals, Systems and Computers*, pages 591–595, San Antonio, TX, USA, 2005.

[19] T. Richardson and R. Urbanke. The capacity of low-density parity-check codes under message-passing decoding. *IEEE Trans. on Information Theory*, 47:595 – 618, February 2001.

[20] A. Shokrollahi. Raptor codes. *IEEE Trans. on Information Theory*, 52(6):2551 – 2567, June 2006.

[21] S. Vangala and H. Pishro-Nik. A highly reliable FSO/RF communication system using efficient codes. In *Proc. IEEE Global Telecommunications Conference (GLOBECOM)*, pages 2232–2236, Washington, DC, USA, November 2007.

[22] S. Vangala and H. Pishro-Nik. Optimal hybrid RF-wireless optical communication for maximum efficiency and reliability. In *Proc. 41st Annual Conference on Information Sciences and Systems (CISS)*, pages 684–689, March 2007.

[23] S.G. Wilson, M. Brandt-Pearce, Q. Cao, and J.H. Leveque, III. Free-space optical MIMO transmission with Q-ary PPM. *IEEE Trans. on Communications*, 53(8):1402 – 1412, August 2005.

[24] E. Yeo, P. Pakzad, B. Nikolic, and V. Anantharam. High throughput low-density parity-check decoder architectures. In *Proc. of IEEE Global Commun. Conf. (GLOBCOM)*, pages 3019 – 3024, San Antonio, TX, USA, Nov.-Dec. 2001.

[25] X. Zhu and M. Kahn. Free-space optical communication through atmospheric turbulence channels. *IEEE Trans. on Communications*, 50(8):1293–1300, August 2002.

[26] Z. Zhuang, D. Ma, and J. Wei. Diversity coding scheme for wireless optical communication with direct detection. *IEE Electronics Letters*, 40:407– 410, 2004.