Error-Correcting Codes
for Fibre-Optic Communication Systems

by

Benjamin P. Smith

A thesis submitted in conformity with the requirements
for the degree of Doctor of Philosophy,
The Edward S. Rogers Sr. Department of
Electrical & Computer Engineering,
University of Toronto

Copyright © 2011 by Benjamin P. Smith
Error-Correcting Codes
for Fibre-Optic Communication Systems

Benjamin P. Smith
Doctor of Philosophy
The Edward S. Rogers Sr. Department of Electrical & Computer Engineering
University of Toronto
2011

Abstract

Electronic signal processing techniques have assumed a prominent role in the design of fibre-optic communication systems. However, state-of-the-art systems operate at per-channel data rates of 100 Gb/s, which constrains the class of communication algorithms that can be practically implemented. Relative to LDPC-like codes, product-like codes with syndrome-based decoding have decoder dataflow requirements that are smaller by more than two orders of magnitude, which strongly motivates the search for powerful product-like codes. This thesis presents a new class of high-rate binary error-correcting codes—staircase codes—whose construction combines ideas from convolutional and block coding. A G.709-compliant staircase code is proposed, and FPGA-based simulation results show that performance within 0.5 dB of the Shannon Limit is attained for bit-error-rates below 10\(^{-15}\). An error-floor analysis technique is presented, and the G.709-compliant staircase code is shown to have an error floor below 10\(^{-20}\). Using staircase codes, a pragmatic approach for coded modulation in fibre-optic communication systems is presented that provides reliable communications to within 1 bit/s/Hz of the capacity of a QAM-modulated system modeled via the generalized non-linear Schrödinger equation. A system model for a real-world DQPSK receiver with correlated bit-errors is presented, along with an analysis technique to estimate the resulting error floor for the G.709-compliant staircase code. By applying a time-varying pseudorandom interleaver of size 2040 to the output of the encoder, the error floor of the resulting system is shown to be less than 10\(^{-20}\).
To my parents
Acknowledgements

To my supervisor, Professor Frank R. Kschischang, thank you for the many opportunities and experiences you have afforded me. I have benefitted immensely from your guidance, your intelligence, and your caring nature. I am greatly privileged to have been your student.

To the external appraiser of this thesis, Professor Erik Agrell, thank you for your careful reading and detailed comments which greatly improved the thesis.

During the course of my studies, I derived significant enjoyment from successful collaborations with Arash Farhood at Cortina Systems Inc., Andrew Hunt at the Communications Research Centre Canada, and René-Jean Essiambre at Alcatel-Lucent Bell Labs.

I would like to thank the Natural Sciences and Engineering Research Council of Canada, and the University of Toronto, for their generous financial support.

Finally, to my parents, Sarah, Peter, Julia, Nicholas, Evelyn, Cathy, Ryan, Sara and Liz, thank you for your love, support and friendship, without which this thesis would not have been possible.
Contents

1 Introduction 1
  1.1 Fibre-optic communication systems 1
    1.1.1 The capacity of fibre-optic communication systems 2
    1.1.2 The role of communication engineering principles 3
  1.2 Overview of the Thesis 4
    1.2.1 Organization of the Thesis 4

2 Background 7
  2.1 Coherent fibre-optic communication systems 7
    2.1.1 The propagation constant $\beta$ 8
    2.1.2 The non-linear Schrödinger equation 9
    2.1.3 Optical amplification and the GNLS equation 9
    2.1.4 System model 11
  2.2 Product codes 12
    2.2.1 Factor graph representations 13
    2.2.2 Bose-Chaudhuri-Hocquenghem codes 15
    2.2.3 Iterative decoding of BCH $\times$ BCH product codes 16
    2.2.4 LDPC vs. Product Codes: Decoder dataflow 16
  2.3 Error-correcting codes for optical transport networks 20
    2.3.1 Net coding gain 20
    2.3.2 ITU-T recommendations G.975 and G.975.1 21

3 Digital backpropagation and memoryless capacity estimations 25
  3.1 A review of known results and techniques 26
    3.1.1 The split-step Fourier method 26
    3.1.2 Single-channel backpropagation 27
### 3.1.3 Memoryless capacity estimations ........................................ 27
### 3.2 Memoryless information rates in the absence of backpropagation .... 30
  3.2.1 Ideal distributed Raman amplification ................................. 31
  3.2.2 Ideal EDFA lumped amplification ....................................... 34
  3.2.3 Revisiting the Gaussian assumption .................................... 37
  3.2.4 Non-monotonicity of achievable rates in the input power ......... 41
### 3.3 Backpropagation step-size and achievable rates ........................ 43
### 3.4 Conclusions ............................................................. 44

### 4 Error-correcting codes for high-speed communications .................. 47
  4.1 Irregular product-like codes .............................................. 48
    4.1.1 Threshold estimation .................................................. 48
    4.1.2 Optimization framework .............................................. 49
  4.2 Staircase codes ............................................................ 53
    4.2.1 Incidence matrix representations .................................... 53
    4.2.2 Matrix encoding rule .................................................. 55
    4.2.3 Properties .............................................................. 57
    4.2.4 Decoding algorithm .................................................... 58
    4.2.5 Multi-edge-type interpretation ...................................... 59
  4.3 A G.709-compatible staircase code ....................................... 60
    4.3.1 Simulation results ..................................................... 61
    4.3.2 Error floor analysis .................................................. 61
  4.4 Generalized staircase codes .............................................. 67
    4.4.1 Mixtures of component codes ....................................... 67
    4.4.2 Multi-block staircase codes ........................................ 67
  4.5 A review of related works ............................................... 68
  4.6 Conclusions ............................................................. 70

### 5 Staircase codes in high-spectral-efficiency fibre-optic systems ........ 72
  5.1 Coded modulation via binary codes ..................................... 72
  5.2 A pragmatic error-correcting system .................................... 74
  5.3 Performance of error-correcting system with staircase codes ......... 77
    5.3.1 Achievable rates ....................................................... 78
    5.3.2 Staircase codes for hard-decision BICM ............................ 82
# List of Tables

2.1 System parameter values .............................................. 12

3.1 Signaling parameter values ........................................... 30

4.1 Parameters of the best bi-regular codes ............................ 51
4.2 Parameters of the best constraint-regular codes .................... 53
4.3 Estimated probabilities of a stall \( s \), assuming that \( 16 - l \) given positions are received in error ................................. 64
4.4 Error floor contributions of \((K,L)\)-stall patterns ................. 66

5.1 Subchannel error rates for equalized 256-QAM, \( L = 500 \) km. .......... 80
5.2 Subchannel error rates for backpropagated 512-CrossQAM, \( L = 500 \) km. 80
5.3 Subchannel error rates for equalized 128-CrossQAM, \( L = 1000 \) km. ... 81
5.4 Subchannel error rates for backpropagated 256-QAM, \( L = 1000 \) km. ... 81
5.5 Subchannel error rates for equalized 64-QAM, \( L = 2000 \) km. ........... 81
5.6 Subchannel error rates for backpropagated 128-CrossQAM, \( L = 2000 \) km. 82
5.7 Staircase codes for hard-decision BICM. ............................ 83
## List of Figures

1.1 Attenuation in fibre as a function of input wavelength, from [1]. 2
1.2 Achievable per-channel spectral efficiencies for a 2000 km WDM fibre-optic system, with and without backpropagation. 3

2.1 Cross-section of an optical fiber. 8
2.2 Coherent fibre-optic communication system model. 12
2.3 A product code $C_{A\otimes B}$. 13
2.4 Factor graph of the product code $C_{A\otimes B}$. 13
2.5 Factor graph of a generalized LDPC code with degree two variable nodes. 14
2.6 Dataflow in an LDPC decoder. 18
2.7 Dataflow in the initial syndrome computing. 19
2.8 Dataflow in a product-code decoder. 20
2.9 Performance of G.975 and G.975.1 codes. 23

3.1 System model for memoryless capacity estimations. 28
3.2 Channel outputs for a fixed-ring input, and back-rotated outputs. 29
3.3 Achievable spectral efficiencies for single-channel transmission with Raman amplification. 32
3.4 Achievable spectral efficiencies for WDM transmission with Raman amplification. 34
3.5 Comparison of achievable spectral efficiencies for WDM and single-channel transmission, without backpropagation. 35
3.6 Achievable spectral efficiencies for single-channel transmission with EDFA amplification. 36
3.7 Achievable spectral efficiencies for WDM transmission with EDFA amplification. 37
3.8 Comparison of achievable spectral efficiencies for WDM and single-channel transmission, without backpropagation ................................................................. 38
3.9 Scatter plots of back-rotated constellation points, for $P = -8$ dBm .......... 39
3.10 Scatter plots of back-rotated constellation points, for $P = -5$ dBm .......... 40
3.11 Noise distribution of back-rotated constellation points on fourth ring, for $P = -8$ dBm ................................................................. 40
3.12 Noise distribution of back-rotated constellation points on sixteenth ring, for $P = -8$ dBm ................................................................. 41
3.13 Noise distribution of back-rotated constellation points on fourth ring, for $P = -5$ dBm ................................................................. 42
3.14 Noise distribution of back-rotated constellation points on sixteenth ring, for $P = -5$ dBm ................................................................. 42
3.15 Scatter plots of back-rotated constellations points from the first, sixteenth and thirty-second rings, after receiver back-propagation. .................. 43
3.16 Achievable spectral efficiencies for WDM transmission ($L = 500$ km) as a function of back-propagation step-size ............................................. 45
3.17 Comparison of achievable spectral efficiencies for WDM transmission ($L = 1000$ km) as a function of back-propagation step-size .................... 46
3.18 Comparison of achievable spectral efficiencies for WDM transmission ($L = 2000$ km) as a function of back-propagation step-size .................... 46

4.1 Idealized form of $Pr[Frame \text{ error occurs}|p_{e,obs}]$ ........................................... 49
4.2 Factor graph of a bi-regular code. .................................................. 50
4.3 Factor graph of a constraint-regular code........................................ 52
4.4 The graphical model corresponding to incidence matrix $I$ ......................... 53
4.5 The incidence matrix $I_p$ for a square product code; only non-zero elements are indicated. By $1_{1 \times n}$ we mean an all-ones row vector of length $n$, and by $I_n$ the $n$-by-$n$ identity matrix. .............................................. 55
4.6 The semi-infinite incidence matrix $I_s$ for a staircase code; only non-zero elements are indicated. .................................................. 56
4.7 The ‘staircase’ visualization of staircase codes. ........................................ 57
4.8 The blocks within an $L = 4$ sliding-window decoder; the black strip represents a codeword that ‘terminates’ in block $B_{i+3}$ ....................... 58
4.9 A multi-edge-type graphical representation of staircase codes. \( \Pi_B \) is a standard block interleaver, i.e., it represents the transpose operation on an \( m \)-by-\( m \) matrix. ................................................................. 60

4.10 Performance of G.975 code, G.975.1 codes, and a G.709-compliant staircase code ................................................................. 62

4.11 A stall pattern for a staircase code with a triple-error correcting component code. Since every involved component codeword has 4 errors, decoding stalls. 62

4.12 A non-minimal stall pattern for a staircase code with a triple-error correcting component code. ................................................................. 65

4.13 Array representation of a BBBC. ......................................................... 69

5.1 Pseudo-Gray labeling of a 32-CrossQAM constellation, and a method for obtaining a \( 2^{M+1} \)-CrossQAM labeling from a Gray-labeled \( 2^M \)-QAM constellation. ................................................................. 75

5.2 Sub-channel capacities of 256-QAM for hard-decision decoder inputs. .. 76

5.3 Capacities of 16-QAM for various forms of coded modulation. ............. 77

5.4 Capacities of 64-QAM for various forms of coded modulation. ............. 77

5.5 Scatter diagram of 64-QAM, before and after phase rotation compensation. 79

5.6 Performance curves for the staircase codes in Table 5.7. ..................... 84

6.1 Model for a DQPSK fibre-optic system. ............................................. 87

6.2 Note that two disjoint runs do not involve common DQPSK symbols. ... 95

6.3 Vertical lines represent the groupings of four rows into distinct 2040-bit interleavers, dots represent rows. ..................................................... 102
Chapter 1

Introduction

State-of-the-art fibre-optic communication systems have the capacity to transmit terabits of data per second, over thousands of kilometres, on a single optical fibre. Advances in physics—the invention of the laser, low-loss optical fibre, and the optical amplifier—have driven the exponential growth in worldwide data communications. However, as these technologies mature, system designers have increasingly focused on techniques from communication theory—error-correcting codes, signal processing, and coherent receivers—to simultaneously increase transmission capacity and decrease transmission costs. While these technologies are, themselves, relatively mature in the domain of wireline and (microwave) wireless communications, new challenges arise due to the tremendous speed at which fibre-optic systems operate. This work focuses on the design of error-correcting codes for fibre-optic communication systems, with an emphasis on designing high-performance codes with efficient hardware realizations.

1.1 Fibre-optic communication systems

Since the early 1980s, when fibre-optic communication systems first became available, tremendous increases in their transmission capacity have primarily been the result of ‘engineering’ a better channel, the end result of which is succinctly captured in Fig. 1.1. The ‘low-loss’ window of optical fibre, extending over input wavelengths from 1.3 \( \mu \text{m} \) to 1.7 \( \mu \text{m} \), corresponds to a bandwidth of 54 THz! Fully exploiting this bandwidth relies on a host of additional technologies, including semiconductor lasers to provide a coherent optical source over the range of input frequencies, and optical amplification—erbium-doped fibre amplifiers (EDFAs), or Raman amplifiers—to compensate for fibre
attenuation in long-haul transmission systems. In principle, a per-fibre capacity of 1 petabit/s, over 2000 km, may be attainable [2].

![Figure 1.1: Attenuation in fibre as a function of input wavelength, from [1].](image)

1.1.1 The capacity of fibre-optic communication systems

Recent progress has been made in estimating the information theoretic capacity of the class of fibre-optic communication systems that are (presently) of commercial interest [1]. In contrast to most classically studied communication channels, optical fibre exhibits significant nonlinearity (in the intensity of the guided light). Furthermore, amplification acts as a source of distributed additive-white-Gaussian noise (AWGN), and fibre chromatic dispersion acts as a distributed linear filter. Complicating matters, these three fundamental effects interact over the length of transmission. For systems of commercial interest with a certain type of ring constellation, the attainable spectral efficiencies do not increase monotonically with input power, but rather exhibit a peak value at some finite average input power, beyond which the effects of nonlinearity dominate, and the spectral efficiency goes to zero, as in Fig. 1.2.
Figure 1.2: Achievable per-channel spectral efficiencies for a 2000 km WDM fibre-optic system, with and without backpropagation.

1.1.2 The role of communication engineering principles

For nearly twenty years, fibre-optic communication systems were operated with on-off-keying (OOK) modulation, direct-detection receivers, and minimal electronic processing in the receiver. However, to reduce system costs, both optical transparency (i.e., the absence of optical-electrical-optical repeaters) and greater per-channel bit rates (both by increasing the symbol rate and the spectral efficiency) are desirable. Furthermore, it is required that the output bit-error-rate (BER) be less than $10^{-15}$. These requirements have led to a renewed interest in communication engineering principles.

The development of the coherent optical receiver [3], in addition to advances in high-speed electronics, have enabled the implementation of many digital signal processing algorithms in the receiver, including: time-domain and frequency-domain algorithms for equalization of chromatic dispersion and polarization mode dispersion [4], optical orthogonal frequency-division-multiplexed (OOFDM) transmission [5], advanced modulation formats [6], and error-correcting coding [7, 8]. However, due to the very high data rates ($\sim 100$ Gb/s) at which electronic signal processing is required to operate, significant implementation challenges arise, necessitating the use of sub-optimal, low-complexity algorithms. Indeed, we devote a significant portion of this thesis to the design of error-correcting codes for 100 Gb/s applications.
State-of-the-art commercial systems

As of 2011, state-of-the-art commercial fibre-optic communication systems provide per-channel information transmission at 100 Gb/s, within the ITU-T standardized 50 GHz channel spacing. The modulation scheme of choice is polarization-duplexed, differential quadrature-phase-shift keying (PD-DQPSK), i.e., DQPSK transmission occurs in parallel over two orthogonal polarizations. Information bits are encoded by a binary error-correcting code of rate $R = \frac{239}{255}$.

1.2 Overview of the Thesis

Despite their remarkable capabilities, fibre-optic communication systems have received little to no attention in most textbooks on digital communications. At least in part, this is likely due to the complicated nature of the channel model, and that until recently, digital signal processing within the receiver was not considered practical. In another light, real-world fibre-optic communication systems were—until recently—within the purview of physicists, and relatively disconnected from communications theory. Although not our primary goal, we hope that this work is a step in bridging the gap.

In this thesis, we take a pragmatic approach to three main problems:

- Understanding the nature of the channel in fibre-optic communication systems, with a focus on the impact of digital backpropagation.

- Designing high-rate, low-error-floor, binary error-correcting codes suitable for high-speed implementations.

- Applying these error-correcting codes to fibre-optic systems with correlated errors, and to systems with multi-level modulation schemes.

1.2.1 Organization of the Thesis

In Chapter 2, we describe the system model of a coherent fibre-optic communication system. We also review product codes, focusing on those with component Bose-Chaudhuri-Hocquenghem (BCH) codes, and we describe their decoding algorithms. We present an analysis of the dataflow requirements for decoding of product and LDPC codes, and show that product codes with syndrome-based decoding have dataflow requirements that are smaller by more than two orders of magnitude, which strongly motivates the work herein.
In Chapter 3, we investigate the impact of digital backpropagation on the (memoryless) information capacity of fibre-optic communication systems, both for lumped and distributed amplification schemes. We show that—despite fibre-nonlinearity—the communication channel induced by real-world fibre-optic communication systems is well-modeled as a linear Gaussian channel for a range of input powers of interest, and that digital backpropagation provides only modest benefits relative to a linear equalizer.

In Chapter 4, we study the design of high-rate, low-error-floor, binary error-correcting codes, with an emphasis on designs that reduce the dataflow requirements within the decoder. We propose staircase codes, a family of codes that provide near-capacity performance, and that can be interpreted as a hybridization of convolutional and block coding techniques. We also provide an interpretation of staircase codes as multi-edge-type codes, emphasizing the ‘irregular’ level of protection afforded to bits within a given decoding window. We propose a G.709-compliant staircase code that provides a 9.45 dB net coding gain at an output error rate of $10^{-15}$, which is 0.46 dB better than the best G.975.1 code, and within 0.52 dB of the Shannon Limit. An error floor estimation technique for staircase codes is presented, and the error floor of the G.709-compliant staircase code is estimated to occur at $4.0 \times 10^{-21}$.

In Chapter 5, we propose a pragmatic approach for coded modulation in fibre-optic communication systems, leveraging the low-complexity of syndrome-based decoding of product-like codes. Similar to the well-known fact that bit-interleaved coded-modulation (BICM) approaches the capacity of a constellation for AWGN, we show that hard-decision BICM closely approaches the hard-decision capacity of the constellation. We simulate dense-wavelength-division-multiplexed (DWDM) Raman-amplified fibre-optic communication systems, using the results to determine the capacity of hard-decision BICM at the operating points (i.e., the input powers) at which the capacity of the system is maximized, and we show that hard-decision BICM performs within 1 bit/s/Hz of the fundamental limit. We design staircase codes for hard-decision BICM, and show that they perform within 0.19 bits/s/Hz of the capacity of hard-decision BICM for linearly equalized systems, and within 0.28 to 0.41 bits/s/Hz of the capacity of hard-decision BICM for digitally backpropagated systems, with error floors below $10^{-25}$.

In Chapter 6, we propose a system model based on a real-world DQPSK receiver, and present an analysis technique to estimate the error floor of the G.709-compliant staircase code in the presence correlated errors. By applying a time-varying pseudorandom interleaver of size 2040 to the output of the encoder, the error floor of the resulting system is
shown to be less than $4.77 \times 10^{-21}$.

In Chapter 7, we summarize the contributions of this work, and we propose some directions for further research.
Chapter 2

Background

In Section 2.1, we describe the system model of a coherent fibre-optic communication system. In Section 2.2, we review product codes, focusing on those with component Bose-Chaudhuri-Hocquenghem (BCH) codes, and we describe their decoding algorithms. We show that the decoder dataflow requirements for syndrome-based decoding of product codes is more than two orders of magnitude smaller than the corresponding dataflow for LDPC codes. In Section 2.3, we review the error-correcting codes standardized by the International Telecommunication Union’s Telecommunication Standardization Sector (ITU-T) for communication over optical transport networks (OTNs).

2.1 Coherent fibre-optic communication systems

Propagation of electromagnetic waves in an optical fibre is governed by Maxwell’s Equations. Figure 2.1 illustrates an optical fiber; the indices of refraction of the core and cladding layers, in addition to their geometry, dictate the properties of the fiber. We denote the longitudinal (i.e., along the length of the fibre) direction by $z$, and the transverse plane (i.e., the cross-section) by the $x - y$ plane.

In this work, we focus on standard single-mode fiber (SSMF)\textsuperscript{1}. It can be shown that the supported mode—the so-called fundamental mode—has a transverse spatial distribution $F(x, y)$ that is constant with propagation. The field distribution $E$ of a mono-chromatic component of the fundamental mode is then

$$E(r, t) = \text{Re} \left[ F(x, y) A e^{j(\omega z - \omega t)} \right],$$

\textsuperscript{1}All existing high-speed, long-haul systems exclusively use single-mode fiber.
where \( \omega \) is the frequency of the field, \( \beta \) is the propagation constant, and \( A \) is a complex scalar \([9]\).

### 2.1.1 The propagation constant \( \beta \)

In general, \( \beta \) is a function of both frequency and intensity, where intensity \( I(z,t) \) is defined as the ratio of the instantaneous power \( P(z,t) \) of the propagating wave to the effective area \( A_{\text{eff}} \) of the fibre. These dependencies give rise to the interesting properties of the fibre-optic channel.

The propagation constant \( \beta \) can be decomposed as \([10]\)

\[
\beta(\omega) = \beta_L(\omega) + \beta_{NL}(\omega) + j\alpha(\omega)/2.
\]

In the expansion of \( \beta(\omega) \), the term \( \beta_L \) accounts for the frequency-dependence of the channel; \( \beta_{NL} \) and \( \alpha \) are essentially constant with frequency. Expanding \( \beta_L \) in a Taylor series about the carrier frequency \( \omega_0 \), we have

\[
\beta_L(\omega) = \beta_0 + \beta_1(\omega - \omega_0) + \beta_2(\omega - \omega_0)^2,
\]

where higher order terms are neglected, since they can be engineered to be zero. The most important parameter is \( \beta_2 \), which reflects the frequency dependence of the speed of propagation of light in the fibre. The dispersion parameter \( D \),

\[
D = -\frac{2\pi c}{\lambda_0^2} \beta_2,
\]

expressed in units of ps/(km-nm), is the difference between the propagation times (in ps) of two spectral components (with wavelengths differing by 1 nm) over a distance of 1 km, where \( \lambda_0 \) is the carrier wavelength and \( c \) is the speed of light in a vacuum. This effect,
termed group velocity dispersion (GVD), results in significant waveform distortion when propagation occurs over long distances.

Due to the optical Kerr effect, the effective refractive index of the fibre depends on the intensity $I(z,t)$ of the propagating waveform. The parameter $\beta_{NL}$,

$$\beta_{NL}(\omega) = n_2(\omega)I(z,t)$$
captures the resulting non-linearity of pulse propagation in optical fibre. The Kerr parameter $n_2$ typically has a value around $2.6 \times 10^{-20} \text{m}^2/\text{W}$, and is nearly constant over a wide range of $\omega$ values of interest.

Finally, the loss parameter $\alpha$ reflects the signal attenuation due to Rayleigh scattering and infrared absorption. As illustrated in Fig. 1.1, standard single-mode fibres (SSMF) have an attenuation of less than 0.2 dB/km near 1.55 $\mu$m.

### 2.1.2 The non-linear Schrödinger equation

Consider the complex baseband representation $A(z,t)$ of an optical signal propagating along the longitudinal direction of the fibre. The non-linear Schrödinger (NLS) equation expresses the evolution of $A(z,t)$, in the absence of noise, as a function of $\beta_2$, $\gamma$ and $\alpha$ [11]:

$$\frac{\partial A}{\partial z} + \alpha A + \frac{j}{2} \beta_2 \frac{\partial^2 A}{\partial t^2} - j\gamma |A|^2 A = 0,$$

where $\gamma = \frac{2\pi n_2}{\lambda_0 A_{\text{eff}}}$. In the stated form, the temporal frame of reference is implicitly assumed to move with the pulse, i.e., it moves at the group velocity.

### 2.1.3 Optical amplification and the GNLS equation

The reach of fibre-optic systems varies from metropolitan systems with reach on the order of $10^2$ km to undersea systems with reach on the order of $10^5$ km. Due to the significant distances spanned by fibre-optic systems, and the significant cost associated with signal regeneration (i.e., optical-electrical-optical conversion), optical amplifiers are a necessary component for multi-channel high-rate communication systems. However, optical amplifiers inevitably contribute an additive noise to the propagating signal via amplified spontaneous emission (ASE); this noise is well-modeled by a spatially and temporally white, circularly symmetric, Gaussian random process. Since the noise co-propagates
with the signal, it is necessary to include its effects in the propagation equation, resulting in the generalized non-linear Schrödinger (GNLS) equation \[11\]

\[
\frac{\partial A}{\partial z} + \frac{\alpha - g(z)}{2} A + j\beta A + j\gamma |A|^2 A = n(z, t),
\]

where the forms of the amplification factor \(g(z)\) and noise \(n(z, t)\) depend on the amplification scheme.

**Lumped amplification**

Erbium-doped fibre amplifiers (EDFA) provide broadband amplification of optical signals. For an amplifier spacing of \(L_A\) km, the (power) gain \(G\) of an EDFA is set to \(G = \exp(\alpha L_A)\), restoring the signal power (in the absence of noise) to its initially transmitted power, i.e.,

\[
g(z) = \alpha L_A \sum_{m=1}^{N} \delta(z - mL_A).
\]

The power spectral density of the noise added per amplifier is

\[
N_{\text{EDFA}} = (G - 1)hv_s n_{sp},
\]

where \(h\) is Planck’s constant, \(v_s = \omega_0/(2\pi)\) is the optical frequency, and \(n_{sp}\) is the spontaneous emission factor. Thus, for amplifiers located at distances \(z = L_A, 2L_A, \ldots, NL_A\), \(n(z, t)\) is a circularly symmetric complex Gaussian noise with autocorrelation \[11\]

\[
\mathcal{E} [n(z, t)n^*(z', t')] = \begin{cases} 
N_{\text{EDFA}} \delta(z - z', t - t') & \text{if } z = mL_A, \text{ for } m \in \{1, 2, \ldots, N\} \\
0 & \text{otherwise},
\end{cases}
\]

where \(\delta(\cdot, \cdot)\) is a two-dimensional generalization of the delta function, such that

\[
\delta(z, t) = 0, \quad z^2 + t^2 \neq 0
\]

and

\[
\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \delta(z, t) \, dz \, dt = 1.
\]

**Distributed amplification**

For a lumped amplification scheme with \(N\) equally-spaced EDFAs, the total noise power spectral density is

\[
N_{\text{EDFA}}^{\text{tot}} = N(G - 1)hv_s n_{sp} = \frac{L}{L_A} (G - 1)hv_s n_{sp},
\]
where $L = NL_A$ is the total transmission length. Now, as $L_A \to 0$ we have $G-1 \to \alpha L_A$, and

$$N_{\text{tot}} \to L\alpha h v_s n_{sp}.$$ 

From a series expansion of $\exp(\alpha L_A)$, it is clear that decreasing the amplifier spacing reduces the total noise power. In the limit, we refer to such amplification schemes as distributed amplification.

In practice, distributed amplification can be accomplished via stimulated Raman scattering (SRS), which has its origins in the non-instantaneous part of the Kerr non-linearity [10]. SRS occurs when the transmission fibre is ‘pumped’ by a high-frequency source; inelastic scattering of this source is then induced by the signals associated with the transmitted data, and energy is transferred to these data-carrying signals. Distributed amplification accomplished via SRS is referred to as Raman amplification.

Ideal distributed Raman amplification maintains signal power throughout transmission, i.e.,

$$g(z) = \alpha,$$

and $n(z,t)$ is a circularly symmetric complex Gaussian noise with autocorrelation [12]

$$\mathcal{E}[n(z,t)n^*(z',t')] = \alpha h v_s K_T \delta(z-z',t-t'),$$

where $K_T$ is the phonon occupancy factor, which replaces $n_{sp}$. It follows that the power spectral density of the total noise added over a system of length $L$ is

$$N_{\text{Raman}}^{\text{tot}} = L\alpha h v_s K_T.$$

### 2.1.4 System model

We consider a coherent fibre-optic communication system as in Fig. 2.2. Within each span, standard-single-mode fiber (SSMF) is used, and a per-span amplification scheme (either EDFA or distributed Raman) is assumed. The complex baseband representation of the signal at the output of the transmitter is $A(0,t)$, and at the input of the receiver is $A(L,t)$, where $L$ is the total system length; note that $A(z,t)$ represents the full field, i.e., in general it represents co-propagating wavelength-division-multiplexed signals. In Table 2.1, we provide parameter values for the system components.
Chapter 2. Background

Figure 2.2: Coherent fibre-optic communication system model

<table>
<thead>
<tr>
<th>Fiber span length $L_A$</th>
<th>50 km</th>
</tr>
</thead>
<tbody>
<tr>
<td>Second-order dispersion $\beta_2$</td>
<td>-21.668 ps$^2$/km</td>
</tr>
<tr>
<td>Loss $\alpha$</td>
<td>$4.605 \times 10^{-5}$ m$^{-1}$</td>
</tr>
<tr>
<td>Nonlinear coefficient $\gamma$</td>
<td>1.27 W$^{-1}$km$^{-1}$</td>
</tr>
<tr>
<td>Center carrier frequency $v_s$</td>
<td>193.41 THz</td>
</tr>
<tr>
<td>Phonon occupancy factor $K_T$</td>
<td>1.13</td>
</tr>
<tr>
<td>Spontaneous emission factor $n_{sp}$</td>
<td>1.0</td>
</tr>
</tbody>
</table>

Table 2.1: System parameter values

2.2 Product codes

Product codes [13, 14], though not capacity-approaching (unlike low-density parity-check (LDPC) codes, and related codes with a sparse graphical representation), possess properties that make them particularly suited to providing error-correction in fibre-optic communication systems.

Consider a linear $[n_A, k_A, d_A]$ code $C_A$ over a finite field $\mathbb{F}$, and a linear $[n_B, k_B, d_B]$ code $C_B$ over $\mathbb{F}$. The set of matrices $M \in \mathbb{F}^{n_A \times n_B}$ such that every column is a codeword in $C_A$, and every row is a codeword in $C_B$, define a $[n_A n_B, k_A k_B, d_A d_B]$ product code $C_{A \otimes B}$. For $C_A$ in systematic form with generator matrix $G_A = [I_{k_A} | P_A]$, and similarly $G_B = [I_{k_B} | P_B]$, the set of valid codewords are $n_A \times n_B$ matrices of the form

$$
\begin{bmatrix}
U & UP_B \\
PT_A & P_T_A U P_B
\end{bmatrix},
$$

illustrated in Fig. 2.3, where $U$ is a $k_A \times k_B$ information matrix.
Chapter 2. Background

2.2.1 Factor graph representations

It is worthwhile to consider the factor graph [15] representation of the product code $C_{A \otimes B}$ in Fig. 2.4, both to recognize their similarities to sparse-graph-based codes, and to appreciate the efficiencies inherent in their construction. Note that product codes are distinguished by their interleaver $\Pi_P$; an edge labeled ‘$i, j$’ connects to the variable node corresponding to the codeword symbol at row $i$, column $j$ in the matrix representation of codewords.
Generalized LDPC codes

The term *generalized* LDPC code [16] refers to an LDPC code for which the single-parity-check constraints are replaced by a (more general) linear code, e.g., the constraint nodes may represent Hamming codes [17]. In the most general case, each constraint may be replaced by a distinct linear code. It is instructive to consider a generalized LDPC code $C_G$ of length $n = n_An_B$, for which each codeword symbol is involved in one constraint enforcing $C_A$ and a second constraint enforcing $C_B$. That is, each codeword symbol is protected by a codeword in $C_A$ and a codeword in $C_B$, analogous to codeword symbols in the product code $C_A \otimes C_B$. Note that each constraint enforcing $C_A$ contributes $r_A = n_A - k_A$ linear constraints, and similarly $r_B = n_B - k_B$ linear constraints are contributed by each constraint enforcing $C_B$. In total, there are thus $2n_An_B - n_Bk_A - n_Ak_B$ linear constraints, and the rate of $C_G$ satisfies

$$R_{CG} \geq 1 - \frac{r_A}{n_A} - \frac{r_B}{n_B}.$$

On the other hand, for the product code $C_A \otimes C_B$, $(n_B - k_B)(n_A - k_A)$ of the $2n_An_B - n_Bk_A - n_Ak_B$ linear constraints are—by construction—redundant, and

$$R_{C_A \otimes C_B} = 1 - \frac{r_A}{n_A} - \frac{r_B}{n_B} + \frac{r_Ar_B}{n_An_B}.$$

Finally, while every linear code has a systematic encoder, the systematic encoder for $C_A \otimes C_B$ is especially simple to implement (it consists of parallel row encodings by $C_A$ followed by parallel column encodings by $C_B$), but the systematic encoder for $R_{CG}$ is, in general, a dense matrix.

![Factor graph of a generalized LDPC code with degree two variable nodes](image)

Figure 2.5: Factor graph of a generalized LDPC code with degree two variable nodes
2.2.2 Bose-Chaudhuri-Hocquenghem codes

Product codes with component Bose-Chaudhuri-Hocquenghem (BCH) codes provide excellent performance, while leveraging the hardware-friendly encoding and decoding algorithms of algebraic codes. In contrast to LDPC codes, for which the single-parity-check component code provides only error detection, BCH codes provide error correction.

Following [18], we briefly review the basics of BCH codes. Consider a binary $n$-tuple $v = (v_0, v_1, \ldots, v_{n-1})$ and the associated polynomial $v(x) = v_0 + v_1x + \cdots + v_{n-1}x^{n-1}$. For a finite field with $n = 2^m$ elements, i.e., $\mathbb{F}_{2^m}$, and $\alpha$ a primitive element of $\mathbb{F}_{2^m}$, a (primitive) binary $t$-error-correcting BCH code is defined as the set of $v$ such that $v(\alpha), v(\alpha^2), v(\alpha^3), \ldots, v(\alpha^{2t})$ are roots. Equivalently, the parity-check matrix is

$$
H = \begin{bmatrix}
1 & \alpha & \alpha^2 & \cdots & \alpha^{n-1} \\
1 & \alpha^2 & (\alpha^2)^2 & \cdots & (\alpha^2)^{n-1} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
1 & \alpha^{2t} & (\alpha^{2t})^2 & \cdots & (\alpha^{2t})^{n-1}
\end{bmatrix}
$$

Since any subset of $2t$ columns of $H$ forms a Vandermonde matrix, the code has minimum distance $d_{\text{min}} \geq 2t + 1$. Finally, the minimal polynomial of $\alpha^i$, $1 \leq i \leq 2t$, necessarily divides any codeword $v(x)$, therefore, the generator polynomial $g(x)$ of a BCH code is

$$
g(x) = \text{LCM}\{m_{\alpha}(x), m_{\alpha^2}(x), \ldots, m_{\alpha^{2t}}(x)\},
$$

where $m_{\alpha^i}(x)$ is the minimal polynomial of $\alpha^i$. Note that the code is cyclic, since $g(x)$ divides $x^n + 1$.

Syndrome decoding

BCH codes are efficiently decodable by syndrome decoding. For a received polynomial $r(x) = r_0 + r_1x + \cdots + r_{n-1}x^{n-1}$, the inner product with each of the rows of $H$ yields the syndrome $S = (r(\alpha), r(\alpha^2), \ldots, r(\alpha^{2t}))$. Now, since $r(x) = v(x) + e(x)$, we have $r(\alpha^i) = e(\alpha^i)$. Therefore, if the received polynomial has $\nu$ errors, in locations $x^{j_l}$, $0 \leq j_l \leq n-1$, $1 \leq l \leq \nu$, then

$$
r(\alpha^i) = (\alpha^i)^{j_1} + (\alpha^i)^{j_2} + \cdots + (\alpha^i)^{j_\nu}.
$$

In the general case, if $\nu \leq t$, the Berlekamp algorithm [19] constructs an error-locator polynomial from $S$, whose roots are the reciprocals of the $\alpha^{j_l}$; a Chien search [20] is typically used for root-finding. In this work, we are primarily interested in BCH codes.
with $t \leq 5$, for which techniques with even greater efficiency exist; direct formulas exist for the coefficients of the error locator polynomial [21], and fast methods for root-finding are available [22, 23]. In Appendix A, we describe these techniques for triple-error-correcting BCH codes.

### 2.2.3 Iterative decoding of BCH $\times$ BCH product codes

When the component codes of a product code can be efficiently decoded via syndromes, there exists an especially efficient decoder for the product code. Briefly, by operating exclusively in the ‘syndrome domain’—which compresses the received signal—and passing at most $t$ binary messages per (component) decoding (for $t$-error-correcting component codes), the implementation complexity of decoding is significantly reduced relative to a soft-decision message passing decoder. The following is a step-by-step description of the decoding algorithm, similar to [24]:

1. From the received data, compute and store the syndrome for each row and column codeword. Store a copy of the received data in memory $R$.

2. Decode those non-zero syndromes corresponding to row codewords. In the event of a successful decoding, set the syndrome to zero, flip the corresponding $t$ or fewer positions in memory $R$, and update the $t$ or fewer affected column syndromes by a masking operation.

3. Repeat Step 2, reversing the roles of rows and columns.

4. If any syndromes are non-zero, and fewer than the maximum number of iterations have been performed, go to Step 2. Otherwise, output the contents of memory $R$.

### 2.2.4 LDPC vs. Product Codes: Decoder dataflow

Although LDPC codes (and related codes) provide the potential for capacity-approaching performance, at 100 Gb/s their implementation presents significant hurdles. While implementations exist at 10 Gb/s (for 10GBase-T ethernet networks), the blocklengths of such implementations ($\sim 500 – 2000$) are too short to provide performance close to capacity; the $(2048, 1723)$ RS-LDPC code is approximately 3 dB from the Shannon Limit at $10^{-15}$ [25], see also [26]. Another significant roadblock is that fibre-optic communication systems are typically required to provide bit-error-rates below $10^{-15}$. It is well-known
that capacity-approaching LDPC codes exhibit error floors, and to achieve the targeted
error rate would likely require concatenation with an outer code (possibly a high-rate
product code, in order to maintain near-capacity performance), thus necessitating the
design of two ‘powerful’ codes.

In this section, we present a high-level view of iterative decoders for LDPC and
product codes. Due to the differences in their implementations, a precise comparison
of their implementation complexities is difficult. Nevertheless, since the communication
complexity of message-passing is a significant challenge in LDPC decoder design, we
consider the decoder dataflow, i.e., the rate of routing/storing messages, as a surrogate
for the implementation complexity.

In the following, we consider a system that transmits information at \( D \) bits/s, using
a binary error-correcting code of rate \( R \), for which hard decisions at \( D/R \) bits/s are
input to the decoder. The decoder operates at a clock frequency of \( C \) Hz. In the
case of LDPC decoding, we assume \( q \)-bit messages internal to the decoder, an average
variable node degree \( d_{av} \), and \( N \) decoder iterations; typically, \( q \) is 4 or 5 bits, \( d_{av} \approx 3 \),
and \( N \sim 15 – 25 \). Iterative decoding of \( \text{BCH} \times \text{BCH} \) product codes is implemented as
described in Section 2.2.3; we assume each row/column codeword is decoded (on average,
over the course of decoding the overall product code) \( v \) times, where typically \( v \) ranges
from 3 to 4.

**LDPC decoder dataflow**

We consider an LDPC decoder that implements sum-product decoding (or some quan-
tized approximation) with a parallel flooding schedule. Initially, hard-decisions are input
to the decoder at a rate of \( D/R \) bits/s and stored in flip-flop registers. At each iteration,
variable nodes compute and broadcast \( q \)-bit messages over every edge, and similarly for
the check nodes, i.e., \( 2qd_{av} \) bits are broadcast per iteration per variable node. Since bits
arrive from the channel at \( D/R \) bits/s, the corresponding internal dataflow per iteration
is then \( \frac{D 2qd_{av}}{R} \), and the total dataflow, including initial loading of 1-bit channel messages,
is

\[
F_{LDPC} = \frac{D}{R} + \frac{2NDqd_{av}}{R} \approx \frac{2NDqd_{av}}{R}.
\]

For \( N = 20, q = 4, d_{av} = 3 \), \( F_{LDPC} \approx \frac{480D}{R} \), which corresponds to a dataflow of more
than 48 Tb/s for 100 Gb/s systems.

![Diagram of LDPC decoder dataflow]

**Figure 2.6: Dataflow in an LDPC decoder**

We note that our definition of decoder dataflow is closely related to the definitions of decoding complexity for graph-based codes in [27, 28, 29], which define the complexity as the total number of messages passed by the decoder. On the other hand, the dataflow is dominated by the rate (which depends on the throughput of the decoder) of the total number of messages passed scaled by the resolution (in bits) of the messages.

**Product-code decoder dataflow**

The analysis of the decoder dataflow for a product code is slightly more involved. We assume that rows are encoded by a $t_1$-error-correcting $(n_1, k_1 = n_1 - r_1)$ BCH code, and the columns are encoded by a $t_2$-error-correcting $(n_2, k_2 = n_2 - r_2)$ BCH code, for an overall rate $R = R_1 R_2$.

The hard-decisions from the channel—at $D/R$ bits/s—are written to a data RAM, in addition to being processed by a syndrome computation/storage device. As opposed to the LDPC decoder dataflow, the clock frequency $C$ plays a central role, namely in the dataflow of the initial syndrome calculation. Referring to Fig. 2.7, and assuming that the bits in a product code are transmitted row-by-row, the input buswidth (i.e., the number of input bits per decoder clock cycle) is $D/(RC)$ bits. Now, assuming these bits correspond to a single row of the product code, each non-zero bit corresponds to some $r_1$-bit mask (i.e., the corresponding column of the parity-check matrix of the row code), the modulo-2 sum of these is performed by a masking tree, and the $r_1$-bit output is masked with the current contents of the corresponding (syndrome) flip-flop register. That is, each clock cycle causes a $r_1$-bit mask to be added to the contents of the corresponding row in the syndrome bank. Of course, each received bit also impacts a distinct column
syndrome, however, the same $r_2$-bit mask is applied (when the corresponding received bit is non-zero) to each of the involved column syndromes; the corresponding dataflow is then $r_2$ bits per clock cycle.

![Figure 2.7: Dataflow in the initial syndrome computing](image)

Once the syndromes are computed from the received data, iterative decoding commences. To perform a row decoding, an $r_1$-bit syndrome is read from the syndrome bank. Since there are $n_2$ row codewords, and each row is decoded on average $v$ times, the corresponding dataflow from the syndrome bank to the row decoder is $r_1 n_2 v D / (R n_1 n_2) = r_1 v D / (R n_1) \text{ bits/s}$. For each row decoding, at most $t_1$ positions are corrected, each of which is specified by $\lceil \log_2 n_1 \rceil + \lceil \log_2 n_2 \rceil$ bits. Therefore, the dataflow from the row decoder to the data RAM is

$$
\frac{t_1 n_2 v D (\lceil \log_2 n_1 \rceil + \lceil \log_2 n_2 \rceil)}{R n_1 n_2} = \frac{t_1 v D (\lceil \log_2 n_1 \rceil + \lceil \log_2 n_2 \rceil)}{R n_1}
$$

bits/s. Furthermore, for each corrected bit, an $r_2$-bit mask must be applied to the corresponding column syndrome, which yields a dataflow from the row decoder to the syndrome bank of $t_1 n_2 r_2 v D / (R n_1 n_2) = t_1 r_2 v D / (R n_1) \text{ bits/s}$. A similar analysis can be applied to column decodings. In total, the decoder dataflow is

$$
F_P = \frac{D}{R} + (r_1 + r_2) \cdot C
+ \frac{D v}{R n_1} \cdot (t_1 \lceil \log_2 n_1 \rceil + t_1 \lceil \log_2 n_2 \rceil + r_1 + t_1 r_2)
+ \frac{D v}{R n_2} \cdot (t_2 \lceil \log_2 n_1 \rceil + t_2 \lceil \log_2 n_2 \rceil + r_2 + t_2 r_1).
$$

In this work, we will focus on codes for which $n_1 = n_2 \approx 1000$, $r_1 = r_2 = 32$, $t_1 = t_2 = 3$, and the decoder is assumed to operate at $C \approx 400$ MHz. For $v = 4$, we then have a dataflow of approximately 293 Gb/s. Note that this is more than two orders of magnitude smaller than the corresponding dataflow for LDPC decoding. Intuitively, the advantage arises from two facts. First, when $R_1 > 1/2$ and $R_2 > 1/2$, syndromes provide
a compressed representation of the received signal. Second, the algebraic component codes admit an economical message-passing scheme, in the sense that message updates are only required for the small fraction of bits that are corrected by a particular (component code) decoding.

### 2.3 Error-correcting codes for optical transport networks

#### 2.3.1 Net coding gain

Many fibre-optic communication systems present a binary symmetric channel (BSC) to the error-correcting code. Indeed, it is commonplace within the fibre-optic community to describe any code on the basis of its net coding gain (NCG), a measure which implicitly assumes an underlying AWGN channel, BPSK modulation, and a slicer in the receiver (i.e., channel messages are quantized with 1 bit). Specifically, consider an uncoded system with BPSK modulation, symbol rate $T_s$, average transmit power $P$, and (real-valued) AWGN with power spectral density $\frac{N_0}{2}$. Defining

$$Q_{\text{unc}} = \sqrt{\frac{PT_s}{N_0}},$$

it can be shown that the error rate at the output of the slicer is

$$p_{\text{unc}} = \frac{1}{2} \text{erfc} (Q_{\text{unc}}).$$

Expressing $Q_{\text{unc}}$ in dB, we have

$$Q_{\text{unc,dB}} = 20 \log_{10} \left( \text{erfc}^{-1}(2p_{\text{unc}}) \right).$$
Next, consider a coded system with a rate $R$ error-correcting code, symbol rate $T_s/R$, and average transmit power $P_c$. Note that in the presence of coding, the error rate of the underlying channel increases if the average transmit power remains constant, since

$$Q_{\text{cod}} = \sqrt{\frac{P_c T_s R}{N_0}}.$$  

To match the error rate (at the output of the channel, but prior to decoding) of an uncoded system, a coded system requires $1/R$ times the average transmit power, but at the output of the error-correcting decoder, we see a channel with an error rate $p_{\text{cod}}$. Now, since an uncoded system requires $Q_{\text{dB}} = 20 \log_{10} \left( \text{erfc}^{-1}(2p_{\text{cod}}) \right)$ to match the error rate at the output of the coded system, but a coded system with input error rate $p_{\text{unc}}$ requires $Q_{\text{dB}} = 20 \log_{10} \left( \text{erfc}^{-1}(2p_{\text{unc}}) \right) + 10 \log_{10} \left( \frac{1}{R} \right)$, the net coding gain is defined as

$$NCG_{\text{dB}} = 20 \log_{10} \left( \text{erfc}^{-1}(2p_{\text{cod}}) \right) - \left( 20 \log_{10} \left( \text{erfc}^{-1}(2p_{\text{unc}}) \right) + 10 \log_{10} \left( \frac{1}{R} \right) \right). \quad (2.1)$$

The NCG is the decrease in required transmit power due to coding, for an error rate $p_{\text{cod}}$.

As standardized in ITU-T Recommendation G.709/Y.1331 [30], the rate for error-correcting codes in optical transport networks (OTNs) is $R = 239/255$. For an output-error-rate of $10^{-15}$, the NCG for a capacity-achieving code (for $p = 7.36 \times 10^{-3}$, $C_{\text{BSC}} = 239/255$) is 9.97 dB. Unless otherwise noted, the ‘gaps to capacity’ reported in this work are with respect to a BPSK-modulated AWGN channel with a slicer, as in the preceding discussion of NCG.

### 2.3.2 ITU-T recommendations G.975 and G.975.1

Early fibre-optic communication systems were operated without any error-correction, largely relying on improvements in the physical devices to provide reliable communication, at least in part because electronics were not available at the required operating speeds. Of course, this implies operation far from the capacity of the underlying channel. One of the first proposals for error-correction in an optical system appeared in [31], which demonstrated a shortened $(224, 216)$ Hamming code implementation at 565 Mbit/s. Since then, more powerful codes have been standardized for fibre-optic communication systems. In this section, we describe these proposals, which provide a performance baseline for the codes we design in Chapter 4.
ITU-T recommendation G.975

The first error-correction code standardized for optical communications—in ITU-T recommendation G.975 [32]—was the \((255, 239)\) Reed-Solomon code, with symbols in \(\mathbb{F}_{2^8}\), capable of correcting up to 8 symbol errors in any codeword. In order to provide improved burst-error-correction, 16 codewords are block-interleaved, providing correction for bursts of as many as 1024 transmitted bits. A framing row consists of \(16 \cdot 255 \cdot 8\) bits, 30592 of which are information bits, and the remaining 2048 bits are parity. The resulting framing structure—a frame consists of four framing rows—is standardized in ITU-T recommendation G.709, and remains the required framing structure for present-day fibre-optic communication systems; as a direct result, the coding rate of any candidate code must be \(R = 239/255\).

In Fig. 2.9, the performance of the \((255, 239)\) RS code is presented, as well as the performance of four representative codes from the ITU-T recommendation G.975.1 [33], which we describe in greater detail below. For an output-error-rate of \(10^{-15}\), the NCG of the RS code is 6.2 dB, which is 3.77 dB from capacity.

ITU-T recommendation G.975.1

As per-channel data rates increased to 10 Gb/s, and the capabilities of high-speed electronics improved, the \((255, 239)\) RS code was replaced with stronger error-correcting codes. In ITU-T recommendation G.975.1, several ‘next-generation’ coding schemes were proposed; among the many proposals, the common mechanism for increased coding gain was the use of concatenated coding schemes with iterative hard-decision decoding. We now describe four of the best proposals, which will motivate our approach in Chapter 4.

In Appendix I.3 of G.975.1, a serially concatenated coding scheme is described, with outer \((3860, 3824)\) binary BCH code and inner \((2040, 1930)\) binary BCH code, which are obtained by shortening their respective mother codes. First, 30592 = \(8 \cdot 3824\) information bits are divided into 8 units, each of which is encoded by the outer code; we will refer to the resulting unit of 30880 bits as a ‘block’. Prior to encoding by the inner code, the contents of consecutive blocks are interleaved (in a ‘continuous’ fashion, similar to convolutional interleavers [34]). Specifically, each inner codeword in a given block involves ‘information’ bits from each of the eight preceding ‘outer’ blocks. Note that the interleaving step increases the effective blocklength of the overall code, but it necessitates a sliding-window style decoding algorithm, due to the continuous nature of
the interleaver. Furthermore, unlike a product code, the parity bits of the inner code are
protected by a single component codeword, which reduces their level of protection. For
an output-error-rate of $10^{-15}$, the NCG of the I.3 code is 8.99 dB, which is 0.98 dB from
capacity.

In Appendix I.4 of G.975.1, a serially concatenated scheme with (shortened versions
of) an outer (1023, 1007) RS code and (shortened versions of) an inner (2047, 1952) binary
BCH code is proposed. After encoding 122368 bits with the outer code, the coded bits
are block interleaved and encoded by the inner BCH code, resulting in a blocklength of
130560 bits, i.e., exactly one G.709 frame. As in the previous case, the parity bits of the
inner code are singly-protected. For an output-error-rate of $10^{-15}$, the NCG of the I.4
code is 8.67 dB, which is 1.3 dB from capacity.

In Appendix I.5 of G.975.1, a serially concatenated scheme with an outer (1901, 1855)
RS code and an inner $(512, 502) \times (510, 500)$ extended-Hamming product code is de-
scribed. Iterative decoding is applied to the inner product code, after which the outer code is decoded; the purpose of the outer code is to eliminate the error floor of the inner code, since the inner code has small stall patterns (see Ch. 4). For an output-error-rate of $10^{-15}$, the NCG of the I.5 code is $8.5$ dB, which is $1.47$ dB from capacity.

Finally, in Appendix I.9 of G.975.1, a product-like code with $(1020,988)$ doubly-extended binary BCH component codes is proposed. The overall code is described in terms of a $512 \times 1020$ matrix of bits, in which the bits along both the rows of the matrix as well as a particular choice of ‘diagonals’ must form valid codewords in the component code. Since the diagonals are chosen to include 2 bits in every row, any diagonal codeword has two bits in common with any row codeword; in contrast, for a product code, any row and column have exactly one bit in common. Note that the I.9 construction achieves a product-like construction (their choice of diagonals ensures that each bit is protected by two component codewords) with essentially half the overall blocklength of the related product code (even so, the I.9 code has the longest blocklength among all G.975.1 proposals). However, the choice of diagonals decreases the size of the smallest stall patterns, introducing an error floor above $10^{-14}$. For an output-error-rate of $2 \cdot 10^{-14}$, the NCG of the I.9 code is $8.67$ dB, which is $1.3$ dB from capacity.
Chapter 3

Digital backpropagation and memoryless capacity estimations

At first glance, the GNLS equation

$$\frac{\partial A}{\partial z} + \alpha - g(z) - \frac{j\beta_2}{2} \frac{\partial^2 A}{\partial t^2} - j\gamma |A|^2 A = n(z,t),$$

discussed in Section 2.1.3, is a rather daunting channel model. Stochastic effects (noise), linear effects (dispersion) and nonlinear effects (Kerr nonlinearity) interact throughout propagation, and—even in the absence of noise—its solution requires numerical techniques. On the other hand, in the absence of noise, the system is invertible, i.e., the transmitted signal $A(0,t)$ can be recovered from the received signal $A(NL_A)$ by inverting the channel. When the channel is inverted by digital signal processing, we say the receiver performs digital backpropagation. In Section 3.1, we review existing applications of backpropagation, with a focus on the recent work of Essiambre et al. [1], which provides estimates for the information capacity of fibre-optic systems that use backpropagation. In Section 3.2, we calculate the achievable rates in the absence of backpropagation, and show that the relative merits of backpropagation depend strongly on whether the system is single-channel or wavelength-division-multiplexed. In particular, we show that for WDM systems, the channel is effectively linear for most input powers of interest. Finally, Section 3.3 studies the tradeoff between achievable rates and the backpropagation step-size, within the regime of input powers for which backpropagation provides some benefit.
3.1 A review of known results and techniques

3.1.1 The split-step Fourier method

The most commonly used numerical method to solve the GNLS equation is the split-step Fourier method [10, 35]. The basic idea is to divide the total fibre length into short segments, then to consider each segment as the concatenation of (separable) nonlinear and linear transforms (for distributed amplification, an additive noise is added after the linear step). In the following, we briefly review the split-step Fourier method. For simplicity of the presentation, we ignore the effects of amplification, which can be incorporated into a numerical solver in an obvious manner.

For a known \( A(z = z_0, t) \), the split-step Fourier method calculates \( A(z = z_0 + h, t) \) as follows. First, in the absence of linear effects, the GNLS has the form,

\[
\frac{\partial A}{\partial z} = j\gamma |A|^2 A,
\]

with solution,

\[
A(z = z_0 + h, t) = A(z = z_0, t) \exp(j\gamma |A(z = z_0, t)|^2 h).
\]

We now use this solution as the input to the linear step, i.e., let

\[
\hat{A}(z = z_0, t) = A(z = z_0, t) \exp(j\gamma |A(z = z_0, t)|^2 h)
\]

be the input to the linear step. The linear form of the GNLS is

\[
\frac{\partial A}{\partial z} = -\frac{\alpha}{2} A - \frac{j\beta_2}{2} \frac{\partial^2 A}{\partial t^2},
\]

which can be efficiently solved in the frequency domain. Defining

\[
A(z, t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \tilde{A}(z, \omega) \exp(j\omega t) d\omega,
\]

it can be shown that

\[
\tilde{A}(z = z_0 + h, \omega) = \tilde{A}(z = z_0, \omega) \exp \left( \left( j\frac{\beta_2}{2} \omega^2 - \frac{\alpha}{2} \right) h \right).
\]  \(\text{(3.1)}\)

Putting this together, we have

\[
A(z = z_0 + h, t) = \mathcal{F}^{-1} \left\{ \mathcal{F} \left\{ \tilde{A}(z = z_0, t) \right\} \exp \left( \left( j\frac{\beta_2}{2} \omega^2 - \frac{\alpha}{2} \right) h \right) \right\},
\]

where \( \mathcal{F} \) is the Fourier transform operator.
3.1.2 Single-channel backpropagation

In the absence of stochastic effects, the effects of propagation over a distance $L$ of optical fibre can be inverted by solving the GNLS in reverse. That is, from $A(z = L, t)$, we can compute $A(z = 0, t)$ by the split-step Fourier method, using a negative step-size $h$. When this process is implemented electronically (via digital signal processing in the receiver), it is referred to as digital backpropagation. Note that, in general, $A(z, t)$ is the complex envelope of a multi-channel optical signal. It then follows that full compensation of channel impairments—even if only a single channel is of interest to the receiver—requires backpropagation to be performed on the multi-channel signal, since nonlinearity induces interaction between signal components at non-overlapping frequencies. However, in practice, receivers operate on a per-channel basis. Even if a multi-channel receiver were available, co-propagating channels may be optically-routed in/out throughout transmission, and thus channels that co-propagated with the desired channel may not even be available at the receiver (and those channels that are available may not have co-propagated with the desired channel). Therefore, we consider single-channel backpropagation, in which the receiver first extracts the channel of interest from $A(z = L, t)$ (via a bandpass filter), and performs digital backpropagation on the corresponding signal. Intuitively, if intra-channel nonlinear effects dominate inter-channel nonlinear effects, then single-channel backpropagation should effectively compensate for channel distortions.

3.1.3 Memoryless capacity estimations

In [1], Essiambre et al. present an estimate of the information theoretic capacity of optical fiber networks. In the remainder of this section, we describe their technique, which we make use of throughout this chapter.

Transmitter

We consider a system that employs pulse-amplitude modulation (PAM) with (orthonormal) sinc pulses. That is, the transmitted signal (corresponding to the baseband representation of the $l$-th channel) is of the form

$$X_l(t) = \sum_{k=-\infty}^{\infty} \phi_{k,l} \frac{1}{\sqrt{T_s}} \text{sinc} \left( \frac{t - kT_s}{T_s} \right),$$
where \( \text{sinc}(\theta) = \frac{\sin \pi \theta}{\pi \theta} \). The \( \phi_{k,l} \) are elements of a discrete-amplitude continuous-phase input constellation \( \mathcal{M} \), i.e., for \( N \) rings, \( \theta \in [0, 2\pi) \), and \( r \geq 0 \),

\[
\mathcal{M} = \{ m \cdot r \exp(j\theta) | m \in \{1, 2, \ldots, N\} \}.
\]

Each ring is assumed equiprobable, and for a given ring, the phase distribution is uniform. This choice of constellation is motivated by the fact that the channel represented by the GNLS can be argued to be statistically rotationally invariant (i.e., for a channel with input \( x \), output \( y \), and conditional distribution \( f(y|x) \), \( f(y|x_0) = f(y \exp(j\theta) | x_0 \exp(j\theta)) \) for \( \theta \in [0, 2\pi) \)) and thus points on the same ring can be considered ‘equivalent’, which reduces the computational requirements in characterizing the channel. Furthermore, for \( N = 32 \), the Shannon Limit of the AWGN channel can be approached to within less than 0.65 dB for spectral efficiencies below 8.5 bits/s/Hz.

In the general case of a multi-channel system with a channel spacing \( 1/T_s \) Hz, the input to the fibre has the general form

\[
A(z = 0, t) = \sum_{k=-\infty}^{\infty} \sum_{l=-M}^{M} \phi_{k,l} \frac{1}{\sqrt{T_s}} \text{sinc} \left( \frac{t - kT_s}{T_s} \right) \exp(j2\pi lt/T_s).
\]

**Receiver**

The operation of a (baseband) digital coherent optical receiver (for the \( l \)-th channel) is illustrated in Fig. 3.1. From the channel output \( A(L, t) \), a bandpass filter centered at \( \omega = \frac{2\pi}{T_s} l \) extracts the \( l \)-th channel, and the corresponding signal is sampled at the rate \( 1/T_s \). The resulting discrete-time signal is then digitally-backpropagated, providing estimates \( \hat{\phi}_{k,l} \) of the transmitted symbols \( \phi_{k,l} \).

![Figure 3.1: System model for memoryless capacity estimations](image)

**Channel Model**

In order to facilitate the capacity estimation, the discrete-time channel is assumed to be *memoryless*, i.e., it is assumed that backpropagation removes any dependence between
received symbols. The (memoryless) conditional distribution of the channel is estimated from numerical simulations.

Since the channel is statistically rotationally invariant, observations of transmitted points from the same ring are first ‘back-rotated’ to the real axis, as illustrated in Fig. 3.2. The back-rotated points are represented by \( \tilde{\phi}_{k,l} \),

\[
\tilde{\phi}_{k,l} = \hat{\phi}_{k,l} \exp (-j(\Phi_{XPM} + \angle \phi_{k,l})) ,
\]

where \( \Phi_{XPM} \) is a constant (input-independent) phase rotation contributed by cross-phase modulation (XPM).

![Figure 3.2: Channel outputs for a fixed-ring input, and back-rotated outputs.](image)

Next, for each \( i \) and a fixed \( l \) (the channel of interest), we calculate the mean \( \mu_i \) and covariance matrix \( \Omega_i \) (of the real and imaginary components) of those \( \tilde{\phi}_{k,l} \) corresponding to the \( i \)-th ring, and model the distribution of those \( \tilde{\phi}_{k,l} \) by \( \mathcal{N}(\mu_i, \Omega_i) \). Finally, from the rotational invariance of the channel, the channel is modeled as

\[
f(y|x = r \cdot i \exp (j\theta)) \sim \mathcal{N}(\mu_i \exp (j\theta), \Omega_i),
\]

where the (constant) phase rotation due to \( \Phi_{XPM} \) is ignored, since it can be canceled in the receiver. Note that this model reduces to an additive ‘noise’ model when \( \mu_i = (r \cdot i, 0) \), but in general this relationship need not be true. In particular, for sufficiently large input powers, we will see in Section 3.2.4 that \( \mu_i \) approaches \( (0, 0) \).
Chapter 3. Digital backpropagation and memoryless capacity estimations

Capacity estimation

The mutual information of the memoryless channel is [36]

\[ I(X;Y) = \int \int f(x,y) \log_2 \frac{f(y|x)}{f(y)} \, dx \, dy, \]

where \( f(x) \) represents the input distribution on \( \mathcal{M} \) with equiprobable rings and a uniform phase distribution, which provides an estimate of the capacity of an optically-routed fibre-optic communication system.

Signaling parameters

We now provide the parameters of the signaling scheme, to be used throughout the remainder of this thesis. In general, further increasing the number of simulated channels has a negligible effect on the capacity estimates.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baud rate (1/T_s)</td>
<td>100 GHz</td>
</tr>
<tr>
<td>Channel bandwidth (W)</td>
<td>101 GHz</td>
</tr>
<tr>
<td>Number of rings (N)</td>
<td>64</td>
</tr>
<tr>
<td>Number of channels</td>
<td>(2M + 1 = 5)</td>
</tr>
</tbody>
</table>

Table 3.1: Signaling parameter values

3.2 Memoryless information rates in the absence of backpropagation

Although many current state-of-the-art systems include some form of electronic dispersion compensation (i.e., equalization) in the receiver, digital backpropagation is significantly more computationally intensive, since many steps—each of which has roughly the complexity of a standard dispersion compensation scheme—of the split-step Fourier method are required to accurately compensate the nonlinear effects (in Section 3.3, we return to this point). Of course, state-of-the-art systems—which employ DP-DQPSK modulation—operate far from the capacity of the channel, i.e., at an input power at which nonlinearity does not significantly affect transmission, and thus linear equalization is sufficient. More generally, in this section we investigate the full range of (memoryless)
achievable information rates when only linear equalization is performed, and compare the achievable rates to those of a system that performs single-channel digital backpropagation. Contrary to [1], which considered only distributed amplification, we also consider systems with lumped amplification, which are of interest since existing installations are predominantly of this type.

From (3.1), we see that the linear effects of the channel are equivalent to an all-pass filter, with a phase quadratic in frequency. Ignoring fiber losses (which are compensated via amplification), the filter with frequency response

\[ H(j\omega) = \exp\left(-j\beta_2\omega^2L\right) \]

fully compensates the dispersion introduced by the fiber over a propagation distance \( L \). Indeed, if \( \gamma = 0 \), this so-called zero-forcing equalizer (which inverts the channel) is the optimal receiver structure [37]. Furthermore, since \( H(j\omega) \) is unitary, linear equalization can equivalently be performed at the transmitter or the receiver. In fact, even in the presence of nonlinearity, the achievable information rates are essentially independent of the choice of pre/post-equalization (or some hybrid of the two), and the results presented herein for linearly-equalized systems should be understood to correspond to any system that electronically compensates fiber dispersion by multiplication with \( H(j\omega) \).

### 3.2.1 Ideal distributed Raman amplification

We first consider the case of ideal distributed Raman amplification. For systems of length \( L = 500 \) km, \( 1000 \) km and \( 2000 \) km, we calculate the achievable information rates for each of the following cases: single-channel transmission with equalization; single-channel transmission with single-channel backpropagation; WDM transmission with equalization; and WDM transmission with single-channel backpropagation.

**Single-channel transmission**

For long-haul fibre-optic communication systems, economic considerations dictate that multiple signals must be wavelength division multiplexed. In practice, the packing of channels is ‘dense’, i.e., the width of guard bands between channels is kept to a minimum. However, since inter-channel nonlinear effects between two channels are strongly dependent on the frequency spacing between them (and we assume that such distortions cannot be compensated in the receiver), it is not immediately clear that a dense packing
is optimal. A single-channel transmission system represents the extreme case, in that there are no inter-channel nonlinear effects. In this case, the benefits of digital backpropagation ought to be greatest, and some conclusions about the potential benefits of a less dense spacing can be made.

![Figure 3.3: Achievable spectral efficiencies for single-channel transmission with Raman amplification](image)

In Fig. 3.3, we present the achievable spectral efficiencies for a single-channel transmission system with ideal distributed Raman amplification, for each of the three investigated transmission lengths $L$. The signal-to-noise ratio (SNR) is defined as

$$SNR = \frac{P}{N_{ASE}W},$$

where $P$ is the average transmitter power, $W$ is the bandwidth occupied by a single channel, and $N_{ASE}$ (where $N_{ASE} = N_{tot}^{\text{EDFA}}$ for a system with lumped amplification, and $N_{ASE} = N_{tot}^{\text{Raman}}$ for a system with distributed amplification) is the power spectral density of the added noise\(^1\). Note that increasing the transmission distance also increases the total noise contributed by amplification, i.e., $N_{ASE}$ implicitly depends on $L$. Specifically, for the same input power, the system with $L = 500$ km has an SNR that is 3 dB larger

\(^1\)In contrast to conventional linear Gaussian channel models, $N_{ASE}$ is fixed by the choices of $L$ and the amplification technique. Therefore, for a fixed system, the SNR can be increased only by increasing the input power.
than the system with $L = 1000$ km, which itself has an SNR 3 dB larger than the system with $L = 2000$ km. Interestingly, the peak spectral efficiencies also exhibit the same dependence, in that the spectral efficiency decreases by 1 bit/s/Hz for every doubling of transmission distance, both with and without digital backpropagation.

For $L = 2000$ km, the peak spectral efficiency is approximately 6.6 bits/s/Hz when only linear equalization is performed, but increases to approximately 10.1 bits/s/Hz for digital backpropagation. For $L = 1000$ km, the peak spectral efficiency is approximately 7.6 bits/s/Hz when only linear equalization is performed, but increases to approximately 11.1 bits/s/Hz for digital backpropagation. Finally, for $L = 500$ km, the peak spectral efficiency is approximately 8.6 bits/s/Hz when only linear equalization is performed, but increases to approximately 12.0 bits/s/Hz for digital backpropagation.

WDM

The opposite extreme of the single-channel case is that of wavelength division multiplexing with zero guard-band between channels. Clearly, the introduction of inter-channel nonlinear effects can only degrade the performance of the system, and the benefits of single-channel digital backpropagation strongly depend on the respective input powers at which intra- and inter-channel nonlinear effects become significant. In Fig. 3.4, we present the achievable spectral efficiencies for a WDM system (with zero guard-band) with ideal distributed Raman amplification, for each of the three investigated transmission lengths $L$.

For $L = 2000$ km, the peak spectral efficiency is approximately 6.45 bits/s/Hz when only linear equalization is performed, but increases to approximately 7.2 bits/s/Hz for digital backpropagation. For $L = 1000$ km, the peak spectral efficiency is approximately 7.4 bits/s/Hz when only linear equalization is performed, but increases to approximately 8.1 bits/s/Hz for digital backpropagation. Finally, for $L = 500$ km, the peak spectral efficiency is approximately 8.45 bits/s/Hz when only linear equalization is performed, but increases to approximately 9.0 bits/s/Hz for digital backpropagation.

Note that the benefits of digital backpropagation are indeed significantly reduced in a WDM system; the benefits of digital backpropagation are only 0.55 to 0.75 bits/s/Hz for the cases considered. From the standpoint of achievable rates, the channel is ‘nearly’ linear for most input powers of interest, in the sense that linear equalization achieves rates that closely approach those achievable via backpropagation. Furthermore, when only
linear equalization is performed, the difference in performance between single-channel and WDM transmission is minimal, as emphasized by Fig. 3.5, which provides similar evidence that inter-channel nonlinearities have a minimal influence in this ‘linear’ regime.

3.2.2 Ideal EDFA lumped amplification

We now consider the case of ideal EDFA lumped amplification, for the same system lengths and cases as for the systems with distributed amplification. There are two essential differences between systems with lumped and distributed amplification: For a fixed average input power $P$, systems with lumped amplification have a lower SNR, but also experience lower nonlinearity since the power of the propagating signal attenuates between amplification sites. We note that for a system with spans of length $L_A$, the ratio of the power spectral densities of the total noises is

$$\frac{N_{\text{tot, EDFA}}}{N_{\text{tot, Raman}}} = \frac{(\exp(\alpha L_A) - 1) n_{sp}}{\alpha L_A k_T},$$

Figure 3.4: Achievable spectral efficiencies for WDM transmission with Raman amplification
which follows from the discussion in Sec. 2.1.3. For the parameters we consider, we have $\frac{N_{\text{tot}}^{\text{EDFA}}}{N_{\text{tot}}^{\text{Raman}}} = 3.459$, or 5.39 dB. However, since the input power is chosen to maximize the spectral efficiency, it is not obvious which of the two amplification schemes results in the greatest spectral efficiency.

### Single-channel transmission

In Fig. 3.6, we present the achievable spectral efficiencies for a single-channel transmission system with ideal lumped EDFA amplification. For $L = 2000$ km, the peak spectral efficiency is approximately 6.25 bits/s/Hz when only linear equalization is performed, but increases to approximately 9.4 bits/s/Hz for digital backpropagation. For $L = 1000$ km, the peak spectral efficiency is approximately 7.35 bits/s/Hz when only linear equalization is performed, but increases to approximately 10.5 bits/s/Hz for digital backpropagation. Finally, for $L = 500$ km, the peak spectral efficiency is approximately 8.3 bits/s/Hz when only linear equalization is performed, but increases to approximately 11.4 bits/s/Hz for digital backpropagation.
As in the case of distributed amplification, we see that the improvement (in bits/s/Hz) is roughly constant with transmission distance. Comparing the achievable rates for distributed amplification to those for lumped amplification, we see that the ultimate spectral efficiency is greater by approximately 0.7 bits/s/Hz for the distributed system, but that for equalization only this improvement reduces to approximately 0.3 bits/s/Hz.

Figure 3.6: Achievable spectral efficiencies for single-channel transmission with EDFA amplification

**WDM**

In Fig. 3.7, we present the achievable spectral efficiencies for a WDM transmission system with ideal lumped EDFA amplification. For $L = 2000$ km, the peak spectral efficiency is approximately 6.1 bits/s/Hz when only linear equalization is performed, but increases to approximately 6.6 bits/s/Hz for digital backpropagation. For $L = 1000$ km, the peak spectral efficiency is approximately 7.1 bits/s/Hz when only linear equalization is performed, but increases to approximately 7.55 bits/s/Hz for digital backpropagation. Finally, for $L = 500$ km, the peak spectral efficiency is approximately 8.05 bits/s/Hz when
only linear equalization is performed, but increases to approximately 8.55 bits/s/Hz for digital backpropagation.

![Figure 3.7: Achievable spectral efficiencies for WDM transmission with EDFA amplification](image)

Similar to the case of distributed amplification, we see that the benefits of digital backpropagation are significantly reduced in a WDM system. For the cases considered, the benefits of digital backpropagation are 0.45 to 0.5 bits/s/Hz. When only linear equalization is performed, the difference in performance between single-channel and WDM transmission is minimal, as emphasized by Fig. 3.8. Finally, comparing the achievable rates for distributed amplification to those for lumped amplification, we see that the ultimate spectral efficiency is greater by approximately 0.55 bits/s/Hz for the distributed system, but that for equalization only this improvement reduces to approximately 0.35 bits/s/Hz.

### 3.2.3 Revisiting the Gaussian assumption

For a fixed input, the fact that the conditional distribution of the output is assumed to be Gaussian distributed is motivated by the fact that amplification contributes additive
white Gaussian noise, as well as the fact that, intuitively, dispersion causes the cumulative
effects of the instantaneous nonlinearity to be ‘averaged’ in the sense of the central limit
theorem. In this section, we illustrate this effect by plotting the conditional distribution
of the outputs, as obtained from the simulations used to perform the capacity estimations.
For brevity, we present results only for the ideal distributed amplification system of length
$L = 2000$ km with WDM transmission, noting that analogous results are observed for
the other systems considered herein.

Both the validity of the Gaussian assumption and the (beneficial) effects of backprop-
agation can be made apparent by plotting the conditional distribution of the received
signal (and its backpropagated counterpart). In the following, we focus on a pair of
input powers, $P = -8$ dBm and $P = -5$ dBm, which correspond to points in Fig. 3.4
with SNR = 20.75 dB and SNR = 23.75 dB, respectively. The corresponding achievable
spectral efficiencies with backpropagation are 6.69 and 7.26 bits/s/Hz, and with equal-
ization are 6.41 and 6.39 bits/s/Hz. In Fig. 3.9 and Fig. 3.10, we present the scatter
diagrams of the received points (points on the same ring are back-rotated to the real

Figure 3.8: Comparison of achievable spectral efficiencies for WDM and single-channel
transmission, without backpropagation
axis) corresponding to the fourth, eighth, twelfth and sixteenth rings of the constellation.

![Figure 3.9: Scatter plots of back-rotated constellation points, for $P = -8$ dBm](image)

To gain a clearer understanding, we investigate the distribution of the noise ‘added’ to the transmitted points, i.e., the distribution of the difference between the transmitted and received (back-rotated) points. From the scatter diagrams, we observe that the variance of the noise increases as the ring radius increases. Thus, in the following we focus on the distribution of the noise for the fourth and sixteenth rings. Note that the distribution along the imaginary axis corresponds to the distribution along the azimuth direction of the non-backrotated points, and similarly the real axis corresponds to the radial direction.

In Fig. 3.11 we present the noise distribution of points on the fourth ring when $P = -8$ dBm, and Fig. 3.12 presents the corresponding noise distribution for the sixteenth ring. Similarly, in Fig. 3.13 we present the noise distribution of points on the fourth ring when $P = -5$ dBm, and Fig. 3.14 presents the corresponding noise distribution for the sixteenth ring.

Note that in each of the cases, the marginal distributions are well-approximated by Gaussian distributions. We also note that the variance along the azimuth direction is larger than along the radial direction, indicating that the dominant nonlinear mechanism
Figure 3.10: Scatter plots of back-rotated constellation points, for $P = -5$ dBm

Figure 3.11: Noise distribution of back-rotated constellation points on fourth ring, for $P = -8$ dBm
Figure 3.12: Noise distribution of back-rotated constellation points on sixteenth ring, for $P = -8$ dBm

is cross-phase modulation (XPM). Finally, the distributions clearly demonstrate the effects of backpropagation, namely, that the ‘noise’ variance contributed by signal-signal nonlinearity is (at least partially) canceled.

### 3.2.4 Non-monotonicity of achievable rates in the input power

The non-monotonicity (in the input power) of the achievable rate curves suggests that compensating the (distributed) interactions of linear, non-linear and stochastic effects becomes more difficult as the input power increases. To provide more insight, Figs. 3.15 presents the scatter plots of back-rotated constellation points on the first, sixteenth and thirty-second rings, for a WDM system of length $L = 500$ km, for a range of input powers. In each case, backpropagation is performed in the receiver, although similar results are observed when linear equalization is performed.

From Fig. 3.15, we see that for sufficiently large input powers, the variance of the ‘noise’ increases faster than the spacing between rings, thus reducing the mutual information. Initially, the dominant increase is in the form of a phase noise, due to uncompensated nonlinearity. For a sufficiently large input power, the output resembles
Figure 3.13: Noise distribution of back-rotated constellation points on fourth ring, for $P = -5$ dBm

Figure 3.14: Noise distribution of back-rotated constellation points on sixteenth ring, for $P = -5$ dBm
a random Gaussian process, and the output becomes independent of the input to the channel. This is not entirely surprising, since even in the absence of nonlinearity, the received signal (prior to dispersion compensation in the receiver, and assuming no dispersion pre-compensation at the transmitter) resembles a random Gaussian process. When the input power is sufficiently large, even the linear effects cannot be compensated (since they cannot be ‘separated’ from the nonlinear effects), and thus the output converges to a random Gaussian process.

![Scatter plots of back-rotated constellations points from the first, sixteenth and thirty-second rings, after receiver back-propagation.](image)

Figure 3.15: Scatter plots of back-rotated constellations points from the first, sixteenth and thirty-second rings, after receiver back-propagation.

### 3.3 Backpropagation step-size and achievable rates

In this section, we examine the tradeoff between achievable rates and the backpropagation step-size. Previously, Ip and Kahn [38, 39] investigated the impact of back-propagation step-size on the error rate of the channel, but our approach is more in keeping with
information theoretic principles.

We consider WDM systems with ideal distributed amplification, since they provide the greatest spectral efficiency among practical systems. Since digital backpropagation consists of a sequence of operations (namely, successive linear filtering and nonlinear phase rotations) that do not commute, the complexity of its implementation is proportional to the step-size used. Because distributed amplification causes the propagating signal to maintain a constant average power, we restrict our attention to schemes in which the step-size is constant throughout backpropagation.

In Fig. 3.16, we present the achievable spectral efficiencies for $L = 500$ km, in Fig. 3.17 for $L = 1000$ km, and in Fig. 3.18 for $L = 2000$ km. In each case, the uppermost curve applies to any constant step-size $s$, $0 < s \leq 2500$ m, i.e., the achievable spectral efficiency is essentially constant for any step-size less than 2.5 km. Increasing the step-size to $s = 5$ km reduces the complexity by a factor of two (relative to $s = 2.5$ km) with a penalty of 0.1 bits/s/Hz for each of the three systems. Similarly, increasing the step-size to $s = 10$ km results in a penalty of 0.4 bits/s/Hz (relative to the maximum), but still outperforms the system with equalization only. Of course, the system with equalization only is equivalent to a system with $s = L$ for which the backpropagation algorithm performs only the linear filtering step. From the curves for $s = 25$ km and $s = 50$ km, we see that the achievable spectral efficiency (of the induced ‘memoryless’ channel) actually decreases below that of equalization only, which indicates that the nonlinear phase rotation compensation actually distorts the signal when the step-size becomes too large.

### 3.4 Conclusions

In this chapter we showed that the communication channel induced by real-world fibre-optic communication systems is ‘nearly’ linear with respect to achievable spectral efficiencies. That is, while nonlinearity limits the maximum achievable spectral efficiency, compensation of nonlinearity by digital backpropagation provides only modest benefits relative to a ‘linear’ solution that relies on a single linear filter, which can be applied at the transmitter.

When the additional gains from backpropagation are desirable, we showed that step-sizes as large as 10 km provide an increase relative to equalization only, and that the ultimate spectral efficiency is attained for step-sizes as large as 2.5 km. We also showed that ideal distributed amplification provides an increase of 0.7 bits/s/Hz in spectral
efficiency relative to ideal lumped amplification with $L_A = 50$ km, which decreases to 0.3 bits/s/Hz when only linear equalization is performed.

From the linearity of the channel and the Gaussian distribution of the additive noise, classical coding methods ought to provide near-capacity reliable communications. However, due to the extremely high per-channel data rates of fibre-optic systems, implementation challenges arise. In the remainder of this work, we focus on the design (and application) of error-correcting codes that provide excellent performance while providing efficient high-speed implementations.
Figure 3.17: Comparison of achievable spectral efficiencies for WDM transmission ($L = 1000$ km) as a function of back-propagation step-size

Figure 3.18: Comparison of achievable spectral efficiencies for WDM transmission ($L = 2000$ km) as a function of back-propagation step-size
Chapter 4

Error-correcting codes for high-speed communications

Since 2004—when the G.975.1 recommendation was finalized—per-channel data rates in fibre-optic communication systems have increased from 10 Gb/s to 100 Gb/s, effectively keeping pace with the corresponding processing improvements due to high-speed electronics. As described in Section 2.2, product-like codes with algebraic component codes admit a significantly easier implementation than LDPC-like codes. In this chapter, we consider the design of codes that improve upon the performance of the G.975.1 codes while maintaining an efficient hardware implementation. We assume that only hard-decisions are available from the channel, and that the resulting channel is well-modeled as a binary symmetric channel.

In Section 4.1, we describe an optimization procedure for determining the choice of the component codes in a product-like code, such that the coding gain is maximized. In Section 4.2 we describe a new class of codes—staircase codes—that can be interpreted as a hybrid of product and convolutional codes, and that provide the best reported performance at $R = 239/255$. In Section 4.3, we devise a method to analytically predict the error floor of iteratively decoded staircase codes. In Section 4.4, we describe two generalizations of the staircase code construction. Finally, in Section 4.5, we compare staircase codes to previously proposed error-correcting codes, in order to emphasize the novelty of the constructions proposed in this work.
4.1 Irregular product-like codes

In the spirit of irregular LDPC codes, we investigate the merits of optimizing over the choice of the component codewords of a product-like code of rate $R = 239/255$, in order to improve upon the performance of the G.975.1 codes. Due to their efficient decoders, and suitability for binary error-correction, we consider BCH component codewords (and extended versions of these codes).

For irregular LDPC codes, the threshold of a family of codes—described by their degree distributions—can be efficiently computed by density evolution [40]. In practice, for a sufficiently long blocklength, the SNR at which the ‘waterfall’ of the bit-error-rate curve occurs is accurately predicted by the computed threshold of the underlying degree distribution. In order to design the code with the best performance, various optimization techniques have been proposed to identify those degree distributions with the highest thresholds (for a fixed rate, maximum node degrees, etc.) [41, 42, 43].

Density-evolution-like techniques (see also EXIT charts for Turbo codes [44]) exploit the fact that messages are likelihoods, and that a straightforward analysis of the evolution of these likelihoods follows when local codewords output ‘extrinsic’ messages via maximum a posteriori (MAP) decoding, and incoming messages to a local codeword are independent. However, these assumptions are not satisfied by syndrome-based decoding of product-like codes, and thus a density-evolution-like analysis is difficult. Nevertheless, sufficiently long product-like codes still exhibit a ‘thresholding’ effect (i.e., they have a very sharp waterfall), which we will exploit to rapidly estimate the threshold via simulation. Finally, we will propose two families of irregular product-like codes, and numerically identify those with the largest thresholds.

4.1.1 Threshold estimation

To motivate our method of threshold estimation, consider the following expression for the frame-error-rate (FER) of some code of length $N$ on a binary symmetric channel:

$$\text{FER} = \int_0^1 \Pr[p_{e,\text{obs}} = \alpha] \cdot \Pr[\text{Frame error occurs} | p_{e,\text{obs}} = \alpha] \, d\alpha,$$

where $p_{e,\text{obs}}$ is the ratio of the number of bit errors (introduced by some realization of the channel) to the blocklength $N$ of the code, i.e., the ‘observed’ error rate. Note that the first term of the integral depends only on the underlying channel, and the second term depends only on the code.
Analogous to the fact that LDPC-like codes exhibit a threshold effect, consider the idealized form of $\text{Pr}[\text{Frame error occurs}|p_{e,\text{obs}}]$ illustrated in Fig. 4.1, for which

$$\text{Pr}[\text{Frame error occurs}|p_{e,\text{obs}}] = 1$$

for $p_{e,\text{obs}} > \hat{p}$, and is zero otherwise. Now, for a binary symmetric channel with crossover probability $p$, the distribution of the number of errors in $N$ channel uses is well-approximated by $\mathcal{N}(Np, Np(1-p))$, and thus

$$\text{FER} = \text{Pr}[\text{Number of errors} > N\hat{p}]$$

$$= \frac{1}{2} \text{erfc} \left( \frac{\sqrt{N}(\hat{p} - p)}{\sqrt{2p(1-p)}} \right).$$

Finally, when $p_{e,\text{obs}}$ exceeds $\hat{p}$, if we assume the decoder gets ‘stuck’ without making any progress (this is analogous to what is observed for LDPC codes, where the dominant contribution to the waterfall is ‘large’ atypical events, see [45]), then we arrive at the following estimate for the bit-error-rate:

$$\text{BER} \approx \frac{\hat{p}}{2} \text{erfc} \left( \frac{\sqrt{N}(\hat{p} - p)}{\sqrt{2p(1-p)}} \right).$$

Despite the assumptions inherent in its derivation, this formula allows an accurate prediction of net coding gain, which we will make use of in comparing the performance of different product-like codes.

### 4.1.2 Optimization framework

Our main goal in this section is to determine the choice of component codewords that maximize the net coding gain of the overall code. By ‘product-like’ codes, we will mean
generalized LDPC codes with algebraic constraints (i.e., BCH rather than single-parity-check). Motivating this choice is that the overall blocklength does not depend on the length(s) of the component code(s), permitting the comparison of different length component codes on an equal footing, and it also admits the generalization to mixtures of component codes and arbitrary variable node degrees. Although others have investigated the problem of optimizing the choice of component codes in a generalized LDPC code when soft-decision decoding is performed [46], our treatment is distinct in that we focus on hard-decision decoding.

**Bi-regular codes**

We first consider the class of product-like codes with the factor graph representation in Fig. 4.2. Constraint nodes with edges to $\Pi_1$ correspond to a (possibly shortened) $(n_1, n_1 - r_1)$ BCH code $C_1$, and constraint nodes with edges to $\Pi_2$ correspond to a (possibly shortened) $(n_2, n_2 - r_2)$ BCH code $C_2$. The overall rate satisfies

$$R \geq 1 - \frac{r_1}{n_1} - \frac{r_2}{n_2},$$

but we will assume that equality holds, since the (expected) fraction of redundant constraints goes to zero as the overall blocklength increases [17].

![Figure 4.2: Factor graph of a bi-regular code.](image)

For a fixed choice of component codewords, we construct a code of blocklength $N = 10^6$, choosing the interleavers at random. In order to estimate the net coding gain of the overall code (and thus the merit of the particular choice of component codewords), we simulate the performance of the code on a ‘constant-error-weight’ channel, that is, a channel that introduces exactly $Np$ errors (randomly among the $N$ positions), varying $p$ (i.e., we perform a 1-D search) until $\hat{p}$ is determined. Of course, in practice the transition is not the idealized step indicated in Fig. 4.1; as a surrogate, we determine the $\tilde{p}$ at which
the frame error rate is $1/2$. Finally, to determine the channel crossover probability $p$ for a target bit-error-rate $p_{\text{cod}}$, we numerically solve (4.1) for $p$ using $\hat{p} = \tilde{p}$ and a desired $N$. The net coding gain then follows by substituting the resulting value of $p$ for $p_{\text{unc}}$ in (2.1).

In order to determine the parameters of the best bi-regular codes, we searched over the space of codes for which $C_1$ is a BCH code of length $n_1 = 2^{m_1} - 1$, $8 \leq m_1 \leq 12$, and similarly for the mother code of $C_2$ (i.e., $C_2$ will be a shortened BCH code, its length shortened to achieve a target rate). The error-correcting parameters $t_1$ and $t_2$ and blocklength $n_2$ are chosen such that the overall rate is $239/255$, but are otherwise unconstrained. For $t \leq 3$, we also consider singly- and doubly-extended codes, i.e., the generator polynomial is multiplied by the factor $(x + 1)$ or $(x + 1)^2$, respectively, which serves to increase the error detection capabilities of the component code. In Table 4.1, we summarize the parameters, the value of $\tilde{p}$, and the estimated net coding gain (for $N = 10^6$, and a target bit-error-rate $10^{-15}$) for the best performing codes.

<table>
<thead>
<tr>
<th>$n_1$</th>
<th>$t_1$</th>
<th>$r_1$</th>
<th>$n_2$</th>
<th>$t_2$</th>
<th>$r_2$</th>
<th>$\tilde{p}$</th>
<th>NCG (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1023</td>
<td>3</td>
<td>32</td>
<td>1018</td>
<td>3</td>
<td>32</td>
<td>$4.66 \times 10^{-3}$</td>
<td>9.29</td>
</tr>
<tr>
<td>1023</td>
<td>4</td>
<td>40</td>
<td>931</td>
<td>2</td>
<td>22</td>
<td>$4.73 \times 10^{-3}$</td>
<td>9.31</td>
</tr>
<tr>
<td>1023</td>
<td>4</td>
<td>40</td>
<td>1481</td>
<td>3</td>
<td>35</td>
<td>$4.77 \times 10^{-3}$</td>
<td>9.32</td>
</tr>
<tr>
<td>1023</td>
<td>4</td>
<td>40</td>
<td>1861</td>
<td>4</td>
<td>44</td>
<td>$4.84 \times 10^{-3}$</td>
<td>9.34</td>
</tr>
<tr>
<td>2047</td>
<td>5</td>
<td>55</td>
<td>1227</td>
<td>4</td>
<td>44</td>
<td>$4.73 \times 10^{-3}$</td>
<td>9.31</td>
</tr>
<tr>
<td>4095</td>
<td>7</td>
<td>84</td>
<td>948</td>
<td>4</td>
<td>40</td>
<td>$4.54 \times 10^{-3}$</td>
<td>9.27</td>
</tr>
</tbody>
</table>

Table 4.1: Parameters of the best bi-regular codes

Note that maximum net coding gain is nearly constant for a diverse set of choices of the component codes $C_1$ and $C_2$. Relative to the best G.975.1 code, an increase in net coding gain of 0.35 dB is possible by optimizing over the choice of the component codes. Furthermore, since the complexity of the BCH decoder is a function of $t$ (and the size of the underlying field), choosing the code with $t_1 = t_2 = 3$ minimizes the maximum complexity of the component BCH decoders. It is interesting that this choice of component codes is essentially the same as in the I.9 code.
Constraint-regular codes

We now consider a different class of codes, so-called constraint-regular codes, which employ a single component code $C_c$ but admit a mixture of variable nodes of degree two and three; see Fig. 4.3. We denote the fraction of variable nodes of degree two by $\Omega_2$, and similarly the fraction of nodes of degree three by $\Omega_3$, where $\Omega_2 + \Omega_3 = 1$. For a $(n_c, n_c - r_c)$ component code, the rate of the overall code satisfies

$$R \geq 1 - \frac{r_c \sum_i \Omega_i \cdot i}{n_c},$$

where again we will assume that equality holds.

![Figure 4.3: Factor graph of a constraint-regular code.]

To determine the parameters of the best constraint-regular codes, we searched over the space of codes for which $C_c$ is a (possibly shortened) BCH code from a mother code of length $n = 2^m - 1$, $8 \leq m \leq 12$; $2 \leq \sum_i \Omega_i \cdot i \leq 3$; and $t_c$ and $n_c$ are such that the overall rate is $239/255$. As before, for $t \leq 3$, we also consider singly- and doubly-extended component codes. In Table 4.2, we summarize the parameters, the value of $\bar{p}$, and the estimated net coding gain (for $N = 10^6$, and a target bit-error-rate $10^{-15}$) for the best performing codes.

Note that the best constraint-regular codes all have $\sum_i \Omega_i \cdot i = 2$; for the sake of interest, we have also provided the performance for codes with $\sum_i \Omega_i \cdot i = 2.5$ and $\sum_i \Omega_i \cdot i = 3$. Note that for $\sum_i \Omega_i \cdot i = 2$, the resulting codes are bi-regular codes for which $C_1 = C_2$ (and $n_1 = n_2$).
In this section, we propose a class of codes, *staircase error-correcting codes*, that can be considered a hybrid of block and convolutional codes, with several similarities to product codes. Intuitively, rather than optimizing the choice of component codewords, we optimize the structure of the graph connecting the component codewords, and obtain performance gains relative to the product-like codes described above. Staircase codes also exploit the streaming nature of communication over OTNs (i.e., data is input to the encoder at a constant rate) to allow the overall code to have an indeterminate blocklength.

### 4.2.1 Incidence matrix representations

By a simple example, we briefly review the notion of an incidence matrix. Consider the graphical model in Fig. 4.4. Our interest is in interpreting such models as representations of error-correcting codes; each circular (variable) node corresponds to a distinct
codeword symbol (from a finite alphabet, typically binary), and the square (constraint) nodes represent local constraints among the connected variable nodes. An assignment of symbols to the variable nodes is a codeword if and only if every constraint node is satisfied. Corresponding to the model in Fig. 4.4 is an incidence matrix,

\[ I = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \end{bmatrix}, \]

where each row corresponds to the constraints imposed by a component codeword, each column corresponds to a variable node, and a non-zero entry indicates the corresponding involvement.

To fully specify the code, it remains to specify the exact nature of the component codeword. For simplicity, we’ll assume that the component codeword is a repetition constraint, i.e., the constraint nodes have an associated parity-check matrix,

\[ h = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}. \]

Finally, the parity-check matrix \( H \) of the overall code is

\[ H = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \end{bmatrix}, \]

which is obtained by replacing the \( i \)th non-zero entry (in the \( k \)th row) of \( I \) with the \( i \)th column of \( h \), while replacing the zero-entries of \( h \) with zero (column) vectors.

**Product codes**

Consider a ‘square’ product code, that is, one with the same (length \( n \)) component codeword associated with both its rows and columns. Its incidence matrix \( I_p \) is then of the form indicated in Fig. 4.5, where only the non-zero elements are indicated. As expected, each row of \( I_p \) has \( n \) non-zero elements. If the positions in the product code are numbered consecutively in a row-by-row manner and associated to the corresponding columns of \( I_p \), then the first \( n \) rows represent the row-codeword constraints, and the remaining \( n \) rows represent the column-codeword constraints.
Staircase codes

Although most naturally described by their matrix encoding rule (see Section 4.2.2), staircase codes were inspired by—and are a special case of—the recurrent codes studied by Wyner and Ash [47], whose description is essentially in the form of an incidence matrix representation\(^1\). Therefore, we prefer to first describe staircase codes\(^2\) in their ‘original’ form. For the same component codeword as the product code (and assuming \(n\) is even), the (semi-infinite) incidence matrix \(I_s\) of a staircase code is of the form indicated in Fig. 4.6. Note that the first \(n/2\) rows have only \(n/2\) non-zero elements, whereas all other rows have \(n\) non-zero elements, i.e., the first \(n/2\) rows are treated as shortened codewords, which provides the corresponding bits with additional protection; this effect, and the fact that it ‘propagates’ in an iterative decoder, provides the superior coding gain relative to product codes.

4.2.2 Matrix encoding rule

Staircase codes are completely characterized by the relationship between successive matrices of symbols. Specifically, consider the (infinite) sequence \(B_0, B_1, B_2, \ldots\) of \(m\)-by-\(m\) matrices \(B_i, i \in \mathbb{Z}^+\). Block \(B_0\) is initialized to a reference state known to the encoder-

---

\(^1\)We note that the codes (and decoding algorithms) explicitly studied by Wyner and Ash were relatively weak; their main goal was to provide a mathematical framework for convolutional codes, in order to derive distance-related bounds.

\(^2\)After completing this work, we were made aware of the block-wise braided block codes [48] (see also braided convolutional codes [49]), which share several similarities to staircase codes; we will return to this point in Section 4.5.
Figure 4.6: The semi-infinite incidence matrix $I_s$ for a staircase code; only non-zero elements are indicated.

dercoder pair, e.g., block $B_0$ could be initialized to the all-zeros state, i.e., an $m$-by-$m$ array of zero symbols.

We select a conventional error-correcting code (e.g., Hamming, BCH, Reed-Solomon, etc.) in systematic form to serve as the \textit{component} code; this code, which we henceforth refer to as $C$, is selected to have blocklength $n = 2m$ symbols, $r$ of which are parity symbols. In order that the rate of the staircase code is positive, we require $r < m$.

Encoding proceeds recursively on the $B_i$. For each $i$, $m(m - r)$ information symbols (from the streaming source) are arranged into the $m-r$ leftmost columns of $B_i$; we denote this sub-matrix by $B_{i,L}$. Then, the entries of the rightmost $r$ columns (this sub-matrix is denoted by $B_{i,R}$) are specified as follows:

1. Form the $m \times (2m-r)$ matrix, $A = [B_{i-1}^T B_{i,L}]$, where $B_{i-1}^T$ is the matrix-transpose of $B_{i-1}$.

2. The entries of $B_{i,R}$ are then computed such that each of the rows of the matrix $[B_{i-1}^T B_{i,L} B_{i,R}]$ is a valid codeword of $C$. That is, the elements in the $j$th row of $B_{i,R}$ are exactly the $r$ parity symbols that result from encoding the $2m - r$ ‘information’ symbols in the $j$th row of $A$.

Generally, successive blocks in a staircase code satisfy the following relation: for any $i \geq 1$, each of the rows of the matrix $[B_{i-1}^T B_i]$ is a valid codeword in $C$. An equivalent description—from which the term ‘staircase codes’ originates—is suggested by Fig. 4.7, in which (the concatenation of the symbols in) every row (and every column) in the ‘staircase’ is a valid codeword of $C$. 
Chapter 4. Error-correcting codes for high-speed communications

4.2.3 Properties

The rate of a staircase code is

\[ R_s = 1 - \frac{r}{m}, \]

since encoding produces \( r \) parity symbols for each set of \( m - r \) ‘new’ information symbols. However, note that the related product code has rate

\[ R_p = \left( \frac{2m - r}{2m} \right)^2 \]
\[ = 1 - \frac{r}{m} + \frac{r^2}{4m^2}, \]

which is greater than the rate of the staircase code. However, for sufficiently high rates, the difference is small, and staircase codes outperform product codes of the same rate.

From the context of transmitter latency—which includes encoding latency and frame-mapping latency—staircase codes have the advantage (relative to product codes) that the effective rate (i.e., the ratio of ‘new’ information symbols, \( m - r \), to the total number of ‘new’ symbols, \( m \)) of a component codeword is exactly the rate of the overall code. Therefore, the encoder produces parity at a ‘regular’ rate, which enables the design of a frame-mapper that minimizes the transmitter latency.

We note that staircase codes can be interpreted as generalized LDPC codes with a systematic encoder and an indeterminate blocklength, which admits decoding algorithms with a range of latencies.

Finally, using arguments analogous to those used for product codes, a \( t \)-error-correcting component code \( C \) with minimum distance \( d_{\text{min}} \) has a Hamming distance between any
two staircase codewords that is at least $d_{\text{min}}^2$.

4.2.4 Decoding algorithm

Staircase codes are naturally unterminated (i.e., their blocklength is indeterminate), and thus admit a range of decoding strategies with varying latencies. That is, decoding can be accomplished in a sliding-window fashion, in which the decoder operates on the received bits corresponding to $L$ consecutively received blocks $B_i, B_{i+1}, \ldots, B_{i+L-1}$. For a fixed $i$, the decoder iteratively decodes as follows: First, those component codewords that terminate in block $B_{i+L-1}$ are decoded (see Fig. 4.8); since every symbol is involved in two component codewords, the corresponding syndrome updates are performed, as in Section 2.2.3. Next, those codewords that terminate in block $B_{i+L-2}$ are decoded. This process continues until those codewords that terminate in block $B_{i}$ are decoded. Now, since decoding those codewords terminating in some block $B_j$ affects those codewords that terminate in block $B_{j+1}$, it is beneficial to return to $B_{i+L-1}$ and to repeat the process. This iterative process continues until some maximum number of iterations is performed, at which time the decoder outputs its estimate for the contents of $B_i$, accepts in a new block $B_{i+L}$, and the entire process repeats (i.e., the decoding window slides one block to the ‘right’).

![Figure 4.8: The blocks within an $L = 4$ sliding-window decoder; the black strip represents a codeword that ‘terminates’ in block $B_{i+3}$](image)

Since symbols in block $B_{i+L-1}$ are (with respect to the current decoding window) only protected by one component codeword, we employ a slightly refined decoding algorithm:
For a fixed $i$, if the decoding of a component codeword terminated in $B_{i+L-1}$ suggests flipping bits associated with an already satisfied syndrome (i.e., an all-zero syndrome) in $B_{i+L-2}$, then the decoding is rejected. This serves as a form of built-in error-detection mechanism, which reduces the frequency of incorrect (component) decodings, and provides a corresponding increase in NCG. In general, it can be valuable to extend this error-detection rule to blocks $B_{i+j}, \ldots, B_{i+L-1}$, where $j \geq 1$ is a parameter to be optimized.

4.2.5 Multi-edge-type interpretation

Staircase codes have a simple graphical representation, which provides a multi-edge-type interpretation of their construction. The term ‘multi-edge-type’ was originally applied to describe a refined class of irregular LDPC codes, in which variable nodes (and check nodes) are classified by their degrees with respect to a set of edge types. Intuitively, the introduction of multiple edge types allows degree-one variable nodes, punctured variable nodes, and other beneficial features that are not admitted by the conventional irregular ensemble. In turn, better performance for finite blocklengths and fixed decoding complexities is possible.

In Fig. 4.9, we present the factor graph representation of a decoder that operates on a window of $L = 4$ blocks; the graph for general $L$ follows in an obvious way. Dotted variable nodes indicate symbols whose value was decoded in the previous stage of decoding. The key observation is that when these symbols are correctly decoded—which is essentially always the case, since the output BER is required to be less than $10^{-15}$—the component codewords in which they are involved are effectively shortened by $m$ symbols. Therefore, the most reliable messages are passed over those edges connecting variable nodes to the shortened (component) codewords, as indicated in Fig. 4.9. On the other hand, the rightmost collection of variable nodes are (with respect to the current decoding window) only involved in a single component codeword, and thus the edges to which they are connected carry the least reliable messages. Due to the nature of iterative decoding, the intermediate edges carry messages whose reliability lies between these two extremes.
4.3 A G.709-compatible staircase code

The ITU-T Recommendation G.709 defines the framing structure and error-correcting coding rate for OTNs. For our purposes, it suffices to know that an optical frame consists of 130560 bits, 122368 of which are information bits, and the remaining 8192 are parity bits, which corresponds to error-correcting codes of rate $R = \frac{239}{255}$. Since $\frac{510-32}{510} = \frac{239}{255}$, we will consider a component code with $m = 510$ and $r = 32$. Specifically, the binary $(n = 1023, k = 993, t = 3)$ BCH code with generator polynomial $(x^{10} + x^3 + 1)(x^{10} + x^3 + x^2 + x + 1)(x^{10} + x^8 + x^3 + x^2 + 1)$ is adapted to provide an additional 2-bit error-detecting mechanism, resulting in the generator polynomial\(^3\)

$$g(x) = (x^{10} + x^3 + 1)(x^{10} + x^3 + x^2 + x + 1)(x^{10} + x^8 + x^3 + x^2 + 1)(x^2 + 1).$$

In order to provide a simple mapping to the G.709 frame, we first note that $2 \cdot 130560 = 510 \cdot 512$. This leads us to define a slight generalization of staircase codes, in which the blocks $B_i$ consist of 512 rows of 510 bits. The encoding rule is modified as follows:

1. Form the $512 \times (512 + 510)$ matrix, $A = \begin{bmatrix} \hat{B}^T_{i-1} & B_{i,L} \end{bmatrix}$, where $\hat{B}^T_{i-1}$ is obtained by appending two all-zero rows to the top of the matrix-transpose of $B_{i-1}$.

2. The entries of $B_{i,R}$ are then computed such that each of the rows of the matrix $\begin{bmatrix} \hat{B}^T_{i-1} & B_{i,L} & B_{i,R} \end{bmatrix}$ is a valid codeword of $C$. That is, the elements in the $j$th row of

\(^3\)This is exactly the code applied to the rows (but not the slopes) of the I.9 code in G.975.1.
$B_{i,R}$ are exactly the 32 parity symbols that result from encoding the 990 ‘information’ symbols in the $j$th row of $A$.

Here, $C$ is the code obtained by shortening the code generated by $g(x)$ by one bit, since our overall codeword length is $510 + 512 = 1022$.

### 4.3.1 Simulation results

Simulation results of the G.709-compatible staircase code (with $L = 6$) were obtained via Altera Stratix IV field-programmable gate arrays, in collaboration with an industrial partner. In Fig. 4.10, we compare the performance of our code to that of the best-performing G.975.1 codes, as well as the $(255, 239)$ Reed-Solomon code, i.e., the G.975 code. Extrapolating the performance curve to $10^{-15}$, the G.709-compatible staircase code provides a 9.45 dB NCG, which is 0.52 dB from the Shannon Limit, and 0.46 dB better than the best G.975.1 code. It is remarkable that performance so close to the Shannon Limit is possible without ‘soft’ messages in the decoder.

### 4.3.2 Error floor analysis

For iteratively decoded codes, an error floor (in the output bit-error-rate) can often be attributed to error patterns that ‘confuse’ the decoder, even though such error patterns could easily be corrected by a maximum-likelihood decoder. In the context of LDPC codes, these error patterns are often referred to as trapping sets [51]. In the case of product-like codes with an iterative hard-decision decoding algorithm, we will refer to them as stall patterns, due to the fact that the decoder gets locked in a state in which no updates are performed, i.e., the decoder stalls, as in Fig. 4.11.

**Definition 1.** A stall pattern is a set $s$ of codeword positions, for which every row and column involving positions in $s$ has at least $t + 1$ positions in $s$.

We note that this definition includes stall patterns that are correctable, since an incorrect decoding may fortuitously cause one or more bits in $s$ to be corrected, which could then lead to all bits in $s$ eventually being corrected. In this section, we obtain an estimate for the error floor by overbounding the probabilities of these events, and

---

4It should be noted that this improvement can be partially attributed to increasing the blocklength. As a rule of thumb, in our regime of interest, doubling the blocklength increases the NCG by approximately 0.05 dB.
Chapter 4. Error-correcting codes for high-speed communications

Figure 4.10: Performance of G.975 code, G.975.1 codes, and a G.709-compliant staircase code

pessimistically assuming that every stall pattern is uncorrectable (i.e., if any stall pattern appears during the course of decoding, it will appear in the final output). The methods presented for the error floor analysis apply to a general staircase code, but for simplicity of the presentation, we will focus on a staircase code with $m = 510$ and doubly-extended triple-error-correcting component codes.

Figure 4.11: A stall pattern for a staircase code with a triple-error correcting component code. Since every involved component codeword has 4 errors, decoding stalls.
A union bound technique

Due to the streaming nature of staircase codes, it is necessary to account for stall patterns that span (possibly multiple) consecutive blocks. In order to determine the bit-error-rate due to stall patterns, we consider a fixed block $B_i$, and the set of stall patterns that include positions in $B_i$. Specifically, we ‘assign’ to $B_i$ those stall patterns that include symbols in $B_i$ (and possibly additional positions in $B_{i+1}$) but no symbols in $B_{i-1}$. Let $S_i$ represent the set of stall patterns assigned to $B_i$. By the union bound, we then have

$$\text{BER}_{\text{floor}} \leq \sum_{s \in S_i} \Pr[\text{bits in } s \text{ in error}] \cdot \frac{|s|}{510^2}.$$ 

Therefore, bounding the error floor amounts to enumerating the set $S_i$, and evaluating the probabilities of its elements being in error.

Bounding the contribution due to minimal stalls

**Definition 2.** A minimal stall pattern has the property that there are only $t + 1$ rows with positions in $s$, and only $t + 1$ columns with positions in $s$.

The minimal stall patterns of a staircase code can be counted in a straightforward manner; the multiplicity of minimal stall patterns that are assigned to $B_i$ is

$$M_{\text{min}} = \binom{510}{4} \cdot \sum_{m=1}^{4} \binom{510}{m} \cdot \binom{510}{4 - m} = \binom{510}{4} \left[ \binom{2 \cdot 510}{4} - \binom{510}{4} \right],$$

and we refer to the set of minimal stall patterns by $S_{\text{min}}$. The probability that the positions in some minimal stall pattern $s$ are received in error is $p^{16}$.

Next, we consider the case in which not all positions in some minimal stall pattern $s$ are received in error, but that due to incorrect decoding(s), all positions in $s$ are—at some point during decoding—simultaneously in error. For some fixed $s$ and $l$, $1 \leq l \leq 16$, there are

$$\binom{16}{l}$$

ways in which $16 - l$ positions in $s$ can be received in error. For the moment, let’s assume that erroneous bit flips occur independently with some probability $\zeta$, and that $\zeta$ does
not depend on \( l \). Then we can \textit{overbound} the probability that a \textit{particular} minimal stall \( s \) occurs by

\[
\sum_{l=0}^{16} \binom{16}{l} p^{16-l} \zeta^l = (p + \zeta)^{16}.
\]

In order to provide evidence in favour of these assumptions, Table 4.3 presents empirical estimates, for \( l = 0, l = 1 \) and \( l = 2 \), of the probability that a minimal stall pattern \( s \) occurs during iterative decoding, given that \( 16 - l \) positions in \( s \) are (intentionally) received in error. The estimates were obtained by intentionally introducing errors in \( 16 - l \) positions of a given stall \( s \), randomly introducing errors (with \( p = 4.8 \times 10^{-3} \)) in the positions ‘outside’ of \( s \), and decoding the resulting observations. Furthermore, an additional such ‘partial’ stall pattern is included every tenth staircase block, in order that the state of the decoder (when the partial stall enters the decoder) be consistent with typical operating conditions.

Note that even if a minimal stall is received, there exists a non-zero probability that it will be corrected as a result of erroneous decodings; we will ignore this effect in our estimation, i.e., we make the worst-case assumption that any minimal stall persists. Furthermore, from the results for \( l = 1 \) and \( l = 2 \), it appears that our stated assumptions regarding \( \zeta \) hold true, and \( \zeta \approx 5.8 \times 10^{-4} \). For \( l > 2 \), we did not have access to sufficient computational resources for estimating the corresponding probabilities. Nevertheless, based on the evidence presented in Table 4.3, the error floor contribution due to minimal stall patterns is estimated as

\[
\frac{16}{510^2} \cdot M_{\text{min}} \cdot (p + \zeta)^{16},
\]

where \( \zeta = 5.8 \times 10^{-4} \) when \( p = 4.8 \times 10^{-3} \).

<table>
<thead>
<tr>
<th>( l )</th>
<th>Estimated probability of minimal stall ( s )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>( 149/150 )</td>
</tr>
<tr>
<td>1</td>
<td>( 1/1725 )</td>
</tr>
<tr>
<td>2</td>
<td>( (1/1772)^2 )</td>
</tr>
</tbody>
</table>

Table 4.3: Estimated probabilities of a stall \( s \), assuming that \( 16 - l \) given positions are received in error
Bounding the contribution due to non-minimal stalls

We now wish to account for the error floor contribution of non-minimal stalls, e.g., the stall pattern illustrated in Fig. 4.12. In the general case, a stall pattern \( s \) includes codeword positions in \( K \) rows and \( L \) columns, \( K \geq 4, L \geq 4 \); we refer to these as \((K, L)\)-stalls. Furthermore, each \((K, L)\)-stall includes \( l \) positions, \( 4 \cdot \max(K, L) \leq l \leq K \cdot L \), where the lower bound follows from the fact that every row and column (in the stall) includes at least 4 positions. Note that there are

\[
A_{K,L} = \binom{510}{L} \cdot \sum_{m=1}^{K} \binom{510}{m} \cdot \binom{510}{K-m}
\]

\[
= \binom{510}{L} \left( 2 \cdot \binom{510}{K} - \binom{510}{K} \right)
\]

ways to select the involved rows and columns.

Figure 4.12: A non-minimal stall pattern for a staircase code with a triple-error correcting component code.

For a fixed \((K, L) \neq (4, 4)\) and a fixed choice of rows and columns, we now proceed to overbound the contributions of candidate stall patterns. Without loss of generality, we assume that \( K \geq L \), and note that there are

\[
\binom{L}{4}^K
\]

ways of choosing \( l = 4K \) elements (in the \( L \cdot K \) ‘grid’ induced by the choice of rows and columns) such that each column includes exactly four elements, and that every stall pattern ‘contains’ at least one of these. Now, since a stall pattern includes \( l \) elements, \( 4 \cdot K \leq l \leq K \cdot L \), the number of stall patterns with \( l \) elements is overbounded as

\[
\binom{L}{4}^K \cdot \binom{K \cdot L - 4 \cdot K}{l - 4 \cdot K}.
\]
For a general \((K, L) \neq (4, 4)\), it follows that the number of stall patterns with \(l\) elements, \(4 \cdot \max(K, L) \leq l \leq K \cdot L\), is overbounded as

\[
\left( \frac{\min(K, L)}{4} \right)^{\max(K, L)} \cdot \left( \frac{K \cdot L - 4 \cdot \max(K, L)}{l - 4 \cdot \max(K, L)} \right).
\]

Finally, over the choice of the \(K\) rows and \(L\) columns, there are

\[
M_{K,L}^l = A_{K,L} \cdot \left( \frac{\min(K, L)}{4} \right)^{\max(K, L)} \cdot \left( \frac{K \cdot L - 4 \cdot \max(K, L)}{l - 4 \cdot \max(K, L)} \right)
\]

\((K, L)\)-stalls with \(l\) elements.

For a fixed \(K\) and \(L\), the contribution to the error floor can be estimated as

\[
\sum_{l=4 \cdot \max(K, L)}^{K \cdot L} \frac{l}{510^2} \cdot M_{K,L}^l \cdot (p + \zeta)^l,
\]

and in Table 4.4, we provide values for various \(K\) and \(L\), when \(\zeta = 5.8 \times 10^{-4}\) and \(p = 4.8 \times 10^{-3}\).

<table>
<thead>
<tr>
<th>(K)</th>
<th>(L)</th>
<th>Contribution to error floor estimate</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>4</td>
<td>(3.55 \times 10^{-21})</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>(7.81 \times 10^{-28})</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>(2.54 \times 10^{-22})</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>(2.21 \times 10^{-28})</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>(1.40 \times 10^{-23})</td>
</tr>
<tr>
<td>6</td>
<td>7</td>
<td>(1.49 \times 10^{-29})</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>(8.53 \times 10^{-25})</td>
</tr>
<tr>
<td>7</td>
<td>8</td>
<td>(1.83 \times 10^{-32})</td>
</tr>
</tbody>
</table>

Table 4.4: Error floor contributions of \((K, L)\)-stall patterns

Note that the dominant contribution to the error floor is due to minimal stall patterns (i.e., \(K = L = 4\)), and that the overall estimate for the error floor of the code is \(3.8 \times 10^{-21}\). Finally, we note that by a similar (but more cumbersome) analysis, the error floor of the G.709-compliant staircase code is estimated to occur at \(4.0 \times 10^{-21}\).
4.4 Generalized staircase codes

We present two generalizations of staircase codes. For simplicity of the presentation, we consider staircase codes with square blocks, but generalizations to non-square blocks (as in Section 4.3) follow in a straightforward fashion. Similarly, although we describe them separately, the generalizations presented in this section can be combined in an obvious way.

4.4.1 Mixtures of component codes

In Section 4.2, we constrained each of the rows of the matrix $[B_{i-1}^TB_i]$ to be a valid codeword in a fixed component code $C$. More generally, the parity positions in $B_i$ can be chosen such that the $i$th row of the matrix is a codeword of length $2m$ in some component code $C_i$ (i.e., the component code may vary among the rows of the same staircase block), resulting in an overall code of rate

$$R = 1 - \sum_{i=1}^{m} \frac{r_i}{m^2},$$

where $r_i$ is the number of parity positions in the code $C_i$.

4.4.2 Multi-block staircase codes

For $m = c \cdot l$, $c, l \in \mathbb{Z}^+$, the rows of $B_i$ can be divided into $c$ blocks of $l$ consecutive rows, i.e.,

$$B_i = [B_{i,1}; B_{i,2}; \cdots; B_{i,c}],$$

where $B_{i,j}, 1 \leq j \leq c$ is the $l$-by-$m$ matrix consisting of rows $1 + (j-1) \cdot c$ to $j \cdot c$ of $B_i$. We can define a multi-block encoding procedure as follows. First, form the matrix

$$D_{i-1} = [B_{i-1,1}; B_{i-2,2}; \cdots; B_{i-c,c}],$$

then select the parity positions in $B_i$ such that the rows of the matrix $[D_{i-1}^TB_i]$ are codewords in some component code $C$. Note that this generalization does not alter the rate of the code, but it increases the ‘distance’ between bits involved in a component codeword.
4.5 A review of related works

We now review earlier proposals from the literature that are related to the staircase construction, in order to emphasize the novelty of the constructions proposed in this work.

First, a general framework for codes with a banded parity-check matrix is provided by the recurrent codes of Wyner and Ash [47], which were in turn inspired by convolutional codes, which themselves have a banded parity-check matrix. For $B_0$ a semi-infinite matrix with $b$ columns, the recurrent codes are the class of codes with parity-check matrix

$$A = [B_0, B_m, B_{2m}, B_{3m}, \ldots],$$

where $m$ is a positive integer, and $B_i$ is the semi-infinite matrix obtained by shifting the rows of $B_0$ down by $i$ rows (where all-zero rows are shifted into the top $i$ rows). In [47], the authors prove several results on the burst and random error-correcting capabilities of recurrent codes, but focus on ‘classical’ bounded-distance style decoding techniques, rather than the more powerful iterative decoding algorithms we apply to staircase codes. Therefore, even though staircase codes have a representation as recurrent codes (via Fig. 4.6), their analysis and performance is more closely related to iteratively-decodeable sparse-graph-based error-correcting codes.

A second class of codes that is related to staircase codes is the LDPC convolutional codes of Feltström and Zigangirov [52]. The LDPC convolutional codes were the first iteratively-decodeable sparse-graph-based error-correcting code with a banded parity-check matrix, i.e., in the style of a recurrent code. Contrary to traditional LDPC codes, the authors argue that the construction admits a sliding-window style decoder that is more amenable to a pipelined implementation, while maintaining the excellent performance of LDPC codes. Of course, this performance comes at the cost of requiring a soft-decision message passing decoder, in contrast to the more efficient hard-decision decoder of staircase codes.

In the context of product-like codes, Justesen [53, 24] proposes various modifications of product codes, including so-called triangular product codes and block product codes. In both cases, the fact that any row and column codewords intersect in a single position is sacrificed in order to provide a desired (shorter) overall blocklength (e.g., a single optical frame), but nevertheless the constructions retain several structural similarities to product codes. In general, the flexibility afforded by these constructions results in a performance
Finally, after completing this work, we were made aware of block-wise braided block codes (BBBCs) [48], which share several similarities to staircase codes. In particular, BBBCs replace the weak component code (the single-parity check code) of LDPC convolution codes with a more powerful algebraically decodeable code, resulting in a construction more closely related to staircase codes. Nevertheless, they have an important structural distinction, which we describe in the remainder of this section.

In Fig. 4.13 (a reproduction of Fig. 5 from [48]), the array representation of BBBCs is given. In general, the $P^{(i)}_j$ are $N$-by-$N$ sparse matrices, referred to as multiple block permutors (MBPs). For $j > 0$, the $P^{(0)}_j$ have $k - r$ ones in every row and column, the $P^{(1)}_j$ have $r$ ones in every row and column, and similarly the $P^{(2)}_j$ have $r$ ones in every row and column; in each case, all other elements are zero. Note that every row and column of the corresponding array representation (illustrated in Fig. 4.13) has $n = k + r$ non-zero elements; each non-zero element of the $P^{(i)}_j$ corresponds to a distinct bit in the overall codeword. Furthermore, it is necessary that $N \geq \max\{r, k - r\}$.

![Figure 4.13: Array representation of a BBBC.](image)

For $j = t$, encoding proceeds by first assigning $N \cdot (k - r)$ information bits to the non-zero positions in $P^{(0)}_t$. Next, for each of the $N$ rows in $P^{(0)}_t$, $r$ bits (corresponding to the non-zero elements in the same row) from $[P^{(2)}_{t-1}]^T$ are concatenated with the $k - r$ bits from $P^{(0)}_t$, and the resulting $k$ bits are encoded by a linear $(n, k)$ code, with the $r$ computed parity bits stored in the non-zero positions in the same row of $P^{(1)}_t$. An analogous procedure is performed for each of the $N$ columns of $P^{(0)}_t$, and the computed parity bits are stored in the columns of $[P^{(2)}_t]^T$. 
In [48], the authors focus on Hamming component codes, due to the fact that they have a relatively efficient a posteriori probability (APP) decoder (i.e., they consider ‘soft’ message-passing decoding). The role of the MBPs is two-fold: since they consider component codes with (relatively short) lengths 15 and 31, increasing \( N \) increases the diameter of the code’s graph, in a similar fashion to the generalization of multi-block staircase codes; the ‘randomness’ introduced by the MBPs beneficially impacts the distance distribution of the overall code, in a similar sense to LDPC codes (in fact, BBBCs are a type of generalized LDPC code). Of course, the relative merits of the MBPs is (at least in part) a function of the fact that they focus on weak and short component codes. We note that the MBPs cause the code to more closely resemble an LDPC-like code, preventing an analytical error floor analysis.

The key distinction between staircase codes and BBBCs lies in the fraction of bits (in a given component codeword) that are contributed from previously encoded component codewords.

First, we note that for the same choice of component codewords, the rate of the overall code is the same in both cases. Next, we recall that the encoding algorithm for staircase codes (with a \((n, k) = (2^m, 2^m - r)\) component code) accepts \( m - r \) ‘new’ information bits, combines them with \( m \) bits from the previously encoded block, and computes \( r \) parity bits. On the other hand, the encoding algorithm for BBBCs accepts \( 2^m - r \) ‘new’ information bits, combines them with \( r \) bits from the previously encoded ‘block’, and computes \( r \) parity bits. The distinction between the two classes of codes arises from this different ‘chaining’ of coded bits. In particular, while both classes of codes are decoded in a sliding-window fashion, the knowledge of the ‘past’ is significantly more useful in the staircase code, since the component codewords are effectively shortened by \( m \) bits rather than \( r \) bits, and \( m \) is much greater than \( r \) for high-rate codes. In effect, the impact of shortening for BBBCs is almost insignificant (with respect to the threshold of the overall code), and the resulting performance of BBBCs is essentially the same as a generalized LDPC code of the same length (with the same component codes).

### 4.6 Conclusions

In this chapter we proposed a method for rapidly estimating the net coding gain of a product-like code. For the bi-regular and constraint-regular classes of product-like codes, we then used this method to determine the choice of component codewords that
maximizes the net coding gain. We identified various bi-regular codes that improve upon
the net coding gain of the best G.975.1 code by approximately 0.35 dB.

In order to further improve upon the net coding gain, we introduced staircase codes,
which are a hybrid of convolutional and block coding techniques, and are closely con-
ected to product codes. We presented a multi-edge-type representation of staircase
codes, which emphasizes the ‘irregular’ level of protection afforded to bits within a given
decoding window, which is responsible for the performance improvements relative to prod-
uct codes. We proposed a G.709-compliant staircase code that provides a 9.45 dB net
coding gain at an output error rate of $10^{-15}$, which is 0.46 dB better than the best G.975.1
code, and within 0.52 dB of the Shannon Limit. An error floor estimation technique for
staircase codes was presented, and the error floor of the G.709-compliant staircase code
was estimated to occur at $4 \times 10^{-21}$. Finally, we presented two generalizations of the
staircase code construction.
Chapter 5

Staircase codes in high-spectral-efficiency fibre-optic systems

As illustrated by the capacity estimations in Chapter 3, higher-order modulation formats are required to approach the capacity of fibre-optic communication systems. In this chapter, we use staircase codes to provide error correction in high-spectral-efficiency systems. In Section 5.1 we review coded modulation techniques that make use of binary error-correcting codes. In Section 5.2 we describe a pragmatic error-correcting system for coded modulation in fibre-optic communication systems with hard-decision decoding, and calculate the maximum achievable spectral efficiencies for this class of system. In Section 5.3 we present simulation results for fibre-optic systems with QAM constellations, operating at those input powers which maximize the capacities calculated in Chapter 3. From these results, we calculate the achievable rates for our proposed system, design corresponding staircase codes, and demonstrate the resulting performance of the system.

5.1 Coded modulation via binary codes

For high-spectral-efficiency communications, the set of channel input symbols must be sufficiently large, and coding is required on the resulting non-binary input alphabet. At first glance, this would seem to require the design of error-correction codes over non-binary alphabets, with a decoding algorithm that accounts for the distance metric implied by the underlying channel. Indeed, this ‘direct’ approach provides motivation for trellis-
Chapter 5. Staircase codes in high-spectral-efficiency fibre-optic systems

coded modulation \cite{54}, in which the code is designed to optimize the minimum Euclidean distance between transmitted sequences. Alternatively, by considering the set of channels induced by the bit-labels of the constellation points, coded modulation via binary codes can be applied with—in principle—no loss of optimality. To see this, consider a $2^M$-point constellation $A$, for which each symbol is labeled with a unique binary $M$-tuple $(b_1, b_2, \ldots, b_M)$. For a channel with input $X \in A$, and an output $Y$, the capacity of the resulting channel is $I(X; Y)$ (maximized over the input distribution $p(x)$), which can be expanded by the chain rule of mutual information:

$$I(X; Y) = I(b_1, b_2, \ldots, b_M; Y) = I(b_1; Y) + I(b_2; Y|b_1) + \cdots + I(b_M; Y|b_1, b_2, \ldots, b_{M-1}) \quad (5.1)$$

Note that each term (i.e., the sub-channels) in the expansion defines a binary-input channel, for which binary error-correction codes can be applied; this approach is referred to as multi-level coding (MLC) \cite{55}. Furthermore, if a capacity-approaching code is applied to each sub-channel, then the capacity of the modulation scheme is achieved, that is, there is no loss in optimality in applying binary coding to each sub-channel. However, from (5.1), it is implied that decoding is performed in stages, since decoded bits from lower-indexed levels provide side information for decoding higher levels; the resulting decoding architecture is referred to as a multi-stage decoder.

Note that the multi-stage architecture introduces decoding latency to the higher levels, requires memory to store channel outputs prior to decoding (since outputs are ‘held’ until decoded bits from the lower levels are available), and requires an individual code for each sub-channel. Clearly, the latency and memory issues can be eliminated simply by ignoring the conditioning in (5.1), and the resulting system has capacity

$$C_{\text{PID}} = I(b_1; Y) + I(b_2; Y) + \cdots + I(b_M; Y),$$

where PID stands for ‘parallel independent decoding’. However, even when capacity-achieving codes (i.e., with rates $I(b_i; Y)$) are applied to each sub-channel, $C_{\text{PID}}$ may be significantly less than $I(X; Y)$. Note that the capacities of the individual bit-channels depend on the constellation labeling; for MLC their overall sum is fixed, regardless of the labeling, but for PID their sum (i.e., $C_{\text{PID}}$) depends on the labeling. In fact, for Gray-labeling, the difference between $C_{\text{PID}}$ and $I(X; Y)$ essentially vanishes, as shown in \cite{56}. Furthermore, even though the capacities of the individual sub-channels are not identical, a single binary error-correcting code (whose rate is less than the capacity of
the binary-input channel whose conditional distribution is the average of the sub-channel conditional distributions) provides near-capacity performance, which addresses the third issue with MLC; this approach is referred to as bit-interleaved coded modulation (BICM).

5.2 A pragmatic error-correcting system

Many existing proposals (in the academic literature) for coded modulation in fibre-optic communication systems amount to using techniques currently used in electrical wire-line and wireless communication systems. For example, in [57], the authors propose a concatenated coding system with inner trellis-coded modulation (for an 8-PSK constellation) and an outer I.9 code, and in [58, 59, 60], the authors propose using LDPC codes for coded modulation. In both cases, the proposals are verified by simulation, where the channel is assumed to be AWGN, but no regard is given to the real-world implementation challenges for the proposed systems.

Other proposals for coded modulation in fibre-optic systems consider simplified channel models, and design codes for the resulting systems. For example, in [61], the authors design a trellis-coded polarization-shift-keying modulation system, but their channel model only considers laser phase noise, i.e., effects related to the propagation over fibre are completely ignored. In [62], the authors consider a nonlinear phase noise (NLPN) channel model studied by [63], and design a multi-level coded modulation system with Reed-Solomon codes at each level. However, this channel model assumes a single-channel dispersion-less system, which is not of practical interest.

In this section, we take a pragmatic approach to coded modulation for fibre-optic systems, that addresses the deficiencies of the aforementioned proposals. Due to the fact that product-like codes with syndrome-based decoding have efficient high-speed decoders, we consider systems with hard-decision decoding. Furthermore, the channel model for which the codes are designed is derived from GNLS-based simulations of the fibre-optic systems, and thus accurately models the non-AWGN channel that occurs in optical communication systems.

We note that for product-like codes (including staircase codes), the gap to capacity (of a BSC) increases as the rate of the code decreases (this remains true even when soft-decision decoding is performed, see [64]). For MLC with set-partitioning labeling, the least reliable sub-channel has rate less than $1/2$ for operating points of interest, while the most reliable sub-channels have almost unit rate. Since a Gray-labeled constellation
tends to induce sub-channels with closer to uniform rates (which maximizes the rate of the worst sub-channel), and admits a lower complexity decoding architecture, we focus on systems with Gray-labeled (or pseudo-Gray-labeled) constellations. Specifically, for QAM constellations we use a Gray labeling, and for CrossQAM constellations we use a pseudo-Gray labeling, since a Gray labeling does not exist [65].

In Fig. 5.1, we illustrate a pseudo-gray labeling for a 32-CrossQAM constellation, as well as the general construction of a $2^{M+1}$-CrossQAM labeling from a $2^M$-QAM Gray labeling. The alphabetic symbols in Fig. 5.1 represent matrices of $M$-bit labels, where $\hat{Z}$ is the matrix obtained by reversing the order of the columns of $Z$, $\bar{Z}$ is the matrix obtained by reversing the order of the rows of $Z$, and $0Z$ is the matrix obtained by prepending a 0 to each $M$-bit label in $Z$.

Figure 5.1: Pseudo-Gray labeling of a 32-CrossQAM constellation, and a method for obtaining a $2^{M+1}$-CrossQAM labeling from a Gray-labeled $2^M$-QAM constellation.

We consider systems with both PID and BICM; in both cases, since hard-decision decoding is assumed, channel outputs are quantized without soft information. For BICM, the error rate of the effective binary channel is the mean of the error rates of the sub-channels. In order to determine the maximum achievable rates for this class of system, we calculate the mutual informations of the corresponding sub-channels, in the spirit of [56]. In Fig. 5.2, the sub-channel capacities for 256-QAM (with hard-decision inputs to the decoder) are plotted; note that a Gray-labeling was obtained by the direct product of a 16-PAM Gray-labeling with itself, thus there are only four distinct sub-channel rates. Using PID, an overall spectral efficiency of 7 bits/symbol is achieved at an SNR of 24.2
dB, and the sub-channels have capacities 0.76, 0.86, 0.92 and 0.96 bits/channel-use.

To determine the loss inherent in assuming hard-decision inputs to the decoder, Figs. 5.3 and 5.4 show the achievable rates for 16-QAM and 64-QAM, respectively, for each of the following cases: constellation capacity with soft quantization, constellation capacity with hard quantization, capacity for PID with hard quantization, and the capacity for BICM with hard quantization. For 16-QAM and a spectral efficiency of 3 bits/symbol, there is a 1.27 dB loss for assuming hard-decisions (relative to the capacity of 16-QAM on an AWGN channel), an additional 0.25 dB loss from using Gray-labeled PID rather than MLC, and an additional 0.05 dB loss for BICM. Similarly, for 64-QAM with a spectral efficiency of 5 bits/symbol, there is a 1.2 dB loss for assuming hard-decisions, an additional 0.33 dB loss from using Gray-labeled PID rather than MLC, and an additional 0.1 dB loss for BICM. In general, for a QAM-like constellation with $2^M$ points and a spectral efficiency $M - 1$ bits/symbol, hard-decision BICM (or hard-decision PID) introduces a loss of approximately 1.55 dB relative to the capacity of the constellation.
5.3 Performance of error-correcting system with staircase codes

We now determine the performance of our pragmatic error-correcting system at the operating points corresponding to the maximum achievable spectral efficiencies of the
fibre-optic communication systems studied in Chapter 3. For the sake of illustration, we present results for the WDM Raman-amplified systems, and the channel parameters (i.e., the crossover probabilities of the sub-channels) are estimated from simulations of fibre-optic systems with QAM input constellations. For each of the three system lengths considered (500 km, 1000 km and 2000 km), the maximum spectral efficiency for a system with linear equalization occurs for an average input power $P = -7$ dBm, and the maximum spectral efficiency for a system with digital backpropagation occurs for an average input power $P = -5$ dBm, thus we will assume the same powers for our simulations with QAM input constellations.

### 5.3.1 Achievable rates

For each of the three system lengths, we now present the achievable rates for hard-decision PID and hard-decision BICM, and compare them to the maximum spectral efficiencies computed in Chapter 3. Whether the system employs linear equalization or digital backpropagation, we assume that the constant phase rotation due to (inter-channel) XPM is canceled in the receiver; to illustrate this effect, Fig. 5.5 presents the scatter diagrams before and after cancellation of XPM-induced phase rotation, for WDM Raman-amplified transmission over $L = 2000$ km using 64-QAM with average input power $P = -7$ dBm.

$L = 500$ km

In Table 5.1, we present the estimated subchannel crossover probabilities for 256-QAM with linear equalization. Relative to the case of AWGN, the absence of pairs of equivalent sub-channels can be attributed to the non-circularity of the channel distortion, although this effect is minimal. Summing the capacities of the individual BSCs, hard-decision PID has a capacity of 7.60 bits/symbol, and from the mean of the subchannel error probabilities, hard-decision BICM has a capacity of 7.58 bits/symbol. From Chapter 3, the estimated capacity for the linearly equalized system is 8.46 bits/symbol.

In Table 5.2, we present the estimated subchannel crossover probabilities for 512-CrossQAM with digital backpropagation. Summing the capacities of the individual BSCs, hard-decision PID has a capacity of 8.36 bits/symbol, and from the mean of the subchannel error probabilities, hard-decision BICM has a capacity of 8.33 bits/symbol. From Chapter 3, the estimated capacity for the digitally backpropagated system is 8.96
Figure 5.5: Scatter diagram of 64-QAM, before and after phase rotation compensation.

In Table 5.3, we present the estimated subchannel crossover probabilities for 128-CrossQAM with linear equalization. Summing the capacities of the individual BSCs, hard-decision PID has a capacity of 6.58 bits/symbol, and from the mean of the subchannel error probabilities, hard-decision BICM has a capacity of 6.57 bits/symbol. From Chapter 3, the estimated capacity for the linearly equalized system is 7.39 bits/symbol.

In Table 5.4, we present the estimated subchannel crossover probabilities for 256-QAM with digital backpropagation. Summing the capacities of the individual BSCs, hard-
Table 5.1: Subchannel error rates for equalized 256-QAM, $L = 500$ km.

<table>
<thead>
<tr>
<th>Sub-channel index</th>
<th>Crossover probability</th>
<th>Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$1.30 \times 10^{-3}$</td>
<td>0.9857</td>
</tr>
<tr>
<td>2</td>
<td>$3.25 \times 10^{-3}$</td>
<td>0.9684</td>
</tr>
<tr>
<td>3</td>
<td>$7.07 \times 10^{-3}$</td>
<td>0.9394</td>
</tr>
<tr>
<td>4</td>
<td>$1.26 \times 10^{-2}$</td>
<td>0.9021</td>
</tr>
<tr>
<td>5</td>
<td>$1.74 \times 10^{-3}$</td>
<td>0.9815</td>
</tr>
<tr>
<td>6</td>
<td>$2.78 \times 10^{-3}$</td>
<td>0.9724</td>
</tr>
<tr>
<td>7</td>
<td>$6.58 \times 10^{-3}$</td>
<td>0.9429</td>
</tr>
<tr>
<td>8</td>
<td>$1.25 \times 10^{-2}$</td>
<td>0.9032</td>
</tr>
</tbody>
</table>

Table 5.2: Subchannel error rates for backpropagated 512-CrossQAM, $L = 500$ km.

<table>
<thead>
<tr>
<th>Subchannel index</th>
<th>Crossover probability</th>
<th>Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$4.39 \times 10^{-3}$</td>
<td>0.9593</td>
</tr>
<tr>
<td>2</td>
<td>$1.85 \times 10^{-3}$</td>
<td>0.9805</td>
</tr>
<tr>
<td>3</td>
<td>$7.12 \times 10^{-3}$</td>
<td>0.9389</td>
</tr>
<tr>
<td>4</td>
<td>$1.11 \times 10^{-2}$</td>
<td>0.9117</td>
</tr>
<tr>
<td>5</td>
<td>$2.05 \times 10^{-2}$</td>
<td>0.8556</td>
</tr>
<tr>
<td>6</td>
<td>$1.95 \times 10^{-3}$</td>
<td>0.9796</td>
</tr>
<tr>
<td>7</td>
<td>$3.54 \times 10^{-3}$</td>
<td>0.9661</td>
</tr>
<tr>
<td>8</td>
<td>$1.06 \times 10^{-2}$</td>
<td>0.9155</td>
</tr>
<tr>
<td>9</td>
<td>$2.50 \times 10^{-2}$</td>
<td>0.8560</td>
</tr>
</tbody>
</table>

decision PID has a capacity of 7.29 bits/symbol, and from the mean of the subchannel error probabilities, hard-decision BICM has a capacity of 7.26 bits/symbol. From Chapter 3, the estimated capacity for the digitally backpropagated system is 8.10 bits/symbol.

$L = 2000$ km

In Table 5.5, we present the estimated subchannel crossover probabilities for 64-QAM with linear equalization. Summing the capacities of the individual BSCs, hard-decision PID has a capacity of 5.57 bits/symbol, and from the mean of the subchannel error probabilities, hard-decision BICM has a capacity of 5.56 bits/symbol. From Chapter 3,
Chapter 5. Staircase codes in high-spectral-efficiency fibre-optic systems

<table>
<thead>
<tr>
<th>Subchannel index</th>
<th>Crossover probability</th>
<th>Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5.70 × 10^{-3}</td>
<td>0.9494</td>
</tr>
<tr>
<td>2</td>
<td>2.50 × 10^{-3}</td>
<td>0.9748</td>
</tr>
<tr>
<td>3</td>
<td>7.07 × 10^{-3}</td>
<td>0.9260</td>
</tr>
<tr>
<td>4</td>
<td>1.36 × 10^{-2}</td>
<td>0.8959</td>
</tr>
<tr>
<td>5</td>
<td>2.82 × 10^{-3}</td>
<td>0.9720</td>
</tr>
<tr>
<td>6</td>
<td>4.30 × 10^{-3}</td>
<td>0.9600</td>
</tr>
<tr>
<td>7</td>
<td>1.23 × 10^{-2}</td>
<td>0.9045</td>
</tr>
</tbody>
</table>

Table 5.3: Subchannel error rates for equalized 128-CrossQAM, \( L = 1000 \) km.

<table>
<thead>
<tr>
<th>Subchannel index</th>
<th>Crossover probability</th>
<th>Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3.31 × 10^{-3}</td>
<td>0.9680</td>
</tr>
<tr>
<td>2</td>
<td>6.47 × 10^{-3}</td>
<td>0.9436</td>
</tr>
<tr>
<td>3</td>
<td>1.23 × 10^{-2}</td>
<td>0.9038</td>
</tr>
<tr>
<td>4</td>
<td>2.53 × 10^{-2}</td>
<td>0.8298</td>
</tr>
<tr>
<td>5</td>
<td>3.32 × 10^{-3}</td>
<td>0.9678</td>
</tr>
<tr>
<td>6</td>
<td>6.35 × 10^{-3}</td>
<td>0.9445</td>
</tr>
<tr>
<td>7</td>
<td>1.23 × 10^{-2}</td>
<td>0.9043</td>
</tr>
<tr>
<td>8</td>
<td>2.53 × 10^{-2}</td>
<td>0.8300</td>
</tr>
</tbody>
</table>

Table 5.4: Subchannel error rates for backpropagated 256-QAM, \( L = 1000 \) km.

The estimated capacity for the linearly equalized system is 6.45 bits/symbol.

<table>
<thead>
<tr>
<th>Subchannel index</th>
<th>Crossover probability</th>
<th>Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3.77 × 10^{-3}</td>
<td>0.9642</td>
</tr>
<tr>
<td>2</td>
<td>7.17 × 10^{-3}</td>
<td>0.9386</td>
</tr>
<tr>
<td>3</td>
<td>1.61 × 10^{-2}</td>
<td>0.8813</td>
</tr>
<tr>
<td>4</td>
<td>3.65 × 10^{-3}</td>
<td>0.9652</td>
</tr>
<tr>
<td>5</td>
<td>7.48 × 10^{-3}</td>
<td>0.9364</td>
</tr>
<tr>
<td>6</td>
<td>1.49 × 10^{-2}</td>
<td>0.8882</td>
</tr>
</tbody>
</table>

Table 5.5: Subchannel error rates for equalized 64-QAM, \( L = 2000 \) km.
In Table 5.6, we present the estimated subchannel crossover probabilities for 128-CrossQAM with digital backpropagation. Summing the capacities of the individual BSCs, hard-decision PID has a capacity of 6.24 bits/symbol, and from the mean of the subchannel error probabilities, hard-decision BICM has a capacity of 6.21 bits/symbol. From Chapter 3, the estimated capacity for the digitally backpropagated system is 7.26 bits/symbol.

<table>
<thead>
<tr>
<th>Subchannel index</th>
<th>Crossover probability</th>
<th>Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$9.63 \times 10^{-3}$</td>
<td>0.9217</td>
</tr>
<tr>
<td>2</td>
<td>$5.75 \times 10^{-3}$</td>
<td>0.9489</td>
</tr>
<tr>
<td>3</td>
<td>$2.06 \times 10^{-2}$</td>
<td>0.8554</td>
</tr>
<tr>
<td>4</td>
<td>$2.75 \times 10^{-2}$</td>
<td>0.8183</td>
</tr>
<tr>
<td>5</td>
<td>$6.07 \times 10^{-3}$</td>
<td>0.9465</td>
</tr>
<tr>
<td>6</td>
<td>$8.36 \times 10^{-3}$</td>
<td>0.9303</td>
</tr>
<tr>
<td>7</td>
<td>$2.78 \times 10^{-2}$</td>
<td>0.8165</td>
</tr>
</tbody>
</table>

Table 5.6: Subchannel error rates for backpropagated 128-CrossQAM, $L = 2000$ km.

As expected, in each of the investigated cases, hard-decision BICM provides essentially the same performance as hard-decision PID; in the following we focus on BICM, since its implementation requires a single error-correcting code. Among the cases considered, the loss in spectral efficiency (relative to the fibre-optic capacity curves) ranges from 0.63 to 1.05 bits/s/Hz. From the calculation in Section 5.2, the expected loss due to hard-decisions is 0.516 bits/s/Hz (or 1.55 dB), however, this loss is relative to the capacity of the QAM constellation, not to the Shannon Limit. That is, the additional loss in spectral efficiency can be attributed to the fact that the ring constellations considered in Chapter 3 provide significant shaping gain (for spectral efficiencies below 8 bits/s/Hz, a 32-ring constellation performs within 0.62 dB of the Shannon Limit).

### 5.3.2 Staircase codes for hard-decision BICM

We now propose staircase codes for each of the investigated systems, and provide simulation results and error floor estimates in each case. Since the input bit-error-rates are larger than those that the G.709-compliant staircase code is able to correct, we will consider BCH component codes with $t = 4$, which will provide better performance.
### Table 5.7: Staircase codes for hard-decision BICM.

In Table 5.7, we provide the parameters of the staircase codes; the terminology used to describe the codes follows Section 4.2.2. In each case, the length of the (mother) BCH component code is the smallest $2^n - 1$ that is greater than or equal to $2^m$. In Fig. 5.6, the bit-error-rate curves are plotted. Since these curves were computed without an FPGA implementation, we were only able to obtain results to approximately $10^{-10}$. Extrapolating the curves to $10^{-15}$, each code is expected to provide an output error rate of better than $10^{-15}$ at the input error rate induced by its corresponding system. The error floor estimates are calculated by the methods outlined in Ch. 4, with $p$ set to the average of the sub-channel error rates. In practice, this assumption may require some degree of interleaving.

For the linearly equalized systems, staircase codes perform within $0.19$ bits/s/Hz of the achievable spectral efficiencies for hard-decision BICM, or approximately $0.6$ dB from the capacity of our pragmatic system. For the digitally backpropagated systems, staircase codes perform within $0.32$ to $0.41$ bits/s/Hz of the achievable spectral efficiencies for hard-decision BICM, or approximately $0.96$ to $1.23$ dB from the capacity of our pragmatic system. We note that the gap to capacity is larger for the digitally backpropagated systems because the corresponding channel error rates are larger; however, this is not a fundamental property of backpropagated system, but rather a function of the system lengths $L$ we have considered.

### 5.4 Conclusions

We proposed a pragmatic approach for coded modulation in fibre-optic communication systems, leveraging the low-complexity implementations of syndrome-decoded product-
like codes. Similar to the well-known fact that BICM approaches the capacity of a constellation for AWGN, we showed that hard-decision BICM closely approaches the hard-decision capacity of the constellation. We simulated WDM Raman-amplified fibre-optic communication systems, using the results to determine the capacity of hard-decision BICM at the operating points (i.e., the input powers) at which the capacity of the system is maximized, and showed that hard-decision BICM performs within 1 bit/s/Hz of the fundamental limit, and we argued that this loss could be reduced via shaping. Finally, we designed staircase codes for hard-decision BICM, and showed that they perform within 0.19 bits/s/Hz of the capacity of hard-decision BICM for linearly equalized systems.
and within 0.28 to 0.41 bits/s/Hz of the capacity of hard-decision BICM for digitally backpropagated systems, with error floors below $10^{-25}$. 
Chapter 6

Staircase Codes for a DQPSK System

We consider the use of the G.709-compliant staircase code in a DQPSK 100 Gb/s system with correlated bit-errors. Since correlated bit-errors increase the error floor at the output of the system, we propose an interleaving strategy to reduce the appearance of multiple correlated bit-errors in the stall patterns of the error-correcting code (ECC), and estimate the error floor of the resulting system. The channel model and ECC-to-DQPSK mapping is closely based on a commercially-available DQPSK receiver, the (private) details of which we obtained during the course of an industrial collaboration.

In Section 6.1 we describe the system model of the DQPSK system, including the channel model, the mapping of bits from the encoder to DQPSK symbols, and we propose the use of pseudorandom interleavers to mitigate the effects of error correlation. In Section 6.2 we present an analysis of the error floor of the resulting system, and show that the error floor is less than $4.77 \times 10^{-21}$.

6.1 Abstraction of a DQPSK system

In order to motivate the analysis, consider the following toy-model of a communication system with correlated errors. Let’s suppose that the channel produces pairs of errored bits, and—due to the specific mapping of bits from the error-correcting code to the transmitted symbols—these pairs of bits map to positions in the same row (or possibly column) of some product-like error-correcting code. In this case, stall patterns may arise with correlated pairs of bit errors, and the probability of a given stall pattern depends
strongly on its ‘composition’. Most importantly, for a fixed bit-error-rate (at the input of the decoder), the probability of observing a stall pattern may be significantly increased, relative to a channel with independent bit errors. In fact, for the case of the G.709-compliant staircase code and a commercially-available DQPSK receiver, this ‘toy-model’ closely mimics reality. In the remainder of this section, we present the details of its system model, represented in Fig. 6.1.

Figure 6.1: Model for a DQPSK fibre-optic system.

### 6.1.1 Channel model

For the commercially-available DQPSK receiver, the decoder sees an overall channel that produces errors in a manner that is well-modeled by a simple memoryless system with three states: \( \{E_{\pi/2}, E_{-\pi/2}, G\} \). These states represent the possible outcomes, i.e., the effects of the channel on a (transmitted) QPSK symbol. If the channel is in state \( G \), the transmitted symbol is received noiselessly; if the state is \( E_{\pi/2} \), the channel rotates the transmitted signal by \( \pi/2 \) radians; if the state is \( E_{-\pi/2} \), the channel rotates the transmitted signal by \(-\pi/2 \) radians. These outcomes have associated probabilities

\[
\begin{align*}
\Pr[E_{\pi/2}] &= \frac{p_s}{2} \\
\Pr[E_{-\pi/2}] &= \frac{p_s}{2} \\
\Pr[G] &= 1 - p_s,
\end{align*}
\]

where \( 0 \leq p_s \leq 1 \). It will be useful to define the event \( B = \{E_{\pi/2}, E_{-\pi/2}\} \), \( \Pr[B] = p_s \), that is, the event that a transmitted symbol is received in error (the ‘bad’ event).

One interpretation of this model is as an approximation (since errors by \( \pi \) radians are not modeled) to a DQPSK receiver in which the output of the channel is quantized (to one of four phases) prior to DQPSK demodulation.

### DQPSK: Modulation

A DQPSK modulator encodes two bits of information (represented by the pair \((A, B)\), \( A, B \in \mathbb{F}_2 \)) into the change in phase between two consecutively transmitted QPSK sym-
bols. By \((A_i, B_i)\) we denote the \(i\)th pair of bits, and by \(T_i\) the \(i\)th transmitted QPSK symbol. The operation of the modulator is described as follows:

\[
\angle T_i = \angle T_{i-1} + \Delta_{T,i},
\]

<table>
<thead>
<tr>
<th>(\Delta_{T,i})</th>
<th>((A_i, B_i))</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>(\pi/2)</td>
<td>(0, 1)</td>
</tr>
<tr>
<td>(\pi)</td>
<td>(1, 1)</td>
</tr>
<tr>
<td>(-\pi/2)</td>
<td>(1, 0)</td>
</tr>
</tbody>
</table>

**DQPSK: Detection**

We consider a receiver that computes the phase difference between consecutive symbols, and subsequently translates this quantity to an estimate \((\hat{A}_i, \hat{B}_i)\) of \((A_i, B_i)\). We denote by \(R_i\) the \(i\)th received QPSK symbol, and model the relationship between input and output symbol angles as

\[
\angle R_i = \angle T_i + N_i,
\]

where \(N_i = \pi/2\) when the channel is in state \(E_{\pi/2}\), \(N_i = -\pi/2\) when the channel is in state \(E_{-\pi/2}\), and \(N_i = 0\) when the channel is in state \(G\). The receiver forms

\[
\Delta_{R,i} = \angle R_i - \angle R_{i-1} = \Delta_{T,i} + (N_i - N_{i-1}),
\]

and determines \((\hat{A}_i, \hat{B}_i)\) by performing the modulator’s inverse mapping. In general, we have

\[
(\hat{A}_i, \hat{B}_i) = (A_i, B_i) + (\hat{e}_{i,1}, \hat{e}_{i,2}),
\]

\(\hat{e}_{i,1}, \hat{e}_{i,2} \in \mathbb{F}_2\) and addition is performed over \(\mathbb{F}_2\).

The following table illustrates the relationship between \(\Delta_{N,i} = N_i - N_{i-1}\) and \((\hat{e}_{i,1}, \hat{e}_{i,2})\):

<table>
<thead>
<tr>
<th>(\Delta_{N,i})</th>
<th>((A_i, B_i) = (0, 0))</th>
<th>((A_i, B_i) = (0, 1))</th>
<th>((A_i, B_i) = (1, 0))</th>
<th>((A_i, B_i) = (1, 1))</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>(0, 0)</td>
<td>(0, 0)</td>
<td>(0, 0)</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>(\pi/2)</td>
<td>(0, 1)</td>
<td>(1, 0)</td>
<td>(1, 0)</td>
<td>(0, 1)</td>
</tr>
<tr>
<td>(\pi)</td>
<td>(1, 1)</td>
<td>(1, 1)</td>
<td>(1, 1)</td>
<td>(1, 1)</td>
</tr>
<tr>
<td>(-\pi/2)</td>
<td>(1, 0)</td>
<td>(0, 1)</td>
<td>(0, 1)</td>
<td>(1, 0)</td>
</tr>
</tbody>
</table>
6.1.2 Mapping: ECC-to-DQPSK

Due to the memory inherent in the detection process, correlated errors occur between temporally-consecutive \((A, B)\) pairs. However, it is not true that consecutive bits (at the output of the ECC encoder) are mapped to consecutive \((A, B)\) pairs. In this section, our main objective is to characterize the set of possible ‘lengths’ separating correlated bit errors, where the (integer) length is defined as their temporal separation with respect to the order of bits at the output of the encoder (equivalently, at the input to the decoder).

**Decomposition of \(\Pi_{DQ}\)**

In practice, \(\Pi_{DQ}\) consists of two mappers. The first implements a multi-channel parallel interface from the OTU4 framing structure to twenty logical lanes (which are implemented via 10 electrical lanes), as standardized in ITU-T Recommendation G.709/Y.1331. Since each electrical lane must periodically ‘see’ the frame alignment sequence (FAS), a rotating assignment of 16-byte units is specified, resulting in a periodically-time-varying mapping. Furthermore, the delays (measured in bits) on the individual electrical lanes are stochastic.

The twenty logical lanes at the output of the first interleaver are then mapped to two ‘physical’ lanes, representing the components of the \((A, B)\) pairs; this second mapper is vendor-specific.

**Properties of \(\Pi_{DQ}\)**

For the commercially-available DQPSK receiver we consider, \(\Pi_{DQ}\) results in correlated bit errors that are separated (in the bit stream input to \(\Pi_{ECC}^{-1}\)) by a number \(l\) of (binary) symbols, where \(l\) is in the vicinity of multiples of 128, and \(l > 512\). However, in order to provide a tractable analysis, the error floor estimation technique presented in Section 6.2 does not require full knowledge of the specific bit-to-symbol mapper \(\Pi_{DQ}\). Rather, the analysis depends only on the following property of \(\Pi_{DQ}\):

1. At the output of \(\Pi_{DQ}^{-1}\), the maximum length between correlated errors is 4608.

**Properties of \(\Pi_{ECC}\)**

In order to randomize the distance between correlated bit errors, the contents of the rows of a staircase block are bit-interleaved prior to the DQPSK mapper/modulator.
 Specifically, after encoding, the rows of the $512 \times 510$ blocks of the G.709-compliant staircase code are first grouped into units of four consecutive rows. Each such grouping of 2040 bits is then passed to $\Pi_{\text{ECC}}$, which performs a 2040-bit pseudorandom interleaving; in practice, the interleaver is programmable, such that a distinct interleaver is applied to each set of four rows within a staircase block.

6.2 Error Floor Estimation

For fixed $\Pi_{\text{ECC}}$ and $\Pi_{\text{DQ}}$, and some fixed stall pattern $s$, the bits in $s$ map to some set of $(A, B)$ pairs. This assignment of bits to $(A, B)$ pairs in turn affects the probability of the bits in $s$ being received in error. Since $\Pi_{\text{ECC}}$ and $\Pi_{\text{DQ}}$ are effectively random, we can only hope to determine the expected value of this probability, over the corresponding class of interleavers. In the following, we thus estimate the expected value of the error floor, for DQPSK transmission with the G.709-compliant code.

6.2.1 An outline of the approach

In order to simplify the analysis, we only consider the contribution of the set of minimal stalls $S$ that lie entirely within a single staircase block, and we assume that a stall occurs only when each of the bits is received in error. As we showed in Chapter 4, minimal stalls provide the dominant contribution to the error floor. Furthermore, since $(p + \zeta)^{16}/p^{16} \approx 6.2$ for our case of interest, accounting for incorrect decodings affects the error floor by less than an order of magnitude.

By the union bound, the expected value of the error floor is then modeled as

$$\mathcal{E}\{\text{BER}_{\text{floor}}\} \approx \frac{16}{510 \cdot 512} \sum_{s \in S} \mathcal{E}\{\Pr[\text{bits in } s \text{ received in error}]\}.$$  

Clearly, our task is to determine (or overbound) $\mathcal{E}\{\Pr[\text{bits in } s \text{ received in error}]\}$, for each of the $s$ in $S$. Due to the many steps involved, we now give an overview of our approach.

Recalling that $\Pi_{\text{ECC}}$ is composed of block interleavers that operate on four consecutive rows of a staircase block, the elements of $S$ can be partitioned according to the number of distinct (4-row) interleavers to which their bits ‘belong’. We will see that a tractable analysis is possible for an $s$ for which all 16 bits belong to a single interleaver. In this case, the assignment of bits in $s$ to $(A, B)$ pairs—over the choice of $\Pi_{\text{ECC}}$ and $\Pi_{\text{DQ}}$—can
be modeled as tossing balls into bins, and a combinatorial analysis follows; the analysis includes a method to determine \( \Pr[\text{bits in } s \text{ received in error}] \) for a known assignment of the bits in \( s \) to \((A, B)\) pairs (see Section 6.2.3), and a method to enumerate the relative frequencies of the various assignments (see Section 6.2.5).

For the general case, it follows from the properties of \( \Pi_{DQ} \) that the elements of \( S \) can be partitioned according to the potential for correlated errors among the bits in the four rows of a stall pattern (e.g., if each of the four rows in some stall \( s \) are separated by 16 or more rows, then a pair of bits—each from distinct rows—cannot be involved in a correlated error). For each of the sets of ‘correlated’ rows, we then argue that our previous analysis provides a worst-case estimation of the contribution to the error floor, and that the independent contributions (of uncorrelated sets of rows) can be multiplied (see Section 6.2.6).

### 6.2.2 Sequence error probabilities

Due to the fact that a single erroneously received symbol \( R_i \) affects the estimation of both \((A_i, B_i)\) and \((A_{i+1}, B_{i+1})\), the errors in their estimation are correlated. For our analysis of error floors, we will be especially interested in determining the total probability of all error sequences that ‘dominate’ an error pattern of interest. We define an error sequence to be a vector of ordered pairs (of elements in \( \mathbb{F}_2 \)), such that each ordered pair includes at least one non-zero element. For \( k \geq 0 \), we will say that

\[
C = [(c_{i,1}, c_{i,2}), (c_{i+1,1}, c_{i+1,2}), \ldots, (c_{i+k,1}, c_{i+k,2})]
\]

dominates

\[
D = [(d_{i,1}, d_{i,2}), (d_{i+1,1}, d_{i+1,2}), \ldots, (d_{i+k,1}, d_{i+k,2})],
\]

(which we denote \( C \succeq_H D \)) if and only if \( d_{j,1} = 1 \) implies \( c_{j,1} = 1 \), and \( d_{j,2} = 1 \) implies \( c_{j,2} = 1 \), \( \forall j \in \{i, i+1, \ldots, i + k\} \). For example, \([1, 0], (1, 1)] \succeq_H [(1, 0), (0, 1)], \) but \([1, 0], (1, 1)] \) does not dominate \([(0, 1), (0, 1)]\).

In this section, we will show that for a run of \( M \) consecutive \((A, B)\) pairs,

\[
[(A_j, B_j), (A_{j+1}, B_{j+1}), \ldots, (A_{j+M-1}, B_{j+M-1})],
\]

any fixed error sequence

\[
E = [(e_{j,1}, e_{j,2}), (e_{j+1,1}, e_{j+1,2}), \ldots, (e_{j+M-1,1}, e_{j+M-1,2})],
\]
and

\[ \hat{E} = [(\hat{A}_j, \hat{B}_j) - (A_j, B_j), \ldots, (\hat{A}_{j+M-1}, \hat{B}_{j+M-1}) - (A_{j+M-1}, B_{j+M-1})], \]

\[ \Pr[\hat{E} \succeq H \ E] \leq \frac{1}{2^M} \sum_{i=0}^{1 + \lceil \frac{M}{2} \rceil} (M + 2 - i) \binom{1}{i} (1 - p_s)^i p_s^{M+1-i}. \]

In words, this is the probability that the detected \((A, B)\) pairs include errors in (at least) those positions specified by the non-zero entries of \(E\).

**One \((A, B)\) pair**

Consider the case of a single pair \((A_i, B_i)\). A priori, each of the four binary pairs is equally likely to have been transmitted. For a fixed \(E = [(e_i, 1), (e_i, 2)]\) such that \(e_{i,1} = 1\) or \(e_{i,2} = 1\) (but not both), and

\[ \hat{E} = [(\hat{A}_i, \hat{B}_i) - (A_i, B_i)], \]

we have:

| \(S_{i-1}\) | \(S_i\) | \(\Pr[\hat{E} \succeq H \ E | S_{i-1}, S_i]\) |
|---|---|---|
| \(G\) | \(G\) | 0 |
| \(G\) | \(B\) | 1/2 |
| \(B\) | \(G\) | 1/2 |
| \(B\) | \(B\) | 1/2 |

where \(S_i \in \{G, B\}\) denotes the state of the channel for the transmission of \(T_i\). Therefore, the marginal probability that a bit is detected in error is

\[ p_b = p_s (1 - p_s) \left( \frac{1}{2} \right) + (1 - p_s) p_s \left( \frac{1}{2} \right) + p_s^2 \left( \frac{1}{2} \right) \]

\[ = p_s - \frac{p_s^2}{2}. \]

For \(p_b\) in our regime of interest, \(p_b \approx p_s\).

**Two \((A, B)\) pairs**

Before addressing the general case, we consider the case of two consecutive pairs, \((A_i, B_i)\) and \((A_{i+1}, B_{i+1})\). Again, for a fixed \(E = [(e_{i,1}, e_{i,2}), (e_{i+1,1}, e_{i+1,2})]\) such that \(e_{i,1} = 1\) or \(e_{i,2} = 1\) (but not both) and \(e_{i+1,1} = 1\) or \(e_{i+1,2} = 1\) (but not both), and

\[ \hat{E} = [(\hat{A}_i, \hat{B}_i) - (A_i, B_i), (\hat{A}_{i+1}, \hat{B}_{i+1}) - (A_{i+1}, B_{i+1})], \]
we have:

| $S_{i-2}$ | $S_{i-1}$ | $S_i$ | $\Pr[\hat{E} \succeq_H \hat{E}|S_{i-2}, S_{i-1}, S_i]$ |
|-----------|-----------|-------|--------------------------------------------------|
| $G$       | $G$       | $G$   | $0$                                               |
| $G$       | $G$       | $B$   | $0$                                               |
| $G$       | $B$       | $G$   | $\left(\frac{1}{2}\right)^2$                    |
| $G$       | $B$       | $B$   | $\left(\frac{1}{2}\right)^2$                    |
| $B$       | $G$       | $G$   | $0$                                               |
| $B$       | $G$       | $B$   | $\left(\frac{1}{2}\right)^2$                    |
| $B$       | $B$       | $G$   | $\left(\frac{1}{2}\right)^2$                    |
| $B$       | $B$       | $B$   | $\left(\frac{1}{2}\right)^2$                    |

Therefore,

$$\Pr[\hat{E} \succeq_H \hat{E}] = \frac{1}{4} \left((1 - p_s)^2 p_s + 3 p_s^2 (1 - p_s) + p_s^3\right).$$

**M** $(A, B)$ **pairs**

For the general case of $M$ consecutive $(A, B)$ pairs,

$$[(A_j, B_j), (A_{j+1}, B_{j+1}), \ldots, (A_{j+M-1}, B_{j+M-1})],$$

we restrict ourselves (for the moment) to error sequences $E$,

$$E = [(e_{j,1}, e_{j,2}), (e_{j+1,1}, e_{j+1,2}), \ldots, (e_{j+M-1,1}, e_{j+M-1,2})],$$

in which each of the elements $(e_{k,1}, e_{k,2})$, $k \in \{j, j + 1, \ldots, j + M - 1\}$, has $e_{k,1} = 1$ or $e_{k,2} = 1$ (but not both). Under this assumption, we note that every sequence of channel states

$$S = [S_{j-1}, S_j, \ldots, S_{j+M-1}],$$

such that no two consecutive elements of $S$ are equal to $G$, satisfies

$$\Pr[\hat{E} \succeq_H \hat{E}|S] = \left(\frac{1}{2}\right)^M,$$

and any $S$ for which there exists two consecutive elements equal to $G$, satisfies

$$\Pr[\hat{E} \succeq_H \hat{E}|S] = 0.$$

That is, the problem is reduced to counting those sequences (with elements from $\{G, B\}$) of length $M + 1$, such that no two consecutive $G$s appear. From Lemma 1, there are
such sequences with exactly \( i \) \( G \)s. Furthermore, no sequence can have more than \( i = 1 + \left\lfloor \frac{M}{2} \right\rfloor \) \( G \)s, otherwise there must be two consecutive \( G \)s. Therefore, we have

\[
\Pr[\hat{\mathbf{E}} \succeq_H \mathbf{E}] = \frac{1}{2^M} \sum_{i=0}^{1+\left\lfloor \frac{M}{2} \right\rfloor} \binom{M+2-i}{i} (1-p_s)^i p_s^{M+1-i}.
\]

**Lemma 1.** There are \( \binom{N-k+1}{k} \) strings of length \( N \) with elements in \( \{0,1\} \) that have exactly \( k \) 1s, but no consecutive 1s.

**Proof.** First, consider the set of strings that do not have have consecutive 1s and that end in 0. Any such string can be parsed uniquely into the phrases \( A = 10 \) and \( B = 0 \). A string of length \( N \) with \( k \) As has \( N - 2k \) Bs, thus there are \( \binom{N-k}{k} \) ways to construct such a string. It remains to count those strings ending in 1. Since the second-to-last position cannot also be 1, the first \( N - 1 \) positions can be parsed uniquely into As and Bs. If the string of length \( N \) has \( k \) 1s, then the first \( N - 1 \) positions include exactly \( k - 1 \) As and \( N - 2k + 1 \) Bs, for which there are \( \binom{N-k}{k-1} \) ways to construct such a string. Therefore, the number of strings of length \( N \) with \( k \) 1s (such that there are no consecutive 1s) is

\[
N_k = \binom{N-k}{k} + \binom{N-k}{k-1} = \binom{N-k+1}{k}.
\]

Finally, suppose that we allow error sequences \( \mathbf{E} \) that include the \((1,1)\) ordered pair. In this case, the set of channel states sequences for which \( \hat{\mathbf{E}} \succeq_H \mathbf{E} \) is strictly reduced, and thus we have the upper bound

\[
\Pr[\hat{\mathbf{E}} \succeq_H \mathbf{E}] \leq \frac{1}{2^M} \sum_{i=0}^{1+\left\lfloor \frac{M}{2} \right\rfloor} \binom{M+2-i}{i} (1-p_s)^i p_s^{M+1-i} \triangleq Q(M).
\]

6.2.3 **Bounding \( \Pr[\text{bits in } s \text{ received in error}] \)**

In order to upperbound \( \Pr[\text{bits in } s \text{ received in error}] \), for fixed \( s \) and a fixed mapping of the bits in \( s \) to \((A,B)\) pairs, we first define an \( M\)-run of errored \((A,B)\) pairs to be a sequence of detected pairs

\[
\Gamma = \{(\hat{A}_{j-1}, \hat{B}_{j-1}), (\hat{A}_j, \hat{B}_j), \ldots, (\hat{A}_{j+M-1}, \hat{B}_{j+M-1})\},
\]

such that

\[
\mathbf{E} = [(e_{j-1,1}, e_{j-1,2}), (e_{j,1}, e_{j,2}), \ldots, (e_{j+M,1}, e_{j+M,2})]
\]
has \((e_{k,1}, e_{k,2}) \neq (0, 0), \forall k \in \{ j, j + 1, \ldots, j + M - 1 \}\), and \((e_{k,1}, e_{k,2}) = (0, 0)\) for \(k \in \{ j - 1, j + M \}\). We can then bound the probability of interest for \(s\) as follows: First, we identify each disjoint run of consecutive \((A, B)\) pairs that involve bits in \(s\). Next, for each run, we use (6.1) to upper-bound the probability that the corresponding bits are received in error, and then we multiply these probabilities to obtain an upper-bound on all 16 bits in \(s\) being received in error. Note that the multiplication operation is justified by the memorylessness of the physical channel, since calculating the probabilities of disjoint runs involves non-intersecting (and thus independently observed) sets of received symbols (see Fig. 6.2).

Figure 6.2: Note that two disjoint runs do not involve common DQPSK symbols.

In the following, we will make use of the concept of ‘run-combining’, by which we mean removing (for the purposes of the calculation of the probability) the (uninvolved) \((A, B)\) pairs that separate one or more of the runs in which the bits in \(s\) are involved. That is, after ‘removing’ the \((A, B)\) pairs separating two runs, the pairs involved in said runs are assumed to be temporally consecutive.

**Definition 3.** For \(A, B \in \mathbb{Z}^N\), \(A \succ B\) implies that the elements of \(A - B\) are non-negative and that \(A - B\) is not the zero vector.

**Lemma 2.** Run-combining strictly increases our upperbound on the probability of receiving the bits in \(s\) in error.

**Proof.** For any non-negative integers \(K\) and \(M\), \(0 < K < M\), the total probability of distinct \(K\)- and \((M - K)\)-runs is upperbounded as \(Q(K) \cdot Q(M - K)\). From the expansion of this expression in terms of powers of \(p_s\) (but without expanding any of the terms of the form \((1 - p_s)^l\)), we denote the coefficient vector (with \(M + 2\) elements, in order of increasing powers of \(p_s\)) by \(C_2^1\). Similarly, we denote the coefficient vector corresponding

\[\text{For example, } (p_s^2 + p_s(1 - p_s) + (1 - p_s)^2)^2 = p_s^4 + 2p_s^3(1 - p_s) + 3p_s^2(1 - p_s)^2 + 2p_s(1 - p_s)^3 + (1 - p_s)^4,\]

and the corresponding coefficient vector is \((1, 2, 3, 2, 1)\).
to $Q(M) \cdot (p_s + (1 - p_s))$ by $C_1$. It can be verified by enumeration that $C_1 \succ C_2$, which implies that the upperbound on the probability strictly increases.

6.2.4 Mapping stall bits to $(A, B)$ pairs

In the general case, the bits in some stall pattern $s$ ‘originate’ from distinct 4-row interleavers (recall, $\Pi_{ECC}$ consists of 128 4-row time-varying interleavers). For a fixed set of $k$ positions at the input of a 4-row interleaver, the output of the 4-row interleaver is modeled (over the random choice of interleaver) by tossing $k$ indistinguishable balls into 2040 distinguishable bins.

In the following, we enumerate the relative frequencies of the assignments (of the bits in $s$ to $(A, B)$ pairs) over the class of random interleavers. For each such assignment, $\Pr[\text{bits in } s \text{ received in error}]$ is upperbounded as described above. In general, this enumeration is a difficult problem; to make progress, we perform the enumeration in a special case, and argue that this approach leads to an upperbound on the error floor estimate.

6.2.5 A worst-case assumption

Consider the case of a stall pattern $s$ wherein all 16 (stall) bits are interleaved by the same 2040-bit interleaver; this requires that the rows (of the stall pattern) occupy four consecutive rows of a staircase block. We will also assume that $\Pi_{DQ}$ is such that the (corresponding) 2040 bits at its input are mapped to 1020 temporally-consecutive $(A, B)$ pairs,

$$\{(A_i, B_i), (A_{i+1}, B_{i+1}), \ldots, (A_{i+1019}, B_{i+1019})\}.$$  

For any such $s$, we wish to over-bound the expected probability that its bits are received in error, over the set of random 2040-bit interleavers. Due to the averaging operation over all interleavers, the specific choice of stall pattern (within this class) is irrelevant. For a fixed stall pattern, each assignment (over the set of random interleavers) of stall pattern positions to the 2040 ‘physical’ bits (by physical bits, we mean the elements of the $(A_i, B_i)$ pairs) is equivalent to an assignment of 16 indistinguishable balls to 2040 distinguishable bins, with at most one ball per bin. There are $\binom{2040}{16}$ ways to do this; we refer to each element as a configuration $c$, and the set of configurations is denoted by $C$.

Intuitively, when the 2040 ‘physical’ bits correspond to 1020 temporally-consecutive $(A, B)$ pairs, the marginal probability that some fixed bit in $c$ is involved in a correlated
error (with other bits in \(c\)) is maximized, since every physical bit (ignoring edge effects) has two neighbouring bits on the left (corresponding to the bits in the previous \((A,B)\) pair), two neighbouring bits on the right (corresponding to the bits in the next \((A,B)\) pair), and one other bit within its \((A,B)\) pair. For a more general correspondence between physical bits and \((A,B)\) pairs, a physical bit may have 0, 1 or 2 neighbouring bits on the left (and similarly for right neighbours), and 0 or 1 neighbouring bits within its \((A,B)\) pair. Since correlation increases the probability of observing a configuration, we conjecture that the greater opportunity for correlation will maximize the upperbound on the probability of observing the configurations, in the spirit of Lemma 2. In the following, for each configuration, we upperbound the probability that the corresponding bits are received in error.

**Partitioning C**

An efficient technique for determining the probability (of receiving the bits in \(c\) in error) for each of the \(\binom{2040}{16}\) configurations is possible by partitioning the set \(C\). Note that any configuration \(c \in C\) can be classified by a 3-tuple \((d, i, \lambda)\), where \(d\) is the number of \((A,B)\) pairs for which both bits are in \(c\), \(i\) is the number of ‘isolated’ \((A,B)\) pairs in \(c\) \([A_i, B_i] \) is isolated if and only if at least one of its bits is in \(c\), and neither \((A_{i-1}, B_{i-1})\) nor \((A_{i+1}, B_{i+1})\) have bits in \(c\), and \(\lambda\) is the set of run-lengths of the non-isolated pairs.

In general, there are many configurations that correspond to a specific \((d, i, \lambda)\); we denote the corresponding subset of configurations by \(C_{(d,i,\lambda)}\). Furthermore, for fixed \(d\) and \(i\), the set of possible \(\lambda\) is denoted by \(\Lambda_{d,i}\). Note that a \(\lambda \in \Lambda_{d,i}\) is a set of run-lengths for the non-isolated (errored) \((A,B)\) pairs. For a fixed \(d\) and \(i\), and \(2n\) total balls (i.e., errors) to assign, \(\lambda\) is a set of integers greater than one, and the sum of its elements is equal to \(N = 2n - (i + d)\). That is, the set \(\Lambda_{d,i}\) is the set of integer partitions of \(N\) with the property that the elements of the partition are greater than one. This set, \(P_N\), can be generated recursively, as follows: First, for \(N = 2\) we have \(P_2 = \{2\}\), and similarly for \(N = 3\) we have \(P_3 = \{3\}\). Now, for \(N \geq 4\), every partition other than the trivial partition (i.e., \(\{N\}\)) includes an element \(j \in \{2, \ldots, \left\lceil \frac{N}{2} \right\rceil\}\). We first list the \(\left\lfloor \frac{N}{2} \right\rfloor - 1\) partitions \(\{j, N - j\}\) for \(j \in \{2, \ldots, \left\lceil \frac{N}{2} \right\rceil\}\). Now, for any partition of \(N\) that includes \(j\), the remaining elements of the partition form a partition of \(N - j\). But since \(P_{N-j}\) is known, we can add the elements \(\{j, \{A\}\}\), \(\forall A \in P_{N-j}\), to \(P_N\). Repeating this for each \(j\), and including the trivial partition, we obtain the desired \(P_N\).
Now, averaging over all configurations (which corresponds to averaging over the set of random interleavers), we have

\[ \mathbb{E}\{ \text{Pr[bits in } s \text{ received in error]} \} = \frac{1}{(\frac{2040}{16})} \sum_{c \in C} \text{Pr[bits in } c \text{ received in error]} \]

We will now partition \( C \) to obtain an expression that can be efficiently evaluated:

\[
\frac{1}{(\frac{2040}{16})} \sum_{d=0}^{8} \sum_{i=0, \ i \neq 16-d-1}^{16-d} \sum_{\lambda \in \Lambda_{d,i}} \text{Pr[bits in } c \text{ received in error]} \\
\leq \sum_{d=0}^{8} \sum_{i=0, \ i \neq 16-d-1}^{16-d} \sum_{\lambda \in \Lambda_{d,i}} \left(\frac{2040}{16}\right) |C_{(d,i,\lambda)}| \cdot \max_{c \in C_{(d,i,\lambda)}} \text{Pr[bits in } c \text{ received in error]} \\
= \sum_{d=0}^{8} \sum_{i=0, \ i \neq 16-d-1}^{16-d} \sum_{\lambda \in \Lambda_{d,i}} \text{Pr}[d] \cdot \text{Pr}[i, \lambda | d] \cdot \max_{c \in C_{(d,i,\lambda)}} \text{Pr[bits in } c \text{ received in error]},
\]

where \( \text{Pr}[d, i, \lambda] \) is the probability that a randomly selected configuration \( c \) is in \( C_{(d,i,\lambda)} \).

**Calculating \( \text{Pr}[d] \)**

The calculation of \( \text{Pr}[d] \) is closely related to the following combinatorial identity.

**Lemma 3.** For \( n, \ M \in \mathbb{Z}^+ \), \( \sum_{i=0}^{n} 2^{2n-2i} \binom{M}{2n-i} \binom{2n-i}{i} = \binom{2M}{2n} \).

**Proof.** The right-hand side of the identity can be interpreted as the number of ways of assigning \( 2n \) indistinguishable balls to \( 2M \) distinguishable bins, with at most one ball per bin. We now give an alternative method to count the number of such configurations. First, group the \( 2M \) distinguishable bins into \( M \) pairs of two (distinguishable) bins; we will refer to these as super-bins. Now, each configuration (from the ‘right-hand’ interpretation) can be classified by the corresponding super-bins it occupies, and which elements within each super-bin are occupied. Note that \( i \) super-bins, \( n \leq i \leq 2n \), will be occupied in a valid configuration. For \( i = 2n \), there are \( \binom{M}{2n} \) ways to select the super-bins, and \( 2^{2n} \) ways to select from among the two distinguishable bins per super-bin. For \( n \leq i \leq 2n-1 \), there are \( \binom{M}{i} \) ways to select the super-bins, \( \binom{i}{2n-i} \) ways to select which super-bins are assigned two balls, and \( 2^{2i-2n} \) ways to select from among the two distinguishable bins in those super-bins occupied by a single ball. Summing over \( i \), we have \( \sum_{i=n}^{2n} 2^{2i-2n} \binom{M}{i} \binom{i}{2n-i} \), which after a change of variable, is equivalent to the left-hand side of the identity. \( \square \)
Following the same reasoning as in Lemma 3, we have
\[
\Pr[d = d_0] = \frac{2^{16-2d_0}}{\binom{1020}{16} \binom{16-d_0}{d_0}} \frac{(16-d_0)}{(2040)_{16}}, \quad 0 \leq d \leq 8.
\]

**Over-bounding \( \Pr[i, \lambda|d] \)**

In this section, we will define \( \alpha(i, d, \lambda) \) and show that it over-bounds \( \Pr[i, \lambda|d] \) for every \((d, i, \lambda)\).

First, for \( d = d_0 \), there are exactly \( 16 - d_0 \) \((A, B)\) pairs with at least one bit in \( c \), where \( c \) is any fixed element of \( C(d = d_0, i, \lambda) \). For \( i = 16 - d_0 \), every \((A, B)\) pair is isolated, thus \( \Lambda(i, d) = \emptyset \). Furthermore, the number of configurations in \( C(d = d_0, i, \emptyset) \) is equivalent to the number of ways of tossing \( i \) indistinguishable balls into 1020 distinguishable (ordered) bins such that no consecutive bins contain a ball, which is
\[
\binom{1020 + 1 - i}{i},
\]
times the number of ways of assigning balls to the selected bins, which is
\[
\binom{16 - d_0}{d_0} 2^{16-2d_0}.
\]

Similarly, for a fixed \( d_0 \), the total number of configurations is
\[
\binom{1020}{16 - d_0} \binom{16-d_0}{d_0} 2^{16-2d_0}.
\]

Therefore, we have
\[
\Pr[i = 16 - d_0, \emptyset|d = d_0] = \frac{\binom{1021-i}{i}}{\binom{1020}{i}},
\]
and we set
\[
\alpha(i = 16 - d_0, d = d_0, \emptyset) = \Pr[i = 16 - d_0, \emptyset|d = d_0].
\]

Note that \( i = 16 - d_0 - 1 \) describes the same case as \( i = 16 - d_0 \), since the remaining single bin is, by definition, also isolated. Therefore, this term is excluded from the summation over \( i \). For \( 0 \leq i < 16 - d_0 - 1 \), we overcount the number of configurations in \( C(i, d = d_0, \lambda = \lambda_a) \), \( \lambda_a \in \Lambda(i, d = d_0) \), as follows. For any \( c \in C(i, d = d_0, \lambda = \lambda_a) \), and \( \lambda_a = \{a_1, a_2, \ldots, a_l\} \), there are \( i+l \) distinct runs of \((A, B)\) pairs that have bits in \( c \). Now, there are at most \( \binom{1020}{i+l} \) choices for the starting position of each such run, and each set of valid starting points gives rise to a number of configurations described by a multinomial
coefficient \( M(i, \lambda) \), times the number of ways of assigning balls to the selected bins, which is
\[
\binom{16 - d_0}{d_0} 2^{16 - 2d_0}.
\]
By \( M(i, \lambda) \), we denote the multinomial coefficient corresponding to the collection of elements
\[
\{1, 1, \ldots, 1, a_1, a_2, \ldots, a_l\},
\]
i.e., it is the number of unique permutations of the elements in the set. Putting this together, we have
\[
\alpha(i, d = d_0, \lambda = \lambda_a) = \frac{(10^{i+l}) \cdot M(i, \lambda_a)}{(10^{20})} \binom{10^{i+l}}{16 - d_0}, \quad 0 \leq i < 16 - d_0 - 1.
\]

**Over-bounding \( \max_{c \in C_{(d,i,\lambda)}} \Pr[\text{bits in } c \text{ received in error}] \)**

For any \( c \in C_{(d,i,\lambda)} \), we over-bound the probability that the bits in \( c \) are received in error. For \( \lambda = \{a_1, a_2, \ldots, a_l\} \), by the arguments in Section 6.2.2, we have
\[
\Pr[\text{bits in } c \text{ received in error}, c \in C_{(d,i,\lambda)}] \leq [Q(1)]^i \prod_{j=1}^l Q(a_j) \triangleq X(i, \lambda).
\]

**Generalization to \( 2n \) balls, \( 1 \leq n \leq 8 \)**

Suppose that we consider the more general case of configurations with \( 2n \) balls. Then, it is straightforward to show that
\[
\frac{1}{\binom{10^{20}}{2n}} \sum_{c \in C} \Pr[\text{bits in } c \text{ received in error}] \leq \sum_{d=0}^n \sum_{i=0}^{2n-d} \sum_{\lambda \in \Lambda_{d,i}} \Pr[d] \cdot \alpha(i, \lambda | d) X(i, \lambda) \triangleq Z(n), \quad (6.2)
\]
where
\[
\Pr[d = d_0] = \frac{2^{2n-2d_0} \binom{10^{20}}{2n-d_0} \binom{2n-d_0}{d_0}}{\binom{10^{20}}{2n}}, \quad 0 \leq d \leq n,
\]
\[
\alpha(i = 2n - d_0, d = d_0, \emptyset) = \frac{10^{2n-i}}{\binom{10^{20}}{i}}, \quad i = 2n - d_0,
\]
\[
\alpha(i, d = d_0, \lambda = \lambda_a) = \frac{(10^{i+l}) \cdot M(i, \lambda_a)}{(10^{20})} \binom{10^{i+l}}{2n-d_0}, \quad 0 \leq i < 2n - d_0 - 1,
\]
and the definition of \( X(i, \lambda) \) is unchanged.
6.2.6 Returning to the general case

In the general case, a stall pattern involves bits in rows \( (R_1, R_2, R_3, R_4) \), \( R_i \in \{1, 2, \ldots, 512\} \), that are not necessarily within the same block interleaver. Furthermore, from the properties of \( \Pi_{\text{DQ}} \), if two rows are sufficiently separated, their errors are uncorrelated. For the moment, we’ll consider the case in which the four rows of \( s \) are sufficiently close to ignore the latter effect. By convention, \( R_i < R_j \) when \( i < j \).

Associated to each row \( R_i \) is the 2040-bit interleaver to which it belongs, and four indistinguishable balls to assign to the corresponding 2040 distinguishable (physical bit) bins. If \( m \) rows belong to the same interleaver, then \( 4m \) balls are assigned to their corresponding 2040 bins. The effective set of bins is the union of the bins corresponding to each of the four rows; the balls, however, are constrained to be assigned to those 2040 bins to which they ‘belong’.

Now, we argue that the expected probability (over the set of configurations) of the bits in \( s \) being received in error is upper bounded by the case considered in Section 6.2.5. Following those arguments, we assume the \( 2040 \cdot l \) (where \( l \) is the number of distinct interleavers in which the four rows are involved) physical-bit bins correspond to \( 1020 \cdot l \) temporally-consecutive \((A, B)\) pairs (although we make no assumption about their relative correspondence to the individual rows). Now, for the 2040 bins corresponding to \( R_1 \), we randomly assign 4 balls. Next, we do the same for each \( i \in \{2, 3, 4\} \). However, note that at each stage, the probability that a ball becomes a neighbour (to previously assigned balls) is no greater than when all 16 balls are assigned to a single set of 1020 temporally-consecutive \((A, B)\) pairs. That is, in the latter case, every ball can become involved in an existing run of errored \((A, B)\) pairs, but the same is not necessarily true in the former case. Therefore, we claim that analysis of assigning 16 balls to 1020 temporally-consecutive \((A, B)\) pairs provides an upper bound to the expected probability.

Relative distances of stall rows

Based on the properties of \( \Pi_{\text{DQ}}^{-1} \), when the difference between the maximum and minimum electrical skew (of the 10 electrical lanes) is less than or equal to 256 (in practice, it is expected to be less than 10), the maximum length between correlated errors (at the input to \( \Pi_{\text{ECC}}^{-1} \)) is 4608. From this, and the fact that \( \Pi_{\text{a}} \) acts on four consecutive rows, we will show that if \( |R_i - R_j| \geq 16 \), then no correlated error can occur between the bits of two such rows. We will use Fig. 6.3 in our argument. Note that between any two
rows with $|R_i - R_j| \geq 16$, there are three blocks of 2040 bits, each input to distinct interleavers. Therefore, the length between any bit in $R_i$ and any bit in $R_j$ is greater than $12 \cdot 510 = 6120$. Since this is greater than the maximum length between correlated errors, we have shown the desired result.

![Diagram](image_url)

Figure 6.3: Vertical lines represent the groupings of four rows into distinct 2040-bit interleavers, dots represent rows.

When $|R_i - R_j| \geq 16$, we can then assume the corresponding rows are ‘independent’ (i.e., no pair of bits, one from each row, can be involved in a correlated error event). Among the four rows in $s$, there exist 14 distinct cases that describe the potential for correlation for bits from distinct rows:

<table>
<thead>
<tr>
<th>Case</th>
<th>Rows</th>
<th>Rel. Freq.</th>
<th>$\mathcal{E}{\Pr[\text{bits in } c \text{ received in error}]}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$\emptyset$</td>
<td>0.689914</td>
<td>$\leq Z^4(2)$</td>
</tr>
<tr>
<td>2</td>
<td>$3 \rightarrow 4$</td>
<td>0.093888</td>
<td>$\leq Z^2(2)Z(4)$</td>
</tr>
<tr>
<td>3</td>
<td>$2 \rightarrow 3$</td>
<td>0.093888</td>
<td>$\leq Z^2(2)Z(4)$</td>
</tr>
<tr>
<td>4</td>
<td>$1 \rightarrow 2$</td>
<td>0.093888</td>
<td>$\leq Z^2(2)Z(4)$</td>
</tr>
<tr>
<td>5</td>
<td>$1 \rightarrow 2, 3 \rightarrow 4$</td>
<td>0.009273</td>
<td>$\leq Z^2(4)$</td>
</tr>
<tr>
<td>6</td>
<td>$2 \rightarrow 3, 3 \rightarrow 4$</td>
<td>0.004815</td>
<td>$\leq Z(2)Z(6)$</td>
</tr>
<tr>
<td>7</td>
<td>$1 \rightarrow 2, 2 \rightarrow 3$</td>
<td>0.004815</td>
<td>$\leq Z(2)Z(6)$</td>
</tr>
<tr>
<td>8</td>
<td>$2 \rightarrow 4$</td>
<td>0.004464</td>
<td>$\leq Z(2)Z(6)$</td>
</tr>
<tr>
<td>9</td>
<td>$1 \rightarrow 3$</td>
<td>0.004464</td>
<td>$\leq Z(2)Z(6)$</td>
</tr>
<tr>
<td>10</td>
<td>$1 \rightarrow 2, 2 \rightarrow 3, 3 \rightarrow 4$</td>
<td>0.000209</td>
<td>$\leq Z(8)$</td>
</tr>
<tr>
<td>11</td>
<td>$1 \rightarrow 2, 2 \rightarrow 4$</td>
<td>0.000100</td>
<td>$\leq Z(8)$</td>
</tr>
<tr>
<td>12</td>
<td>$1 \rightarrow 3, 3 \rightarrow 4$</td>
<td>0.000100</td>
<td>$\leq Z(8)$</td>
</tr>
<tr>
<td>13</td>
<td>$1 \rightarrow 3, 2 \rightarrow 4$</td>
<td>0.000098</td>
<td>$\leq Z(8)$</td>
</tr>
<tr>
<td>14</td>
<td>$1 \rightarrow 4$</td>
<td>0.000084</td>
<td>$\leq Z(8)$</td>
</tr>
</tbody>
</table>

The notation $2 \rightarrow 4$ implies $|R_4 - R_2| < 16$, which then implies both $|R_4 - R_3| < 16$ and $|R_3 - R_2| < 16$. The relative frequencies of the various cases are estimated via Monte
Carlo simulation. Finally, upper bounds on the expected probabilities are obtained using (6.2) and the row correlations implied in the second column of the Table. We are now in a position to upperbound the contribution of received minimal stall patterns (that lie entirely within one staircase block) to the expected bit-error-rates:

\[
\mathcal{E}\{\text{BER}_{\text{floor}}\} \leq \frac{16}{510 \cdot 512} |S| \left( 0.689914 \cdot Z^4(2) + 0.281664 \cdot Z^2(2) \cdot Z(4) + 0.009273 \cdot Z^2(4) \\
+ 0.018558 \cdot Z(2) \cdot Z(6) + 0.000591 \cdot Z(8) \right) = 4.77 \times 10^{-21}
\]

### 6.3 Conclusions

For a system model based on a real-world DQPSK receiver, we proposed an analysis technique to estimate the error floor of the G.709-compliant staircase code in the presence of correlated errors. By applying a time-varying pseudorandom interleaver of size 2040 to the output of the encoder, the error floor of the resulting system was shown to be less than \(4.77 \times 10^{-21}\).
Chapter 7

Conclusions

We now review the main contributions of this thesis and propose several directions for future research.

7.1 Summary of Contributions

This thesis contributed to the state-of-the-art in error-correcting codes for fibre-optic communication systems.

We showed that, despite fibre-nonlinearity, the communication channel induced by real-world fibre-optic communication systems is surprisingly similar to the classical linear Gaussian channel, and that digital backpropagation provides only modest benefits relative to a linear equalizer. For a WDM Raman-amplified system, we showed that the full benefits of backpropagation can be attained for step-sizes as large as 2.5 km.

Despite the fact that LDPC-like codes promise reliable communications at rates approaching capacity, we showed that product-like codes with syndrome-based decoding have an architecture whose dataflow requirements are more than two orders of magnitude smaller. Since the per-channel data rates of fibre-optic communication systems are increasing at a similar rate to improvements in high-speed electronics, the relevance of the dataflow benefits of product-like codes is likely to remain true for the near future.

These practical considerations motivated the search for product-like codes with error floors below $10^{-15}$ that maximize the net coding gain, and culminated in the invention of staircase codes, a hybrid of block and convolutional coding methods. A G.709-compliant staircase code was proposed, and FPGA-based simulation results were presented to validate its performance within 0.52 dB to the Shannon limit. An error floor analysis
technique was introduced, and the error floor of the G.709-compliant staircase code was shown to be below $10^{-20}$.

To approach the maximal spectral efficiency of fibre-optic systems, a pragmatic coded-modulation scheme was proposed. Using a staircase code with hard-decision bit-interleaved coded-modulation (HD-BICM), performance to within 0.19 bits/s/Hz of the capacity of HD-BICM was achieved for linearly equalized systems, and to within 0.28 to 0.41 bits/s/Hz for digitally backpropagated systems, with error floors below $10^{-25}$.

Finally, we studied a real-world DQPSK receiver, for which correlated errors occur. We proposed a system model and an analysis technique to estimate the error floor of the G.709-compliant staircase code, in the presence of a time-varying pseudorandom interleaver at the output of the encoder. For an interleaver of size 2040, the error floor of the resulting system was shown to be less than $4.77 \times 10^{-21}$.

### 7.2 Directions for Future Research

We now propose several directions in which the work presented herein could be extended.

On the subject of the optical channel, this work ignored polarization-dependant effects, due to the computational complexity associated with their inclusion in a capacity analysis. Is it sufficient to compensate the linear polarization effects (such as polarization-mode dispersion), that is, is the capacity limited by inter-channel nonlinearities (which cannot be compensated), as it is when polarization-dependant effects are ignored? Looking beyond backpropagation and (assumed) memoryless channels, is the true capacity of the fibre-optic channel larger than that predicted by existing methods?

Staircase codes provide near-capacity performance for high rates, but the performance gap increases as the rate decreases. Is this fundamental to hard-decision decoded codes? Interestingly, LDPC codes with Gallager B decoding also exhibit the same effect, even though LDPC codes with soft-decision decoding can be made to approach capacity at arbitrary rates.

For high-spectral efficiency communications, it is interesting to investigate the benefits of shaping in the context of the pragmatic hard-decision BICM system. Using a mixed Gray/set-partitioning labeling, an overall Gray-labeling can be maintained, and a shaping algorithm (such as trellis shaping) applied to the set-partioned bit-channels.

The analysis of the effects of correlated errors in a DQPSK system assumed an interleaver of size 2040. It would be interesting to present an analysis that allows the
determination of the smallest interleaver for a given target error floor.
Appendix A

Efficient decoding for triple-error correcting binary BCH codes

This section summarizes known techniques for efficiently decoding triple-error correcting binary BCH codes. The formulas for the reciprocal error-locator polynomial coefficients are given in [21], and the methods for solving quadratic and cubic equations are described in [23, 22].

 Syndromes

Transmit \( c(x) \), satisfying \( c(\alpha) = c(\alpha^3) = c(\alpha^5) = 0 \), where \( \alpha \) is a primitive element of \( F_q \) (\( q = 2^m \)) and receive \( r(x) = c(x) + e(x) \). Compute the syndromes

\[
S_1 = r(\alpha) = e(\alpha), \\
S_3 = r(\alpha^3) = e(\alpha^3), \\
S_5 = r(\alpha^5) = e(\alpha^5),
\]

and the quantities

\[
D_3 = S_1^3 + S_3, \\
D_5 = S_1^5 + S_5.
\]

Four cases arise:

\[
t = 0 : \quad e(x) = 0 \\
t = 1 : \quad e(x) = x^i \quad \quad 0 \leq i \leq q - 2 \\
t = 2 : \quad e(x) = x^i + x^j \quad 0 \leq i < j \leq q - 2 \\
t = 3 : \quad e(x) = x^i + x^j + x^k \quad 0 \leq i < j < k \leq q - 2
\]
Appendix A. Efficient decoding for triple-error correcting binary BCH codes

These relations distinguish the cases:

\[ t = 0 \colon S_1 = S_2 = S_3 = 0 \]
\[ t = 1 \colon S_1 \neq 0, D_3 = D_5 = 0 \]
\[ t = 2 \colon S_1 \neq 0, D_3 \neq 0, S_1 D_5 = S_3 D_3 \]
\[ t = 3 \colon D_3 \neq 0, t \neq 2 \]

Reciprocal Error-Locator Polynomial

Define the reciprocal error-locator polynomial \( \tilde{\sigma}(x) \) as follows:

\[ t = 1 : \tilde{\sigma}(x) = x + \alpha^{t} \]
\[ t = 2 : \tilde{\sigma}(x) = (x + \alpha^{t})(x + \alpha^{j}) \]
\[ t = 3 : \tilde{\sigma}(x) = (x + \alpha^{t})(x + \alpha^{j})(x + \alpha^{k}) \]

The coefficients of the \( \tilde{\sigma}(x) \) are given as:

\[ t = 1 : \tilde{\sigma}(x) = x + S_1 \]
\[ t = 2 : \tilde{\sigma}(x) = x^2 + S_1 x + D_3/S_1 \]
\[ t = 3 : \tilde{\sigma}(x) = x^3 + S_1 x^2 + b x + S_1 b + D_3 \]

where

\[ b = (S_1^2 S_3 + S_5)/D_3. \]

Note that, in case of \( t = 2 \), all of the coefficients of \( \tilde{\sigma}(x) \) are nonzero.

Reductions

In case \( t = 0 \) and \( t = 1 \), it is trivial to determine the error location. (For \( t = 1 \), \( L_1 = S_1 \).)

Quadratics

If \( f_X(x) = x^2 + ax + b \) with \( a \neq 0 \), substitute \( x = ay \) to obtain

\[ f_Y(y) = a^2(y^2 + y + b/a^2). \]

If \( f_Y(r) = 0 \) then \( f_X(ar) = 0 \). Thus the problem of finding roots of \( f_X(x) \) reduces the problem of finding roots of the suppressed quadratic \( f_Y(y) \), which can be solved by table-lookup.
Appendix A. Efficient decoding for triple-error correcting binary BCH codes

In our case \( a = S_1, b = D_3/S_1 \), hence

\[
b/a^2 = D_3/S_1^3.
\]

Denoting the table-lookup operations by \( g_1(\cdot) \) and \( g_2(\cdot) \), we obtain reciprocal error-locators

\[
L_1 = a \cdot g_1(D_3/S_1^3), \quad L_2 = a \cdot g_2(D_3/S_1^3).
\]

Cubics

If \( f_X(x) = x^3 + ax^2 + bx + c \) substitute \( x = y + a \) to obtain

\[
f_Y(y) = y^3 + (a^2 + b)y + ab + c.
\]

Note that \( yf_Y(y) \) is a linearized polynomial with respect to \( F_2 \) and hence the set of zeros of \( yf_Y(y) \) is a vector space over \( F_2 \). In particular, the roots of \( yf_Y(y) \), if distinct, are of the form \( \{0, r_1, r_2, r_1 + r_2\} \). Thus only \( r_1 \) and \( r_2 \) need to be stored in the lookup table.

Two cases arise, depending on the value of \( a^2 + b = D_5/D_3 \).

If \( D_5 = 0 \), so that \( a^2 + b = 0 \), then \( f_Y(y) = y^3 + ab + c \), and the roots can be found by finding the cube roots of \( ab + c \). In our case, \( a = S_1 \) and \( ab + c = D_3 \). Denoting the cube root table lookup operations by \( h_1(\cdot) \) and \( h_2(\cdot) \), we have

\[
L_1 = S_1 + h_1(D_3), \quad L_2 = S_1 + h_2(D_3), \quad L_3 = S_1 + L_1 + L_2.
\]

If \( D_5 \neq 0 \), so that \( a^2 + b \neq 0 \), substitute \( y = (a^2 + b)^{1/2}z \) to obtain

\[
f_Z(z) = (a^2 + b)^{3/2}(z^3 + z + (ab + c)/(a^2 + b)^{3/2}).
\]

The roots of the suppressed cubic \( f_Z(z) \) can be found by table-lookup. Note that

\[
\frac{ab + c}{(a^2 + b)^{3/2}} = \left( \frac{D_5^5}{D_3^3} \right)^{1/2}.
\]

Thus, if we denote the two table lookup operations as \( j_1(\cdot) \) and \( j_2(\cdot) \), we have

\[
r_1 = j_1 \left[ \left( \frac{D_5^5}{D_3^3} \right)^{1/2} \right], \quad r_2 = j_2 \left[ \left( \frac{D_5^5}{D_3^3} \right)^{1/2} \right],
\]

and

\[
L_1 = S_1 + (D_5/D_3)^{1/2}r_1, \quad L_2 = S_1 + (D_5/D_3)^{1/2}r_2, \quad L_3 = S_1 + L_1 + L_2.
\]
Combining the Quadratic and Cubic Cases

The quadratic case can be combined with the cubic case by introducing an additional error, and then ignoring the additional error location. This can be done, for example, by introducing a zero root (multiplying the quadratic $f(x)$ by $x$), and then manipulating the resulting cubic into a suppressed cubic as above.

An equivalent formulation would be to introduce an extra error in position $S_1$ (which, in case of two errors, is certainly *not* one of the error locations). Let $S_1$, $S_3$, $S_5$, $D_3$, and $D_5$ be as given (and resulting in a quadratic). Introducing a new error in position $S_1$ results in new “primed” values

$$S'_1 = S_1 + S_1 = 0$$
$$S'_3 = S_3 + S_1^3 = D_3$$
$$S'_5 = S_5 + S_1^5 = D_5$$
$$D'_3 = (S'_1)^3 + S'_3 = D_3$$
$$D'_5 = (S'_1)^5 + S'_5 = D_5$$

These “primed” values (which are already available without further computation) can be passed to the cubic solver, which would return three error locations including $S_1$. The $S_1$ location must be ignored.
Bibliography


