

Secure Coding Schemes and Code Design for the Wiretap Channel

Ke Zhang

July 1, 2014

Dedicated to My parents

Jury Members

Dr. António Beça Gonçalves Porto

Professor Catedrático

Faculdade de Ciências da Universidade do Porto (Presidente)

Dr. Marco Alexandre Cravo Gomes

Professor Auxiliar

Departamento de Engenharia Eletrotécnica e de Computadores

Faculdade de Ciência e Tecnologia

Universidade de Coimbra

Dr. Atílio Manuel da Silva Gameiro

Professor Associado

Departamento de Eletrónica

Telecomunicações e Informática

Universidade de Aveiro

Dr. Luís Filipe Coelho Antunes

Professor Associado

Faculdade de Ciências

Universidade do Porto

Dr. Pedro Miguel Aives Brandão

Professor Auxiliar

Faculdade de Ciências

Universidade do Porto

Dr. Miguel Raúl Dias Rodrigues

Professor Associado Convidado

Faculdade de Ciências da Universidade do Porto (Orientador)

Acknowledgments

This dissertation would not be possible without the contributions of several people. First and foremost, I would like to thank from my heart to my supervisor, Dr. Miguel Rodrigues, whose encouragement, guidance and support from the initial to the final stages has enabled me to develop understanding of the subject. His perfectionism and passion for research are always inspirational. I am especially grateful for all the opportunities that he provided me.

I am especially indebted to Prof. Martin Tomlinson in University of Plymouth, who has contributed to the supervision of my research work during the four years Ph.D period. I would like to thank Martin for his mentoring, encouragement and constant support.

My stay with the group of Prof. Martin Tomlinson and Prof. Mohamed zaki Ahmed at the University of Plymouth was a great experience. The six months visiting research period in Plymouth have led to various research directions that considerably shaped this dissertation. I thank Professor Ahmed for receiving me and nice discussions with me, and the group members Mubarak Jibril, Nadia Awad and Salah Al-Hassan for such a nice environment.

I am very grateful to my collaborators, Munnujahan Ara, Hugo Reboredo and Francesco Renna, who always offered help to me. I would like to thank all my lab mates who have made a comfortable working environment. My gratefulness to Alex and Elimary, who have been very kind and helpful throughout my stay at DCC (Departamento de Ciência de Computadores) and extended all administrative support.

I would also like to thank to IT (Instituto de Telecomunicações) and FCT (Fundação para a Ciência e a Tecnologia) for the financial support during my PhD¹.

As always, my greatest gratitude is to my parents and my boyfriend. I dedicate this

¹This work was supported by Fundação para a Ciência e a Tecnologia (the Portuguese National Science Foundation) under grant SFRH/BD/69852/2010. Part of this work was also supported by Fundação para a Ciência e a Tecnologia, Portugal through the research project PTDC/EEATEL/100854/2008 and CMU-PT/SIA/0026/2009.

dissertation to them with love.

Finally, I would like to say ‘Thank you!’ to everyone in Departamento de Ciência de Computadores, Faculdade de Ciências da Universidade do Porto. Your kindnesses have made Porto a warm home for me.

Resumo

Como as redes sem fio continuam a florescer, a nível mundial, as questões de segurança nessas redes desempenham um papel cada vez mais importante. Ao contrário dos serviços de segurança tradicionais, que se baseiam em esquemas de criptografia na camada de aplicação, a abordagem baseada em teoria da informação abre uma direção promissora para atingir comunicação sigilosa na camada física. Esta tese baseia-se em fundamentos de segurança da teoria da informação para analisar e desenvolver novos esquemas de codificação confiáveis e seguros, e novas técnicas de “code design” na camada física.

Em primeiro lugar, é analisada a confiabilidade e a segurança de códigos ortogonais, utilizando uma estrutura de codificação de Wyner, na presença de um canal de escuta Gaussiano. Tanto os resultados analíticos como os de simulação mostram que com os códigos ortogonais se pode conseguir uma comunicação confiável e secreta quando o canal de escuta é Gaussiano com degradação, em regime de banda larga.

Em segundo lugar, é proposto um codificador simplificado para o esquema de “syndrome coding”, o que torna o esquema prático para os códigos em que o comprimento dos bits de paridade é longo. É descrita uma forma recursiva eficiente para a avaliação da ambiguidade no esquema de “syndrome coding” tradicional, que fornece indicações sobre formas de construir melhores códigos. A melhor técnica de “code design” é proposta para o esquema de “syndrome coding”, e é avaliada com uma métrica de segredo da informação: a equivocação. Mostra-se que os melhores códigos de equivocação têm um melhor desempenho no sigilo para o esquema de “syndrome coding” do que os códigos conhecidos anteriormente.

Em terceiro lugar, é estudado o desempenho, em termos de segurança, dos códigos aleatórios no esquema de “syndrome coding” tradicional. Além de uma análise de Monte Carlo, é proposta uma forma teórica para avaliar a equivocação média de códigos aleatórios lineares. Mostra-se que esta forma teórica é muito precisa, e para códigos longos é muito mais rápida e prática do que a simulação de Monte Carlo. Os resultados mostram que, quando o comprimento do código aumenta, a taxa de informação se aproxima da

capacidade de sigilo .

Em quarto lugar, são propostos dois esquemas de “syndrome coding” baseados em cadeias para um canal com escutas binário e simétrico. Mostra-se que com o esquema baseado em cadeias no qual se utilizam uma função de “hash”, para um valor suficientemente grande do tamanho da cadeia, se consegue um elevado nível de segurança.

Por fim, é analisada a equivocação do conhecido sistema de criptografia de chave pública McEliece em diferentes ataques, o que mostra que há considerável fuga de informações para um espião. É proposta uma modificação ao sistema de codificação de chave pública de McEliece para reduzir a fuga de informação, que apresenta uma fuga nula no caso de um ataque de Brickell, sendo o custo para esse acréscimo de segurança uma taxa de comunicação inferior.

Abstract

As wireless networks continue to flourish worldwide, the issues of security play an increasingly important role in those communication networks. Unlike the traditional security services, which are based on some cryptographic schemes in the application layer, the information theoretic security approach opens a promising direction to achieve the secrecy communication in the Physical layer. This PhD thesis builds upon the information-theoretic security foundations to analyse and develop novel reliable and secure coding schemes and the code design technique in the Physical layer.

First, the reliability and the security of the orthogonal codes using in Wyner's coding structure have been analysed in the Gaussian wiretap channel. Both the analytic and the simulation results show that the orthogonal codes can achieve reliable and perfect secret communication in degraded Gaussian wiretap channel in wideband regime.

Second, a simplified encoder for the syndrome coding scheme has been proposed, which makes the scheme practical for the codes with long lengths of the parity bits. An efficient recursive evaluation of the equivocation for the traditional syndrome coding scheme has been described, which gives an insight to construct the best codes. The best code design technique has been proposed for the syndrome coding scheme, as measured by an information secrecy metric: the equivocation. It is shown that the Best Equivocation Codes exhibit better security properties than the best known codes, when used in conjunction with syndrome coding schemes.

Third, the security performance of the random codes in the traditional syndrome coding scheme has been studied. Besides the Monte Carlo analysis, a theoretical way has been proposed to evaluate the average equivocation of the random linear codes. It is shown that random codes with large values of m , the number of parity bits, may be analysed theoretically even though Monte Carlo analysis is impractical. The results give that as the code length increases, the information rate is getting closer to the secrecy capacity.

Fourth, two chain based syndrome coding schemes have been proposed in the binary symmetric wiretap channel. It has been shown that the chain based syndrome coding

scheme with a hash function can achieve high security level with some large enough value of the chain depth.

Finally, the equivocation of the popular McEliece public key encryption system has been analysed in the presence of different attacks, showing that there can be considerable information leakage to the eavesdropper. A modified McEliece public-key cryptosystem has been proposed to reduce the information leakage, which exhibits zero information leakage under Brickell's attack and the price for this higher security is a lower information rate.

Contents

Acknowledgments	iii
Resumo	v
Abstract	vii
Contents	ix
List of Figures	xiv
List of Tables	xvii
Abbreviations	xviii
Notation	xix
1 Introduction	1
1.1 Motivation	1
1.2 Thesis organization	5
1.3 Thesis contribution	7
2 Information-Theoretic Security and Coding for Secure Communications	11
2.1 Information theoretic security	11
2.1.1 Information theoretic security versus computational security . . .	11

2.1.2	Shannon's notion of perfect secrecy	14
2.1.3	Wiretap channel model	15
2.1.4	Reliability and security measures	17
2.1.5	Secrecy capacity	17
2.1.5.1	Binary erasure wiretap channel	18
2.1.5.2	Binary symmetric wiretap channel	19
2.1.5.3	Gaussian wiretap channel	20
2.2	Some basic coding schemes and code design for the wiretap channels . .	22
2.2.1	Linear block codes	23
2.2.2	Wyner's coding scheme	24
2.2.3	Syndrome coding scheme	26
2.2.3.1	Encoder	27
2.2.3.2	Legitimate decoder	27
2.2.3.3	Eavesdropper decoder	27
3	Orthogonal Codes for Secure Communication in the Wideband Regime	29
3.1	Introduction	29
3.2	The Gaussian wiretap channel in the wideband regime	30
3.2.1	Tradeoff C_s vs. ξ_b/N_0 in the wideband regime	30
3.2.2	Wyner's coding scheme in the wideband regime	32
3.3	Code construction	34
3.4	Error probability analysis	35
3.5	Equivocation analysis	36
3.6	Simulation results	37
3.7	Conclusions	40
4	Syndrome Coding Based on Best Equivocation Code Construction	41

4.1	Introduction	41
4.2	Background of code design	43
4.3	Syndrome coding scheme without look-up table	46
4.3.1	Encoder	47
4.3.2	Legitimate decoder	47
4.3.3	Eavesdropper decoder	48
4.4	Calculation of the equivocation of a syndrome coding scheme	48
4.4.1	Analysis of the syndrome probability distribution	49
4.4.1.1	Traditional evaluation method of the syndrome probability distribution	49
4.4.1.2	Recursive evaluation of the syndrome probability distribution	50
4.4.2	Code construction by code extension	51
4.5	Code design for syndrome coding	52
4.6	Results	54
4.7	Conclusions	59
5	Syndrome Coding Based on Random Linear Binary Codes	63
5.1	Introduction	63
5.2	Random linear binary codes in syndrome coding	65
5.2.1	Random codes	65
5.2.2	Equivocation analysis	65
5.3	Average equivocation evaluation of random codes in syndrome coding . .	66
5.3.1	Using Monte Carlo experiments to derive statistics of random codes	66
5.3.2	Theoretical analysis	67
5.3.3	Examples	70
5.4	Simulation results	71

5.5	Conclusions	77
6	A Chain Based Syndrome Coding Scheme	79
6.1	Introduction	79
6.2	A chain based syndrome coding scheme	80
6.2.1	The encoder and the decoder description	80
6.2.1.1	Encoder	80
6.2.1.2	Legitimate decoder	82
6.2.1.3	Eavesdropper decoder	82
6.2.2	Decoder output equivocation analysis	83
6.3	A chain based syndrome coding scheme with a hash function	84
6.3.1	The encoder and the decoder description	84
6.3.1.1	Encoder	84
6.3.1.2	Legitimate decoder	85
6.3.1.3	Eavesdropper decoder	85
6.3.2	Decoder output equivocation analysis	86
6.3.3	Simulation results	87
6.4	Conclusions	88
7	A Modified McEliece Public Key Encryption System with a Higher Security Level	91
7.1	Introduction	91
7.2	McEliece Public Key Encryption System	93
7.2.1	System Description	93
7.2.1.1	Key generation	93
7.2.1.2	Message encryption	93
7.2.1.3	Message decryption	94

7.2.2	Popular Attacks	94
7.2.3	Equivocation Analysis	95
7.3	Modified McEliece Public Key Encryption System with Higher Security Level	99
7.3.1	System Description	99
7.3.1.1	Key generation	99
7.3.1.2	Message encryption	99
7.3.1.3	Message decryption	101
7.3.2	Equivocation Analysis	101
7.3.3	Information Rate Analysis	104
7.4	Conclusions	104
8	Conclusions and Future Work	107
8.1	Conclusions	107
8.2	Future work	109
	References	113
A	Proof of the MAP Decoder for the Syndrome Coding Scheme	123
B	Best Equivocation Codes Table	129

List of Figures

1.1	Layered protocol architecture	3
2.1	Channel model in cryptographic	13
2.2	Information theoretic security	14
2.3	Shannon's cryptosystem	14
2.4	The wiretap model	16
2.5	The Binary erasure wiretap model	19
2.6	The Binary symmetric wiretap model	20
2.7	Gaussian wiretap channel	20
2.8	Typical transmission system	22
2.9	Simplified model of a coded system	23
2.10	Binning structure in Wyner's coding scheme	25
3.1	Tradeoff between C_s and ξ_b/N_0 for various values of α . Solid lines represent the true curve. Dashed lines represent the affine expansion.	31
3.2	Codeword error probability between the legitimate parties for $\alpha = 0.08$	38
3.3	Eavesdropper equivocation lower bound for $\alpha = 0.08$. Solid lines correspond to the equivocation lower bound. Dashed lines correspond to the message entropy.	39
3.4	Normalized eavesdropper equivocation gap upper bound γ for $\alpha = 0.08$. γ is obtained by using the equivocation lower bound in $\frac{H(W)-H(W \mathbf{Y}_E)}{H(W)}$	39

4.1	Comparison of the normalised equivocation of best equivocation and best known codes having 26 parity bits	61
4.2	Normalised equivocation vs. information rate for Best Equivocation Codes as a function of m	61
5.1	The theoretical pmf of $\sum_{w=1}^n p(S)_w$, when $p(S)_1 = 0$	72
5.2	The theoretical pmf of $\sum_{w=1}^n p(S)_w$, when $p(S)_1 = \alpha \cdot (1 - \alpha)^{n-1}$	72
5.3	The simulated pmf of $\sum_{w=1}^n p(S)_w$, when $p(S)_1 = 0$	73
5.4	The simulated pmf of $\sum_{w=1}^n p(S)_w$, when $p(S)_1 = \alpha \cdot (1 - \alpha)^{n-1}$	73
5.5	The cumulative distribution probability of $\sum_{w=1}^n p(S)_w$, when $p(S)_1 = 0$	74
5.6	The cumulative distribution probability of $\sum_{w=1}^n p(S)_w$, when $p(S)_1 = \alpha \cdot (1 - \alpha)^{n-1}$	74
5.7	The normalised average equivocation vs. the code length for the random codes ($m = 20$, sample number=10000)	75
5.8	The standard deviation of the normalised equivocation vs. the code length for the random codes ($m = 20$)	76
5.9	The normalised average equivocation vs. the information rate for the random codes with different values of m	76
6.1	The encoder model of the chain based syndrome coding scheme	81
6.2	The encoder model of the chain based syndrome coding scheme with a hash function	85
6.3	Normalised equivocation vs. α for the (14,8) repeated Hamming code	88
6.4	KL distance vs. α for the (14,8) repeated Hamming code	89
7.1	Wiretap scenario	93
7.2	Normalised equivocation vs. information rate (k/n)	97
7.3	Work factor vs. information rate (k/n)	97
7.4	Main blocks of the Modified McEliece cryptosystem	99

8.1	Code concatenation model for BSC	110
-----	--	-----

List of Tables

3.1	Tradeoff between rate, reliability and security	40
4.1	Best Equivocation Codes that achieve 90% secrecy in syndrome coding for $\alpha = .05$	56
4.2	Well packed BKC	60
5.1	w-error events statistic ($\alpha = 0.05$)	71
7.1	Information rate for traditional McEliece system and modified McEliece system	105
B.1	Best Equivocation Code Table (BEC)	129

Abbreviations

GSM	Global System For Mobile Communication
MAC	Multiple Access Channel
SNR	Signal-to-Noise Ratio
LDPC	Low Density Parity Check
DMC	Discrete Memoryless Channel
BEWC	Binary Erasure Wiretap Channel
BSWC	Binary Symmetric Wiretap Channel
i.i.d.	Independent and identically distributed
MAP	Maximum a Posteriori Probability
BEC	Best Equivocation Codes
BKC	Best Known Codes
AWGN	Additive White Gaussian Noise
KL	Kullback-Leibler
pmf	probability mass function
CDF	Cumulative Distribution Function
KS	Kolmogorov-Smirnov

Notation

$\mathbf{H}^T$	Conjugate transpose (Hermitian) of matrix $\mathbf{H}$.
$\mathbb{E}[\cdot]$	Statistical expectation.
$\log_2[\cdot]$	Logarithm in base 2.
$p_z(\cdot)$	Z transform.
C_M	Channel capacity of the main channel.
C_E	Channel capacity of the eavesdropper channel.
C_s	Secrecy capacity.
$I(X; Y)$	Mutual informations between the random variables X and Y .
S_c	Channel output equivocation.
S_d	Decoder output equivocation.
R_e	Equivocation rate.
R	Information rate.
$\mathbf{I}$	The identity matrix.
P_e	Codeword error probability.
$[x]^+$	Positive value of x , i.e. $\max(0, x)$.
α	Cross-over probability of the binary symmetric channel.
$a \rightarrow 0$	The limit of a approaches zero.
$>$	Greater than.
$<$	Less than.
$\geq$	Greater than or equal to.
$\leq$	Less than or equal to.
$\in$	Belongs to.
$l!$	The factorial of l .
$\min(x, y)$	The smallest value function.
$\lfloor x \rfloor$	The largest integer not greater than x .

Chapter 1

Introduction

1.1 Motivation

As the wireless communication networks flourish worldwide, security is becoming more and more important in wireless communications. Because of the inherent openness of wireless media, security in the wireless communications usually includes these three issues: (i) confidentiality, (ii) authentication, and (iii) integrity [1]. Confidentiality guarantees that the legitimate receiver obtains the transmitted messages successfully, while keeping the eavesdroppers' complete ignorance about the useful information. Authentication ensures that the receiver is able to identify the sender of the message. Integrity relates that the original information transmitted by the legitimate party is not modified by other malicious attackers during its transmission [2].

Historically, the solutions for those security issues are arranged on different layers in the layered protocol architecture (Figure 1.1). For example, spread-spectrum modulation techniques in the physical layer are employed to increase resistance to natural interference, noise and jamming, to prevent detection. Authentication mechanisms are implemented in the link layer to prevent unauthorized access. Cryptographic schemes are used in the application layer to guarantee the eavesdroppers' ignorance about the useful information.

Regarding the confidentiality, currently most of the security services of the wireless communications are based on some cryptographic schemes, in which security is always viewed as an independent feature addressed above the physical-layer, and all widely used cryptographic protocols (e.g. AES-CCMP for 802.11, A5 for GSM) are designed and

implemented, assuming that the physical-layer has already been established providing an error free link [3], which is guaranteed by the channel coding schemes in the physical layer. Conventional techniques for achieving security in communication networks are based on cryptographic encryption, which assume the hardness of computational problems in number theory such as the integer-factoring problem (e.g. RSA [4]) or the problem of computing discrete logarithms in certain finite cyclic groups (e.g. Diffie-Hellman [5]). Such a cryptosystem falls under the domain of *computational security*. In encryption, a transmitter uses a key to encrypt source information into ciphertext. The legitimate receiver extracts the original information from the ciphertext by a corresponding key. If an eavesdropper has access to the ciphertext, he can not obtain the source information without the corresponding decryption key, e.g. for RSA, if the key is large enough, only someone with knowledge of the key can feasibly decode the message. Encryption includes two principal types of algorithms: public-key encryption algorithms and secret-key encryption algorithms. For example, RSA is one of the first practicable public-key cryptosystems and is widely used for secure data transmission [4]. In such a cryptosystem, the encryption key is public and differs from the decryption key which is kept secret. In secret-key cryptosystem, e.g. AES, the same key is used for both encrypting and decrypting the data [6]. Compared to public-key algorithms, secret-key algorithms are computationally efficient, while presenting challenges for key management, such as secure key storage and distribution. Public-key algorithms exhibit a simple key management procedure, but require considerable computational resources, are not provably perfectly secure and are vulnerable to the man-in-the-middle attack [2, 7].

Up to date, no practical cipher has been proven computationally secure. It is an inherent fact that computational security can only hold under certain assumptions on the eavesdropper's computer resources and also relies on the assumptions of the intractability of some computational problems. Moreover, a computationally infinitely powerful opponent could presumably break these cryptosystem by exhaustive search over the key space [8]. On the contrary, *information theoretical security*, which is based on information theory rather than complexity theory, does not require those assumptions on the computer resources and is very promising.

An information-theoretic treatment of a simple cryptosystem was initiated by Shannon in [9]. The important notion of the equivocation was introduced to measure the uncertainty of the eavesdropper about the transmitted message, which is intimately associated with the security of the communication. The study of the information theoretic security is based on Shannon's idea of the perfect secrecy, which is defined as the case of that when

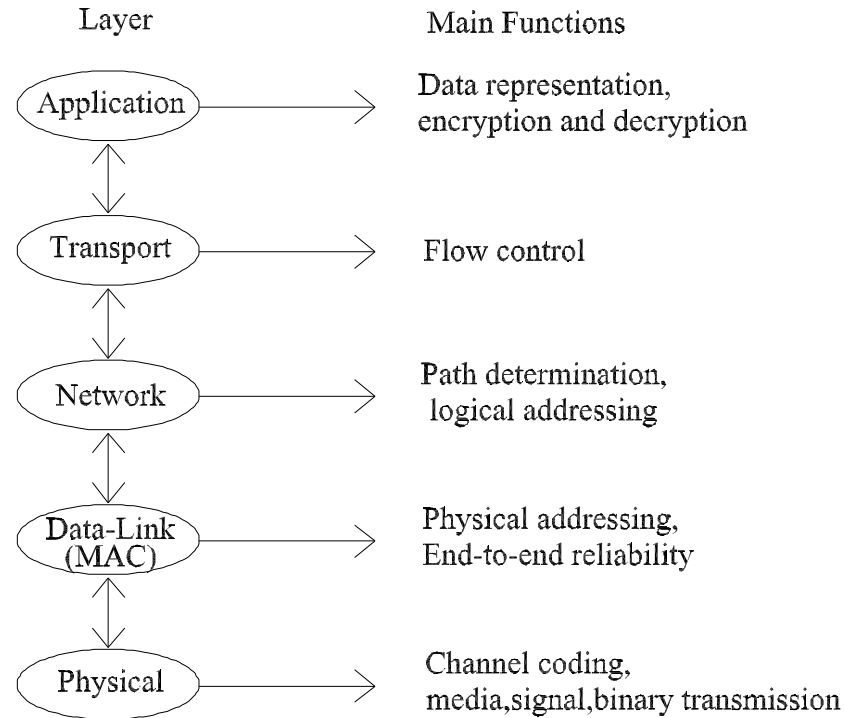


Figure 1.1: Layered protocol architecture

the equivocation of the eavesdropper is equal to the entropy of the transmitted message.

Traditionally, the coding schemes in the physical layer are only to guarantee the reliability of the communication, regardless of the security. In 1970's, the physical layer security approach has emerged through the works of Wyner [10], Csiszár and Körner [11], who demonstrated that messages can be transmitted both reliably and security without using an encryption key, by solely exploiting the characteristics of the physical channel. Compared to cryptographic encryption and cryptosystems, information theoretic security approaches to guarantee secrecy have the advantages of eliminating the key management issues, such as: having a significantly lower complexity and savings in resources. Furthermore, the information theoretic security approaches can still achieve provable security for those very powerful eavesdroppers who possess unlimited computational resources, full knowledge of the coding and decoding algorithms and the channel state information.

From that time, the information theoretic security approach (also known as the physical layer security) to achieve secure communication has led a promising new direction to strengthen the security of digital communication systems. The basic principle of

information-theoretic security - widely accepted as the strictest notion of security - is to exploit the randomness of communication channels (including noises and channel fluctuations due to fading) and the difference between the channel to a legitimate receiver and the channel to an eavesdropper to guarantee that the messages cannot be decoded by the eavesdropper on the communications medium [2].

The wiretap channel, which was introduced by Wyner [10], is a basic physical-layer model that captures the essence of communication security. In this model, a transmitter wishes to send confidential information to a legitimate receiver in the presence of an eavesdropper. Wyner characterized the rate-equivocation region of the degraded wiretap channel as well as its secrecy capacity. One of the most important secure coding schemes in the physical layer is Wyner's coding scheme [10]. In this approach, a transmitter intentionally adds structural randomness to prevent potential eavesdroppers and attackers from intercepting useful information while guaranteeing that a legitimate receiver can obtain the information. The syndrome coding scheme was proposed by Wyner in 1978 for a specific wiretap channel where the main channel is error-free and the eavesdropper channel is binary symmetric channel [12]. Wyner showed that as long as the size of the syndrome space is chosen to be smaller than the Shannon entropy of the binary symmetric channel, there exist codes that leak a vanishing proportion of information to the eavesdropper as the code length extends to infinity. Later on, the syndrome coding scheme became an important secure coding scheme in the physical layer and has been widely employed by lots of researchers [13–16].

More recently, there has been a renewed interest in information-theoretic security, arguably due to the work of Maurer [17], who proved that even when the legitimate users have a worse channel than the eavesdropper, it is possible to generate a secret key through public communication over an insecure yet authenticated channel. The advent of wireless communications, which are particularly susceptible to eavesdropping because of the broadcast nature of the transmission medium, has also motivated a closer analysis of the secrecy potential of wireless networks. Of particular relevance, Barros and Rodrigues [18] characterized in detail the outage secrecy capacity of slow fading channels, showing that fading alone guarantees information-theoretic security even when the eavesdropper has a better signal-to-noise ratio (SNR) on average than the legitimate receiver without the need for public communication over a feedback channel. Li *et al.* [19], Liang *et al.* [20], and Gopala *et al.* [21] also characterized the ergodic secrecy capacity of wireless fading channels.

To date, research on physical-layer security has mainly sought to establish the fundamental reliability and security limits of various communications channels, by computing the secrecy capacity of various single-user channel models or the secrecy capacity region of some multi-user channel models [22–25]. Moreover, a number of attempts have been made in designing code scheme for the secure communication over the wiretap channels. For example, Wei [26] studies channel coding for reliable and secure communications over wiretap channels, showing how to encode secret information using cosets of certain linear block codes. From an arbitrary given channel code over a discrete or Gaussian memoryless channel, Hayashi [27] constructs a wiretap code with the strong security. Other coding schemes for wiretap channels exploit low-density parity-check (LDPC) codes [28–30] or polar codes [31–34]. More recently, Belfiore and Oggier propose a wiretap lattice code for Gaussian wiretap channels [35] and the Rayleigh fading Wiretap Channel [36], introducing the notion of secrecy gain as a new code design criterion.

Currently, the coding schemes designed for the wiretap channels have been employed for enhancing the security of the communication in the following three ways: (i) the wiretap channel coding scheme is used to enhance the protection of the secret key employed in the encryption algorithm [37, 38]; (ii) the wiretap channel coding scheme is employed as the encoder of the message to enhance the security of the transmitted codewords in a noisy eavesdropper channel [39, 40]; (iii) the wiretap channel coding is employed to increase the security of a Data Encryption Standard (DES) block cipher operating in the ciphertext feedback mode [41–44].

In spite of the numerous of these contributions motioned above, the general problem of design of physical-layer coding schemes for secure communication over wiretap channels is still open at large.

1.2 Thesis organization

In this dissertation, we propose three schemes—the orthogonal coding scheme, the chain based syndrome coding scheme and the modified McEliece public-key cryptosystem—to achieve a higher security level in wiretap channels. Moreover, we make a detailed research on the traditional syndrome coding scheme. We present a design technique to produce the best codes for the syndrome coding scheme as measured by an information theoretic secrecy metric, the equivocation rate. We also propose a theoretical analysis

method to calculate the average equivocation of the random linear binary codes in the syndrome coding scheme.

In Chapter 2, we briefly review the basis of information theoretic security and some popular secure coding schemes for the wiretap channel as well as prior work in the area and its relation to the work in this dissertation.

In Chapter 3, we consider the communication in the Gaussian wiretap channel in the wideband regime. Based on Wyner's code construction, an orthogonal coding scheme was proposed. By the theoretical analysis on the error probability of the main channel and the equivocation of the eavesdropper channel and some simulation results, it is shown that the proposed orthogonal codes can achieve a reliable and perfectly secure communication in the Gaussian wiretap channel in the asymptotic wideband regime, which entails communication at a vanishing rate.

In Chapter 4, we propose an efficient recursive method to determine the equivocation rate for the Binary Symmetric Channel (BSC) and any linear binary code. Based on this practical algorithm, a code design technique is presented to produce the best codes for the syndrome coding scheme as measured by an information secrecy metric: the equivocation rate. Some Best Equivocation Codes (BEC) for the syndrome coding scheme are published. It is shown that the Best Equivocation Codes have a better performance on the secrecy for the syndrome coding scheme than the best known codes (BKC), which were designed for the error correcting scheme.

In Chapter 5, we analyse the security performance of the random linear codes in Wyner's syndrome coding scheme from the viewpoint of the information theoretical security. We propose a theoretical way to calculate the average equivocation of the (n, k) random codes, and compare it with the results from the Monte Carlo experiments. Various examples are given to show the accuracy of our analysis. It is shown that this theoretical method works very well for the codes with the large values of $n - k$, which are hardly calculated by the Monte Carlo method. Using our theoretical method, the relation between the code length and the equivocation and the relation between information rate and equivocation are explored.

In Chapter 6, we propose a chain based syndrome coding scheme for the secure communication in a specific wiretap channel, where the main channel is noiseless and the

eavesdropper channel is binary symmetric, which represents a generalization of Wyner's conventional syndrome coding scheme. By using the channel decoder output equivocation of the eavesdropper to measure the security of the scheme, extensive analytic and numerical results demonstrate that the chain based syndrome coding scheme with a hash function exhibits significantly lower information leakage than the conventional syndrome coding scheme.

In Chapter 7, we analyse the equivocation of the eavesdropper under different attacks when the legitimate parties employ the McEliece public-key cryptosystem in the traditional wiretap scenario. It is shown that there is considerable information leakage to the eavesdropper under Brickell's attack. A modified McEliece public-key cryptosystem is proposed to reduce the information leakage under Brickell's attack. It is proved that the modified system exhibits zero information leakage under Brickell's attack, and the price for this higher security is a lower information rate.

In Chapter 8, we give the conclusions of this dissertation, summarize the main contributions and suggestions for future work.

1.3 Thesis contribution

This dissertation focuses on the design of the secure physical layer coding schemes for different types of wiretap channels, based on Wyner's coding scheme. As the syndrome coding scheme has been applied widely, it is relevant to study the impact of different code designs for the syndrome coding scheme, including random designs and best equivocation designs. To consider the first problem, in this dissertation we set up a putative code model to study the security performance of the random codes in the syndrome coding scheme. As we know, most of the code design methods are for the error correcting system [45–49], which aim to guarantee the reliability of the communication. To consider the issue of the security, firstly we have to give a quantified analysis on the security level of the syndrome coding scheme for different codes employed, which has been addressed in this dissertation: we propose an efficient recursive method for determining the equivocation for the Binary Symmetric Channel (BSC) and any linear binary code in the syndrome coding scheme. Based on this efficient way to calculate the equivocation of the code, we propose a code design method to construct the best code for the syndrome coding scheme. Finally, we apply the information-theoretic analysis method to study the

secrecy of McEliece public key encryption system, and propose a modified system to combat some attack.

The major scientific contributions of the dissertation are outlined as follows.

- **A study of the reliability and security of orthogonal codes in Gaussian wiretap channels in wideband regime (Chapter 3).** We prove that this codes can achieve reliable and perfect secrecy communication in the Gaussian wiretap channel in wideband regime, by analysis on the error probability of the main channel and the equivocation of the eavesdropper channel. We also show various simulation results that corroborate the analysis.
- **A simplified encoder for the syndrome coding scheme (Chapter 4).** In the traditional syndrome coding scheme, there is the need to generate an error-syndrome table as a look-up table for the encoder, but this also requires a considerable memory to store this table for long codes. We propose a way to encode the message without a look-up table, which makes the syndrome coding scheme more practical for the long codes.
- **An efficient recursive calculation method for the equivocation of the syndrome coding scheme (Chapter 4).** The equivocation is an important metric to quantify the security level of the communication. The traditional calculation method is not efficient and practical for the long codes, so we introduce a very efficient way to calculate the equivocation in the syndrome coding scheme, which is also important for the code design and the performance analysis of the random codes in the following chapters.
- **Best equivocation codes design for the syndrome coding scheme (Chapter 4).** Based on the efficient recursive calculation of the equivocation, we also propose a code design technique to construct the best codes for the syndrome coding scheme. The target of this code design is to construct the codes with best security performance, which is different from the code design for the error correcting scheme.
- **Analysis of the average equivocation of the random codes in the syndrome coding scheme (Chapter 5).** A theoretical model is proposed to analyse the security performance of syndrome coding schemes with random codes, which also shows the potential of the scheme for secret communication.
- **A chain based syndrome coding scheme (Chapter 6).** As the conventional syndrome coding scheme is widely employed, we propose a chain based syndrome coding scheme as a generalization of the syndrome coding scheme and a chain

based syndrome coding scheme with a hash function to improve the security level of the communication. The new scheme with a hash function has increased secrecy levels than the traditional one, as the value of the chain depth increases.

- **A modified McEliece public-key cryptosystem with higher security level (Chapter 7).** We propose a modified cryptosystem to achieve a higher security level of the communication.

These contributions led to the following publications:

1. K. Zhang, M. Tomlinson, M. Z. Ahmed, Marcel Ambroze and M. R. D. Rodrigues, Best Binary Equivocation Code Construction for Syndrome Coding, *IET communications*, May 2014.
2. K. Zhang, M. Tomlinson, M. Z. Ahmed, A Chain Based Syndrome Coding Scheme for Secure Communication in the Wiretap Channel, *IEEE Vehicular Technology Conference spring*, Soul, South Korea, May, 2014.
3. K. Zhang, M. Tomlinson, M. Z. Ahmed and Marcel Ambroze, The Average Equivocation of Random Linear Binary Codes in Syndrome Coding, *International Conference on Telecommunications*, Lisbon, Portugal, May, 2014.
4. K. Zhang, M. Tomlinson, M. Z. Ahmed A Modified McEliece Public Key Encryption System with a Higher Security Level, *IEEE International Conference on Information Science and Technology (ICIST 2013)*, Yangzhou, China, March, 2013.
5. K. Zhang, M. R. D. Rodrigues, M. Z. Ahmed, M. Tomlinson and F. Cercas, Orthogonal Signalling in the Gaussian Wiretap Channel in the Wideband Regime, *IEEE Vehicular Technology Conference Spring*, Yokohama, Japan, May 2012.

Chapter 2

Information-Theoretic Security and Coding for Secure Communications

This chapter briefly reviews the basis of information-theoretic security and physical-layer security, together with some popular transmission schemes for secure communication. In particular, we introduce Shannon's idea of perfect secrecy, the wiretap channel model, the measures of reliability and security and the secrecy capacity of different wiretap channels. We also describe Wyner's coding scheme and the syndrome coding scheme, which will be considered later in this thesis.

2.1 Information theoretic security

There are two main notions of security: the security of a cryptographic system can rely either on the computational infeasibility of breaking it (computational security notion), or on the theoretical impossibility of breaking it, even using infinite computing power (information theoretic or unconditional security notion).

It is important to understand the differences between these two notions and how they affect the choice of the technology in a practical scenario.

2.1.1 Information theoretic security versus computational security

To guarantee the confidentiality of the information transmitted by the transmitter, most of the cryptographic schemes, which are widely applied in modern communication sys-

tems, are based on computational security and rely on assumptions on the limitations of eavesdropper's computing resources. Computational security introduces two relaxations to perfect security: (1) security is only against the adversaries with limited computer resources; (2) a small probability of success is allowed. Actually, the computational security is based on complexity theory. Usually, the system is deemed as secure, if the computation complexity of the decoding algorithm is equal to that required for solving "hard" mathematical problems (e.g. NP-hard problems). The proved lower bounds on the running time of algorithms solving specific problems are either rather weak or valid only in special computational models. For example, the cryptographic algorithm like RSA and AES are deemed secure for a large number of applications, because so far no efficient attacks have been proposed. So-called provable computational security is always conditional, unreasonable with today's fast improved technology and not reliable in the long term. Computing power increases very fast, so the brute-force attacks, which were considered infeasible, become possible later. It has been shown that the integer-factoring as well as the discrete logarithm problem can be solved in polynomial-time by quantum computer [50].

In general, the security of the computational secure system is measured by whether it survives a set of attacks or not and there are no precise metrics to compare the security level of different ciphers. Using the security metric in the information theory, which was introduced by Shannon, it can be shown that the cryptographic paradigm can never provide information theoretic security, since the communication channel between the friendly parties and the eavesdropper is error free. Consequently, practical computational security is always conditional and additionally faces the risk of being broken by progress in the theory of efficient algorithms or in hardware engineering [8]. Therefore, it is desirable from both a scientific and practical point of view to design cryptosystems whose security is not based on any assumptions and can be proven rigorously.

To avoid these assumptions associated with the computational security, we can consider another criterion based on an information theoretic measure, which is the metric of secrecy in information theoretic security. We treat the messages and the encrypted code-words as random variables denoted by M and V , respectively, and measure the security of the communication by the entropy of the message given the received codeword $H(M|V)$. If this conditional entropy $H(M|V)$ is equal to the entropy of the message $H(M)$, we deem the communication to be perfect secrecy, which is due to the fact that $H(M|V) = H(M)$ – the eavesdropper can not extract any useful information about the transmitted message M from the received vector V – means that $p(M|V) = p(M)$, so the best the eavesdropper can do is to guess the message. The main advantages of information theoretic security derive

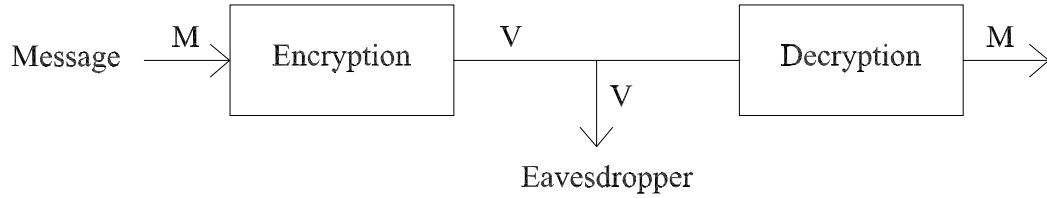


Figure 2.1: Channel model in cryptographic

from the fact that it does not impose computational restrictions on the eavesdropper and that the information leakage to the eavesdropper can be precisely measured as a function of the channel quality and the block length of the message. However, there exist few practical scheme, that achieve perfect secrecy to date.

The channel model mentioned above (Figure 2.1) is too pessimistic from the viewpoint of security, because practical communications channels always exhibit various type of impairments, such as noise, that can be leveraged to secure information further. The incorporation of such impairments onto the communications channel model leads to the *wiretap channel*, which is widely employed in information theoretic security. The basic idea of the information theoretic approach to securely transmit confidential messages (without using an encryption key) to a legitimate receiver is to use the inherent randomness of the physical medium. In this approach, a transmitter intentionally adds structural randomness to prevent potential eavesdroppers and attackers from intercepting useful information while guaranteeing that a legitimate receiver can obtain the information. Figure 2.2 shows the relation between the information theoretical security approach and the conventional encryption method. The information theoretical security approach combines the 'encryption' and 'encoding' into a single block 'secure coding' to guarantee both reliability and security of the communication [2]. In fact, the information theoretic security has already been realized in practice through quantum key distribution and it has been proved that suitable long codes can come exponentially close to perfect secrecy [51].

Based on the comparisons above, the information theoretic security algorithms in a classical system exploit physical layer attributes of channel randomness for secure communications, and provide an additional layer of protection for information transmission, which gives a very promising direction of the research on the communication security.

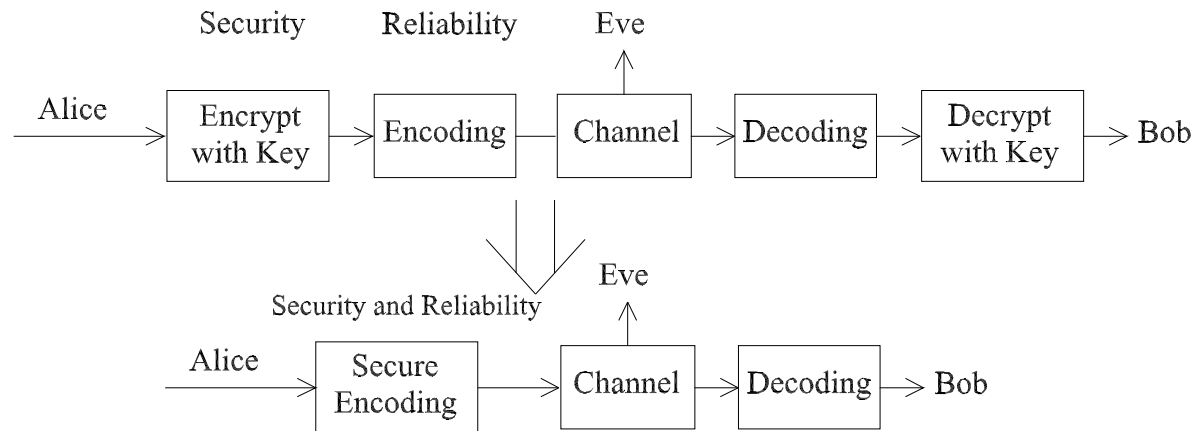


Figure 2.2: Information theoretic security

2.1.2 Shannon's notion of perfect secrecy

The fundamental principle of the information theoretic security is based on Shannon's idea of perfect secrecy, which was introduced by Shannon in 1949 [9]. The following cryptosystem is considered (Figure 2.3): the sender wants to send a message M to the legitimate receiver secretly by encrypting it into a cipher-text V using a secret key, K . The eavesdropper observes this cipher-text V without any degradation, which corresponds to a scenario where the channel between the transmitter and the eavesdropper is noiseless, and tries to obtain the original message M . The security of this system relies on the secret key, K , which is shared only between the transmitter and the legitimate receiver.

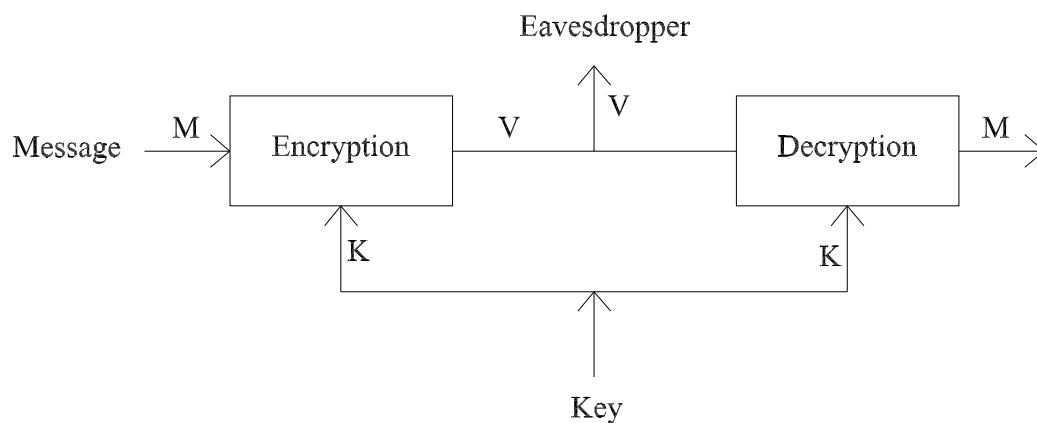


Figure 2.3: Shannon's cryptosystem

Shannon formalized the notion of secrecy by quantifying the average uncertainty of the eavesdropper. In terms of information theory, the secrecy is measured by the conditional entropy of the original message given the cipher text, denoted as $H(M|V)$, in which M and V are seen as random variables, also known as the equivocation of the eavesdropper. Perfect secrecy is achieved if the equivocation of the message equals to the entropy of the message, that is:

$$H(M|V) = H(M) \quad (2.1)$$

equivalently, the mutual information between the message and the cipher text is zero,

$$I(M; V) = H(M) - H(M|V) = 0 \quad (2.2)$$

which implies that the cipher text V is statistically independent of the message M .

Shannon's perfect secrecy is also known as '*strong secrecy*', which entails $H(M|V) = H(M)$.

In Shannon's cryptosystem, perfect secrecy can be achieved by means of a simple procedure called a one-time pad with low complexity, however, its applicability is limited by the following requirements [51]: (a) the legitimate partners must generate and store long keys consisting of random bits, (b) each key can only be used once, otherwise the cryptanalyst has a fair chance of discovering the key, and (c) the key must be shared over a secure channel.

It has been proved that the perfect secrecy can only be achieved when $H(K) \geq H(M)$ for Shannon's cryptosystem, i.e. the uncertainty about the key should be at least as high as the uncertainty about the message, which also implies that the size of the key has to be at least as large as the size of the message. Actually, the one-time pad encryption is impractical unless there exist efficient means of distributing secret keys between the legitimate parties.

2.1.3 Wiretap channel model

The information theoretic security approach is based on Shannon's perfect secrecy idea, which considers the cryptosystem as Figure 2.3 with noiseless channels. However, in reality, random noise is an intrinsic element of almost all physical communication channels.

In order to understand the role of the noise in the security of communication systems, Wyner introduced the wiretap channel, which is the most basic physical layer model that captures the essence of communication security (see Figure 2.4). A transmitter, Alice, wishes to transmit a m -bit message M to a legitimate receiver, Bob, and wishes to keep this message as secret as possible from an eavesdropper, Eve. The transmitted message M is uniformly distributed over a message set $\mathcal{M}$. The encoder at the transmitter maps each message M to a n -bit codeword X , and sends this codeword over discrete memoryless channels (DMC) with transition probabilities $P_{Y_M|X}$ and $P_{Y_E|X}$. The output of the main channel, Y_M , is received by the legitimate receiver and decoded to an estimated message $\hat{M}_M$. The output of the eavesdropper channel, Y_E , is received by the eavesdropper. It is assumed that the main channel is less noisy than the eavesdropper channel.

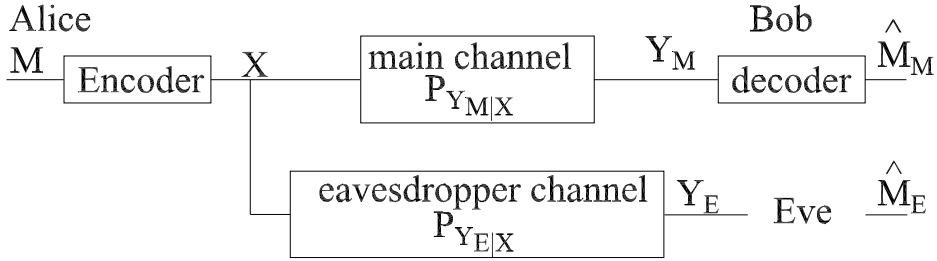


Figure 2.4: The wiretap model

The key difference between the wiretap channel and Shannon's cryptosystem relates to the fact that: in the wiretap channel the legitimate receiver and the eavesdropper observe different noisy versions of the transmitted codeword and usually the main channel is less noisy than the eavesdropper channel, while the channels are error free in Shannon's model. Moreover, the security of Shannon's cryptosystem relies on the use of a secret key, while the information theoretic security approaches designed for the wiretap channel derive the benefits from the difference of the noise levels between the main channel and the eavesdropper channel to guarantee the communication security without any keys.

'Weak secrecy', has been introduced as another secrecy condition, which is different from (2.1), in that the equivocation rate $R_e = \frac{1}{n}H(M|Y_E)$ is arbitrarily close to the entropy rate of the original message $R = \frac{1}{n}H(M)$, for large enough value of n [52–54]. The maximum transmission rate achievable under this condition is defined as *secrecy capacity*, which will be introduced later.

Wyner's wiretap channel is a very basic model for the design of information theoretic security schemes. Other generalizations of this conventional wiretap channel model include binary symmetric wiretap channel [10], binary erasure wiretap channel [12], Gaussian wiretap channel [55].

2.1.4 Reliability and security measures

The source message M is randomly and uniformly distributed over a message set $\mathcal{M}$ with a number of $|\mathcal{M}|^1$ messages. The legitimate transmitter, Alice, wishes to convey a message $M \in \mathcal{M}$ to the legitimate receiver, Bob. The legitimate transmitter-receiver pair use a code of length n (assumed to be known also to the eavesdropper), a stochastic encoding function that maps the message M into a length n transmitted codeword $X = [X(1) X(2) \cdots X(n)]$, and a decoding function that maps the received codeword $Y_M = [Y_M(1) Y_M(2) \cdots Y_M(n)]$ into the message estimate $\hat{M}_M$. The information rate of this code is:

$$R = \frac{\log_2 |\mathcal{M}|}{n} \quad (2.3)$$

The average (message) error probability of the length n code is used to measure the reliability of the communication, which is defined as follows:

$$P_e = \frac{1}{|\mathcal{M}|} \sum_M \Pr(\hat{M}_M \neq M|M) \quad (2.4)$$

When $P_e \rightarrow 0$, the communication is said to be completely reliable.

In the eavesdropper channel, the eavesdropper receives a length n codeword $Y_E = [Y_E(1) Y_E(2) \cdots Y_E(n)]$ and uses some decoder function to get the estimated message $\hat{M}_E$. The equivocation of the eavesdropper is an important security metric, which is given by:

$$S_c = H(M|Y_E) \quad (2.5)$$

and the equivocation rate is $R_e = \frac{1}{n} S_c$.

2.1.5 Secrecy capacity

The *rate-equivocation pair* (R, R_e) is achievable if there exists a sequence of codes with rate R and length n , and encoder-decoder pairs such that the following reliability and

¹ $|\cdot|$ denotes the cardinality of a set.

security criteria are satisfied:

$$P_e \rightarrow 0 \quad (2.6)$$

$$R_e \leq \liminf_{n \rightarrow \infty} \frac{1}{n} H(M|Y_E) \quad (2.7)$$

as n goes to infinity.

The *capacity-equivocation region* ζ is defined to be the closure of all achievable pairs (R, R_e) . In this dissertation, the wiretap channel we consider is restricted to the model that the main channel is always less noisy than the eavesdropper channel. The capacity-equivocation region ζ of this wiretap channel is given by [2]:

$$\zeta = \cup_{P_X P_{Y_M|Y_E|X}} \left(\begin{array}{c} (R, R_e) \\ 0 \leq R \leq I(X; Y_M) \\ 0 \leq R_e \leq R \\ R_e \leq I(X; Y_M) - I(X; Y_E) \end{array} \right) \quad (2.8)$$

The highest R , when the pair (R, R_e) is achievable, is defined as *secrecy capacity*, C_s . In this case, $R = R_e$, which is perfect secrecy. The secrecy capacity of a *less noisy* channel² is given by:

$$C_s = \max_{P_X} I(X; Y_M) - I(X; Y_E) \quad (2.9)$$

where the maximum is taken over all possible input distributions p_X .

We now introduce specific wiretap channel models.

2.1.5.1 Binary erasure wiretap channel

In binary erasure wiretap channel (BEWC), the main channel is noiseless and the eavesdropper channel is the binary erasure channel with an error probability of p (Figure 2.5).

The secrecy capacity of this binary erasure wiretap channel is $C_s = p$.

Many code schemes have been studied in BEWC [12, 15, 28, 56]. For example, in [12], Ozarow and Wyner consider error correcting code design for a type II BEWC based on a coset coding scheme. In [28], Thangaraj *et al.* proposed using a duals of low density parity check codes (LDPC) in a coset coding scheme to achieve weak secrecy on the BEWC.

²A wiretap channel is the less noisy channel if the channel to the legitimate receiver is *more capable* than the channel to the eavesdropper, i.e. $I(X; Y_M) \geq I(X; Y_E)$ [2].

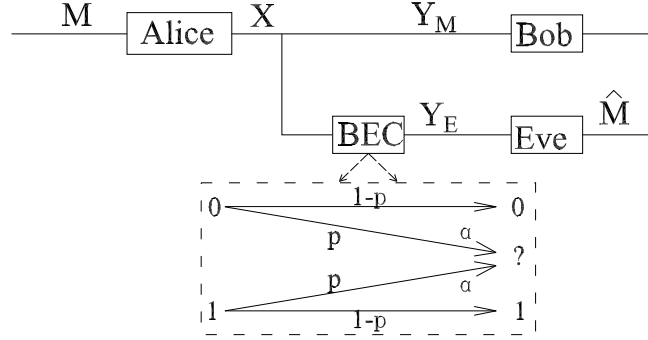


Figure 2.5: The Binary erasure wiretap model

2.1.5.2 Binary symmetric wiretap channel

Wyner introduced the wiretap channel in [10], where the main channel is error-free and the eavesdropper channel is binary symmetric with a 'crossover probability' α (see Figure 2.6), given by:

$$Y_M = X \quad (2.10)$$

$$Y_E = X + e \quad (2.11)$$

where Y_M and Y_E denote the main channel output and the eavesdropper channel output at time i respectively, and e is the error generated by the binary symmetric channel.

This represents an abstraction of a typical bugging situation where the legitimate transmitter, Alice, and the legitimate receiver, Bob employ a robust channel, and the eavesdropper, Eve, is listening using a bug with limited transmission power and antenna gain with non line of sight radio propagation conditions so that she receives the transmissions with errors due to the poor communication channel.

As given in [10] the secrecy capacity of this wiretap channel is:

$$C_s = -\alpha \cdot \log_2 \alpha - (1 - \alpha) \cdot \log_2(1 - \alpha) \quad (2.12)$$

which is the highest transmission rate that can be achieved whilst keeping communications secure from eavesdropping for this channel.

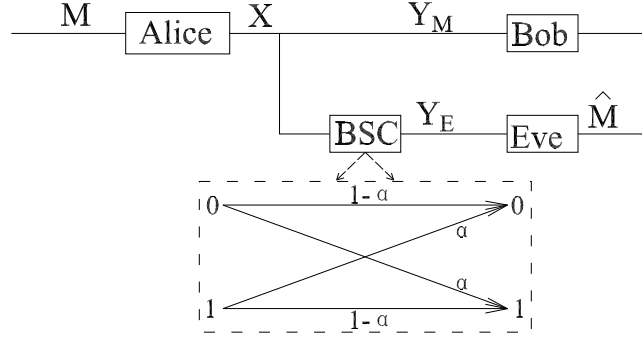


Figure 2.6: The Binary symmetric wiretap model

2.1.5.3 Gaussian wiretap channel

The Gaussian wiretap channel is shown in Figure 2.7, where two legitimate parties, Alice and Bob, communicate in the presence of an eavesdropper, Eve, given by:

$$Y_M(i) = X(i) + N_M(i), \quad i = 1, \dots, n \quad (2.13)$$

$$Y_E(i) = X(i) + N_E(i), \quad i = 1, \dots, n \quad (2.14)$$

where $Y_M(i)$ and $Y_E(i)$ correspond to the main channel and the eavesdropper channel outputs at time i , respectively, $X(i)$ corresponds to the channel input at time i , and $N_M(i)$ and $N_E(i)$ are independent and identically distributed (i.i.d.) Gaussian random variables with mean zero and variance $N_M = N_{M0} \cdot B = N$ and $N_E = N_{E0} \cdot B = \frac{1}{\alpha} \cdot N$, respectively, where $N_{M0} = N_0$ and $N_{E0} = \frac{1}{\alpha} \cdot N_0$ are one-sided noise power spectral densities and B is the effective bandwidth of the channel. In particular, we consider the case when the main channel is less noisy than the eavesdropper channel, i.e. $0 \leq \alpha \leq 1$.

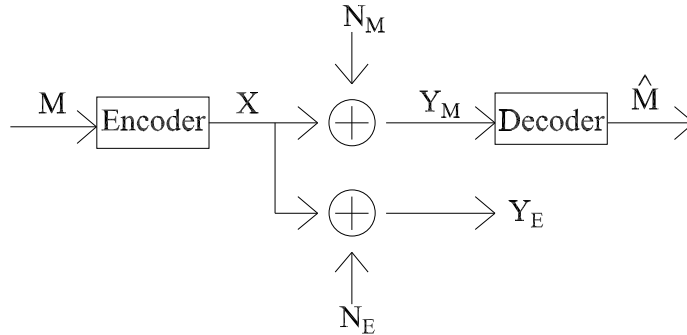


Figure 2.7: Gaussian wiretap channel

We denote for simplicity the set of channel inputs (the transmit codeword) by $X =$

$[X(1) X(2) \cdots X(n)]$, the set of channel outputs (the receive codewords) by $Y = [Y(1) Y(2) \cdots Y(n)]$, and the set of noise values by $N = [N(1) N(2) \cdots N(n)]$.³ We also assume the average power constraint:

$$\frac{1}{n} \sum_{i=1}^n |X(i)|^2 \leq P \quad (2.15)$$

The secrecy capacity C_s of the Gaussian wiretap channel, which corresponds to the supremum of achievable perfect secrecy rate, is given by:

$$C_s = \sup_{p_X(x): \mathbb{E}\{|X|^2\} \leq P} I(X; Y_M) - I(X; Y_E) \quad (2.16)$$

where $p_X(x)$ denotes the probability distribution of the random variable X .

The secrecy capacity, which in the degraded Gaussian wiretap channel is achieved by circularly symmetric complex Gaussian inputs, admits the closed form expression [55]:

$$C_s = [C_M - C_E]^+ = \left[\log_2 \left(1 + \frac{P}{N} \right) - \log_2 \left(1 + \alpha \cdot \frac{P}{N} \right) \right]^+ \quad (2.17)$$

where $[x]^+ = \max(0, x)$, $C_M = \log_2(1 + P/N)$ is the capacity of the main channel and $C_E = \log_2(1 + \alpha \cdot P/N)$ is the capacity of the eavesdropper channel.

It is important to note that the discrete-time wiretap channel model in (2.13) and (2.14) arises from the continuous-time model:

$$y_M(t) = x(t) + n_M(t) \quad (2.18)$$

$$y_E(t) = x(t) + n_E(t) \quad (2.19)$$

where $y_M(t)$ and $y_E(t)$ are the receive signals at the output of the main and eavesdropper channels, respectively, $x(t)$ is the transmit information-bearing signal, and $n_M(t)$ and $n_E(t)$ are white Gaussian noise with double-sided power spectral density $\frac{N_{M0}}{2}$ and $\frac{N_{E0}}{2}$, respectively. By letting the effective duration and bandwidth of the transmit signal be T and B then passing the receive signals through an orthonormal transformation with $n = TB$ complex dimensions is sufficient to preserve all the information (asymptotically). Note that the rate in bits per second per hertz is equal to the number of bits per dimension, i.e.,

$$R = \frac{k}{TB} = \frac{k}{n} \quad (2.20)$$

³We use subscripts to indicate whether the vector Y and the vector N refer to the main or the eavesdropper channel.

where k denotes the number of bits conveyed by the information bearing signal $x(t)$. Note also that the power is given by:

$$P = \frac{\xi_s}{T} = \frac{k \cdot \xi_b}{T} = R \cdot \xi_b \cdot B \quad (2.21)$$

where ξ_s is the signal energy and ξ_b is the energy per bit.

2.2 Some basic coding schemes and code design for the wiretap channels

A typical transmission system can be represented by the block diagram shown in Figure 2.8. There are two pairs of encoder-decoder. The *source encoder* is to transform the source output into a sequence of vectors, M . The *channel encoder* is to transform M into a discrete encoded sequence X called *codeword*. The issues of the reliability and the security are considered in channel coding. To focus attention on the channel coding, the information source and the source encoder can be combined into a *digital source* with output M , and the source decoder and the destination can be combined into a *digital sink* with input $\hat{M}$ (Figure 2.9).

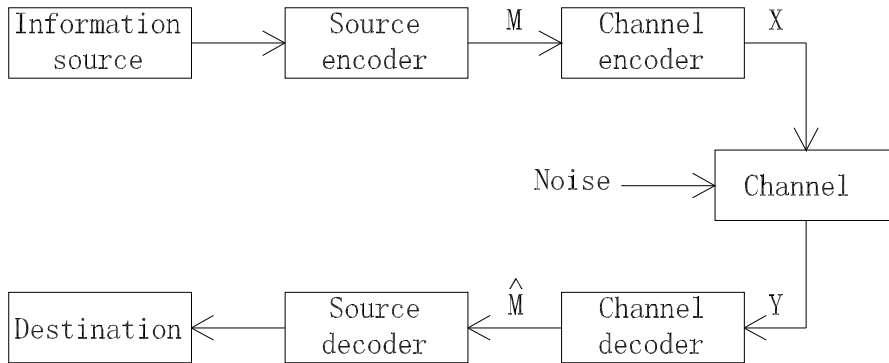


Figure 2.8: Typical transmission system

Traditionally the design and implementation of the channel coding in physical layer is to combat the noise in the channel and to guarantee the reliability of the communication. 1948, Shannon [57] demonstrated that by proper encoding of the information, errors induced by a noisy channel can be reduced to any desired level without sacrificing the rate of information, as long as the information rate is less than the capacity of the channel.

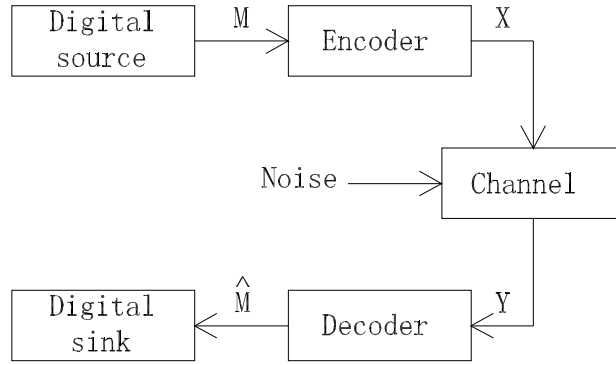


Figure 2.9: Simplified model of a coded system

Error-correcting coding scheme is the main scheme to correct the errors and to guarantee the reliability of the communication.

Wyner's coding scheme was introduced to guarantee the security of the communication in the wiretap channel and syndrome coding scheme is a specific case of Wyner's coding scheme, which is an important coding scheme in the physical layer and has been widely studied for various applications [28, 58–60].

In this section, we describe some important secure coding schemes and some code design techniques in the physical layer, which will be used later.

2.2.1 Linear block codes

One of the most useful codes is block codes [61], which can be used in both error-correcting coding schemes and the Wyner's coding scheme. In block coding, the binary information sequence is segmented into *message blocks* of fixed length k ; each message block, denoted by M , is a k -bit vector. There are 2^k distinct messages in total. The encoder transforms each message M into a n -bit codeword X , according to certain rules. The linear block codes are defined as follows:

Definition 1. A block code of length n and 2^k codewords is called a linear (n, k) code if and only if the modulo-2 sum of two codewords is also a codeword.

Linear block codes are defined and described in terms of a $k \times n$ generator matrix $\mathbf{G}$ or by a $m \times n$ parity check matrix $\mathbf{H}$. If M is a k -bit message to be encoded, the corresponding

n -bit codeword is: $X = M \cdot \mathbf{G}$, which is transmitted over a noisy channel. The output of the channel is a n -bit vector $Y = X + e$, in which e is a n -bit error vector induced by the channel.

When Y is received, the decoder computes a $(n - k)$ -bit vector: $S = Y \cdot \mathbf{H}^T$, which is called the *syndrome* of Y . $S = 0$ if and only if Y is a codeword, and $S \neq 0$ if and only if Y is not a codeword. According to S , some error patterns can be detected and corrected.

Definition 2. *The minimum distance of a block code C , denoted by d , is defined as:*

$$d = \min\{d(x_1, x_2) : x_1, x_2 \in C, x_1 \neq x_2\} \quad (2.22)$$

in which $d(x_1, x_2)$ is the Hamming distance between two different codeword in C : x_1 and x_2 , which is defined as the number of places where they differ.

The error-detecting and error-correcting capabilities of a block code are determined by the minimum distance of the code. A block code with minimum distance d is capable of detecting all the error patterns of $d - 1$ or fewer errors and correcting all the error patterns of $t = \lfloor (d - 1)/2 \rfloor$ or fewer errors.

2.2.2 Wyner's coding scheme

There are two goals of the code scheme design in the wiretap channel: (i) maximize the information rate of reliable communication between the transmitter and the legitimate receiver (reliability condition); (ii) guarantee that the eavesdropper knows the original information as little as possible (secrecy condition).

In wiretap channel, the channel between legitimate parties has a higher capacity (lower noise) than that in the eavesdropper channel. Appropriate coding techniques exploit the difference of noise levels in order to guarantee confidential communications between legitimate parties. Wyner's coding scheme gives the basic structure of the secrecy capacity achieving coding scheme put forth in [10] for the wiretap channel. The basic idea of Wyner's coding scheme is to add randomness in the encoding process, i.e. to assign multiple codewords to each message. The scheme achieves the reliability and security objectives by incorporating redundancy as well as randomness in a binning structure, as illustrated in Figure 2.10.

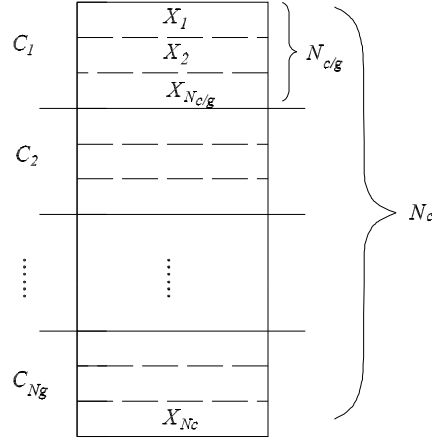


Figure 2.10: Binning structure in Wyner's coding scheme

In particular, the coding scheme, and the encoding and decoding procedure operate as follows:

- *Codebook*: The codebook $\mathcal{C}$ is composed of a number of equal size sub-codebooks $\mathcal{C}_1, \dots, \mathcal{C}_{N_g}$. The number of sub-codebooks is $N_g = 2^{nC_s} = 2^{n(C_M - C_E)}$. The total number of codewords in the codebook is $N_c = 2^{nC_M}$ and the total number of codewords in each sub-codebook is $N_{c/g} = 2^{nC_E}$, in which C_s is the secrecy capacity of the wiretap channel, C_M is the channel capacity of the main channel and C_E is the channel capacity of the eavesdropper channel. $X_1, \dots, X_{N_c}$ denote all N_c codewords in the codebook.
- *Encoding*: The transmitter maps the message M into a randomly chosen codeword from the corresponding sub-codebook. This codeword is transmitted over the channel.
- *Decoding*: The legitimate receiver can successfully decode over the codebook $\mathcal{C}$, but the eavesdropper receiver can not. This ensures reliability and perfect secrecy.

Wyner's coding construction, which revolves around the notion of stochastic encoding, suggests a basic structure to design secrecy capacity achieving codes for reliable and secure communication over the wiretap channel.

2.2.3 Syndrome coding scheme

Even though the Wyner's code construction does not provide any practical method for constructing a real code, the binning structure is reproduced by many researchers, attempting to construct secret codes for the wiretap channel. Wyner and Ozarow used the following syndrome coding strategy (also known as coset encoding or nested code) [10, 12] to show that perfect secrecy can be achieved when the main channel is error free and the eavesdropper channel is binary symmetric (see Figure 2.6).

Alice wishes to convey a sequence of independent and uniformly distributed m -bit binary messages to the legitimate receiver, $M(1), \dots, M(b)$, in which b is the block length of the messages. This sequence of m -bit messages $M(1), \dots, M(b)$ is encoded into a sequence of n -bit codewords $X(1), \dots, X(b)$. Bob observes the sequence of n -bit codewords $Y_M(1), \dots, Y_M(b)$ where

$$Y_M(i) = X(i), \quad i = 1, \dots, b \quad (2.23)$$

and Eve observes the sequence of n -bit codewords $Y_E(1), \dots, Y_E(b)$ where

$$Y_E(i) = X(i) + e(i), \quad i = 1, \dots, b \quad (2.24)$$

in which '+' is mod 2 addition and $e(i)$ represents a n -bit error vector arising from the binary symmetric channel. Bob produces the sequence of original messages $M(1), \dots, M(b)$ from the main channel output sequence $Y_M(1), \dots, Y_M(b)$, whereas Eve produces a sequence of message estimates $\hat{M}(1), \dots, \hat{M}(b)$ from the eavesdropper channel output sequence $Y_E(1), \dots, Y_E(b)$.

The syndrome coding scheme uses a $(n, k, 2t+1)$ linear block code capable of correcting t errors, defined either by a $k \times n$ generator matrix $\mathbf{G}$ or by a $(n-k) \times n$ parity check matrix $\mathbf{H}$ [61]. It is a property of any linear block code that there exist 2^{n-k} distinct minimum weight error patterns in which each pattern produces a distinct syndrome of the total 2^{n-k} syndromes. These minimum weight error patterns are the coset leaders of the code and may be represented by a error syndrome table. For perfect codes all error patterns in the table are the n -bit binary vectors with a weight $w \leq t$. For non-perfect codes some of the error patterns in the table are n -bit binary vectors with a weight $w \leq t$ and the remaining error patterns are n -bit binary vectors with a weight $w > t$. Regardless of whether or not the code is perfect, in syndrome coding all 2^{n-k} syndromes are used to send messages. This 'error-syndrome table', which is key to the operation of syndrome based coding schemes, is assumed to be known by Alice, Bob and Eve.

The encoding and decoding processes are as follows.

2.2.3.1 Encoder

Alice produces the n -bit vector $X(i)$ from the $m = (n - k)$ -bit message $M(i)$ at time i :

- Alice generates a n -bit codeword $C_T(i)$ from a random k -bit vector $D_R(i)$:

$$C_T(i) = D_R(i) \times \mathbf{G} \quad (2.25)$$

- Alice equates the m bit message to an m bit syndrome $S_T(i)$ and uses the error syndrome table to find the n -bit error pattern $E_T(i)$, $S_T(i) = M(i)$.
- Alice then generates the n -bit vector $X(i)$ by the adding the n -bit codeword $C_T(i)$ to the n -bit error pattern $E_T(i)$, i.e.,

$$X(i) = C_T(i) + E_T(i) \quad (2.26)$$

The information rate of the syndrome coding scheme is $R = \frac{n-k}{n} = \frac{m}{n}$.

2.2.3.2 Legitimate decoder

We assume that Bob uses look up table syndrome decoding to determine the original messages. By computing the syndrome associated with the received vector, which is given by:

$$S_{Bob}(i) = Y_M(i) \times \mathbf{H}^T = X(i) \times \mathbf{H}^T = S_T(i) \quad (2.27)$$

and then the original message is immediately as follows:

$$M(i) = S_{Bob}(i) = S_T(i) \quad (2.28)$$

2.2.3.3 Eavesdropper decoder

We also assume that Eve – as Bob – uses look up table syndrome decoding to estimate the original messages. By computing the syndrome associated with the received vector as follows:

$$S_{Eve}(i) = Y_E(i) \times \mathbf{H}^T = (X(i) + e(i)) \times \mathbf{H}^T = S_T(i) + S_e(i) \quad (2.29)$$

then the estimate of the message is given by:

$$\hat{M}(i) = S_{Eve}(i) = M(i) + S_e(i) \quad (2.30)$$

The syndrome coding scheme has been widely studied by employing different kinds of codes. For example, Thangaraj *et al.* considered nested LDPC codes for the case when the main channel is noiseless, and showed that this syndrome coding scheme achieves both the reliability and the security criterion [28]. Liu *et al.* also considered noiseless main channels in [58], with a BEC, BSC, or an AWGN channel to the eavesdropper. In [59] Liu *et al.* considered nested codes designed for the BEWC used over general binary input symmetric channels for transmission at rates below the secrecy capacity. In [60], Chen and Vinck showed that using random linear codes in syndrome coding scheme can achieve the secrecy capacity over the binary symmetric wiretap channel and an upper bound on the information leakage was derived. Moreover, syndrome coding scheme by polar codes are secrecy capacity achieving for the wiretap channel, which has been shown by several research groups independently [31, 34].

For the case that both of the main channel and the eavesdropper channel are noisy, channel noise and the resulting error rate could have non-negligible effect on both decoding messages by the legitimate receiver and attack by the eavesdropper. In [43] and [44], Yin *et al.* have studied the effects of channel errors on security and decoding error probability of some encryption systems when the main channel and the eavesdropper channel are both binary symmetric channels, and have suggested the tradeoff between communication reliability and security. Wiretap channel coding for securing noisy communications has also been studied in [28, 62, 63]. For example, in [28], Thangaraj *et al.* employ a code concatenation scheme with a combination of the error-correcting coding scheme and the syndrome coding scheme in the wiretap channel, in which both the main channel and the eavesdropper channel are noisy. In this scheme, when the main channel is less noisy than the eavesdropper channel, the errors in the main channel can be cleaned up, which means the main channel becomes error free, but there are still some errors in the eavesdropper channel. In [62], the transmitter ensures secrecy of communication by utilizing some of the available power to produce “artificial noise”, such that only the eavesdropper’s channel is degraded.

Chapter 3

Orthogonal Codes for Secure Communication in the Wideband Regime

This chapter investigates the use of orthogonal codes in the structure of Wyner's coding scheme to realize a reliable and secure communication in the wideband regime. In particular, the error probability of main channel and the equivocation of the eavesdropper have been analysed and simulated. It shows that orthogonal codes achieve the secrecy capacity in Gaussian wiretap channel in the wideband regime.

3.1 Introduction

Wyner proposed an very important secure coding strategy (called Wyner's coding scheme) in wiretap channel, suggesting that encoding for reliability and security should partition the codebook into sub-codebooks, which provides a basic code design strategy for secure communication in wiretap channel [26,28,35]. Based on the structure of Wyner's coding scheme, a number of attempts have been made in designing coding schemes for secure communication over wiretap channels. For example, Liu *et al.* [58] propose the use of secure nested coding schemes for secure communication over a type II Gaussian wiretap channel, based on cosets of a good code sequence or cosets of the dual of a good code sequence.

Orthogonal codes have been widely studied for various applications. It has been proven

that the error probability of the orthogonal codes can be made arbitrarily small by making the number of the codewords tends to infinity for fixed information rate R , and orthogonal codes achieve the secrecy capacity bound as the number of codeword goes to infinity in the Additive white Gaussian noise (AWGN) channel in wideband regime, when the information rate $R < C_\infty$, in which C_∞ is the capacity of the infinite-bandwidth AWGN channel [64]. It is interesting to study the reliability and security performance of the orthogonal codes, when they are used in Wyner's coding structure in the AWGN wiretap channel.

This chapter focuses on secure communication over degraded Gaussian wiretap channels in the wideband regime, which has been studied in detail in [65]. In particular, by capitalizing on Wyner's code construction, the objective is to show that an orthogonal coding scheme is able to achieve the secrecy capacity of the Gaussian wiretap channel in the wideband regime.

3.2 The Gaussian wiretap channel in the wideband regime

The channel we consider in this chapter is the Gaussian wiretap channel, which has been introduced in Chapter 2 (Figure 2.7). The secrecy capacity of the Gaussian wiretap channel is given in (2.17). In this section, we analyse the properties of Gaussian wiretap channel in wideband regime, in which the bandwidth is infinite. In [66], it shows that determining the infinite-bandwidth capacity is equivalent to finding the minimum energy per bit required to transmit information reliably.

3.2.1 Tradeoff C_s vs. ξ_b/N_0 in the wideband regime

The tradeoff between the secrecy capacity C_s and the ratio between the energy per bit to the one-sided noise spectral density ξ_b/N_0 , which mirrors the tradeoff between bandwidth and power, is a useful characteristic of the secrecy capacity of the Gaussian wiretap channel. Such a characterization is particularly important in the wideband regime, in order to capture the interplay between the notions of power, bandwidth and rate [66]. In view of the fact that $P = RB\xi_b$, the secrecy capacity in (2.17) is also given by:

$$C_s = \log_2 \left(1 + \frac{\xi_b}{N_0} \cdot R \right) - \log_2 \left(1 + \alpha \cdot \frac{\xi_b}{N_0} \cdot R \right) \quad (3.1)$$

so that, by assuming that the communication rate is equal to the highest possible secrecy rate, i.e., $R = C_s$, one obtains immediately the tradeoff C_s vs. ξ_b/N_0 as follows:

$$\frac{\xi_b}{N_0} = \frac{2^{C_s} - 1}{C_s} \cdot \frac{1}{(1 - \alpha \cdot 2^{C_s})} \quad (3.2)$$

Figure 3.1 shows the tradeoff between C_s and ξ_b/N_0 for various values of $\alpha \in [0, 1]$. The figure portrays well the price of secrecy in the sense that the energy required to achieve a certain secrecy capacity is higher for a wiretap channel ($\alpha > 0$) than for the standard Gaussian channel ($\alpha = 0$). In particular, the higher the value of α the higher the value of ξ_b/N_0 required to achieve a target C_s .

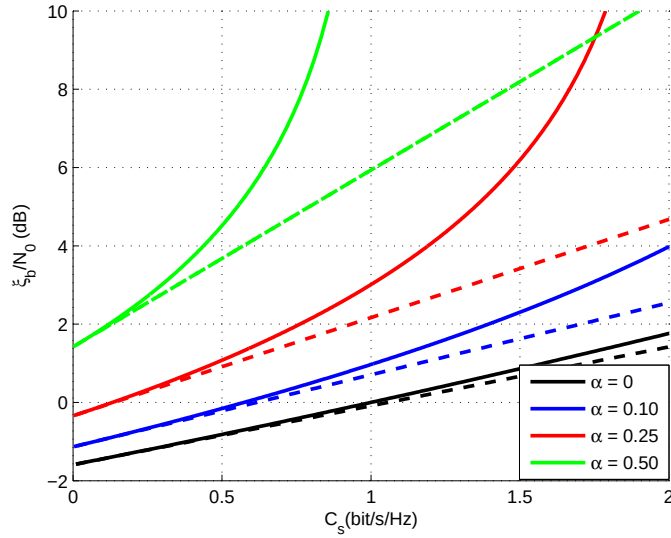


Figure 3.1: Tradeoff between C_s and ξ_b/N_0 for various values of α . Solid lines represent the true curve. Dashed lines represent the affine expansion.

Of particular relevance is also the characterization of the tradeoff between C_s and ξ_b/N_0 in the wideband regime, where $R \rightarrow 0$. The appropriate characterization expands the secrecy capacity as follows [65, 66]:

$$C_s = \mathcal{S}_0 \cdot \left(\frac{\left. \frac{\xi_b}{N_0} \right|_{\text{dB}}}{3 \text{ dB}} - \frac{\left. \frac{\xi_b}{N_0} \right|_{\text{min dB}}}{3 \text{ dB}} \right) + o \left(\left. \frac{\xi_b}{N_0} \right|_{\text{dB}} - \left. \frac{\xi_b}{N_0} \right|_{\text{min dB}} \right) \quad (3.3)$$

where $\left. \frac{\xi_b}{N_0} \right|_{\text{dB}}$ is the (transmitted) energy per information bit $\frac{\xi_b}{N_0}$ in dB, $\left. \frac{\xi_b}{N_0} \right|_{\text{min dB}}$ is the minimum (transmitted) energy per information bit $\frac{\xi_b}{N_0 \text{ min}}$ required for reliable and secure communication in dB and $\mathcal{S}_0$ is the slope therein in bit/s/Hz/(3 dB).

The two fundamental wideband performance measures, $\frac{\xi_b}{N_0 \min}$ and $\mathcal{S}_0$, can be obtained directly from (3.2), as follows:

$$\frac{\xi_b}{N_0 \min} = \lim_{C_s \rightarrow 0} \frac{2^{C_s} - 1}{C_s} \cdot \frac{1}{(1 - 2^{C_s} \cdot \alpha)} = \frac{\ln 2}{1 - \alpha} \quad (3.4)$$

$$\mathcal{S}_0 = \lim_{C_s \rightarrow 0} \frac{1}{\frac{d}{dC_s} 10 \log_{10} \frac{\xi_b}{N_0}(C_s)} \cdot 10 \log_{10} 2 = 2 \cdot \frac{1 - \alpha}{1 + \alpha} \quad (3.5)$$

Alternatively, these wideband measures can also be obtained from a characterization of the secrecy capacity in terms of the signal-to-noise ratio [65]. Note that when $\alpha = 1$ then $\frac{\xi_b}{N_0 \min} = +\infty$ and $\mathcal{S}_0 = 0$, so that, as expected, secure communication is not possible; on the other hand, when $\alpha = 0$ the quantities $\frac{\xi_b}{N_0 \min}$ and $\mathcal{S}_0$ reduce to the corresponding wideband measures of the standard Gaussian channel. The affine expansions are also depicted in Figure 3.1.

To establish a reliable and secure code scheme in wideband regime, we have to prove that for a given value of α , the lower bound of the required $\frac{\xi_b}{N_0}$ is equal to $\frac{\xi_b}{N_0 \min}$ in (3.4) to guarantee that the codeword error probability tends to zero and the equivocation rate tends to the information rate, which is equal to C_s in wideband regime, as the length of the codeword, n , tends to infinity.

It will also be instructive to re-examine the structure of Wyner's coding scheme in the wideband regime.

3.2.2 Wyner's coding scheme in the wideband regime

Let's consider communication in the wideband regime at a rate $R = C_s$ with $\frac{\xi_b}{N_0} \Big|_{\text{dB}} \rightarrow \frac{\xi_b}{N_0 \min} \Big|_{\text{dB}}$. Let us write:

$$\frac{\xi_b}{N_0} \Big|_{\text{dB}} = \frac{\xi_b}{N_0 \min} \Big|_{\text{dB}} + \Delta \Big|_{\text{dB}} \quad (3.6)$$

Therefore, as $\Delta \Big|_{\text{dB}} \rightarrow 0$ the secrecy capacity in (3.3) can be written as follows:

$$C_s = \mathcal{S}_0 \cdot \frac{\Delta \Big|_{\text{dB}}}{3 \text{ dB}} + o(\Delta \Big|_{\text{dB}}) \quad (3.7)$$

As $\Delta|_{\text{dB}} \rightarrow 0$ the capacities of the main and eavesdropper channel also admit the expansions:

$$\begin{aligned} C_M &= \log_2 \left(1 + \mathcal{S}_0 \cdot \frac{\Delta|_{\text{dB}}}{3 \text{ dB}} + o(\Delta|_{\text{dB}}) \cdot \frac{\xi_b}{N_0} \cdot \left(1 + \frac{\ln 10}{10} \Delta|_{\text{dB}} + o(\Delta|_{\text{dB}}) \right) \right) \\ &= \frac{1}{\ln 2} \cdot \mathcal{S}_0 \cdot \frac{\Delta|_{\text{dB}}}{3 \text{ dB}} \cdot \frac{\xi_b}{N_{0 \min}} + o(\Delta|_{\text{dB}}) \end{aligned} \quad (3.8)$$

where (3.8) follows from Taylor's expansion $\log_2(1+x) = \frac{1}{\ln 2} \cdot x + o(x)$ as $x \rightarrow 0$, and

$$C_W = \log_2 \left(1 + C_s \cdot \frac{\xi_b}{N_0} \cdot \alpha \right) \quad (3.9)$$

$$= \log_2 \left(1 + \mathcal{S}_0 \cdot \frac{\Delta|_{\text{dB}}}{3 \text{ dB}} + o(\Delta|_{\text{dB}}) \cdot \frac{\xi_b}{N_0} \cdot \left(1 + \frac{\ln 10}{10} \Delta|_{\text{dB}} + o(\Delta|_{\text{dB}}) \right) \cdot \alpha \right) \quad (3.10)$$

$$= \frac{\alpha}{\ln 2} \cdot \mathcal{S}_0 \cdot \frac{\Delta|_{\text{dB}}}{3 \text{ dB}} \cdot \frac{\xi_b}{N_{0 \min}} + o(\Delta|_{\text{dB}}) \quad (3.11)$$

The structure of Wyner's coding scheme as $\frac{\xi_b}{N_0}|_{\text{dB}} \rightarrow \frac{\xi_b}{N_{0 \min}}|_{\text{dB}}$ follows from the expansions in (3.7), (3.8) and (3.11). In particular, the total number of sub-codebooks as $\Delta|_{\text{dB}} \rightarrow 0$ is given by:

$$N_g = 2^{n C_s} = 2^{n \cdot \mathcal{S}_0 \cdot \frac{\Delta|_{\text{dB}}}{3 \text{ dB}} + o(\Delta|_{\text{dB}})} \approx 2^{n \cdot \mathcal{S}_0 \cdot \frac{\Delta|_{\text{dB}}}{3 \text{ dB}}} \quad (3.12)$$

The total number of codewords in the codebook and the total number of codewords per sub-codebook as $\Delta|_{\text{dB}} \rightarrow 0$ are given by:

$$N_c = 2^{n C_M} = 2^{n \cdot \frac{1}{\ln 2} \cdot \mathcal{S}_0 \cdot \frac{\Delta|_{\text{dB}}}{3 \text{ dB}} \cdot \frac{\xi_b}{N_{0 \min}} + o(\Delta|_{\text{dB}})} \approx N_g^{\frac{1}{1-\alpha}} \quad (3.13)$$

$$N_{c/g} = 2^{n C_W} = 2^{n \cdot \frac{\alpha}{\ln 2} \cdot \mathcal{S}_0 \cdot \frac{\Delta|_{\text{dB}}}{3 \text{ dB}} \cdot \frac{\xi_b}{N_{0 \min}} + o(\Delta|_{\text{dB}})} \approx N_g^{\frac{\alpha}{1-\alpha}} \quad (3.14)$$

3.3 Code construction

We use n orthogonal codewords with length n and with average power P to construct the coding scheme:

$$\begin{aligned} X_1 &= [\sqrt{nP} \ 0 \ \cdots \ 0 \ 0] \\ X_2 &= [0 \ \sqrt{nP} \ 0 \ \cdots \ 0] \\ &\vdots \\ X_n &= [0 \ 0 \ \cdots \ 0 \ \sqrt{nP}] \end{aligned}$$

Based on Wyner's code construction, we divide the set of codewords – the codebook $\mathcal{C}$ – into several equal size sub-sets of codewords – the sub-codebooks $\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_{N_g}$. Sub-codebook $\mathcal{C}_1$ consists of orthogonal codewords $X_1, X_2, \dots, X_{N_{c/g}}$; sub-codebook $\mathcal{C}_2$ consists of orthogonal codewords $X_{N_{c/g}+1}, X_{N_{c/g}+2}, \dots, X_{2 \cdot N_{c/g}}$; etc. Message $W = 1$ is mapped with equal probability to any of the codewords in sub-codebook $\mathcal{C}_1$; message $W = 2$ is mapped with equal probability to any of the codewords in sub-codebook $\mathcal{C}_2$; etc.

The energy per codeword of the orthogonal code is: $\xi_s = nP/B$ and the energy per bit is: $\xi_b = \frac{\xi_s}{\log_2 N_g}$.

In the wideband regime as $\frac{\xi_b}{N_0}|_{\text{dB}} \rightarrow \frac{\xi_b}{N_{0 \min}}|_{\text{dB}}$, the total number of codewords in the codebook, the total number of codewords per sub-codebook, and the number of sub-codebooks, obey the relations put forth in (3.7), (3.13) and (3.14). It is also important to note that due to the orthogonality constraint the total number of codewords is equal to the length of the codewords, i.e., $N_c = n$.

It is also possible to show that communication in the wideband regime at a rate $R = C_s$ with $\frac{\xi_b}{N_0}|_{\text{dB}} \rightarrow \frac{\xi_b}{N_{0 \min}}|_{\text{dB}}$, in view of (3.7), (3.13) and (3.14), requires orthogonal codewords with length n such that:

$$\frac{1}{n} \cdot \log_2 n \approx \frac{1}{1 - \alpha} \cdot \mathcal{S}_0 \cdot \frac{\Delta|_{\text{dB}}}{3 \text{ dB}} \quad (3.15)$$

This implies, as expected, that as $\frac{\xi_b}{N_0}|_{\text{dB}} \rightarrow \frac{\xi_b}{N_{0 \min}}|_{\text{dB}}$ then $n \rightarrow \infty$.

3.4 Error probability analysis

We analyze the error probability between the legitimate transmitter and the legitimate receiver, by assuming that the receiver is equipped with the optimal detector for a set of orthogonal codewords. The optimal detector selects the codeword that leads to the largest cross-correlation between the receive codeword and each of the possible transmit codewords, i.e.,

$$\hat{X} = \arg \max_{X \in \{X_1, X_2, \dots, X_n\}} |C(X, Y_M)| \quad (3.16)$$

where

$$C(X, Y_M) = \sum_{i=1}^n Y_M(i) X(i) \quad (3.17)$$

Due to the nature of the encoding and decoding process, where at the transmitter the original message W is mapped randomly to a codeword within sub-codebook $\mathcal{C}_W$ and at the receiver a codeword estimate within sub-codebook $\mathcal{C}_{\hat{W}}$ is mapped into the message $\hat{W}$, an error occurs when the sub-codebook of the codeword estimate differs from the sub-codebook of the original codeword. The error probability between the original message at the legitimate transmitter and the estimate of the message at the legitimate receiver can then be expressed as follows:

$$P_e = \Pr(e) = \frac{1}{n} \sum_{i=1}^n \Pr(e|X = X_i) \quad (3.18)$$

and the conditional error probabilities, which are equal due to symmetry considerations, can be expressed as follows:

$$\Pr(e|X = X_1) = \Pr(\hat{X} \neq X_1 \wedge \dots \wedge \hat{X} \neq X_{N_{c/g}} | X = X_1) \quad (3.19)$$

Let us now note that:

$$\Pr(\hat{X} \neq X_1 \wedge \dots \wedge \hat{X} \neq X_{N_{c/g}} | X = X_1) \quad (3.20)$$

$$= 1 - \Pr(\hat{X} = X_1 \vee \dots \vee \hat{X} = X_{N_{c/g}} | X = X_1) \quad (3.21)$$

$$\leq 1 - \Pr(\hat{X} = X_j | X = X_1) \quad (3.22)$$

for any $j = 1, \dots, N_{c/g}$. Consequently, the error probability can be upper bounded as follows:

$$P_e \leq 1 - \Pr(\hat{X} = X_j | X = X_1) \quad (3.23)$$

for any $j = 1, \dots, N_{c/g}$.

The probability in (3.23) can be further upper bounded by using the techniques in [64], which leads to the upper bound to the probability of error given by:

$$P_e \leq \begin{cases} 2e^{-k_1(\gamma_b - 2\ln 2)/2}, & \ln n \leq \frac{\gamma}{4N} \\ 2e^{-k_1(\sqrt{\gamma_b} - \sqrt{\ln 2})^2}, & \frac{\gamma}{4N} \leq \ln n \leq \gamma \end{cases} \quad (3.24)$$

in which $\gamma = \frac{\xi_s}{N_0} = \frac{nP}{N}$ and $\gamma_b = \frac{\gamma}{\log_2 n}$, we have:

$$P_e \leq \begin{cases} 2e^{-k_1\left(\frac{nP}{\log_2 n} \cdot \frac{1}{N} - 2\ln 2\right)/2}, & \ln n \leq \frac{nP}{4N} \\ 2e^{-k_1\left(\sqrt{\frac{nP}{\log_2 n} \cdot \frac{1}{N}} - \sqrt{\ln 2}\right)^2}, & \frac{nP}{4N} \leq \ln n \leq \frac{nP}{N} \end{cases} \quad (3.25)$$

where $k_1 = \log_2 n$. Note that the first upper bound is loose for large values of n , whereas the second upper bound is tight for large values of n .

It follows immediately that as $n \rightarrow \infty$ then $P_e \rightarrow 0$ provided that:

$$\frac{nP}{\log_2 n} \cdot \frac{1}{N} > \ln 2 \quad (3.26)$$

or, using the fact that $nP = B\xi_s = B\xi_b \log_2 N_g$, $N = N_0B$, $N_c = n$, $R = C_s$ and the relations in (3.13) and (3.14), provided that:

$$\frac{\xi_b}{N_0} > \frac{\ln 2}{1 - \alpha} = \frac{\xi_b}{N_{0\min}} \quad (3.27)$$

3.5 Equivocation analysis

We analyze the eavesdropper equivocation to determine its uncertainty about the message. Let us determine a lower bound to the eavesdropper equivocation as follows:

$$\begin{aligned} H(W|Y_E) &= H(W, Y_E) - H(Y_E) \\ &= H(W, X, Y_E) - H(X|W, Y_E) - H(Y_E) \\ &= H(X) + H(Y_E|X) + H(W|X, Y_E) \\ &\quad - H(X|W, Y_E) - H(Y_E) \\ &\geq H(X) - H(X|Y_M) - H(Y_E) + H(Y_E|X) \\ &\quad - H(X|W, Y_E) \\ &= I(X; Y_M) - I(X; Y_E) - H(X|W, Y_E) \end{aligned} \quad (3.28)$$

Now, based on (3.1) and (3.12), it is possible to show that in the wideband regime the mutual information between the input and output of the main channel and the mutual information between the input and the output of the eavesdropper channel for the orthogonal code construction admit the expansions:

$$I(X; Y_M) \approx \log_2 N_g \cdot \frac{\xi_b}{N_0} \cdot \frac{1}{\ln 2} \quad (3.29)$$

$$I(X; Y_E) \approx \log_2 N_g \cdot \frac{\xi_b}{N_0} \cdot \frac{1}{\ln 2} \cdot \alpha \quad (3.30)$$

so that

$$I(X; Y_M) - I(X; Y_E) \approx \log_2 N_g \cdot \frac{\xi_b}{N_0} \cdot \frac{1}{\ln 2} \cdot (1 - \alpha) \quad (3.31)$$

It is also possible to show, by using Fano's inequality [67], that:

$$H(X|W, Y_E) \leq 1 + \Pr(\hat{X} \neq X|W) \cdot \log_2 N_{c/g} \quad (3.32)$$

where $\Pr(\hat{X} \neq X|W)$ represents the probability of error at the eavesdropper given the message. Consequently, one immediately infers from (3.31) and (3.32) that in the wideband regime as $\frac{\xi_b}{N_0}|_{\text{dB}} \rightarrow \frac{\xi_b}{N_{0\min}}|_{\text{dB}}$, which also implies $n \rightarrow \infty$ via (3.15), the eavesdropper equivocation rate obeys:

$$\frac{1}{n} \cdot H(W|Y_E) \geq \frac{1}{n} \cdot \log_2 N_g - \epsilon = C_s - \epsilon \quad (3.33)$$

where ϵ is negligible.

To conclude, we observe that in the wideband regime as $\frac{\xi_b}{N_0}|_{\text{dB}} \rightarrow \frac{\xi_b}{N_{0\min}}|_{\text{dB}}$ the orthogonal code construction defined by (3.13), (3.14) and (3.15) guarantees communication at a rate $R = C_s$ (with $C_s \rightarrow 0$) with perfect reliability and security.

3.6 Simulation results

We provide various simulation results to illustrate the performance of the orthogonal code construction in the Gaussian wiretap channel. Figure 3.2 plots the (codeword) error probability between the legitimate transmitter and the legitimate receiver *vs.* ξ_b/N_0 for various values of N_g with $\alpha = \frac{N_{M0}}{N_{E0}} = 0.08$. We observe that the higher the value of N_g (and hence n) the lower the value of ξ_b/N_0 required to achieve a target (message) error probability. In particular, in accordance with the analysis, one would obtain that $P_e \rightarrow 0$

as $N_g \rightarrow \infty$ (and hence $n \rightarrow \infty$) provided that $\frac{\xi_b}{N_0} \geq \frac{\xi_b}{N_{0 \min}} = \frac{\ln 2}{1-\alpha} = -1.23$ dB. Figure 3.3 plots the eavesdropper equivocation lower bound using (3.28) and (3.32) vs. ξ_b/N_0 for various values of N_g with $\alpha = \frac{N_{M0}}{N_{E0}} = 0.08$. In particular, one observes that the eavesdropper equivocation tends to the message entropy in a vicinity of $\frac{\xi_b}{N_0} = \frac{\xi_b}{N_{0 \min}}$ for large values of N_g (and hence n). This is further emphasized in Figure 3.4, which depicts the gap (denoted by γ) between the message entropy and the eavesdropper equivocation lower bound, normalized to the message entropy.

These simulation results suggest that the orthogonal code construction achieve the secrecy capacity of the Gaussian wiretap channel in the wideband regime as $\frac{\xi_b}{N_0} \rightarrow \frac{\xi_b}{N_{0 \min}}$. The price, of course, is reliable and secure communications at a vanishing information rate.

Finally, Table 3.1 depicts the tradeoff between rate, reliability and security for $P_e = 10^{-2}$ and $P_e = 10^{-3}$. As expected, as the information rate tends to zero the higher the reliability (i.e. the lower the ξ_b/N_0 necessary to attain a target error probability) and the higher the security (i.e. the lower the equivocation upper bound). We also observe that for a certain target information rate the higher the reliability level the lower the security level.

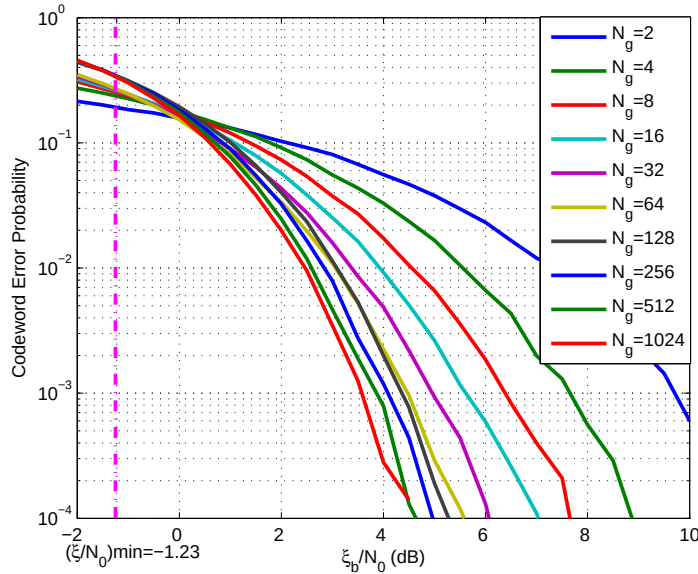


Figure 3.2: Codeword error probability between the legitimate parties for $\alpha = 0.08$.

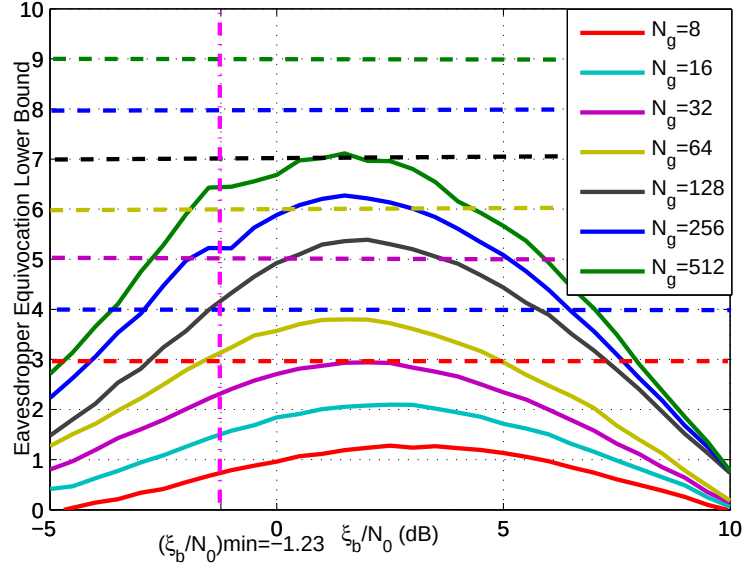


Figure 3.3: Eavesdropper equivocation lower bound for $\alpha = 0.08$. Solid lines correspond to the equivocation lower bound. Dashed lines correspond to the message entropy.

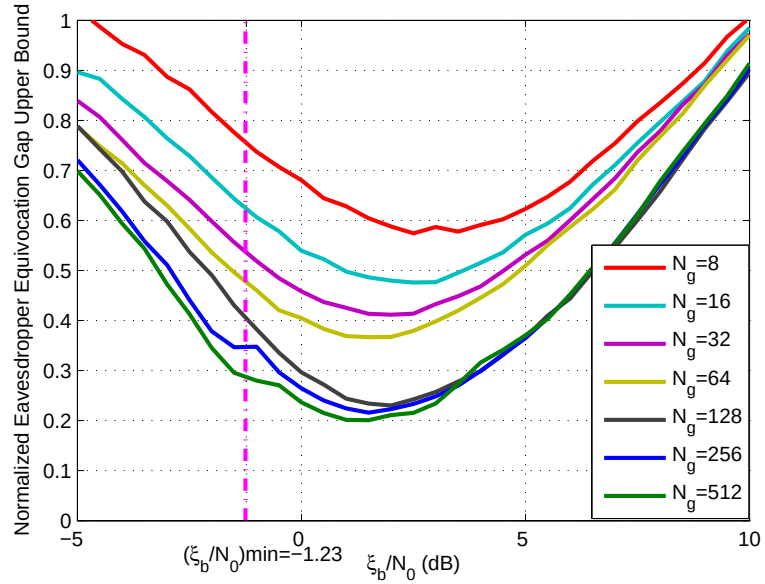


Figure 3.4: Normalized eavesdropper equivocation gap upper bound γ for $\alpha = 0.08$. γ is obtained by using the equivocation lower bound in $\frac{H(W) - H(W|Y_E)}{H(W)}$.

Table 3.1: Tradeoff between rate, reliability and security

N_g	8	16	32	64	128	256	512
R	0.375	0.250	0.156	0.094	0.055	0.031	0.018
ξ_b/N_0 (dB)@ $P_e = 10^{-2}$	4.4	3.8	3.5	3.1	2.9	2.8	2.6
ξ_b/N_0 (dB)@ $P_e = 10^{-3}$	6.2	5.6	4.9	4.6	4.3	4.2	4
γ @ $P_e = 10^{-2}$	0.6	0.51	0.45	0.4	0.25	0.24	0.22
γ @ $P_e = 10^{-3}$	0.7	0.6	0.53	0.48	0.32	0.31	0.3

3.7 Conclusions

In this chapter, we have considered communication in the Gaussian wiretap channel in the wideband regime. In particular, it has been shown by analysis and suggested by simulation that the use of orthogonal codes, which follow a construction *akin* to Wyner's code construction, can achieve the secrecy capacity of the Gaussian wiretap channel in the asymptotic wideband regime, which entails communication at a vanishing rate. The results can also be used to infer the error probability between the legitimate parties or bound the information rate leaked to the eavesdropper due to the use of the orthogonal coding construction in non-asymptotic regimes.

Chapter 4

Syndrome Coding Based on Best Equivocation Code Construction

In this chapter, we propose a design technique to produce the best codes for the syndrome coding scheme as measured by an information secrecy metric: the equivocation. An efficient recursive method to determine the equivocation for the Binary Symmetric Channel and any linear binary code is also presented. Some Best Equivocation Codes for the syndrome coding scheme are thus produced. It is shown that the Best Equivocation Codes have a better secrecy performance for the syndrome coding scheme than the best known codes.

4.1 Introduction

Wyner proposed a syndrome coding scheme (also known as 'coset coding scheme') to guarantee the security of the communication for BSWC [12]. Wyner showed that as long as the size of the syndrome space is chosen to be smaller than the Shannon entropy of the binary symmetric channel, there exist codes that leak a vanishing proportion of information to the eavesdropper as the code length approaches infinity. In [68], Bennett strengthens Wyner's result in the sense that the channels are more general and the estimate of the number of information bits leaked to Eve is stronger. Cohen and Zemor [69] also provide a more refined analysis of the information leakage of syndrome coding for the wiretap channel. Recently, Cassuto and Bandic contribute to the study of wire-tap II codes in two directions: for the case of error-free main channels, two families of codes are constructed with optimal encoding and decoding complexities for their wire-tap security;

for the case of main channels with errors, two concatenation types are studied for the wire-tap and error correcting codes [70].

The syndrome coding scheme is an important physical layer scheme to guarantee the secure communication that has been widely studied for various applications. For example, Cohen and Zemor [13] studied a generalization of Wyner's syndrome coding scheme which is applicable to noisy main and eavesdropper channels, for biometric applications. Cohen and Zemor also prove that this scheme achieves the Shannon-capacity of the system. Chabanne *et al.* have extended the analysis of the syndrome coding as a non-malleable code, following a security analysis close to the one of Secure Network Coding [71]. Luo *et al.* study the equivocation of the data symbols to the eavesdropper who has more power than that of Wyner, and introduce some new concepts to analyse the secrecy on the wire-tap channel of type II [72]. Reddy *et al.* [14] studied low-density parity-check (LDPC) codes for syndrome coding with applications to video coding systems. Suresh *et al.* [15] showed that duals of certain LDPC codes, when used in a syndrome coding scheme, provide strong secrecy over the binary erasure wiretap channel. Salim and Emina showed that network security can be achieved by using syndrome coding as an additional layer to a network code [16]. In [73], Song *et al.* propose a secret key agreement scheme which adopts the idea of the syndrome coding scheme to achieve perfect secrecy over the binary erasure wiretap channel of type II.

For the syndrome coding scheme, it is important to construct the good codes to guarantee the security of the communication. Traditionally, good codes are designed for an error correcting system to combat noisy transmission channels and achieve reliable communication for different systems [74, 75], which focus on designing codes with maximum Hamming distance, such as BKC [45]. Codes designed for error correction can also be used in syndrome coding, but it is shown below that the best performance is not generally achieved and better codes exist. The information rate of a syndrome coding scheme using an (n, k, d) linear code is $\frac{n-k}{n}$ and all possible binary vectors of length n may be transmitted. This is quite different from error correction coding applications where only codewords are transmitted and the information rate is k/n . For the syndrome coding, performance is measured in terms of the level of information theoretic secrecy, the equivocation [12].

In this chapter we restrict the choice of codes to binary, linear codes. We present a numerical design technique to produce the best codes for the syndrome coding scheme as measured by the equivocation rate and compare these codes to the best known error

correcting codes. The technique is limited practically to codes having around 35 parity bits, but this is not a major limitation as near 100% secrecy is attainable with code lengths less than 256 bits [76]. The BKC error correcting code tables list codes less than 256 bits long, which make the comparisons possible. We also present an efficient recursive method to analyse the secrecy performance of any linear binary code, which is used in syndrome coding for the Binary Symmetric Channel. Finally, we present performance results of some best equivocation codes for the syndrome coding scheme and compare these to the best known codes. In almost all cases there is a substantial improvement.

4.2 Background of code design

Code design is one of the important topics in reliable communications. Numerous contributions have been made in designing good code families such as BCH and Goppa codes [77] or by modifying good codes by concatenation [46], extension [47–49, 78] or shortening [77]. Traditionally, good codes are designed for an error correcting system to combat noisy transmission channels and achieve reliable communication for different systems [74, 75].

Historically many code design methods for linear codes, which have been presented in the literature, are based on construction of the parity check matrix, $\mathbf{H}$ [47–49]. Construction X and its variations increase the length of a code in order to enlarge the minimum weight, which use two or more codes of the same length where one code contains the other codes which have a higher minimum weight [48, 49]. Construction Y_1 is a construction method which decreases both the length and the dimension of a code [77].

Any column of a parity check matrix of a $(n, k, d = 2t + 1)$ code, $\mathbf{H}$, can be represented by an integer, b_i , in the range 0 to $2^{n-k} - 1$, in which i is the index of the column and $0 \leq i < n$. The parity check matrix of a $(n, k, d = 2t + 1)$ code is defined by n integers, referred to as packed integers, and these integers can be in any order since the corresponding codes will all be equivalent [77]. Any code design of length n may be represented by these n packed integers. Since any parity check matrix can be put into reduced echelon form by elementary row and column matrix operations, the first m packed integers are fixed and the code design reduces to the determination of $n - m$

packed integers. Let us start with a general binary, reduced echelon parity check matrix:

$$\mathbf{H} = \begin{pmatrix} 1 & 0 & \cdots & 0 & a_{m0} & \cdots & a_{(n-1)0} \\ 0 & 1 & \cdots & 0 & a_{m1} & \cdots & a_{(n-1)1} \\ \vdots & \vdots & \cdots & \vdots & \vdots & a_{ij} & \vdots \\ 0 & 0 & \cdots & 1 & a_{m(m-1)} & \cdots & a_{(n-1)(m-1)} \end{pmatrix} \quad (4.1)$$

in which $a_{i,j}$ takes a value of 0 or 1, i and j denote the index of the rows and the columns respectively, $0 \leq j \leq m-1$ and $m \leq i \leq n-1$. Each column may be represented as a packed integer defined as $b_i = \sum_{j=0}^{m-1} a_{ij} \cdot 2^j$. The systematic packed integer form of the parity check matrix is $[1, 2, \dots, 2^{m-1}, b_m, \dots, b_{n-1}]$. For example, the parity check matrix of the (15,7,5) BCH code is:

$$\mathbf{H} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{pmatrix} \quad (4.2)$$

which in packed integer representation is:

$$\mathbf{H} = [1, 2, 4, 8, 16, 32, 64, 128, 27, 47, 117, 202, 166, 71, 149] \quad (4.3)$$

A central property of an error correcting code is the minimum Hamming distance, d , which determines the number of independent errors that can be corrected and the resulting reliability of communication. The main objective in error correcting code design is to optimise one of the parameters n , k and d for given values of the other two [78]. Usually this optimization is related to the following functions: $n(k, d)$ - the minimum length of linear codes for given minimum distance d and dimension k and $d(n, k)$ the largest value of the minimum Hamming distance for which a binary (n, k, d) code exists. The covering radius R_c is usually not considered a design parameter in traditional error correcting code design, but it is an indicator of codeword packing [77].

Based on this central target of error correcting, several techniques have been used to solve different error correcting problems. For example, in [79], Dontas and Jong describe some experiments on the use of genetic algorithm to discover maximal distance codes for the

problems of knowing limited characteristics of the codes. In [80], scatter search and memetic algorithms have been used to solve the error correcting code problem. In [81], Miller and Wolf describe a method of using a combination of interleaved Reed-Solomon codes interspersed with uncoded bits to achieve a higher code rate than the classical error correcting codes solution. Evolutionary algorithms have been also used to get optimum error correcting codes by Alba and Khuri [82, 83].

Based on different techniques of code design, optimum binary codes are published in an online database by Grassl as Best Known Codes (BKC) in the form of tables of lower and upper bounds to d [45] for different code rates and code lengths up to 256 bits. With so much research attention historically, improvements to these tables are relatively rare.

There is a well known relationship between the parity check matrix and the minimum Hamming distance of the code from first studies of error correction coding [84]:

Property 1. *A linear code has d minimum distance if and only if its parity check matrix has d linearly dependent columns but no set of $d-1$ or fewer, linearly dependent columns.*

Based on *Property 1*, any packed integer of the parity check matrix cannot be equal to the modulo 2 sum of any combination of $d-2$ packed integers or fewer integers of the parity check matrix. This is apparent by observing that the syndrome of a codeword is equal to zero. If any packed integer of the parity check matrix were equal to the modulo 2 sum of any combination of $d-2$ packed integers or fewer integers then the resulting combination would have a syndrome equal to zero and therefore be a codeword of weight less than d , an obvious contradiction.

Some code extension method are based on *Property 1*. For example, Bouyukliev's code construction method extends a code by excluding all linear combinations of up to $d-2$ columns of the parity check matrix [78] exploiting *Property 1*. The inverting construction Y_1 , proposed by Edel adds an additional parity bit and extends a (n, k, d) code to a $(n+x, k+x-1, d)$ code, by extending the parity check matrix [47] similar to Bouyukliev.

Until now, most of the code design methods only consider about the reliable communication. The secure coding schemes, which are designed for the wiretap channel, usually employ the error correcting code. However, there are rare research on the design of the best code to guarantee the security of the communication, which is very necessary for achieving higher levels of communication security and is a promising new direction on the field of the physical layer security. This is the main target of this chapter: we aim to

construct the best codes for the syndrome coding scheme, which have the highest value of equivocation rate for a given code length and code rate. It is convenient to use the parameter $m = n - k$, to represent the number of parity bits of the code. A best code for syndrome coding is represented by the functions: $n(m, R_e)$ - the shortest code for a given R_e and m and $R_e(n, m)$ - the highest equivocation rate for which a binary $(n, k = n - m)$ code exists. In contrast, a best code for error correction is represented by the function $n(m, d, R_c)$ - the longest code for a given d , m and R_c . Based on a list of parity check matrices of best performing codes for the syndrome coding scheme for given m and n , we aim to generate a Best Equivocation Codes (BEC) database in the form of n , m and $R_e(n, m)$.

4.3 Syndrome coding scheme without look-up table

Traditionally, the syndrome coding scheme uses a $(n, k, d = 2t + 1)$ linear block code, which is capable of correcting t errors, defined by a $m \times n$ parity check matrix $\mathbf{H}$ [61]. It is a property of any linear block code that there exist 2^m distinct n -bit minimum weight error patterns, in which each pattern e_j produces a distinct syndrome S_j of the total 2^m syndromes, based on $S_j = e_j \times \mathbf{H}^T$, where $0 \leq j \leq 2^m - 1$. These minimum weight error patterns, e_j , are the coset leaders of the code and may be represented in a table of 2^m syndromes, S_j , and associated minimum weight error patterns, e_j . The covering radius, R_c , is equal to the maximum weight of all 2^m coset leaders. For perfect codes, the error patterns, e_j , in the syndrome table are all of the n -bit binary error vectors with a weight $w \leq t$, so $R_c = t$. In the traditional syndrome coding, it is necessary to store an error pattern-syndrome look up table, which can be accessed by Alice, Bob and Eve. However, it is shown below that such a look up table is unnecessary and the parity check matrix of the code is sufficient.

With some elementary row and column operations, every parity check matrix can be represented in a reduced echelon form [77]. In the following it is assumed that the parity check matrix of the code is in the reduced echelon form with an identity sub-matrix of m parity bits followed by k information bit columns.

We consider the binary symmetric wiretap channel, in which the eavesdropper channel is a binary symmetric channel with an crossover probability α (Figure 2.6). The encoding and decoding processes are as follows.

4.3.1 Encoder

Alice equates the m -bit messages to m -bit syndromes $S_T(i) = M(i)$ and produces n -bit vectors $X(i)$ from each m -bit message $M(i)$ at time i such that $X(i) \times \mathbf{H}^T = M(i)$ in three steps:

- Alice generates a random n -bit codeword $C_T(i)$ from a random, uniformly distributed k -bit vector $D_R(i)$:

$$C_T(i) = D_R(i) \times \mathbf{G} \quad (4.4)$$

- Alice forms an n -bit zero padded message vector, $V(i)$, consisting of the m -bit message $M(i)$ followed by k 0's.
- Alice then generates the transmitted n -bit vector $X(i)$ by adding the n -bit codeword $C_T(i)$ to the n -bit zero padded message vector

$$X(i) = C_T(i) + V(i) \quad (4.5)$$

In this way Alice produces n -bit vectors with the property that $X(i) \times \mathbf{H}^T = M(i)$.

Proof. The syndrome of $X(i)$, $S_T(i)$ satisfies

$$\begin{aligned} S_T(i) &= X(i) \times \mathbf{H}^T \\ &= C_T(i) \times \mathbf{H}^T + V(i) \times \mathbf{H}^T \\ &= 0 + M(i) \end{aligned} \quad (4.6)$$

$V(i) \times \mathbf{H}^T$ produces $M(i)$, which is due to the structure of $V(i)$ and $\mathbf{H}$. $\square$

The information rate of the syndrome coding scheme is $R = \frac{m}{n}$.

4.3.2 Legitimate decoder

Bob calculates the syndromes of the received vectors, $Y_M(i)$ to determine the original messages:

$$S_{Bob}(i) = Y_M(i) \times \mathbf{H}^T \quad (4.7)$$

As there are no transmission errors in Bob's channel $Y_M(i) = X(i)$ and

$$S_{Bob}(i) = M(i) \quad (4.8)$$

4.3.3 Eavesdropper decoder

Eve computes the syndromes associated with the vectors, $Y_E(i)$, as follows:

$$S_{Eve}(i) = Y_E(i) \times \mathbf{H}^T = X(i) \times \mathbf{H}^T + e(i) \times \mathbf{H}^T = M(i) + S_e(i) \quad (4.9)$$

Eve's estimate of the message is given by:

$$\hat{M}(i) = S_{Eve}(i) = M(i) + S_e(i) \quad (4.10)$$

4.4 Calculation of the equivocation of a syndrome coding scheme

To simplify the calculation, the decoder output equivocation is employed as the metric of the security, which is given by:

$$S_d = H(M|\hat{M}(i)) \quad (4.11)$$

In [69], it has been proved that $H(M|Y_E) = H(M|\hat{M}(i))$, meaning that there is no advantage for the eavesdropper in possessing $Y_E(i)$ on top of its syndrome. We also give our proof that eavesdropper's decoder is maximum a posteriori probability (MAP) decoder for the syndrome coding scheme in Appendix A.

The eavesdropper decoder output equivocation can be calculated as follows:

$$H(M(i)|\hat{M}(i)) = H(M(i), \hat{M}(i)) - H(\hat{M}(i)) \quad (4.12)$$

$$= H(M(i)) - H(\hat{M}(i)) + H(\hat{M}(i)|M(i)) \quad (4.13)$$

$$= H(M(i)) - H(M(i) + S_e(i)) + H(M(i) + S_e(i)|M(i)) \quad (4.14)$$

$$= m - m + 0 + H(S_e(i)|M(i)) \quad (4.15)$$

$$= H(S_e(i)) \quad (4.16)$$

$$= - \sum_{j=0}^{2^m-1} p(S_j) \log_2 p(S_j) \quad (4.17)$$

in which, S_j denotes one of the total 2^m syndromes. The simplifications in equations (4.15) and (4.16) are due to $M(i)$ being uniformly distributed and independent of $S_e(i)$. The equivocation is calculated after deriving the probability mass function of the syndromes due to errors from the BSC, $p(S_j)$ and is a function of the parity check matrix of the code.

4.4.1 Analysis of the syndrome probability distribution

Equation (4.17) shows that it is necessary to get the probability mass function of the syndromes, $p(S_j)$, for the calculation of the equivocation in BSC. There are two ways to evaluate $p(S_j)$, in which $0 \leq j \leq 2^m - 1$.

4.4.1.1 Traditional evaluation method of the syndrome probability distribution

There are 2^n possible ways in which error patterns, $e(i)$, occur in each transmitted n -bit vector. These error patterns occur with probability:

$$p(e(i)) = \alpha^{w(e(i))} \cdot (1 - \alpha)^{n-w(e(i))} \quad (4.18)$$

where $w(e(i))$ is the weight of $e(i)$. Each error pattern results in one of the 2^m syndromes, which is produced as follows:

$$S_e(i) = e(i) \times \mathbf{H}^T \quad (4.19)$$

Any error pattern, which is a codeword, has a syndrome equal to zero. As the code is linear, for each particular syndrome, S_j , there are 2^k different error patterns, $e(i)$, that produce the same syndrome and the probability of each syndrome due to all possible such error patterns is given by

$$p(S_j) = \sum_{e(i)=0}^{2^n-1} p(e(i)) \delta(S_e(i) - S_j) \quad (4.20)$$

where S_j denotes a specific syndrome vector in all of the 2^m syndromes, $0 \leq j \leq 2^m - 1$, and $\delta()$ is the Dirac function. We have

$$H(S_e(i)) = - \sum_{j=0}^{2^m-1} p(S_j) \log_2 p(S_j) \quad (4.21)$$

This method for evaluating the equivocation is computationally manageable for short codes ($n < 40$), but for the long codes it is not practical because it involves the evaluation of 2^n error patterns, an exponential function of the code length. However the probability distribution of the syndromes may be determined recursively and this also provides some insight into code construction by code extension.

4.4.1.2 Recursive evaluation of the syndrome probability distribution

Let us represent the parity check matrix of the code in the packed integer format. The packed integers directly correspond to syndrome values in that a single bit error in a transmitted codeword results in a syndrome equal to the packed integer of the column corresponding to the bit error position. As the codes are linear, any combination of bit errors produces a syndrome equal to the modulo 2 sum of the packed integers corresponding to each column of the parity check matrix in which a bit error occurs.

The following theorem shows that the probability mass function of the syndrome, $p(S_j)$, is a function of the parity check matrix of the code and the crossover probability of the BSC.

Theorem 1. *The probability mass function (pmf) of S_j for $j = 0$ to $2^m - 1$ may be defined as $p(S_j) = \beta(j)$ where $\beta(j)$ are coefficients of the probability generating function using the Z transform, $p_z(S)$ and $p_z(S)$ only depends on the columns of the parity check matrix and α .*

$$p_z(S) = \sum_{j=0}^{2^m-1} \beta(j) Z^j = \prod_{i=0}^{n-1} ((1 - \alpha) + \alpha Z^{b_i}) \quad (4.22)$$

where b_i are the packed integer representations of the columns of the parity check matrix and exponent sums of powers of Z are modulo 2 sums.

Proof. Any error pattern may be represented as a sum of single bit error events:

$$\begin{aligned} e(i) &= [e^1 \ e^2 \ \dots \ e^n] \\ &= [e^1 \ 0 \ \dots \ 0] + [0 \ e^2 \ \dots \ 0] + \dots + [0 \ 0 \ \dots \ e^n] \end{aligned} \quad (4.23)$$

in which each term denotes a single bit error event where $e^i = 1$ with probability α and $e^i = 0$ with probability $1 - \alpha$.

The linearity of the syndrome coding scheme means that the syndrome resulting from any error pattern is the sum of the syndromes for each bit error position:

$$S_e(i) = e(i) \times \mathbf{H}^T = [e^1 \ e^2 \ \dots \ e^n] \times \mathbf{H}^T \quad (4.24)$$

$$= b_1 \delta(e^1 - 1) \oplus b_2 \delta(e^2 - 1) \dots \oplus b_n \delta(e^n - 1) \quad (4.25)$$

in which $\oplus$ denotes the modulo 2 sum. Since the probabilities of $e^1, e^2, \dots, e^n$ are independent, the probability of $S_e(i)$ is the product of the probabilities of n separate error events. Adding together the coefficients of same powers of Z results in the coefficients, β_j and reduces the number of terms from 2^n to 2^m . $\square$

Theorem 1 gives us some insight into code construction by code extension.

4.4.2 Code construction by code extension

A corollary of *Property 1* is that every error pattern of weight less than $t + 1 = \lfloor \frac{d+1}{2} \rfloor$ has a distinct syndrome. For syndrome coding this means that the syndrome probability mass function considering only error events with weight less than $t + 1$ will be maximally flat. However as higher weight error events all the way up to weight n can occur, other properties of the code other than d are involved.

The code construction method for obtaining codes with good equivocation is based on the observation that the syndrome probability mass function of a code extended in length is a function of the probability mass function of the original code and good equivocation codes produce good extended codes and this is elaborated below. BKC's have also been obtained by code extension [45].

Theorem 1 gives us some insight into code construction by code extension. If the columns of the parity check matrix of the shortened code with a length r are taken consecutively from $i = 0$ to $r - 1$, in which i is the index of the column, and the Z transform the probability generating function of the shortened code is

$$p_z(\mathcal{S}_r) = \prod_{i=0}^{r-1} ((1 - \alpha) + \alpha Z^{b_i}) \quad (4.26)$$

then the Z transform of the probability generating function the code of length $r + 1$ is

$$p_z(\mathcal{S}_{r+1}) = \prod_{i=0}^r ((1 - \alpha) + \alpha Z^{b_i}) = p_z(\mathcal{S}_r)((1 - \alpha) + \alpha Z^{b_r}) \quad (4.27)$$

By denoting the coefficients of the shortened code of length r , as $\beta_r(i)$ then

$$p_z(\mathcal{S}_r) = \sum_{j=0}^{2^m-1} \beta_r(j) Z^j \quad (4.28)$$

and

$$p_z(\mathcal{S}_{r+1}) = (1 - \alpha) \sum_{j=0}^{2^m-1} \beta_r(j) Z^j + \alpha \sum_{j=0}^{2^m-1} \beta_r(j) Z^{j \oplus b_r} \quad (4.29)$$

which simplifies to

$$p_z(\mathcal{S}_{r+1}) = (1 - \alpha)p(\mathcal{S}_r) + \alpha \sum_{j=0}^{2^m-1} \beta_r(j) Z^{j \oplus b_r} \quad (4.30)$$

It is apparent that the syndrome probability generating function in Z transform of the code of length $r + 1$ is equal to the syndrome probability generating function in Z transform of the shortened code of length r , weighted by $1 - \alpha$ plus a permuted syndrome probability generating function in Z transform of the shortened code of length r , weighted by α . The permutation arises from the results of the modulo 2 additions $j \oplus b_i$. In general, the more uniform the syndrome pmf of a code is, the more uniform the syndrome pmf of the extended code will be.

This leads to the conclusion that the syndrome pmf of the code may be obtained recursively, starting with the generating function $p_z(\mathcal{S}_1)$ determining $p_z(\mathcal{S}_2)$ then $p_z(\mathcal{S}_3)$ through to $p_z(\mathcal{S}_n)$. It is also apparent that good equivocation codes will also produce good equivocation codes when extended in length.

4.5 Code design for syndrome coding

We are now in a position to consider code design for a syndrome coding scheme for the wiretap channel. To construct the best codes for the syndrome coding scheme, we want the equivocation of the eavesdropper to be as high as possible, i.e. the pmf of the syndromes should be as uniform as possible. Since the eavesdropper channel is a binary symmetric channel, for relatively low values of the error probability, α , the equivocation is dominated by error events of low weight. We note the following design considerations:

1. For the BSC, low weight error events have higher probabilities, so they dominate the pmf of the syndrome. If the low weight error events produce distinct syndromes, this helps the pmf of the syndromes to become more uniform. This has a connection to the minimum Hamming distance of the code, d , because error events of weight up to $\frac{d-1}{2}$ all have distinct syndrome sums. Best known codes, which have the highest d , are likely to give good performance in the syndrome coding scheme, but are not necessarily the best codes, because all error patterns, regardless of weight, contribute towards the syndrome pmf of the code.
2. With the parity check matrix in reduced echelon form, that is in systematic packed integer form, the packed integers of any of the information bits cannot have a weight less than $d - 1$; otherwise the codeword formed from that information bit alone will have weight less than d , where d is the minimum Hamming distance of the code.
3. There is at least one syndrome that is produced by a weight R_c error event and

no lower weight error event, where R_c is the covering radius of the code. A good equivocation code will tend to have low covering radius compared to the minimum Hamming distance, d . Perfect codes have low covering radius and are good equivocation codes.

4. A shortened best known code has a bounded covering radius $R_c \geq (d-1)$, because there exists at least one coset leader of the shortened code of weight $d-1$ or greater; otherwise the extended code formed by adding this coset leader to the generator matrix of the code will produce a code with minimum Hamming distance less than d , a contradiction. It is unlikely that shortened best known codes will be best equivocation codes.
5. If columns of the parity matrix are repeated, a weight 2 error event will produce a zero syndrome, which makes the pmf of the syndrome non-uniform. To get a good equivocation code, the parity check matrix should have no repeated integers, unless the code is very long compared to the number of parity bits.

Following these observations, a code design algorithm has been formulated, that produces best equivocation codes as follows:

INPUT: C_{in} -Set of l inequivalent ¹ highest equivocation rate (n, m) codes with parity check matrices in systematic packed integer form $\mathbf{H}$.

OUTPUT: C_{out} - Set of all extended inequivalent $(n+1, m)$ codes, which are ranked by equivocation in descending order.

For each code C in C_{in} , the extension steps are as follows:

1. Preset a status array $a[i] = 1$, where $1 \leq i \leq 2^{n-k}$.
2. If i is equal to any integer in $\mathbf{H}$, set $a[i] = 0$.
3. For all j such that $a[j] = 1$, extend $\mathbf{H}$ with one integer, equal to j , to $\mathbf{H} = [\mathbf{H}, j]$. Eliminate all equivalent codes and evaluate the equivocation² for each remaining code.

¹If the codes have different equivocation rates, they are inequivalent codes.

²The equivocation is evaluated for the binary symmetric eavesdropper channel for a given error probability.

4. Rank the inequivalent codes by their equivocation in descending order, C_{out} , and select the first l codes. This is termed the best codes subset. These are used as the INPUT for the next extension round.

For a given m , all possible $(m + 1, 1)$ codes are used as the initial INPUT with the parity check matrices: $[1, 2, \dots, 2^{m-1}, i]$, for all i , $1 \leq i \leq 2^{n-k}$. For each round as n is incremented, the best equivocation codes are obtained, provided that the best codes subset is sufficiently large. The syndrome probability generating function in Z transform of each (n, m) code, $p_z(\mathcal{S})_{in}$, is stored and the syndrome probability generating function in Z transform for each extended $(n + 1, m)$ code, $p_z(\mathcal{S})_{out}$, is determined using equation (4.30) which makes for a fast algorithm.

Usually, the best extended code will be derived from the highest ranked code from the previous round but this is not always the case. The size of the best codes subset, l , needs to be large enough that no better extended code is missed. For short codes it is possible to store all inequivalent codes in the best codes subset and it is proven that the highest ranked code is the best (n, m) equivocation code.

To estimate the size required of the best codes subset, for a given number of parity bits, l is assigned (e.g. $l = 20000$) and the code is extended in length by several bits. For the highest ranked code of length n , the ranked position p_{n-1} is determined of its parent code of length $n - 1$ from the previous round. Statistics of p_{n-1} are collected for each code length and the subset size l is maintained to be greater by a significant margin to the maximum value of p_{n-1} , i.e. $l \geq \max p_{n-1}$. For $m = 6$ and $m = 7$, for all code lengths of $n \leq 70$, the maximum p_{n-1} values are 25 and 275 respectively. It is observed that the greater the length of the code, the higher the possibility of $p_{n-1} = 1$, i.e. the extended best equivocation code of length n produces the best equivocation code of length $n + 1$.

4.6 Results

Following the procedure above, the best equivocation codes have been determined for a given number of parity bits and code length. There are too many codes to present, so we only present here some representative codes. The codes are listed in Appendix B and an on-line database [76] in packed integer format. The minimum Hamming distance, covering radius and equivocation rate for a BSC error probability of $\alpha = 0.05$ are given for each code.

As examples, Table 4.1 lists the highest rate codes, in packed integer format, which provide at least 90% secrecy when used in syndrome coding for a BSC error probability of $\alpha = 0.05$. Interestingly, in general nearly all of the best equivocation codes do not coincide with the best known codes. However these best equivocation codes do have a respectable minimum Hamming distance and covering radius.

With reference to Appendix B, there are a few best known codes that are also best equivocation codes, notably all of the perfect codes. The perfect codes are the repetition codes, the Hamming codes and the Golay code. The reason for this coincidence is that these codes have the highest minimum Hamming distance, d , and are optimally packed codes. It is notable that there is a small number of other best known codes that are also best equivocation codes and we term these best known codes as well packed codes. A feature of these well packed codes is that for a given number of parity bits there is no known longer code with the same minimum Hamming distance, d . Table 4.2 gives the code parameters of the well packed BKC for various values of m . We have observed that some of the shorter well packed BKC coincide with the best equivocation codes for given m and given k .

It is also interesting to note that each well packed BKC of length n is followed by a poorly packed BKC of length $n + 1$. These BKC's are termed the worst packed BKC's. It is found that the corresponding BEC's have much higher equivocation rates than that of the worst packed BKC's.

Figure 4.1 compares the normalised equivocation of best known codes and best equivocation codes for increasing code rate, for codes with 26 parity bits at different values of α . It shows that the BECs have higher equivocation than BKC's not only for $\alpha = 0.05$ but also for other values of α . The dashed line gives the secrecy capacity of the channel. It is readily apparent that the BEC's outperform the BKC's particularly in the lower rate region.

As the number of parity bits increase, codes get longer for the same information rate and the equivocation moves closer to the secrecy capacity limit, C_s , which is applicable to infinite length codes. This is shown in Figure 4.2 for $m = 6$ through to $m = 26$.

Table 4.1: Best Equivocation Codes that achieve 90% secrecy in syndrome coding for $\alpha = .05$

m	n	d	R_c	R_e	Packed integer parity check matrix
4	25	2	1	0.906251	1 2 4 8 1 2 3 3 4 5 5 6 6 7 8 9 10 11 11 12 12 13 14 15 15
5	28	3	2	0.907646	1 2 4 8 16 3 5 6 7 9 11 12 13 14 15 17 18 19 20 22 23 24 25 26 27 28 29 30
6	31	4	3	0.908721	1 2 4 8 16 32 7 11 13 14 19 21 22 25 26 28 31 35 37 38 41 42 44 47 49 50 52 56 59 61 62
7	35	3	3	0.907273	1 2 4 8 16 32 64 7 14 25 26 28 31 38 43 51 53 56 62 70 75 83 85 88 93 100 103 104 110 112 118 121 122 124 127
8	38	4	3	0.904458	1 2 4 8 16 32 64 128 26 35 46 61 79 83 86 89 98 101 105 126 131 143 156 166 171 183 185 193 198 200 203 205 223 231 236 243 244 250
9	41	4	3	0.900208	1 2 4 8 16 32 64 128 256 31 45 60 100 113 122 143 153 156 181 191 194 204 214 218 281 291 302 331 332 344 370 383 394 397 403 418 427 440 478 485 488
10	45	4	3	0.904383	1 2 4 8 16 32 64 128 256 512 29 87 109 191 203 221 229 301 344 350 379 409 426 435 467 468 495 501 527 539 596 613 637 661 675 716 755 787 820 858 879 897 969 994 1008
11	48	4	3	0.902358	1 2 4 8 16 32 64 128 256 512 1024 82 95 103 137 170 340 376 415 416 456 486 579 647 734 754 811 829 887 964 977 1079 1082 1227 1239 1302 1370 1420 1505 1546 1596 1710 1828 1863 1929 1938 1975 2041
12	51	4	3	0.900428	1 2 4 8 16 32 64 128 256 512 1024 2048 108 214 225 327 415 460 777 814 862 939 978 998 1059 1116 1206 1450 1606 1663 1677 1845 2243 2355 2456 2554 2951 2964 2985 3119 3414 3467 3582 3731 3802 3812 3819 3907 3928 3950 4049
13	55	6	5	0.905225	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 309 345 618 687 690 981 1207 1236 1374 1380 1653 1897 1962 2373 2414 2472 2639 2748 2760 3306 3794 3924 4315 4735 4746 4807 4828 4887 4944 5278 5496 5520 6225 6612 7263 7273 7411 7449 7588 7848 8007 8137

Continued on next page

Table 4.1 – Continued from previous page

m	n	d	R_c	R_e	Packed integer parity check matrix
14	58	5	4	0.902823	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 463 659 1306 1903 2017 2420 2623 2681 2774 3009 3018 3372 3462 3727 3848 4807 4978 5102 5286 5495 5755 6833 7557 7706 8435 9814 10158 10732 10769 10991 11355 11837 12664 12788 13759 13762 13880 15043 15411 15498 15640 15910 16068 16352
16	64	5	4	0.9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 2174 6623 9685 10416 10555 11116 11896 12221 12999 14403 14586 14926 19995 20228 20788 23508 23688 23953 25340 27319 27881 28822 30545 31261 31842 32143 33695 35079 35300 40115 41850 42821 43146 46587 46930 48617 48950 50224 50393 52924 53375 53724 55136 57952 61511 62572 65419
18	71	6	5	0.903035	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 11955 13936 32282 32999 38800 39250 42924 47840 55393 56951 58478 60921 69584 75429 81270 92272 93719 98994 101001 102495 107633 112602 118747 135116 148158 154496 159264 161706 168209 175488 176485 180456 187783 193685 194276 195402 203311 206292 207540 207680 208910 210560 225118 230979 235516 238390 247552 252600 254562 254936 256876 258694 260096
20	77	8	7	0.900739	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 41772 53076 70933 85654 95837 102185 121984 143743 159055 160966 162356 177792 186954 191057 197822 208965 259410 268572 272682 296292 308553 313067 320896 366336 371328 405616 442424 445952 450779 457021 475870 491392 532261 539043 560237 602893 616832 625083 632370 646431 661559 680024 692328 692902 701447 716691 741146 746583 749617 756858 782531 799616 814830 923790 985856 996729 1044736

Continued on next page

Table 4.1 – Continued from previous page

m	n	d	R_c	R_e	Packed integer parity check matrix
22	84	8	7	0.903293	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 14191 111264 120844 135402 156066 193837 297854 309690 330532 338475 430909 567303 682894 714276 780924 869995 905402 923298 982920 1351071 1376188 1380394 1419610 1487155 1551602 1884223 1964241 2073026 2141889 2289013 2314096 2350121 2376601 2501526 2638285 2645293 2687586 2703451 2758616 2779212 2825279 2893319 3047974 3082337 3148814 3167196 3172106 3212496 3244862 3367783 3378339 3595701 3671892 3675563 3727842 3733852 3753502 3954669 3966529 3984378 3994343 4013090
24	90	7	6	0.901784	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 573185 576055 605300 649314 701023 902444 935950 972344 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 3857484 4410637 4447108 4642815 4754320 4865882 4933529 4981695 5087490 5170378 5176601 5211409 5493873 6813668 6982303 7139876 7894909 7933395 8894748 9354907 9466165 10033669 10290550 10320865 10887849 11095914 11204875 11347996 11485683 11727110 11796941 12233544 12889002 13314240 13707676 14157669 14737865 14795911 14837286 14923311 15030096 15150513 15341837 15855350 15923713 15972347 16494372 16540142 16554318

Continued on next page

Table 4.1 – Continued from previous page

m	n	d	R_c	R_e	Packed integer parity check matrix
26	97	8	6	0.904462	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 3419307 4492287 9395871 10093907 11068401 11840460 12139956 14534737 14816701 17641732 17992524 18692090 20131180 20858618 22460065 22562674 23680920 23791323 24279912 25579531 26280684 26831716 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 36286430 38278580 39088097 40262360 41667867 43800122 44014004 44811375 46058364 46251188 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 52561368 53663432 54332396 54744803 54780620 55996779 56474461 56772244 57104360 57872920 58140380 58689469 60935573 61409218 62624620 63555923 64592814 65128072 65473972

4.7 Conclusions

We considered the code design problem for syndrome coding from the information theoretic security view and presented a code construction method for best equivocation codes. In this way, an table of constructed BEC's in packed integer format has been produced (Appendix B). Code examples have been given of the highest rate BEC's that achieve 90% secrecy to an eavesdropper using the BSC with an error probability of 0.05. Unlike the traditional case, the presented encoding method does not require a syndrome look up table. This encoding method also provides a new interpretation of syndrome coding in which the message is impressed on the parity bits of a random codeword. In effect the m parity bits of an n -bit random codeword are used to scramble an m -bit message necessitating the error free recovery of all n bits for an eavesdropper in order to determine the m bit message.

An efficient recurrent method was also presented for the calculation of the pmf of the

Table 4.2: Well packed BKC

m	Well packed BKC, (n,k,d)
4	(5,1,5), (8,4,4), (15,11,3)
5	(6,1,6), (16,11,4), (31,26,3)
6	(7,1,7), (8,2,5), (32,26,4)
7	(8,1,8), (9,2,6), (11,4,5), (64,57,4)
8	(9,1,9), (10,2,6), (11,3,5), (128,120,4), (255,247,3)
9	(10,1,10), (11,2,7), (18,9,6), (23,14,5)
10	(11,1,11), (12,2,8), (15,5,7), (24,14,6), (33,23,5)
11	(12,1,12), (16,5,8), (23,12,7), (34,23,6), (47,36,5)
12	(13,1,13), (14,2,9), (24,12,8), (48,36,6), (65,53,5)
13	(14,1,14), (15,2,10), (25,12,8), (27,14,7), (66,53,6), (81,68,5)
14	(15,1,15), (16,2,10), (17,3,9), (28,14,8), (31,17,7), (82,68,6), (128,114,5)
15	(16,1,16), (17,2,11), (18,3,10), (20,5,9), (32,17,8), (37,22,7), (129,114,6), (151,136,5)
16	(17,1,17), (18,2,12), (21,5,10), (23,7,9), (38,22,8), (47,31,7), (152,136,6)
17	(18,1,18), (19,2,12), (20,3,11), (24,7,10), (27,10,9), (48,31,8), (63,46,7)
18	(19,1,19), (20,12,13), (21,3,12), (23,5,11), (28,10,10), (31,13,9), (64,46,8), (68,50,7)
19	(20,1,20), (21,2,14), (24,5,12), (26,7,11), (32,13,10), (35,16,9), (69,50,8), (88,69,7)
20	(21,1,21), (22,2,14), (27,7,12), (31,11,11), (36,16,10), (41,21,9), (89,69,8), (95,75,7)
21	(22,1,22), (23,2,15), (24,3,13), (32,11,12), (33,12,11), (42,21,10), (45,24,9), (96,75,8), (128,107,7)
22	(23,1,23), (24,2,16), (25,3,14), (27,5,13), (34,12,12), (36,14,11), (46,24,10), (49,27,9), (129,107,8), (155,133,7)
23	(24,1,24), (25,2,16), (28,5,14), (29,6,13), (37,14,12), (47,24,11), (50,27,10), (54,31,9), (156,133,8), (162,139,7)
24	(25,1,25), (26,2,17), (27,3,15), (30,6,14), (32,8,13), (48,24,12), (55,31,10), (64,40,9), (163,139,8)
25	(26,1,26), (27,2,18), (28,3,16), (31,6,15), (33,8,14), (34,9,13), (49,24,12), (65,40,10), (72,47,9)
26	(27,1,27), (28,2,18), (32,6,16), (35,9,14), (38,12,13), (50,24,12), (51,25,11), (73,47,10), (77,51,9)
27	(28,1,28), (29,2,19), (33,6,16), (35,8,15), (39,12,14), (40,13,13), (52,25,12), (63,36,11), (78,51,10), (94,67,9)
28	(29,1,29), (30,2,20), (31,3,17), (36,8,16), (37,9,15), (41,13,14), (44,16,13), (64,36,12), (69,41,11), (95,67,10), (128,100,9)
29	(30,1,30), (31,2,20), (32,3,18), (38,9,16), (40,11,15), (45,16,14), (46,17,13), (70,41,12), (71,42,11), (129,100,10), (135,106,9)
30	(31,1,31), (32,2,21), (33,3,18), (34,4,17), (41,11,16), (44,14,15), (47,17,14), (48,18,13), (72,42,12), (136,106,10), (142,112,9)

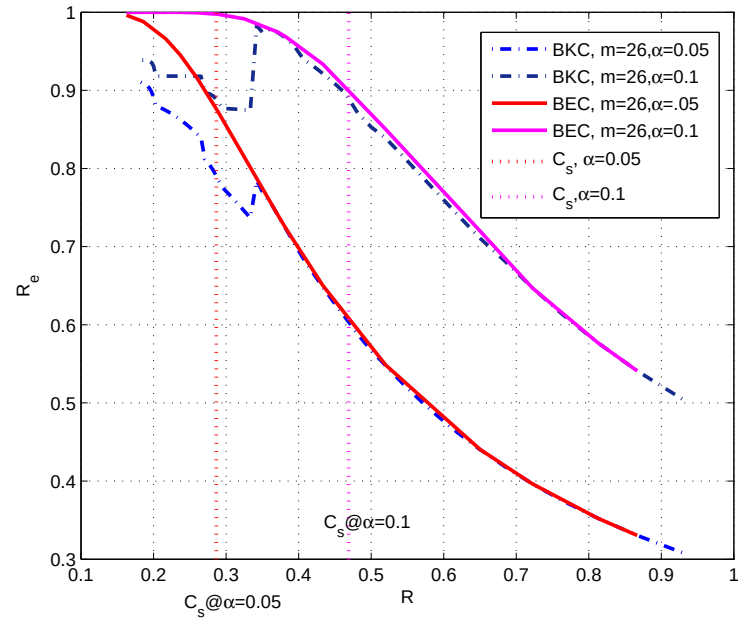


Figure 4.1: Comparison of the normalised equivocation of best equivocation and best known codes having 26 parity bits

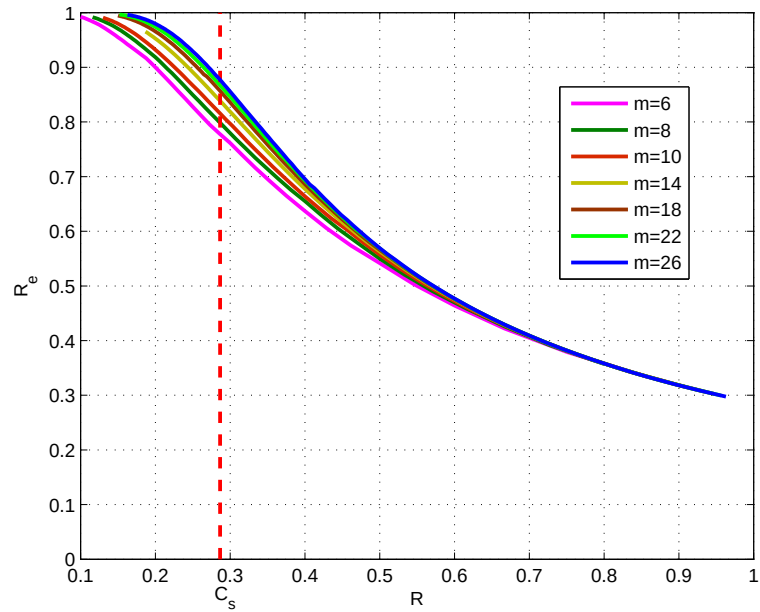


Figure 4.2: Normalised equivocation vs. information rate for Best Equivocation Codes as a function of m

syndromes of a code from the parity check matrix of the code, which in turn enables the secrecy achieved by the code, the equivocation rate, to be determined. It was shown that apart from the perfect codes, the best equivocation codes rarely coincide with the best known error correcting codes and that usually a best equivocation code will provide greater secrecy than using a best known error correcting code with the same parameters.

Chapter 5

Syndrome Coding Based on Random Linear Binary Codes

This chapter studies the security performance of the random codes when used in conjunction with a syndrome coding scheme. A theoretical analysis is proposed to calculate the average equivocation of the (n, k) random codes and compared with the simulation results generated from Monte Carlo analysis, which shows that the theoretical method is precise. It also shows that the theoretical analysis works well for long codes having large values of $n - k$, which are not amenable to Monte Carlo analysis.

5.1 Introduction

The performance of random codes is one of the important topics in information theory following derived from Shannon's random code ensemble analysis in 1948. He showed that in an ensemble of random codes there must be at least one code from the random code ensemble that approaches channel capacity as the code length approaches infinity [57]. Most of the researches focus on the reliability of the random codes and apply the random codes in the error correcting scheme to study the average error probability of the communication. For example, in 1973, Gallager constructed the random coding bound by upper bounding the average of a channel's error probability over a random ensemble of codes and proved that the random coding bound is exponentially tight for the random code ensemble at all rates [85, 86]. MacMullan and Collins calculated new bounds on the size of the performance gap between random codes and the best possible codes for the binary symmetric channel, the binary erasure channel and the additive white Gaussian

noise channel [87]. In 2002, Barg and Forney studied the minimum distances, distance distributions and error exponents on the binary symmetric channel for the typical codes from Shannon's random code ensemble and for typical codes from a random linear code ensemble [88]. Recently, in [89], Domb *et al.* revisited Gallager's proof and constructed the new lower bound and proved that random coding bound is tight at all rates for the random linear ensemble. The research work on the random codes mentioned above all focus on the reliability of the communication for the random codes.

Besides transmission reliability, physical layer security is also very important in communications nowadays, which is usually considered in the wiretap model introduced by Wyner in 1975 [10]. The security performance of random codes in the syndrome coding scheme was studied by Csiszár and Körner for the broadcast channel in [11]. Cohen and Zemor showed that a random choice of code in the syndrome coding scheme suffices to guarantee the secure communication, by showing that the most likely syndrome does not have too high probability of occurrence [69], and also propose the generalised syndrome coding schemes to the case when both the main channel and the eavesdropper channel are noisy, which achieves secrecy capacity for randomly chosen codes [13]. Recently, in [60], Chen and Vinck prove that the error probability tends to zero and the equivocation rate tends to one, which means both reliability and security can be achieved, as the code length tends to infinity, using random linear codes in syndrome coding scheme over the binary symmetric wiretap channel.

To study the security performance of the random codes more completely, it is important and useful to determine the security level of the random codes for specific values of code length and information rate. This can be measured in terms of the level of security, the average equivocation achieved by the random code ensemble. To date, this has not been done and giving results for this case is the main aim for this chapter.

This chapter is organized as follows: Section 5.2 briefly reviews the definition of the random linear code and the expression of the equivocation in binary symmetric channel. In Section 5.3, we propose a theoretical method and a simulation way to calculate the average equivocation of the (n, k) random linear codes employed in syndrome coding scheme in binary symmetric wiretap channel, and also take some (n, k) random code as an example to calculate the equivocation in both the theoretical analysis and the simulation way to show our theoretical analysis is sharp. Section 5.4 gives some examples of (n, k) random codes to show the relation between the average equivocation and the code length for the random linear codes and the relation between the average equivocation and

the information rate. Finally, we summarise the main contributions of this chapter.

5.2 Random linear binary codes in syndrome coding

The syndrome coding scheme without look-up table has been introduced in Chapter 4, which is suitable for the codes with long code length and is also considered in this Chapter. In this chapter, we employ the random linear code in conjunction with the syndrome coding scheme.

5.2.1 Random codes

We use (n, k) random linear binary codes in the syndrome coding scheme. Each code can be defined by its $m \times n$ parity check matrix $\mathbf{H}$, in which $m = n - k$. As introduced in Chapter 4, the systematic format of $\mathbf{H}$ for the (n, k) random linear code can be represented by the following integer sequence:

$$[1, 2, 4, \dots, 2^{m-1}, b_m, \dots, b_{n-1}] \quad (5.1)$$

in which the first m integers denote the identity matrix and the residual integers of $b_m, \dots, b_{n-1}$ are generated randomly between 1 and $2^m - 1$ with the constraint that there are no repeated integers in the sequence $1, 2, 4, \dots, 2^{m-1}, b_m, \dots, b_{n-1}$, which implies that the minimum Hamming distance $d \geq 3$.

5.2.2 Equivocation analysis

Alice wants to sent a sequence of independent and random m -bit messages to Bob, $M(1), \dots, M(i), \dots, M(b)$, in which b is the block length of the messages, through an error free channel. Alice employs the (n, k) random codes in conjunction with the syndrome coding scheme, which has been described in Chapter 4, to encode the sequence of the messages into a sequence of n -bit vectors, $X(1), \dots, X(i), \dots, X(b)$. Bob receives $Y_M(1), \dots, Y_M(i), \dots, Y_M(b)$, in which $Y_M(i) = X(i)$, and calculates the corresponding syndromes to get the original messages. Meanwhile, an eavesdropper, Eve, observes a corrupted version of $X(i)$, $Y_E(i) = X(i) + e(i)$, through a binary symmetric channel with a crossover probability α , where $e(i)$ is the n -bit error vector generated by the channel. Then Eve employs the syndrome decoder to get the estimated messages: $\hat{M}(1), \dots, \hat{M}(i), \dots, \hat{M}(b)$, in which $\hat{M}(i) = Y_E(i) \times \mathbf{H}^T = M(i) + S_e(i)$, where $S_e(i) = e(i) \times \mathbf{H}^T$.

As introduced in Chapter 4, in syndrome coding scheme, the secrecy of a specific code is measured by the eavesdropper decoder output equivocation (Equation (4.17)), which is rewritten here:

$$H(M(i)|\hat{M}(i)) = H(S_e(i)) = - \sum_{j=0}^{2^m-1} p(S_j) \log_2 p(S_j) \quad (5.2)$$

in which, S_j denotes one of the total 2^m syndromes, and $p(S_j)$ is the probability mass function of the syndromes. The equivocation is calculated after deriving the probability mass function of the syndromes due to errors from the BSC, $p(S_j)$ and is a function of the parity check matrix of the code.

5.3 Average equivocation evaluation of random codes in syndrome coding

To evaluate the average equivocation of the (n, k) random linear codes, a direct method is to use Monte Carlo analysis to generate a large number of (n, k) codes randomly, do the exhaustive calculation of the equivocation for each code and average the evaluated equivocation values to get the mean. Monte Carlo analysis can also provide an idea of that calculating the average equivocation of the random codes based on the probability mass function (pmf) of the syndrome probabilities, which is the basic idea of our theoretical analysis.

5.3.1 Using Monte Carlo experiments to derive statistics of random codes

A large number, $|C|$, of randomly chosen (n, k) codes are generated randomly. We can get the average equivocation of these random codes by Monte Carlo experiments as follows:

$$\mathbb{E}[H] \approx \frac{1}{|C|} \sum_{t=1}^{|C|} H_t(M(i)|\hat{M}(i)) \quad (5.3)$$

$$= \frac{1}{|C|} \sum_{t=1}^{|C|} \left(- \sum_{j=0}^{2^m-1} p_t(S_j) \cdot \log_2 p_t(S_j) \right) \quad (5.4)$$

in which t is the index of the random codes, $|C|$ is the number of the codes, $H_t(M(i)|\hat{M}(i))$ is the entropy of the t -th code, S_j is one of the total 2^m syndromes and $p_t(S_j)$ is the

probability of S_j for the t -th code. Expanding (5.4):

$$\mathbb{E}[H] \approx -|S| \times \frac{1}{|S| \times |C|} \sum_{j=0}^{2^m-1} \sum_{t=1}^{|C|} p_t(S_j) \cdot \log_2 p_t(S_j) \quad (5.5)$$

in which $|S| = 2^m$ is the number of different syndromes. In (5.5), we collect all $2^m \times |C|$ values of $p_t(S_j)$ for all 2^m syndromes for all $|C|$ random codes to get the probability mass function (pmf), $p(p(S))$, of $p(S)$. Then we have:

$$\mathbb{E}[H] \approx -2^m \times \sum_l p(p(S)(l)) \times p(S)(l) \cdot \log_2 p(S)(l) \quad (5.6)$$

in which the probability of $p(S)(l)$ is:

$$p(p(S)(l)) = \sum_{j=0}^{2^m-1} \sum_{t=1}^{|C|} \frac{1}{2^m \times |C|} \cdot \delta(p_t(S_j) - p(S)(l)) \quad (5.7)$$

in which $\delta(\cdot)$ is the Kronecker Dirac function, l is the index of all $p(S)(l)$ which have $\sum_l p(p(S)(l)) = 1$.

The method of Monte Carlo experiments works well for the short random codes, but for the random codes with a large value of m ($m \geq 30$) this method will be impractical, because the exhaustive calculation of the equivocation for each code takes extremely long time to analyse all 2^m syndromes. However, (5.6) gives us an insight to evaluate the average equivocation of (n, k) random codes based on the pmf of $p(S)$, $p(p(S))$.

5.3.2 Theoretical analysis

The approach is to derive a putative (n, k) code which has the same pmf of syndrome probabilities as (5.6), which can be calculated from Monte Carlo experiments. It is shown below that the pmf of syndrome probabilities may be modelled as a number of Poisson distributions which may be used in turn to derive the putative code. On this basis the putative code for other code lengths n , and values of k , may be derived and used to predict the average equivocation of randomly chosen (n, k) codes. The equivocation of the (n, k) putative code is:

$$\mathbb{E}[H] \approx - \sum_{j=0}^{2^m-1} p(S_j) \cdot \log_2 p(S_j) \quad (5.8)$$

$$= -|S| \times \sum_l p(p(S)(l)) \times p(S)(l) \cdot \log_2 p(S)(l) \quad (5.9)$$

It is important to analyse the pmf of $p(S)$.

For a binary symmetric channel with an error probability, α , 2^m syndromes ($m=n-k$), $S_e(i)$, are contributed by 2^n error patterns, $e(i)$, with a probability,

$$p(e(i)) = \alpha^{w(e(i))} \times (1 - \alpha)^{n-w(e(i))} \quad (5.10)$$

in which $w(e(i))$ is the weight of $e(i)$, based on the following expression:

$$S_e(i) = e(i) \times H^T \quad (5.11)$$

Each of the 2^n error patterns has a syndrome value determined by the parity check matrix of the code. As there are only 2^m distinct syndromes and it is a linear code, the same syndrome is produced by 2^{n-m} distinct error patterns. Accordingly the probability of each syndrome is determined by the combined probability of 2^{n-m} distinct error patterns.

We divide those 2^n error patterns into different weight error events: 0-error event, 1-error events, $\dots$, w -error events, $\dots$, n -error events, $0 \leq w = w(e(i)) \leq n$. For each weight, all the error patterns with that weight have the same probability of (5.10).

Now let us analyse the pmf of $p(S)$ based on different error events. The probability of the syndrome, $p(S)$, is contributed by all w -error events and expressed as follows:

$$p(S) = \sum_w p(S)_w \quad (5.12)$$

in which $0 \leq w \leq n$, so the pmf of $p(S)$ can be calculated as follows:

$$p(p(S)) = p\left(\sum_w p(S)_w\right) \quad (5.13)$$

$$= p(p(S)_0) * \dots * p(p(S)_w) * \dots * p(p(S)_n) \quad (5.14)$$

in which $p(S)_w$ denotes the additive probability contributed by the corresponding w -error events, $x * y$ denotes the convolution of x and y and $0 \leq w \leq n$.

Next we analyse the pmf's of the additive probabilities $p(S)_w$, $0 \leq w \leq n$.

For $w = 0$, there is only one 0-error event, which has a probability of $(1 - \alpha)^n$ and map to $S = [0]$. There are two possible values of $p(S)_0$: 0 and $(1 - \alpha)^n$ with the probabilities

of $\frac{2^m-1}{2^m}$ and $\frac{1}{2^m}$ respectively. Therefore, the pmf of $p(S)_0$ is:

$$p(p(S)_0(l) = 0) = \frac{2^m - 1}{2^m} \quad (5.15)$$

$$p(p(S)_0(l) = (1 - \alpha)^n) = \frac{1}{2^m} \quad (5.16)$$

For $w = 1$, there are n 1-error events, in which each of them has a probability of $\alpha \cdot (1 - \alpha)^{n-1}$. Because of $d \geq 3$, these n 1-error events map to distinct n syndromes and it is impossible for 1-error events mapping to $S = [0]$. There are two possible values of $p(S)_1$: 0 and $\alpha \cdot (1 - \alpha)^{n-1}$ with the probabilities of $\frac{2^m-n-1}{2^m-1}$ and $\frac{n}{2^m-1}$ respectively. The pmf of $p(S)_1$ is:

$$p(p(S)_1(l) = 0) = \frac{2^m - n - 1}{2^m - 1} \quad (5.17)$$

$$p(p(S)_1(l) = \alpha \cdot (1 - \alpha)^{n-1}) = \frac{n}{2^m - 1} \quad (5.18)$$

For w -error events, in which $2 \leq w \leq n$, it is possible for them to map any of 2^m syndromes without any restriction. There are $\binom{n}{w}$ error patterns each leading to an w -error event. Each of these maps into one syndrome out of 2^m syndromes. All w -error events are binomially distributed. There are so many events that the Poisson distribution [90] is applicable to describe the number of times l that a particular syndrome will arise from an w -error events as follows:

$$p(l) = \frac{\lambda^l \cdot e^{-\lambda}}{l!} \quad (5.19)$$

in which $\lambda = \frac{\binom{n}{w}}{2^m}$ is the average number of the occurrence of an w -error events mapping into each syndrome, $p(l)$ is the probability mass function of the occurrence l ¹, e is the base of the natural logarithm ($e = 2.71828$) and $l!$ is the factorial of l .

Each time an w -error event maps into a syndrome, the additive probability of this syndrome is $\alpha^w \times (1 - \alpha)^{n-w}$. Therefore, the probability mass function (pmf), $p(p(S)_w)$, of the additive probability, $p(S)_w$, of the syndrome from w -error events as follows:

$$p(p(S)_w(l)) = p(l) = \frac{\lambda^l \cdot e^{-\lambda}}{l!} \quad (5.20)$$

in which

$$p(S)_w(l) = l \cdot \alpha^w \times (1 - \alpha)^{n-w} \quad (5.21)$$

¹we consider all values of l , which satisfies: $\sum_l p(l) \approx 1$

Based on the analysis above, we can get all pmf's, $p(p(S)_w)$ of $p(S)_w$ for all $0 \leq w \leq n$ and use them in (5.14) to get the pmf, $p(p(S))$, of $p(S)$. Using the results in (5.9), we can get the average equivocation finally.

5.3.3 Examples

Considering random codes with parameters (100,85) with ($\alpha = 0.05$), we can compare the theoretical pmf of $p(S)$ with the results obtained from Monte Carlo experiments.

Table 5.1 shows a statistics table of different error events for (100,85) random codes when $\alpha = 0.05$. It shows that considering 0 to 14 error events has a cumulative probability of 0.999864 and so error events of weight more than 14 have a negligible contribution to the pmf.

Based on the previous analysis, we can generate the pmf of additive probability of each syndrome for different error events. Some results from the Monte Carlo experiments and from the theoretical are shown and compared here. Since 0-error event and zero syndrome are special and easy to analyse separately, we only show some results related to non 0-error events and non-zero syndromes here.

We calculate the convolution of pmf's of the additive probabilities from 1 to 14 error events together. In order to show the details, we divide the convolution into two parts based on the values of $\sum_{w=1}^n p(S)_w$. Figure 5.1 shows the part of the pmf of $\sum_{w=1}^n p(S)_w$, when $p(S)_1 = 0$. Figure 5.2 shows the part of the pmf of $\sum_{w=1}^n p(S)_w$, when $p(S)_1 = \alpha \cdot (1 - \alpha)^{n-1}$. The corresponding pmf's, which are generated from Monte Carlo experiments, are given in 5.3 and 5.4.

The Kolmogorov-Smirnov test (K-S test) can be used to quantify a distance between two probability distributions and it is sensitive to differences in both location and shape of the cumulative distribution functions of the two samples [91]. We use K-S test to measure the difference between the theoretical analysis and Monte Carlo results for two parts of pmf of $\sum_{w=1}^n p(S)_w$. Figure 5.5 and Figure 5.6 give the cumulative distribution probability of Figure 5.1 and $D_n = 0.016887$, the cumulative distribution probability of Figure 5.2 and $D_n = 0.023036$, respectively, in which D_n is the maximum difference between the theoretical result and the Monte Carlo results vindicating the theoretical approach. It shows that the difference between two curves is very small, which means

that the theoretical results is very close to the simulation results.

Calculating the normalised average equivocation based on the theoretical pmf of $p(S_e(i))$ produces $\mathbb{E}[R_e] = \mathbb{E}[H]/m = 0.98862$. The average equivocation rate from Monte Carlo analysis is 0.99048. The difference between the theoretical result and the simulation result is 0.00186, caused by neglecting the error events with $w > 14$. More examples are given in next section.

Table 5.1: w -error events statistic ($\alpha = 0.05$)

w	$\binom{n}{w}$	$p(e(i))$	$\frac{\binom{n}{w}}{2^m} \cdot p(e(i))$	$\binom{n}{w} \times p(e(i))$	Cumulative probability
0	1	5.92053E-03	1.8068E-07	5.92053E-03	0.005921
1	100	3.11607E-04	9.5095E-07	0.03116068	0.037081
2	4950	1.64004E-05	2.4775E-06	0.081181772	0.118263
3	161700	8.63177E-07	4.2595E-06	0.139575678	0.257839
4	3921225	4.54304E-08	5.4365E-06	0.178142642	0.435981
5	75287520	2.39107E-09	5.4937E-06	0.180017827	0.615999
6	1192052400	1.25846E-10	4.5781E-06	0.150014856	0.766014
7	16007560800	6.62300E-12	3.2356E-06	0.106025537	0.87204
8	1.86088E+11	3.49000E-13	1.9797E-06	0.064870888	0.93691
9	1.90223E+12	1.80000E-14	1.0651E-06	0.034901296	0.971812
10	1.73103E+13	9.65661E-16	5.1013E-07	0.016715884	0.988528
11	1.416298046436E+14	5.08242E-17	2.1967E-07	0.007198228	0.995726
12	1.050421051106701E+15	2.67496E-018	8.5749E-08	0.002809834	0.998536
13	7.110542499799197E+015	1.40788E-19	3.0550E-08	0.001001075	0.999537
14	4.41869426773236E+15	7.40986E-21	9.9920E-09	3.274191E-04	0.999864

5.4 Simulation results

We want to know the relation between the code length and the normalised average equivocation for the random codes given fixed m when $\alpha = 0.05$. Figure 5.7 shows the normalised average equivocation and $\mathbb{E}[R_e] \pm 3\sigma$ vs. the code length for the random codes with $m = 20$ (99% codes has the equivocation rate between $\mathbb{E}[R_e] - 3\sigma$ and $\mathbb{E}[R_e] + 3\sigma$), in which σ is the standard deviation of the equivocation rate. It is shown that the longer the code the higher the average equivocation.

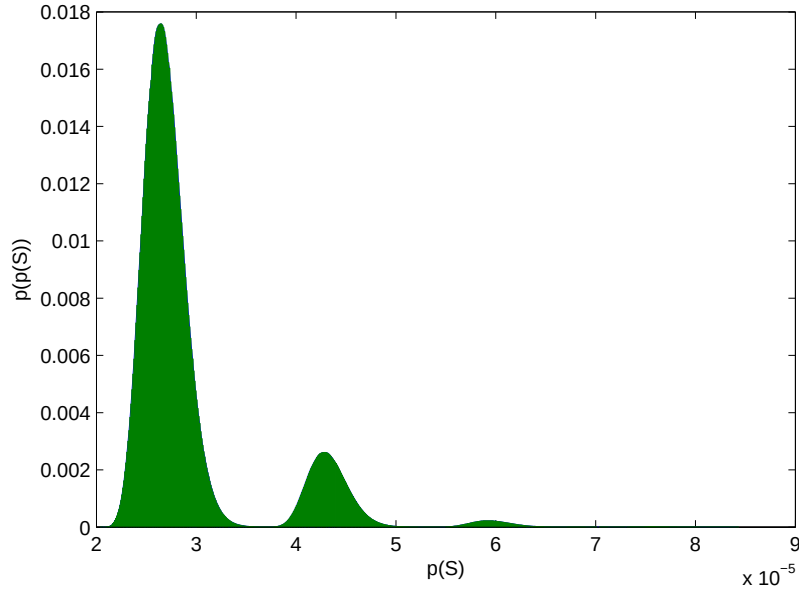


Figure 5.1: The theoretical pmf of $\sum_{w=1}^n p(S)_w$, when $p(S)_1 = 0$.

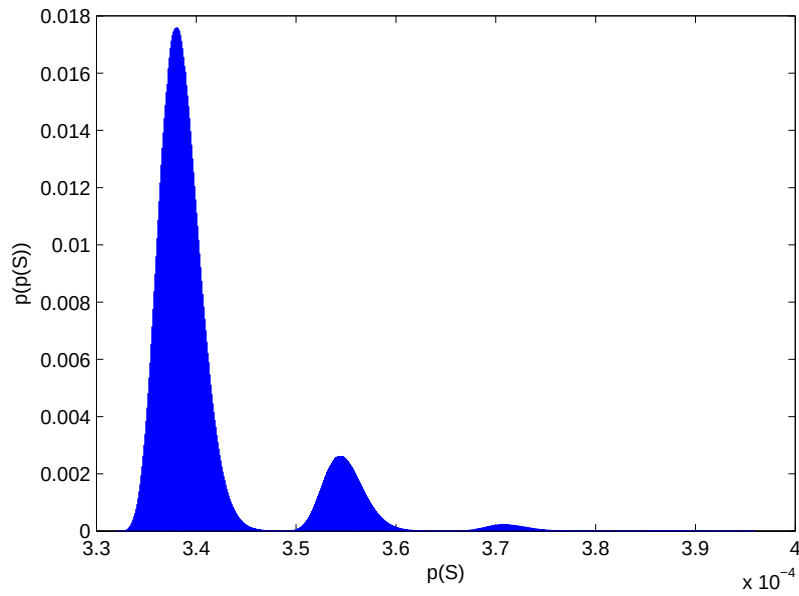


Figure 5.2: The theoretical pmf of $\sum_{w=1}^n p(S)_w$, when $p(S)_1 = \alpha \cdot (1 - \alpha)^{n-1}$.

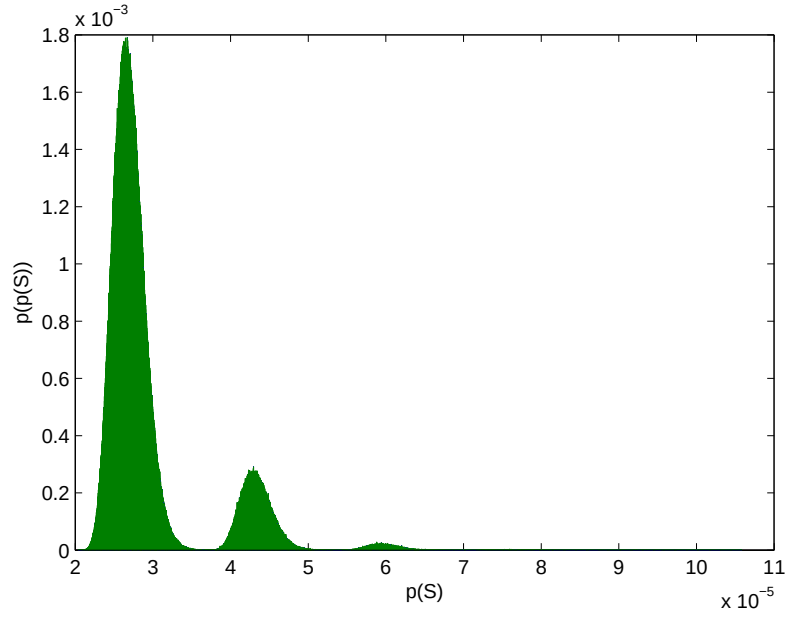


Figure 5.3: The simulated pmf of $\sum_{w=1}^n p(S)_w$, when $p(S)_1 = 0$.

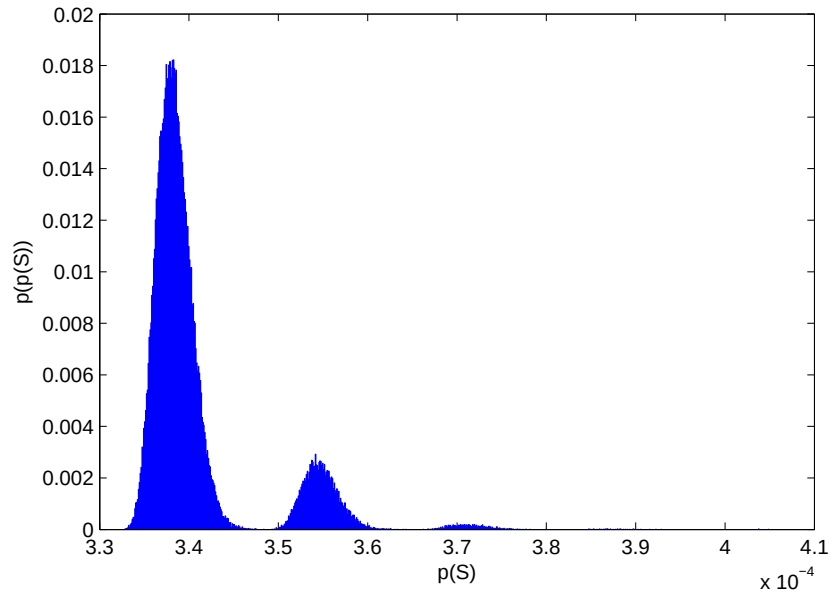


Figure 5.4: The simulated pmf of $\sum_{w=1}^n p(S)_w$, when $p(S)_1 = \alpha \cdot (1 - \alpha)^{n-1}$.

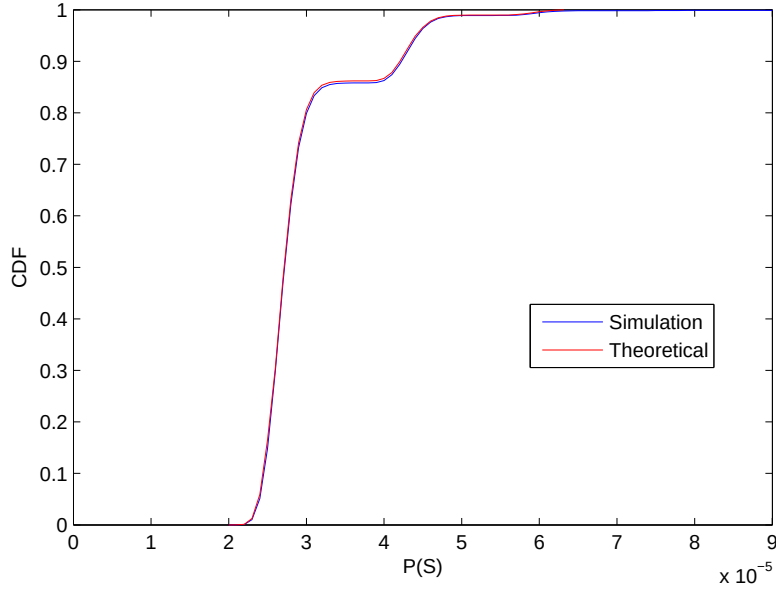


Figure 5.5: The cumulative distribution probability of $\sum_{w=1}^n p(S)_w$, when $p(S)_1 = 0$.

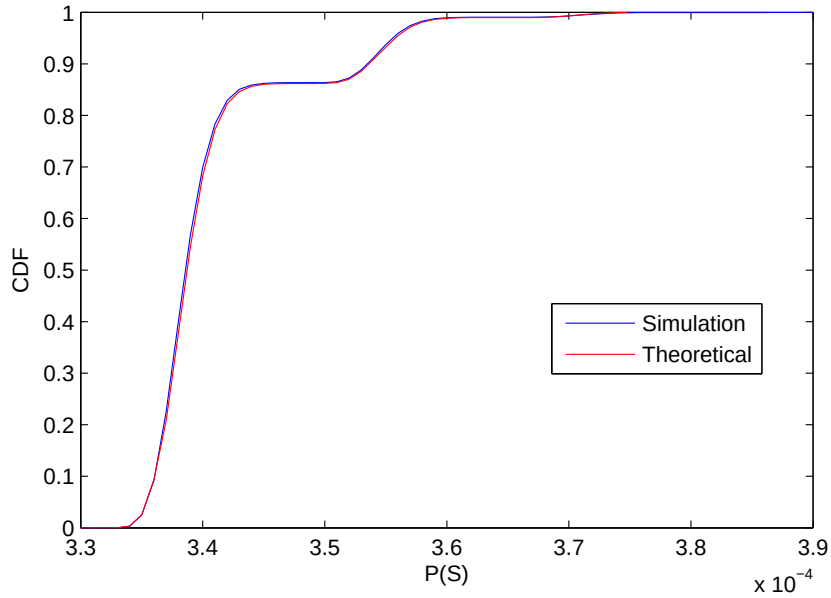


Figure 5.6: The cumulative distribution probability of $\sum_{w=1}^n p(S)_w$, when $p(S)_1 = \alpha \cdot (1 - \alpha)^{n-1}$.

Figure 5.8 shows that as the code length increases, the standard deviation of the normalised equivocation decreases. For codes longer than 150 bits or so any randomly chosen code achieves almost perfect secrecy.

Figure 5.9 plots the normalised average equivocation vs. the information rate for the random codes with different values of m . The points of the pink star are generated by the theoretical way, which are quite close to the Monte Carlo simulation results (sample number=10000). The differences between theoretical analysis and Monte Carlo analysis for random codes with $m = 20$ are all less than 0.3%. For random codes with $m \geq 30$, Monte Carlo analysis requires powerful computer resources and takes a long time, but the codes are still amenable to theoretical analysis. It may be observed that good results may be obtained with randomly chosen codes and that it is not necessary to use a large number of parity bits or excessive code length to attain almost perfect secrecy. When $\alpha = 0.05$, the brown line gives the secrecy capacity of the channel: 0.2864. We observe that for the random codes with $R > 0.2864$, the normalised average equivocation is far away from 1, i.e. it is impossible to achieve perfect secrecy for those codes. It also shows that for the codes of the given length the higher the value of the information rate, the lower the value of the normalised average equivocation, i.e. to get a higher security level, we have to pay a price of a lower rate.

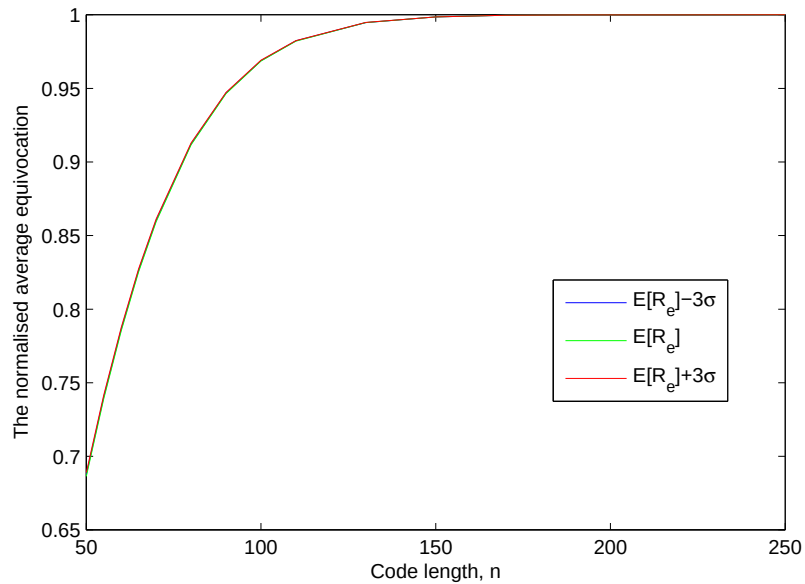


Figure 5.7: The normalised average equivocation vs. the code length for the random codes ($m = 20$, sample number=10000)

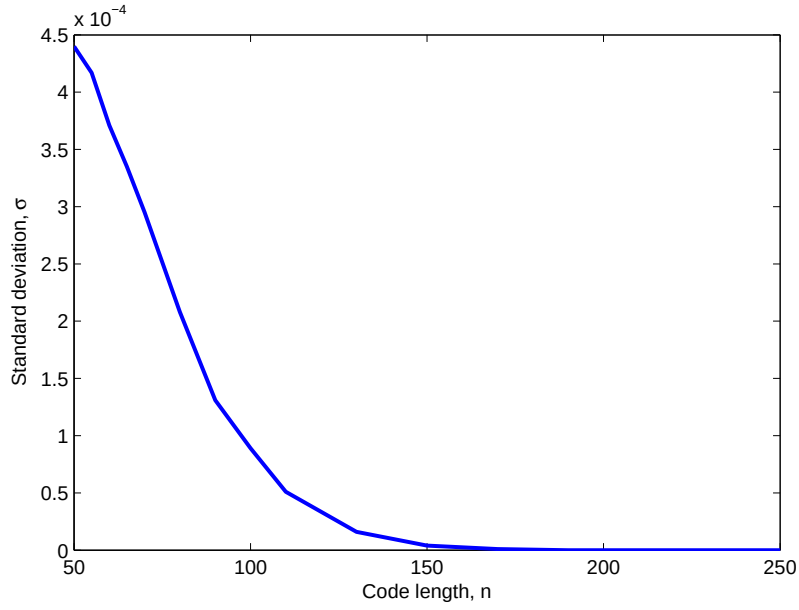


Figure 5.8: The standard deviation of the normalised equivocation vs. the code length for the random codes ($m = 20$)

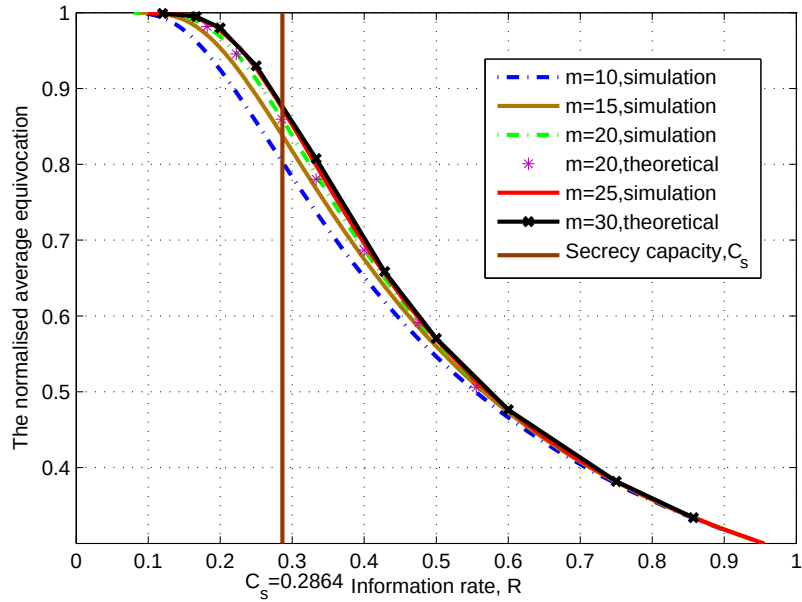


Figure 5.9: The normalised average equivocation vs. the information rate for the random codes with different values of m

5.5 Conclusions

In this chapter, we analyse the security performance of (n, k) random linear binary codes in Wyner's syndrome coding scheme from the viewpoint of the information theoretical security as a function of code length and code rate. An important contribution in this chapter is the proposal of theoretical analysis based on a putative (n, k) code which has the same distribution of syndrome probabilities as the ensemble of all (n, k) random codes having the same parameters to calculate the average equivocation of the (n, k) random codes, and the comparison with the results obtained by Monte Carlo analysis. Specific examples are given to show the accuracy of the theoretical analysis. It is shown that random codes with large values of m , the number of parity bits, may be analysed theoretically even though Monte Carlo analysis is impractical.

Many examples of the random codes are given to show that the longer the length of the code the higher value of the average equivocation and the lower the value of the standard deviation of the equivocation, and the higher the value of the information rate, the lower the value of the average equivocation. Possibly the most important conclusion is that almost perfect secrecy may be obtained for syndrome coding using a randomly chosen code of length greater than 150 bits and reasonable code rate.

Chapter 6

A Chain Based Syndrome Coding Scheme

This chapter proposes and studies two variants of the traditional syndrome coding scheme for secure communication in a wiretap channel: a chain based syndrome coding scheme and a chain base syndrome coding scheme with a hash function. By using the eavesdropper decoder output equivocation to measure the security of these two schemes, it is shown that the first variant of the chain based syndrome coding scheme can not enhance the security of the communication, but numerical simulation results demonstrate that the chain based syndrome coding scheme with a hash function exhibits significantly lower information leakage than the conventional syndrome coding scheme.

6.1 Introduction

Wyner's syndrome coding scheme has been proposed to achieve the secure communication for the wiretap model. In the previous chapters, the security levels of the syndrome coding scheme for random codes have been analysed and the construction technique of best equivocation code has been proposed. In order to attain a higher level of the security, it is interesting to explore some extensions of the syndrome coding scheme.

Hash functions are primarily used to generate fixed-length output data which is a shortened reference to the original data of arbitrary length. One of the important properties of hash functions is that hash functions are not invertible, i.e. it is impossible to reconstruct the input data from hash output value [92]. Hash functions are mainly used in hash tables to locate the index of some input data. In this chapter, hash functions are employed to

compress and randomize the input data to a fixed length data.

In view of the fact that to achieve strong secrecy [52], it is necessary for the distribution of messages given the relevant observations (e.g., the eavesdropper channel output or the eavesdropper decoder output) to be uniform, we put forth a generalization of Wyner's conventional syndrome coding scheme by employing a syndrome-chain technique. We also utilise a hash function in our proposed chain technique syndrome coding scheme. By using the eavesdropper decoder output equivocation to measure the security of the scheme, we demonstrate that the new scheme provides much higher level of the security than that the conventional syndrome coding scheme offers.

This chapter is organized as follows: Section 6.2 introduces the chain based syndrome coding scheme and also analyses its security. Section 6.3 introduces the chain based syndrome coding scheme with a hash function and provides some simulation results to show the security of the communication. Finally, we summarize the main contributions of the chapter in Section 6.4.

6.2 A chain based syndrome coding scheme

We now introduce the proposed chain based syndrome coding scheme. We consider a wiretap channel model where the main channel is an error free channel and the eavesdropper channel are binary symmetric with a 'crossover probability' α , which has been introduced in Chapter 2 (Figure 2.6).

6.2.1 The encoder and the decoder description

6.2.1.1 Encoder

The chain based syndrome coding scheme represents a generalization of the conventional syndrome coding scheme [10]. We consider this generalization for an underlying $(n, k, 2t+1)$ linear block code capable of correcting t errors with a $k \times n$ generator matrix $\mathbf{G}$ and a $(n-k) \times n$ parity check matrix $\mathbf{H}$, which is the same as the code employed in the traditional syndrome coding scheme. The exact operation of the chain based syndrome coding scheme, including the encoding and decoding processes, is now described in the sequel.

Alice produces the n -bit codeword $X(i)$ from the $m = (n-k)$ -bit message $M(i)$ at time i as follows:

- Alice generates a n -bit vector $C_T(i)$ from a random k -bit binary vector $D_R(i)$ as follows:

$$C_T(i) = D_R(i) \times \mathbf{G} \quad (6.1)$$

- Alice also generates a n -bit error pattern $E_T(i)$ from the m -bit syndrome $S_T(i)$ in accordance with the 'error-syndrome table', where the current syndrome is set to be equal to the sum of the current message and the previous d syndromes as follows:

$$S_T(i) = M(i) + S_T(i-1) + \dots + S_T(i-d) \quad (6.2)$$

where d is the chain depth.

- Alice then generates the n -bit codeword $X(i)$ by the adding the n -bit vector $C_T(i)$ to the n -bit error pattern $E_T(i)$, i.e.,

$$X(i) = C_T(i) + E_T(i) \quad (6.3)$$

Note that this operational procedure requires setting the values of the m -bit initial conditions $S_T(i)$, $i = 1-d, \dots, 0$. We assume that these initial conditions, which are relevant not only for the encoding but also the decoding processes, are the key shared only by the legitimate parties: Alice and Bob. The eavesdropper, Eve, has no information about the initial conditions. This operational procedure is succinctly depicted in Figure 6.1.

The information rate of this scheme is $R = \frac{n-k}{n} = \frac{m}{n}$.

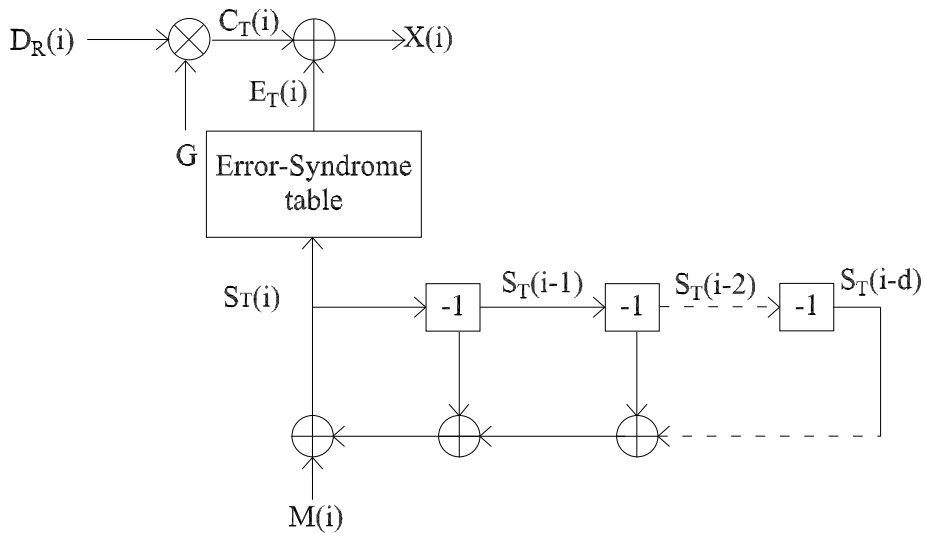


Figure 6.1: The encoder model of the chain based syndrome coding scheme

6.2.1.2 Legitimate decoder

We assume that Bob uses a conventional syndrome decoding scheme to estimate the original messages. By computing the syndrome associated with the receive codeword, which is given by:

$$S_{Bob}(i) = Y_M(i) \times \mathbf{H}^T = X(i) \times \mathbf{H}^T = S_T(i) \quad (6.4)$$

it is possible – in view of (6.2) – to obtain the original message immediately as follows:

$$\begin{aligned} M(i) &= S_{Bob}(i) + S_{Bob}(i-1) + \dots + S_{Bob}(i-d) \\ &= S_T(i) + S_T(i-1) + \dots + S_T(i-d) \end{aligned} \quad (6.5)$$

6.2.1.3 Eavesdropper decoder

We also assume that Eve – as Bob – uses a conventional syndrome decoding scheme to estimate the original messages. By computing the syndrome associated with the receive codeword as follows:

$$S_{Eve}(i) = Y_E(i) \times \mathbf{H}^T = (X(i) + e(i)) \times \mathbf{H}^T = S_T(i) + S_e(i) \quad (6.6)$$

where $S_e(i)$ is the corresponding syndrome of the additional error $e(i)$ in the eavesdropper channel. Since Eve doesn't know the initial conditions, he applies the following decoder to get the estimated message at time i : when $i \leq d$,

$$\begin{aligned} \hat{M}(i) &= S_{Eve}(i) + \dots + S_{Eve}(1) \\ &= M(i) + S_e(i) + \dots + S_e(1) + S_T(0) + \dots + S_T(i-d) \end{aligned} \quad (6.7)$$

and when $i > d$,

$$\begin{aligned} \hat{M}(i) &= S_{Eve}(i) + \dots + S_{Eve}(i-d) \\ &= M(i) + S_e(i) + \dots + S_e(i-d) \end{aligned} \quad (6.8)$$

Note that in the traditional syndrome coding scheme the encoding and decoding processes are memoryless, which is a special case of the chain based syndrome coding scheme where the chain depth $d = 0$. In contrast, in this chain based syndrome coding scheme the encoder and decoder exhibit memory: in particular, Alice, Bob and Eve have to use d delay registers to store the previous d transmitted or received syndromes.

6.2.2 Decoder output equivocation analysis

Since Eve doesn't know the initial conditions, the decoder output equivocation of Eve is as follows:

$$S_d = H(M(1), \dots, M(b) | \hat{M}(1), \dots, \hat{M}(b)) \quad (6.9)$$

$$= \sum_{i=1}^b H(M(i) | \hat{M}(1), \dots, \hat{M}(b), M(i-1), \dots, M(1)) \quad (6.10)$$

$$= \sum_{i=1}^d H(\hat{M}(i) + S_e(i) + \dots + S_e(1) + S_T(0) + \dots + S_T(i-d) | \hat{M}(1), \dots, \hat{M}(b), M(i-1), \dots, M(1)) \\ + \sum_{i=d+1}^b H(\hat{M}(i) + S_e(i) + \dots + S_e(i-d) | \hat{M}(1), \dots, \hat{M}(b), M(i-1), \dots, M(1)) \quad (6.11)$$

$$= d \times m + \sum_{i=d+1}^b H(\hat{M}(i) + M(i-1) + \hat{M}(i-1) + S_e(i) + S_e(i-d-1) \\ + \hat{M}(1), \dots, \hat{M}(b), M(i-1), \dots, M(1)) \quad (6.12)$$

$$= d \times m + \sum_{i=d+1}^b H(S_e(i) + S_e(i-d-1)) \quad (6.13)$$

(6.12) is due to the fact that the eavesdropper doesn't know the initial conditions $S_T(0) + \dots + S_T(i-d)$. In (6.13), only the first term depends on the value of d . The second term of (6.12) is generated as follows:

$$\hat{M}(i-1) = M(i-1) + S_e(i-1) + \dots + S_e(i-d) + S_e(i-d-1) \quad (6.14)$$

which leads to:

$$S_e(i-1) + \dots + S_e(i-d) = M(i-1) + \hat{M}(i-1) + S_e(i-d-1) \quad (6.15)$$

Applying (6.15) in (6.11), we will get (6.12) finally.

This scheme has a higher normalised equivocation, $R_e = \frac{S_d}{b \times m}$, than that of the traditional syndrome coding scheme. As d increases, the value of the first term in (6.13) increases, so R_e increases. However, for the case of $b \gg d$, the value of the chain depth has very limited effect on the normalised equivocation, since $\frac{d \times m}{b} \rightarrow 0$ and the value of the second term in (6.13) is independent of the value of the chain depth. We wish the chain depth has stronger effect on the equivocation, i.e. as the chain depth increases, the equivocation increases. Another chain based syndrome coding scheme with a hash function is introduced in next section to achieve this target.

6.3 A chain based syndrome coding scheme with a hash function

We employ a hash function in the chain based syndrome coding scheme.

6.3.1 The encoder and the decoder description

The exact operation of the chain based syndrome coding scheme, including the encoding and decoding processes, is described below.

6.3.1.1 Encoder

Alice produces the n -bit codeword $X(i)$ from the $m = (n - k)$ -bit message $M(i)$ at time i as follows:

- Alice generates a n -bit vector $C_T(i)$ from a random k -bit binary vector $D_R(i)$:

$$C_T(i) = D_R(i) \times \mathbf{G} \quad (6.16)$$

- Alice also generates a n -bit error pattern $E_T(i)$ from the m -bit syndrome $S_T(i)$ in accordance with the 'error-syndrome table', where the current syndrome is set to be equal to the sum of the current message and the previous d syndromes as follows:

$$S_T(i) = M(i) + f(S_T(i-1), \dots, S_T(i-d)) \quad (6.17)$$

where $f(\cdot)$ is a hash function with $d \times m$ -bit input vectors and m -bit output vectors and d is the chain depth. When the length, $d \times m$, of the input is fixed, we generate a hash table including all possible $2^{d \times m}$ input vectors, in which each of them maps to a randomly generated m -bit vector (those 2^m possible output vectors are uniformly distributed). This hash table is shared by Alice, Bob and Eve.

- Alice then generates the n -bit codeword $X(i)$ by the adding the n -bit vector $C_T(i)$ to the n -bit error pattern $E_T(i)$, i.e.,

$$X(i) = C_T(i) + E_T(i) \quad (6.18)$$

We assume that these initial conditions are known only by Alice and Bob. Eve doesn't know any information of the initial condition. This encoder procedure is depicted in Figure 6.2.

The information rate of this scheme is $R = \frac{n-k}{n} = \frac{m}{n}$.

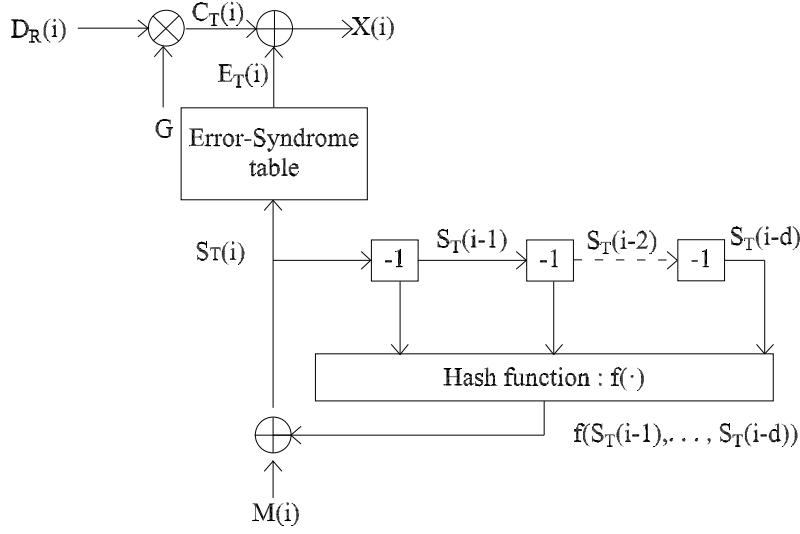


Figure 6.2: The encoder model of the chain based syndrome coding scheme with a hash function

6.3.1.2 Legitimate decoder

We assume that Bob uses a conventional syndrome decoding scheme to estimate the original messages. By computing the syndrome associated with the received codeword, which is given by:

$$\begin{aligned} S_{Bob}(i) &= Y_M(i) \times \mathbf{H}^T \\ &= X(i) \times \mathbf{H}^T = S_T(i) \end{aligned} \quad (6.19)$$

it is possible – in view of (6.17) – to obtain the original message immediately as follows:

$$\begin{aligned} M(i) &= S_{Bob}(i) + f(S_{Bob}(i-1), \dots, S_{Bob}(i-d)) \\ &= S_T(i) + f(S_T(i-1), \dots, S_T(i-d)) \end{aligned} \quad (6.20)$$

6.3.1.3 Eavesdropper decoder

We also assume that Eve – like Bob – uses a conventional syndrome decoding scheme to estimate the original messages. By computing the syndrome associated with the receive codeword as follows:

$$\begin{aligned} S_{Eve}(i) &= Y_E(i) \times \mathbf{H}^T \\ &= (X(i) + e(i)) \times \mathbf{H}^T \end{aligned} \quad (6.21)$$

$$= S_T(i) + S_e(i) \quad (6.22)$$

where $S_e(i)$ is the corresponding syndrome of the additional error $e(i)$ in the eavesdropper channel, then the estimate of the message at time i when $i \leq d$ is given by:

$$\begin{aligned}\hat{M}(i) &= S_{Eve}(i) + f(S_{Eve}(i-1), \dots, S_{Eve}(1)) \\ &= M(i) + S_e(i) + f(S_T(i-1), \dots, S_T(i-d)) \\ &\quad + f(S_{Eve}(i-1), \dots, S_{Eve}(1))\end{aligned}\quad (6.23)$$

and the estimate of the message at time i when $i > d$ is given by:

$$\begin{aligned}\hat{M}(i) &= S_{Eve}(i) + f(S_{Eve}(i-1), \dots, S_{Eve}(i-d)) \\ &= S_T(i) + S_e(i) + \dots + f(S_{Eve}(i-1), \dots, S_{Eve}(i-d)) \\ &= M(i) + S_e(i) + f(S_T(i-1), \dots, S_T(i-d)) \\ &\quad + f(S_{Eve}(i-1), \dots, S_{Eve}(i-d))\end{aligned}\quad (6.24)$$

Alice, Bob and Eve have to use d delay registers to store the previous d transmitted or received syndromes.

6.3.2 Decoder output equivocation analysis

We measure the security associated with this scheme, by considering the eavesdropper decoder output equivocation given by:

$$S_d = H(M(1), \dots, M(b) | \hat{M}(1), \dots, \hat{M}(b)) \quad (6.25)$$

where $H(\cdot|\cdot)$ denotes the conditional entropy.

This metric measures the eavesdropper uncertainty about the transmitted message $M(1), \dots, M(b)$ given the estimated message $\hat{M}(1), \dots, \hat{M}(b)$. It is known that the higher the value of the equivocation the lower the information leakage. The maximum value that the equivocation can take $S_d = mb$ is associated with communication that is perfectly secure.

We can calculate S_d based on the following expressions:

$$S_d = H(M(1), \dots, M(b) | \hat{M}(1), \dots, \hat{M}(b)) \quad (6.26)$$

$$= H(M(1), \dots, M(b), \hat{M}(1), \dots, \hat{M}(b)) - H(\hat{M}(1), \dots, \hat{M}(b)) \quad (6.27)$$

$$= H(M(1), \dots, M(b), \hat{M}(1), \dots, \hat{M}(b)) - b \times m \quad (6.28)$$

$$= - \sum_{X;Y} p(X, Y) \cdot \log_2 p(X, Y) - b \times m \quad (6.29)$$

in which $X = M(1), \dots, M(b)$ denotes the original transmitted sequence of messages, $Y = \hat{M}(1), \dots, \hat{M}(b)$ denotes the estimated sequence of messages, and $p(X; Y)$ is the joint probability distribution of X, Y . (6.28) is due to the fact of that $\hat{M}(1), \dots, \hat{M}(b)$ are uniformly distributed. Essentially, we want this distribution, $p(X, Y)$, to be as uniform as possible in order to get a higher equivocation.

The chain technique helps to increase the uncertainty of the eavesdropper about the transmitted information. As the chain depth, d , increases, the equivocation is increasing. Some results obtained by simulation are given in Section 6.3.3.

6.3.3 Simulation results

We simulated our chain based technique syndrome coding scheme using a (14,8) code derived from a repeated (7,4) Hamming code with $m = 3$ for different values of $d = 0, 1, 2$ at the BSC with a cross-over probability of α . We now present a series of results to showcase the behaviour of the eavesdropper decoder output normalized equivocation given by:

$$R_e = S/(m \cdot b) \quad (6.30)$$

in which $0 \leq R_e \leq 1$. When the value of R_e is 1, there is perfect secrecy with Eve gaining zero information from Alice.

The hash function is generated as a large look up table and shared by all the parties. We generate each message, $M(i)$, randomly, encode it to a n -bit vector, $X(i)$, and send this vector through a binary symmetric channel to Eve. Eve receives a corrupted version of X , $Y_E(i)$, and decodes it to $\hat{M}(i)$. We store all information of $M(i)$ and $\hat{M}(i)$, to estimate the probability distribution of $p(X, Y)$. At the end of the simulation $p(X, Y)$ is used to calculate $H(X, Y)$ and used in turn to calculate $H(X|Y)$ based on (6.29). Due to computer memory limitations and time constraints we only present results for codes with small values of m .

Figure 6.3 shows the equivocation rate for different chain depths ($d = 0$ corresponds to conventional syndrome-coding) as a function of the BSC crossover probability α for the (14,8) repeated Hamming code for different values of d . We observe that – for a specific block-length n – the higher the chain depth the higher the eavesdropper decoder output equivocation rate. The effectiveness of the chain depth is apparent. For $d = 2$ almost perfect secrecy is achieved when the BSC bit error probability, $\alpha > 0.1$.

The Kullback-Leibler divergence (KL distance) is a metric, which is used to measure the

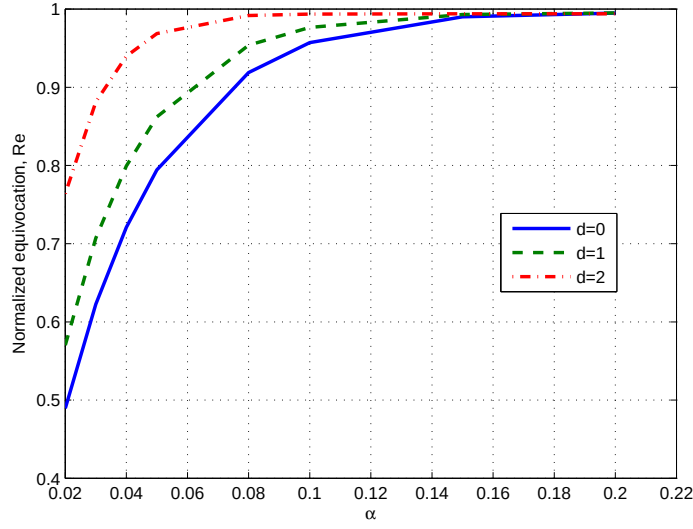


Figure 6.3: Normalised equivocation vs. α for the (14,8) repeated Hamming code

difference between two probability distributions [93] and is defined as follows:

$$D(P||Q) = \sum_{x \in X} P(x) \log_2 \frac{P(x)}{Q(x)} \quad (6.31)$$

where $D(P||Q)$ is the KL distance between the distribution P and Q , $P(x)$ and $Q(x)$ are the probabilities of x for two distribution. In our case, Q is the wanted distribution, the uniform distribution, and P is the distribution of $p(X, Y)$.

Figure 6.4 shows the KL distance between $p(X, Y)$ and the uniform distribution for the (14,8) repeated Hamming code. We observe that the KL-distance decreases, as the chain depth d increases for given values of the BSC cross-over probability α , i.e. the chain technique allows $p(X, Y)$ to get closer to the uniform distribution.

6.4 Conclusions

It has been shown that the chain based syndrome coding scheme results in less information available to an eavesdropper at the decoder output than the conventional syndrome coding scheme and the chain technique with a hash function improves the equivocation as the chain depth increases. Simulation results demonstrate that the chain depth has a profound effect on the eavesdropper decoder output equivocation rate in the chain based syndrome coding scheme with a hash function: the higher the value of the chain depth

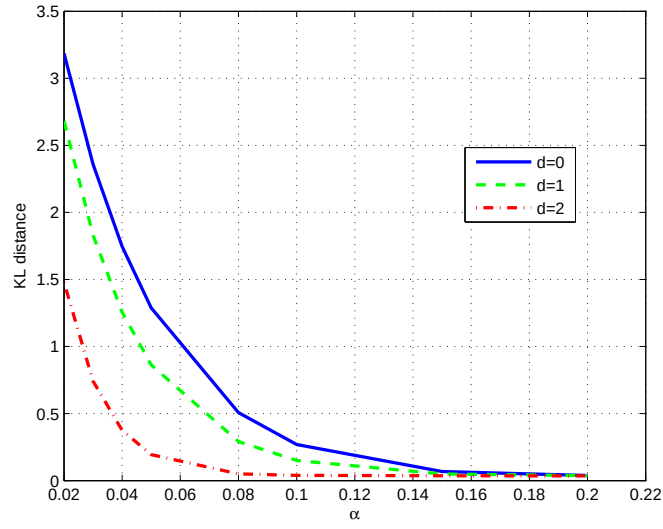


Figure 6.4: KL distance vs. α for the (14,8) repeated Hamming code

the higher the value of the equivocation rate. Our results also demonstrate that the lower the information rate the higher the equivocation rate. We anticipate that different choices of code, hash function and chain depth will lead to additional gains in secrecy.

Chapter 7

A Modified McEliece Public Key Encryption System with a Higher Security Level

This chapter analyses the equivocation of the eavesdropper under different attacks when the legitimate parties employ the McEliece public-key cryptosystem in the traditional wiretap scenario. The analysis is not on the best attack to break the cryptosystem but on the information leakage. We show that there is considerable information leakage to the eavesdropper under Brickell's attack. We also propose a modified McEliece public-key cryptosystem to reduce the information leakage under Brickell's attack. We prove that the modified system exhibits zero information leakage under Brickell's attack, and the price for this higher security is a lower information rate.

7.1 Introduction

McEliece proposed the first public key cryptosystem based on algebraic coding theory, the McEliece cryptosystem, using Goppa codes in 1978 [94]. The crux of the idea is based on the hardness of decoding an arbitrary linear code, which is known to be an NP-hard problem [95]. Compared with other public key cryptosystems, like RSA [96] the McEliece system has the advantage of high-speed encryption and decryption and is generally accepted to be quantum computer secure, and with the growth of the key size, the security grows much faster. In addition, the McEliece system uses probabilistic encryption, which is better than deterministic encryption in resisting to known plain text attacks [97]. However, there exist some drawbacks in the McEliece system, such as the

long public key, throughput efficiency and residual information leakage. To reduce the key size, Barbier and Barreto [98] exhibit a keysize gain of about 4% for the standard McEliece cryptosystem and up to 21% for the adjusted quasi-dyadic variant, by using list decoding algorithm [99]. Some research work points out the weak keys for McEliece system to help the system avoiding the weak keys [100, 101].

Security is an important issue in a communication system. In the past, various attacks were proposed to break the McEliece system. The most effective attacks on McEliece system are based on information-set decoding algorithms, such as Brickell's attack [102] and Stern's attack [103]. Berson's attack [104] considers the cases with resent-message or related-message. In 2008, Bernstein Lange and Peters [105] described a practical attack on the original McEliece cryptosystem, based on finding low-weight code words using an algorithm published by Jacques Stern in 1989 [103], and also gave a defending way for this attack. Various modifications of the McEliece cryptosystem have been proposed to prevent such attacks [97, 106, 107]. For example, Sun [97] proposes some variants of the McEliece public key cryptosystem to prevent Berson's attacks [104]. Baldi *et al.* [108] introduce a variant of the McEliece cryptosystem including low-density parity-check codes, which are used in many telecommunication standards and applications, to make the system practical and estimate the trade-off between security and complexity.

In this chapter, we view the McEliece cryptosystem from an information theory perspective, in particular the information leakage under Brickell's attack. Although Brickell's attack is not the best attack to break the cryptosystem, it is interesting from an information theory viewpoint. For the traditional wiretap scenario, we analyse the information leakage of the McEliece cryptosystem under two specific eavesdropper attacks: the Brickell's attack and the full-code attack. We also propose a modification to a common systematic version of the McEliece cryptosystem, which employs a syndrome-coding method [69] based on the $(23, 12, 7)$ Golay code – a perfect code – corrupted by random errors from a constrained probability distribution. Using equivocation analysis, we show that the modified system is able to achieve perfect secrecy under a specific Brickell's attack, assuming that the attack does not break the key. The price for this improvement in secrecy is a lower information rate.

7.2 McEliece Public Key Encryption System

7.2.1 System Description

We consider a standard wiretap scenario, where a transmitter, Alice, sends a (uniformly distributed) binary message M to a legitimate receiver, Bob, in the presence of an eavesdropper, Eve (see Figure 7.1). Alice and Bob employ the McEliece public-key cryptosystem to exchange the message. The key generation, message encryption and message decryption processes are described below.

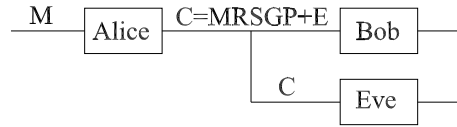


Figure 7.1: Wiretap scenario

7.2.1.1 Key generation

Bob selects a $(n, k, 2t + 1)$ binary Goppa code capable of correcting t errors with a $k \times n$ generator matrix $\mathbf{G}$ and also selects randomly a $k \times k$ non-singular matrix $\mathbf{S}$ and a $n \times n$ permutation matrix $\mathbf{P}$. Bob also determines a $k \times k$ matrix $\mathbf{R} \neq \mathbf{S}^{-1}$ that – by row additions and column swaps – produces a reduced echelon form of $\mathbf{SGP}$, i.e. $\mathbf{RSGP} = [\mathbf{I}_k | \mathbf{Q}]$ where $\mathbf{I}_k$ is the $k \times k$ identity matrix and $\mathbf{Q}$ is a $k \times (n - k)$ parity check matrix. The public key, which is known to everyone, is the $k \times n$ matrix $\mathbf{RSGP}$. The private key, which is only known to Bob, is the set of matrices $(\mathbf{R}, \mathbf{S}, \mathbf{G}, \mathbf{P})^1$.

7.2.1.2 Message encryption

Alice generates the n -bit cryptogram C from the k -bit binary message M as follows:

- i) Alice generates the n -bit vector $C' = MRSGP$ by using the public key as a generator matrix;
- ii) Alice generates the n -bit cryptogram $C = C' + E = MRSGP + E$, by adding a random n -bit vector E of weight t .

¹We consider this systematic version, since the information theoretic properties of $\mathbf{SGP}$ and $\mathbf{RSGP}$ are identical and it is easier to store the public key associated with the systematic format [109].

7.2.1.3 Message decryption

Bob recovers the k -bit message M from the n -bit cryptogram C as follows:

- i) Bob computes the n -bit vector $CP^{-1} = MRS G + EP^{-1}$ by using the inverse of P ;
- ii) Bob then computes the correct k -bit vector MRS by using the algebraic decoding algorithm of the Goppa code;
- iii) Multiplying by the inverses of the non-singular matrix S and the matrix R finally yields the k -bit message M .

It is interesting to note that a traditional way to characterize the security of the McEliece cryptosystem involves the calculation of the work factor w , which is the average number of operations required to successfully get the message.

7.2.2 Popular Attacks

There are several ways of attacking the McEliece cryptosystem. We briefly introduce four popular attacks.

Attack 1 (Brickell's Attack): The best known attack has been proposed by Lee and Brickell in [102]. This attack involves observing k “full-rank” components of the cryptogram given by:

$$[C]_k = M[RS GP]_k + [E]_k = [C']_k + [E]_k \quad (7.1)$$

where $[C]_k$ denotes the k components of C , $[C']_k$ denotes the k components of C' , $[E]_k$ denotes the k components of E , and $[RS GP]_k$ is a full-rank $k \times k$ sub-matrix of the $k \times n$ matrix $RS GP$.

Since $[C]_k + [E]_k = M[RS GP]_k$, then

$$(C_k + E_k)([RS GP]_k)^{-1} = M \quad (7.2)$$

Note that the eavesdropper can recover the message M when the k components of the error pattern E are zeros. The total expected work factor for this attack – in view of the

fact that matrix inversion requires k^α steps – is given by [110]:

$$w = k^\alpha \binom{n}{k} / \binom{n-t}{k}. \quad (7.3)$$

in which $2 < \alpha \leq 3$.

Attack 2 (Full-Code Attack): Another possible attack involves observing all the n components of the cryptogram. This requires a very powerful computer to obtain the message from the cryptogram via numerous attempts.

Attack 3 (Message-Resend Attack): Assume that a message M is encrypted twice due to some accident or a special action of Eve. Since Eve knows $C_1 = MRSGP + E_1$ and $C_2 = MRSGP + E_2$, then Eve also knows $E_1 + E_2 = C_1 + C_2$. It turns out that – via the analysis of the weight of $C_1 + C_2$ – Eve requires fewer attempts to get the correct message compared to the full-code or Brickell’s attack [104].

Attack 4 (Related-Message Attack): Assume that two messages M_1 and M_2 are encrypted. Assume also that Eve knows a linear relation between the messages, e.g. $M_1 + M_2$. Since Eve knows $C_1 = M_1RSGP + E_1$, $C_2 = M_2RSGP + E_2$ and $(M_1 + M_2)RSGP$, then Eve also knows $E_1 + E_2 = C_1 + C_2 + (M_1 + M_2)RSGP$. It also turns out that – via the analysis of the weight of $E_1 + E_2$ – Eve requires fewer attempts to get the correct message compared to the full-code or Brickell’s attack [104].

We restrict our attention to *Attack 1* and *Attack 2* only.

7.2.3 Equivocation Analysis

We use equivocation to measure the uncertainty of the eavesdropper.

Attack 1: In Brickell’s attack, the equivocation of Eve is given by:

$$\begin{aligned} H(M|[C]_k) &= H(M, [C]_k) - H([C]_k) \\ &= - \sum_{M, [C]_k} p([C]_k|M) \cdot p(M) \cdot \log(p([C]_k|M) \cdot p(M)) \\ &\quad + \sum_{[C]_k} p([C]_k) \cdot \log p([C]_k) \end{aligned} \quad (7.4)$$

where

$$p(M) = \frac{1}{2^k} \quad (7.5)$$

is the probability mass function of the k -bit uniformly distributed message M ,

$$\begin{aligned} p([C]_k|M) &= p([C']_k + [E]_k|M) \\ &= p(M[\mathbf{RSGP}]_k + [E]_k|M) = p([E]_k) \end{aligned} \quad (7.6)$$

if M maps to the $[C']_k$ upon which adding $[E]_k$ produces $[C]_k$, and

$$p([C]_k|M) = 0 \quad (7.7)$$

otherwise, is the probability mass function of the k components of the cryptogram C given a specific message M and

$$p([C]_k) = \sum_M p([C]_k|M) \cdot p(M) \quad (7.8)$$

is the probability mass function of the k components of the cryptogram C .

Let us now explain how to calculate the probability mass functions $p([C]_k)$ and $p([C]_k|M)$, which are necessary to compute the equivocation. These depend directly on the probability mass function $p([E]_k)$. Since the weight of the n -bit error pattern E is t , we have that the number of distinct k -bit error patterns $[E]_k$ is $\sum_{s=0}^t \binom{k}{s}$, where s is the weight of $[E]_k$. The probability of the k -bit error pattern $[E]_k$ depends on its weight only, since each cryptogram of length n bits contains exactly t errors. The probability of a particular k -bit error pattern with a weight s is $\binom{n-k}{t-s} / \binom{n}{t}$. Note that $p([E]_k)$ does not depend on the choice of the k components.

We calculate Eve's normalised equivocation $H(M|[C]_k)/H(M)$ and Eve's work factor w for various sets of parameters for the McEliece system (see Figure 7.2 and Figure 7.3). We observe that there is considerable information leakage in the McEliece cryptosystem. In particular, the higher the information rate the lower the normalised equivocation; in contrast, the work factor achieves the highest value when the information rate is approximately $R = 0.64$. We also observe that the higher the cryptogram length the higher the work factor but the lower the normalised equivocation.

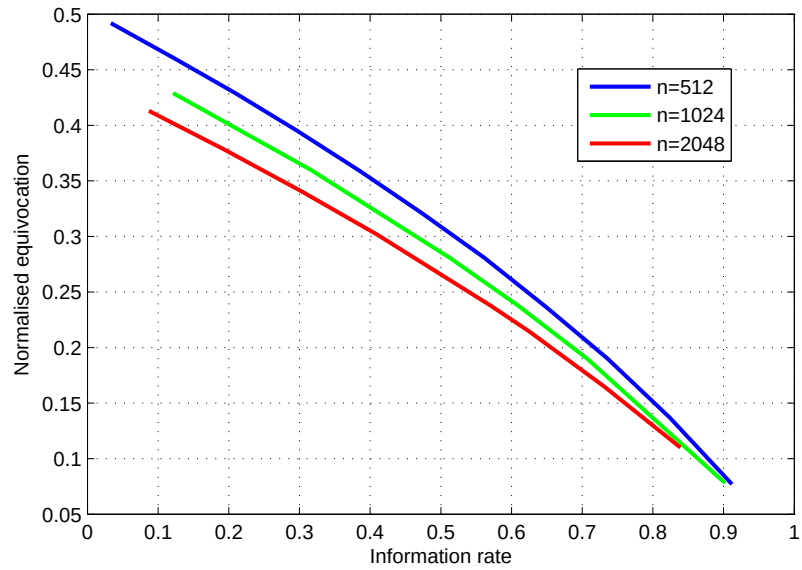


Figure 7.2: Normalised equivocation vs. information rate (k/n)

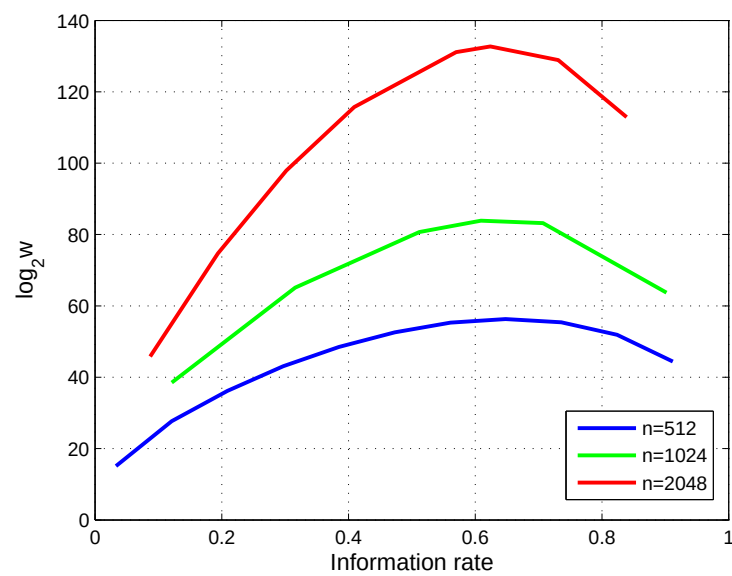


Figure 7.3: Work factor vs. information rate (k/n)

Attack 2: In the full-code attack, the equivocation of Eve is given by:

$$\begin{aligned}
 H(M|C) &= H(M, C) - H(C) \\
 &= - \sum_{M, C} p(C|M) \cdot p(M) \cdot \log(p(C|M) \cdot p(M)) \\
 &\quad + \sum_C p(C) \cdot \log p(C)
 \end{aligned} \tag{7.9}$$

where $p(M) = \frac{1}{2^k}$ is the probability mass function of the k -bit uniformly distributed message M , $p(C|M) = p(C' + E|M) = p(M\mathbf{RSGP} + E|M)$ is the probability mass function of the cryptogram C given a specific message M and $p(C) = \sum_M p(C|M) \cdot p(M)$ is the probability mass function of the cryptogram C .

Let us now calculate the probability mass functions $p(C)$ and $p(C|M)$, which are also necessary to compute the equivocation. We note that a single M maps onto a unique C' ; in fact, the mapping M to C' is one-to-one. We also note that a single C' maps onto $\binom{n}{t}$ distinct C , because there are $\binom{n}{t}$ distinct error patterns E . Since the minimum Hamming distance of the Goppa code is $2t + 1$, we also observe the set of cryptograms produced by the set of messages are all different. Therefore,

$$p(C|M) = \frac{1}{\binom{n}{t}} \tag{7.10}$$

if M maps to the C' upon which adding E produces C , and

$$p(C|M) = 0 \tag{7.11}$$

otherwise. In turn,

$$p(C) = \frac{1}{\binom{n}{t} \times 2^k} \tag{7.12}$$

By using $p(M) = \frac{1}{2^k}$, (7.10), (7.11) and (7.12) in (7.9), we can calculate the equivocation as follows:

$$H(M|C) = k \cdot \log_2 \binom{n}{t} - k \cdot \log_2 \binom{n}{t} = 0 \tag{7.13}$$

We observe that, under the full-code attack, Eve's equivocation is zero, i.e. Eve can recover the exact message at the expense of a very high computational complexity.

7.3 Modified McEliece Public Key Encryption System with Higher Security Level

7.3.1 System Description

We now modify the McEliece cryptosystem in order to reduce the information leakage of the communication. The new system differs from the traditional system in two aspects:

- i) In the traditional system the binary message M is processed directly by the McEliece cryptosystem; in the new system the binary message M is first processed using a syndrome coding method [69] based on the (23, 12, 7) Golay code and then by the McEliece cryptosystem;
- ii) In the traditional system we add a random n -bit error vector E with weight t ; in the new system we add a random n -bit error vector E with weight t that obeys some additional constraints.

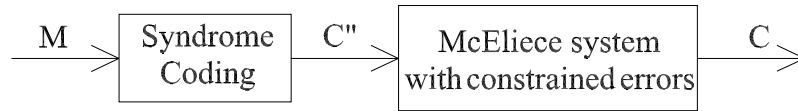


Figure 7.4: Main blocks of the Modified McEliece cryptosystem

The key generation, message encryption and message decryption processes are described below.

7.3.1.1 Key generation

The generation process of the public key and private key in the new system is identical to that in the traditional McEliece cryptosystem. The public key is the matrix $RS GP$ and the private key is the set of matrices (R, S, G, P) .

7.3.1.2 Message encryption

Alice generates the n -bit cryptogram C from a k -bit vector C''' , which relates to the message M , as follows:

- i) Alice generates the n -bit vector $C' = C''\mathbf{RSGP}$ by using the public key as a generator matrix;
- ii) Alice then generates the n -bit cryptogram $C = C' + E = C''\mathbf{RSGP} + E$, by adding a random n -bit vector E of weight t that obeys some additional constraints.

We will now illustrate how to generate the k -bit vector C'' . We partition the message M into l blocks of 11 bits, i.e. $M = [M_1, M_2, \dots, M_l]$, and the k -bit vector C'' into l blocks of 23 bits plus an additional block of $k - 23 \times l$ random data bits, i.e. $C'' = [C''_1, C''_2, \dots, C''_l, DR_0]$ where DR_0 is a $k - 23 \times l$ -bit random data. The 11-bit message block M_i then maps to the 23-bit block C''_i by using a syndrome coding method based on the (23, 12, 7) Golay code. In particular, $C''_i = C''_{0i} + E'_i$ with $C''_{0i} = DR_i \times G_0$ where G_0 is the 12×23 generator matrix of the (23, 12, 7) Golay code, DR_i is a 12-bit random data vector and E'_i is a 23-bit error pattern that relates to the 11-bit message M_i . Note that – in view of the fact that the (23, 12, 7) Golay code is perfect – the mapping between weight 0, 1, 2 and 3 error patterns E'_i and syndromes S_i is one-to-one. We obtain E'_i from M_i by setting $S_i = M_i$.

We will now illustrate how to generate the n -bit vector E of weight t . We write $E = [E_1, E_2, \dots, E_l, E']$ where $E_i, 1 \leq i \leq l$, are 23-bit error vectors, which correspond to the Golay codewords, E' is a $n - 23 \times l$ -bit error vector. There are two possible ways to define the n -bit vector E , depending on the choice of the number of Golay coded blocks l .

- *The basic scheme:* In this scheme, the number of Golay coded blocks, l , is set to be equal to $\min\left(\left\lfloor \frac{k}{23} \right\rfloor, \left\lfloor \frac{t}{3} \right\rfloor\right)^2$, which ensures that we can fit 3 errors in all blocks if required. In particular, Alice selects the weight of $E_i, w(E_i)$, to be 0 to 3 randomly and the weight of $E', w(E')$, to be $t - \sum_{i=1}^l w(E_i)$. For the (1024, 524, 101) standard McEliece system, where $t = 50, l = 16$ and the number of information bit per cryptogram is $16 \times 11 = 176$.
- *The adaptive scheme:* In this scheme, the number of Golay coded blocks l adapts to the number of errors added to the previous blocks (note that $l \leq \left\lfloor \frac{k}{23} \right\rfloor$). In particular, Alice selects the weight of $E_i, w(E_i)$, to be 0 to 3 randomly, until the t errors run out. Alice continuously calculates the number of added errors to decide about the number of the blocks l . Since Bob can get the error pattern E , he also

² $\min(x, y)$ is the smallest value function. $\lfloor x \rfloor$ is the largest integer not greater than x .

knows the number of Golay coded blocks present in the cryptogram. For the (1024, 524, 101) standard system, the number of Golay coded blocks, l , ranges from 17 to 22, depending on the outcome of the uniform distribution for E_i , and the number of information bit per cryptogram ranges from $16 \times 11 = 176$ to $22 \times 11 = 242$.

7.3.1.3 Message decryption

Bob recovers the message vector M from the n -bit cryptogram C as follows:

- i) Bob computes the n -bit vector $CP^{-1} = C''RSG + EP^{-1}$ by using the inverse of P ;
- ii) Bob then computes the correct k -bit vector $C''RS$ by using the algebraic algorithm of the Goppa code;
- iii) Multiplying by the inverses of the nonsingular matrix S and the matrix R finally yields the k -bit vector C'' ;
- iv) Finally, Bob calculates the syndromes of C''_i , $1 \leq i \leq l$, to obtain the messages M_i , $1 \leq i \leq l$.

7.3.2 Equivocation Analysis

We now analyse the equivocation of the modified McEliece cryptosystem under the Brickell's and the full-code attack.

Attack 1: In the Brickell's attack, Eve observes k fixed components of the cryptogram. We concentrate on a specific form of Brickell's attack where Eve observes only the first k components of the cryptogram, because – in view of the fact that $RS GP = [I_k | Q]$ – the information bits are potentially exposed. Thus, Eve's equivocation is given by³:

$$H(M|[C]_k) = H(M|[C]_{k'}) \quad (7.14)$$

$$\begin{aligned} &= H(M) + H([C]_{k'}|M) - H([C]_{k'}) \\ &= H(M) - H([C]_{k'}) + H([E]_{k'}|M) + H([C]_{k'}|[E]_{k'}, M) \\ &\quad - H([E]_{k'}|[C]_{k'}, M) \end{aligned} \quad (7.15)$$

$$\begin{aligned} &= H(M) - H([C]_{k'}) + H([E]_{k'}) + H([C]_{k'}|M) \\ &\quad - H([E]_{k'}|[C]_{k'}, M) \end{aligned} \quad (7.16)$$

³ $[V]_{k'}$ and $[V]_k$ refer to the fixed k' and k elements of vector V

where (7.14) is due to the fact that Eve's equivocation based on the observation of the first k bits of the cryptogram is equal to Eve's equivocation based on the observation of the first $k' = 23 \times l$ bits of the cryptogram because the last $k - 23 \times l$ bits of the first k bits are filled with random data, (7.15) is due to

$$\begin{aligned} H([C]_{k'}, [E]_{k'} | M) &= H([E]_{k'} | M) + H([C]_{k'} | [E]_{k'}, M) \\ &= H([C]_{k'} | M) + H([E]_{k'} | [C]_{k'}, M) \end{aligned}$$

and (7.16) is due to

$$H([E]_{k'} | M) = H([E]_{k'})$$

and

$$\begin{aligned} H([C]_{k'} | [E]_{k'}, M) &= H([C']_{k'} + [E]_{k'} | [E]_{k'}, M) \\ &= H([C']_{k'} | [E]_{k'}, M) = H([C']_{k'} | M) \end{aligned}$$

The calculation of the various entropies is based on the following observations:

- i) A single fixed message M maps into $2^{12 \times l}$ distinct $[C'']_{k'}$ with equal probability, due to the syndrome coding method.
- ii) A single fixed $[C'']_{k'}$ maps into a single $[C']_{k'}$ (in fact, $[C']_{k'} = [C'']_{k'}$, due to the fact that $\mathbf{RSGP} = [\mathbf{I}_k | \mathbf{Q}]$).
- iii) A single fixed $[C']_{k'}$ maps into $2^{11 \times l}$ distinct $[C]_{k'}$ with equal probability, due to the error pattern generation method.
- iv) In the view of i), ii) and iii) and the syndrome coding method, a single fixed message M maps into $2^{23 \times l}$ distinct $[C]_{k'}$ with equal probability, that cover the space because $k' = 23 \times l$. This also implies that $[C]_{k'}$, unconditioned on M , follows a uniform distribution.

Therefore, we have that $H(M) = 11 \times l$, $H([C]_{k'}) = 23 \times l$, $H([E]_{k'}) = 11 \times l$, $H([C']_{k'} | M) = 12 \times l$ and $H([E]_{k'} | [C]_{k'}, M) = 0$. and so we also have that

$$H(M | [C]_k) = H(M) \tag{7.17}$$

We observe that Eve's equivocation is equal to $H(M)$, i.e. Eve can not recover any useful information under this particular Brickell's attack⁴.

Attack 2: In the full-code attack, Eve observes all the components of the cryptogram. Thus, the equivocation is given by:

$$\begin{aligned} H(M|C) &= H(M) + H(C|M) - H(C) \\ &= - \sum_M p(M) \cdot \log(p(M)) - \sum_{M,C} p(C|M) \cdot p(M) \\ &\quad \cdot \log(p(C|M)) + \sum_C p(C) \cdot \log p(C) \end{aligned} \quad (7.18)$$

where $p(M) = \frac{1}{2^{11 \times l}}$ is the probability mass function of the $11 \times l$ -bit uniformly distributed message M , $p(C|M) = p(C' + E|M) = p(C''RSGP + E|M)$ is the probability mass function of the cryptogram C given a specific message M and $p(C) = \sum_M p(C|M) \cdot p(M)$ is the probability mass function of the cryptogram C .

Let us now calculate the probability mass functions $p(C)$ and $p(C|M)$ which are necessary to compute the equivocation. Note that:

- i) A single message M maps into $2^{12 \times l + k - 23 \times l} = 2^{k - 11 \times l}$ distinct C'' with equal probability.
- ii) A single C'' maps into a single C' (in fact, the mapping C'' to C' is one-to-one).
- iii) A single C' maps into N_E distinct C with equal probability, where N_E is the number of distinct error patterns.

Therefore,

$$p(C|M) = \frac{1}{2^{k - 11 \times l} \times N_E} \quad (7.19)$$

if M maps to the C' upon which adding E produces C , and

$$p(C|M) = 0 \quad (7.20)$$

otherwise. In turn,

$$p(C) = \frac{1}{2^k \times N_E} \quad (7.21)$$

⁴One can argue through that Eve is able to extract information based on the observation of another set of k components.

By using $p(M) = \frac{1}{2^{11 \times l}}$, (7.19), (7.20) and (7.21) in (7.18), we can calculate the equivocation as follows:

$$H(M|C) = 11 \times l + k - 11 \times l + \log N_E - k - \log N_E = 0$$

We get that Eve's equivocation is zero under the full-code attack, i.e. Eve can recover the message at the expense of a very high computational complexity.

As the analysis above, the modified McEliece cryptosystem works well to defend Brickell's attack. However, for the full code attack, the equivocation is also zero, which is not surprise, since the channel is error free and the secrecy capacity of this model is zero.

7.3.3 Information Rate Analysis

In our modified McEliece system, the price for the higher level of security is a lower information rate. We now compare the information rate of the modified system with that of the traditional McEliece system. The information rate for the traditional McEliece system is: $R_t = k/n$.

In turn, the information rate for the modified McEliece system is $R_m = l \times 11/n$, in which, l is the number of blocks which carries the message and constrained by: $l \leq \lfloor \frac{k}{23} \rfloor$ and $\sum_{i=1}^l Ne_i \leq t$.

Table 7.1 shows that the information rate of the modified system for the adaptive scheme is about half that of the traditional system.

7.4 Conclusions

In this chapter, we analysed the equivocation rate of the McEliece cryptosystem for the full-code attack and Brickell's attack, showing that there is considerable information leakage. In order to reduce this we propose a modified system, which uses a syndrome-coding method based on the perfect Golay (23,12,7) code and constrain the probability distribution of the added random errors. It has been shown that under a particular Brickell's attack that examines the potentially unprotected "systematic" bits the information

Table 7.1: Information rate for traditional McEliece system and modified McEliece system

n	k	t	R (traditional)	R_m (basic)	R_m (adaptive)
1024	724	30	0.707	0.107	0.107-0.333
1024	524	50	0.512	0.172	0.172-0.236
1024	324	70	0.316	0.150	0.150
2048	1498	50	0.731	0.086	0.086-0.349
2048	838	110	0.409	0.193	0.193
2048	398	150	0.194	0.091	0.091

leakage of a "systematic" version of the modified McEliece cryptosystem is zero. The price is a penalty in the information rate.

Chapter 8

Conclusions and Future Work

This chapter summarizes our main conclusions and points out some directions for future work.

8.1 Conclusions

Physical layer security, which is based on the idea of that the inherent randomness and noises of the physical medium are the resources for information-theoretic security, has a significant potential to enhance the security level of the current communication systems: hence it advocates a paradigm shift in cryptography and calls for a cross-layer design of security scheme. Although compared to computational security, the physical layer security has a main advantage due to the fact that there are no computation restrictions on the eavesdropper, there exist few practical scheme to guarantee the perfect secrecy in physical layer.

This dissertation has been concerned with the security analysis and the design of physical layer coding schemes that aim to improve the reliability and security of information transmitted over the wireless medium. Based on the combination of Shannon's notion of perfect secrecy and coding theory, this dissertation gives a deep analysis on the security of some traditional coding schemes in the physical layer and also proposes some novel practical coding schemes and a code design technique to strengthen the communication security.

Wyner's coding scheme provides a basic structure for secure communications that in-

volves the addition of extra randomness to confuse the eavesdropper about the useful information in the wiretap channel. Chapter 3 designs the orthogonal codes based on Wyner's coding structure. Both the theoretical analysis and the simulation results show that this scheme can achieve the reliable and perfect secure communication in the Gaussian wiretap channel in wideband regime.

Based on Wyner's code structure, syndrome coding scheme has been introduced and widely applied in information-theoretic security. This dissertation gives a deep analysis on the security of the traditional syndrome coding scheme. To study the security performance of the syndrome coding scheme, Chapter 4 introduces an efficient and practical evaluation method of the equivocation, which is a metric of secrecy. It is important to study the impact of different codes used in the syndrome coding scheme. In particular, random codes and best equivocation codes have been analysed. Chapter 4 proposes a Best Equivocation Code construction technique to design the optimum codes for the syndrome coding scheme to enhance the security of communication. It is shown that the Best Equivocation Codes can provide a higher level of the security for the syndrome coding scheme than that the best known codes provide. Moreover, Chapter 5 analyses the security performance of the random linear codes in the syndrome coding scheme. A theoretical analysis method has been proposed to model (n, k) random codes with a putative (n, k) code, which has the same pmf of syndrome probabilities as the ensemble of all (n, k) random codes, to evaluate the average equivocation of the random linear codes. It has been shown that our theoretical analysis is practical for codes with large values of m , the number of parity bits, even though Monte Carlo analysis is impractical. Similar with Shannon's theorem, it shows that there must be at least one code from the random code ensemble in syndrome coding scheme that approaches the secrecy capacity and the longer the code is, the closer it can be.

BEC, which have been proposed in Chapter 4, can achieve the highest security level for the traditional syndrome coding scheme. In order to enhance the security of the communication, Chapter 6 proposes and studies two variants of traditional syndrome coding scheme: a chain based syndrome coding scheme and a chain based syndrome coding scheme with a hash function. It is shown that the chain technique with a hash function helps to increase the uncertainty of the eavesdropper without decreasing the information rate. The chain based syndrome coding scheme with a hash function has a low computation complexity and exhibits significantly lower information leakage than the traditional syndrome coding scheme. Actually, we can also use BEC in this novel scheme to strengthen the security of communication further.

In addition, we employ the secrecy metric in information-theoretic security, the equivocation, to measure the security level of the McEliece public key encryption system for different attacks in Chapter 7. We propose a modified McEliece system defend Brickell's attack, which can achieve zero information leakage communications.

Overall, the communication security can be enhanced by different coding schemes and code design technique we have proposed in this dissertation for the physical layer, which is based on the exploration on the characteristics of the physical channel to get benefits from the difference of the noise levels between the main channel and the eavesdropper channel.

8.2 Future work

The work presented in this dissertation could be extended in many interesting directions.

- **A study of strong secrecy for orthogonal codes:** For the orthogonal codes, this dissertation concentrates on the analysis on the weak secrecy. It is necessary to do some extended analysis on the strong secrecy of the orthogonal codes, measured as variational distance [111], in Gaussain wiretap channel in wideband regime.
- **Non-binary best codes design for the syndrome coding scheme:** This dissertation designs the best equivocation code for the syndrome coding scheme in the binary field. Actually, the main idea of our code construction technique can also be employed for the q -ary field by using the conventional evaluation method for the syndrome probability distribution calculation. It will be interesting to generalize the code design technique to non-binary and possibly non-linear code constructions.
- **Security analysis on the random codes without any restriction:** This dissertation studies the average equivocation of the the random code, which has a restriction of that there is no repeated integers in the parity check matrix of the codes. We can use similar method to analyse the random codes without any restriction. It has to be careful to analyse the 1-error events, which are possible mapping to the same syndrome.

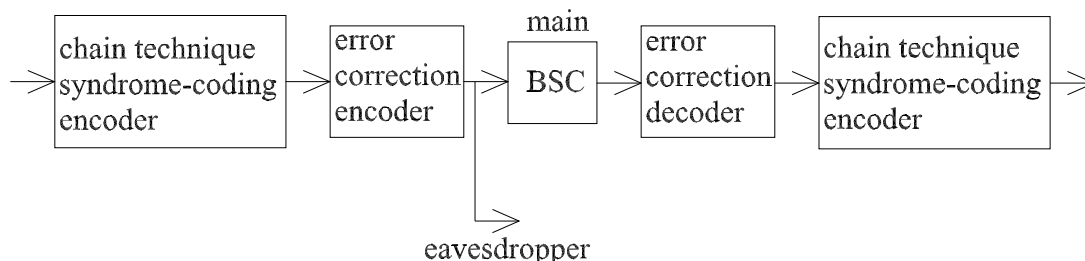


Figure 8.1: Code concatenation model for BSC

- **A general model for the syndrome coding scheme:** For both the conventional syndrome coding and the chain based syndrome coding scheme, the considered channel is a specific wiretap channel where the main channel is error free and the eavesdropper channel is binary symmetric. In practice, the main channel always has some errors, so it is important to generalize the results from the noiseless main channel-binary symmetric eavesdropper channel model to a model where both the main and the eavesdropper channels are binary symmetric. To solve this problem, we can use the code concatenation technique proposed by Cassuto and Bandic [70]. In particular, this entails using a standard error correction code as an inner code to transform the noisy main channel in a nearly noiseless channel and our chain based syndrome coding approach as an outer code to provide security (see Figure 8.1). This approach could pave the way to address a number of real applications in wired and wireless settings for syndrome coding secrecy.
- **Cross layer security schemes:** Information-theoretic security can enhance the security of the communication, which doesn't rely on the assumptions of limited computation resources, but it has some assumptions on the communication channels which may not be extremely accurate in practice. In practice, we can employ the information-theoretic security schemes in the physical layer to be part of security solutions, together with the cryptographic techniques in other layers. This cross layer security schemes, which combine the information-theoretic security and the computational security, could be very promising to guarantee secure communications in the future.

- **Variations of McEliece encryption system to defend other attacks:** This dissertation gives a solution to defend Brickell's attack for the McEliece public key encryption system to achieve zero information leakage. It is necessary to consider the equivocation for other attacks and try to design some variations to defend those attacks to guarantee a certain level of secrecy.
- **Multi-user information-theoretic security:** This dissertation and most of current work of the information-theoretic security focus on the wiretap channel, but rare attempt has been considered for multi-user networks. It is useful to generalise the secure coding schemes in physical layer proposed for point-to-point communications to multi-user network communication.

References

- [1] A. E. Earle , “Wireless Security Handbook”, *Taylor and Francis group, 6000 Broken Sound Parkway NW, Suite 300, Boca Raton, FL 33487-2748: Auerbach Publications*, 2006.
- [2] Y. Liang, H. V. Poor and S. Shamai, “Information Theoretic Security”, *Foundations and Trends in Communications and Information Theory*, vol. 5, pp. 355-580, 2008.
- [3] M. Debbah, H. El-Gamal, H. V. Poor, S. Shamai, “Wireless Physical Layer Security”, *Hindawi Publishing Corporation, EURASIP Journal on Wireless Communications and Networking*, 2009.
- [4] R.L. Rivest, A. Shamir, L. Adleman, “A method for obtaining digital signatures and public-key cryptosystems”, *Journal of Communications of the ACM*, vol. 21, no. 2, pp. 120-126, 1978.
- [5] W. Diffie and M. E. Hellman, “New directions in cryptography”, *IEEE Transactions on Information Theory*, vol.22, no.6, pp. 644-654, 1976.
- [6] H. Delfs, H. Knebl, “Symmetric-key encryption”, *Introduction to Cryptography: Principles and Applications*, Springer, pp. 11-32, 2007.
- [7] A. J. Menezes. P.C. van Oorschot and S. A. Vanstone, “Handbook of Applied Cryptography”, *Boca Raton, FL, USA: CRC Press*, 1996.
- [8] S. Wolf, “Unconditional security in cryptography”, *Lectures on Data Security*, Springer Berlin Heidelberg, pp 217-250, 1999.
- [9] C. E. Shannon, “Communication theory of secrecy systems”, *The Bell Syst. Tech. J.*, vol. 28, pp. 656-715, 1948.
- [10] A. D. Wyner, “The wire-tap channel”, *Bell Syst. Tech. J.*, vol. 54, pp. 1355-1367, 1975.

References

- [11] I. Csiszár and J. Körner, “Broadcast channels with confidential messages”, *IEEE Transactions on Information Theory*, vol. 24, no. 3, pp. 339-348, May 1978.
- [12] L. H. Ozarow and A. D. Wyner, “Wire-tap channel II”, *Bell Syst. Tech. J.*, vol. 63, no. 10, pp. 2135-2157, Dec. 1984.
- [13] G. Cohen and G. Zemor, “Generalized coset schemes for the wire-tap channel: application to biometrics”, *Proceedings. International Symposium on Information Theory 2004*, Chicago, pp. 46, Jun. 2004.
- [14] S. Reddy, P. Aparna, S. David, “Syndrome coding of video with LDPC codes”, *9th International Conference on Signal Processing, 2008*, Beijing, pp. 1985-1988, Oct. 2008.
- [15] A. T. Suresh, A. Subramanian, A. Thangaraj, M. Bloch, S. W. McLaughlin, “Strong secrecy for erasure wiretap channels”, *Proceedings of 2010 IEEE Information Theory Workshop (ITW'10)*, Dublin, pp. 1-5, Sep. 2010.
- [16] E. R. A. Salim and S. Emina, “On wiretap networks II”, *IEEE International Symposium on Information Theory, 2007*, Nice, pp. 551-555, Jun. 2007.
- [17] U. Maurer, “Secret key agreement by public discussion from common information”, *IEEE Transactions Information Theory*, vol. 39, no. 3, pp. 733-742, 1993.
- [18] J. Barros and M. R. D. Rodrigues, “Secrecy capacity of wireless channels”, *Proc. IEEE International Symposium on Information Theory*, Seattle, WA, USA, Jun. 2006.
- [19] Z. Li, R. Yates, and W. Trappe, “Secrecy capacity of independent parallel channels”, *Securing Wireless Communications at the Physical Layer*, Springer US, pp 1-18, 2010.
- [20] Y. Liang, H. V. Poor, and S. Shamai, “Secrecy capacity region of fading broadcast channels”, *Proc. of IEEE International Symposium on Information Theory*, Nice, France, Jun. 2007.
- [21] P. K. Gopala, L. Lai, and H. El-Gamal, “On the secrecy capacity of fading channels”, *Proc. of IEEE International Symposium on Information Theory*, Nice, France, Jun. 2007.
- [22] Y. Liang and H. V. Poor, “Multiple access channels with confidential messages”, *IEEE Transactions on Information Theory*, vol. 54, no. 3, pp. 976-1002, Mar. 2008.

References

- [23] Y. Liang, H. V. Poor and S. Shamai, "Secure communication over fading channels", *IEEE Transactions on Information Theory-Special Issue on Information Theoretic Security*, vol. 54, no. 6, pp. 2470-2492, 2008.
- [24] F. Oggier and B. Hassibi, "The secrecy capacity of the MIMO wiretap channel", *Proc. IEEE International Symposium on Information Theory*, Toronto, Canada, vol. 57, no. 8, pp. 4961-4972, July 2008.
- [25] G. Bagherikaram, A. S. Motahari, A. K. Khandani, "The secrecy capacity region of the degraded vector Gaussian broadcast channel", *IEEE International Symposium on Information Theory*, Seoul, pp. 2772-2776 Jul. 2009.
- [26] V. K. Wei, "Generalized hamming weights for linear codes", *IEEE Transactions on Information Theory*, vol. 37, no. 5, pp. 1412-1418, Sep. 1991.
- [27] M. Hayashi, R. Matsumoto, "Construction of wiretap codes from ordinary channel codes", *Proc. 2010 IEEE International Symposium on Information Theory*, Austin, TX, pp. 913-917, Jun. 2010.
- [28] A. Thangaraj, S. Dihidar, A. R. Calderbank, S. McLaughlin and J-M. Merolla, "Application of LDPC codes to the wiretap channel", *IEEE Transactions on Information Theory*, vol. 53, no. 8, pp. 2933-2945, Aug. 2007.
- [29] D. Kline, Jeongseok Ha, S. W. McLaughlin, J. Barros, K. Byung-Jae, "LDPC codes for the Gaussian wiretap channel", *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 6, pp. 532-540, Sep. 2011.
- [30] V. Rathi, M. Andersson, R. Thobaben, J. Kliewer, M. Skoglund, "Two edge type LDPC codes for the wiretap channel", *2009 Conference Record of the Forty-Third Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, CA, pp. 834-838, Nov. 2009.
- [31] H. MahdaviFar and A. Vardy, "Achieving the Secrecy Capacity of Wiretap Channels Using Polar Codes", *IEEE Transactions on Information Theory*, vol. 57, no. 10, pp. 6428-6443, Oct. 2011.
- [32] E. Arikan, "Channel polarization: a method for constructing capacity achieving codes for symmetric binary-input memoryless channels", *IEEE Transactions on Information Theory*, vol. 55, pp. 3051-3073, Jul. 2009.
- [33] E. Hof, S. Shamai, "Secrecy-achieving polar-coding", *IEEE Information Theory Workshop*, Dublin, pp. 1-5, Aug. 2010 .

References

- [34] O. O. Koyluoglu, H. E. Gamal, “Polar coding for secure transmission and key agreement”, *IEEE International Symposium on Personal Indoor and Mobile Radio Communications*, Istanbul, pp. 2698-2703, Sept. 2010.
- [35] J. C. Belfiore and F. Oggier, “Secrecy gain: a wiretap lattice code design”, *IEEE Information Theory and its Applications*, Taichung, pp. 174-178, Oct. 2010.
- [36] J. C. Belfiore, F. Oggier, “Lattice code design for the Rayleigh fading wiretap Channel”, *2011 IEEE International Conference on Communications Workshops (ICC)*, Kyoto, pp. 1-5, June 2011.
- [37] M. J. Mihaljević and H. Imai, “An approach for stream ciphers design based on joint computing over random and secret data”, *Computing*, vol. 85, no. 1, pp. 153-168, Jun. 2009.
- [38] F. Oggier and M. J. Mihaljević, “An information-theoretic security evaluation of a class of randomized encryption schemes”, *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 2, pp. 158-168, Jun. 2014.
- [39] M. J. Mihaljević and H. Imai, “A security evaluation of certain stream ciphers which involve randomness and coding”, *2010 International Symposium on Information Theory and its Applications (ISITA)*, Taichung, pp. 789-794, Oct. 2010.
- [40] M. J. Mihaljević, “An approach for light-weight encryption employing dedicated coding”, *IEEE Global Communications Conference*, Anaheim, CA, pp. 874-880, Dec. 2012.
- [41] Y. S. Khiabani, S. Wei, J. Yuan, and J. Wang, “Enhancement of secrecy of block ciphered systems by deliberate noise”, *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 5, pp. 1604-1613, Oct. 2012.
- [42] Y. S. Khiabani, S. Wei, J. Yuan, and J. Wang, “Linear cryptanalysis against block ciphered system under noisy ciphertexts”, *IEEE Global Communications Conference*, Anaheim, CA, pp. 856-861, Dec. 2012.
- [43] R. Yin, S. Wei, J. Yuan, X. Shan, and X. Wang, “Tradeoff between reliability and security in block ciphering systems with physical channel errors”, *IEEE Military Communications Conference*, vol. 8, no. 4, pp. 2156-2161, Nov. 2010.
- [44] S. Wei, J. Wang, R. Yin, and J. Yuan, “Trade-off between security and performance in block ciphered systems with erroneous cipher texts”, *IEEE Transactions on Information Forensics Security*, vol. 8, no. 4, pp. 636-645, Apr. 2013.

References

- [45] M. Grassl, “Bounds on the minimum distance of linear codes and quantum codes”, online, Available: <http://www.codetables.de>, 2007.
- [46] E. L. Blokh and V. V. Zyablov, “Coding of generalized concatenated codes”, *Problems in Information Transmission*, vol. 10, no. 3, pp. 218-222, 1974.
- [47] Y. Edel, “Inverting construction Y1”, *IEEE Transactions on Information Theory*, , vol. 44, no. 5, Sep, 1998
- [48] W. O. Alltop, “A method for extending binary linear codes”, *IEEE Transactions on Information Theory*, vol. 30, no. 6, pp. 871-872, Nov. 1984.
- [49] N. J. A. Sloane, S. M. Reddy, C.-L. Chen, “New binary codes”, *IEEE Transactions on Information Theory*, vol. 18, no. 4, pp.503-510, 1972.
- [50] P. W. Shor, “Algorithm for quantum computation: discrete log and factoring”, *Proceedings of the 35th IEEE Symposium on the Foundations of Computer Science*, Santa Fe, NM, pp. 124-134, 1994.
- [51] M. Bloch and J. Barros, “Physical-Layer Security: From Information Theory to Security Engineering”, *Cambridge University Press*, Nov. 2011.
- [52] A. Subramanian, A. T. Suresh, S. Raj, A. Thangaraj, M. Bloch, S. McLaughlin, “Strong and weak secrecy in wiretap channels”, *2010 6th International Symposium on Turbo Codes and Iterative Information Processing*, Brest, France, pp. 30-34, Sep. 2010.
- [53] U. M. Maurer and S. Wolf, “Information-theoretic key agreement: From weak to strong secrecy for free”, *Advances in Cryptology - Eurocrypt 2000, Lecture Notes in Computer Science. B. Preneel*, pp.356-373, 2000.
- [54] I. Csiszàr, “Almost independence and secrecy capacity”, *Problems of Information Transmission*, vol. 32, no. 1, pp. 40-47, Jan.-Mar. 1996.
- [55] S. K. Leung-Yan-Cheong and M. E. Hellman, “The Gaussian wire-tap channel”, *IEEE Transactions on Information Theory*, vol. 24, no.4, pp.451-456, Jul. 1978.
- [56] A. Subramanian, A Thangaraj, M. Bloch, S. W. McLaughlin, “Strong secrecy for erasure wiretap channel using large-girth LDPC codes”, *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 3, pp. 585-594, Sep. 2011.
- [57] C. E. Shannon, “A mathematical theory of communication”, *The Bell Syst. Tech. J.*, pp. 379-423; pp. 623-656, July 1948.

References

- [58] R. Liu, Y. Liang, H. V. Poor, and P. Spasojevic, "Secure nested codes for type II wiretap channels", *Proc. 2007 IEEE Information Theory Workshop*, Tahoe City, CA, pp. 337-342, Sep. 2007.
- [59] R. Liu, H. V. Poor, P. Spasojevic, and Y. Liang, "Nested codes for secure transmission", *IEEE 19th International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC 2008*, Cannes, pp. 1-5, Sep. 2008.
- [60] Y. Chen and A. J. H. Vinck, "On the binary symmetric wiretap channel", *Proc. International Zurich Seminar on Communications*, pp. 17-20, Mar. 2010.
- [61] S. Lin and D. J. Costello, "Error Control Coding (second edition)", *Pearson Education*, 2004, pp. 100-231.
- [62] S. Goel and R. Negi, "Guaranteeing secrecy using artificial noise", *IEEE Transactions on Wireless Communications*, vol. 7, no. 6, pp. 2180-2189, Jun. 2008.
- [63] J. Vilela, M. Bloch, J. Barros, and S. McLaughlin, "Wireless secrecy regions with friendly jamming", *IEEE Transactions on Information Forensics Security*, vol. 6, no. 2, pp. 256-266, Jun. 2011.
- [64] John G. Proakis, "Digital Communications", *McGraw-Hill Science/Engineering/Math* Third Edition, 1995.
- [65] M. C. Gursoy, "Secure communication in the low-SNR regime: a characterization of the energy-secrecy tradeoff", *IEEE International Symposium on Information Theory*, Seoul, Korea, pp.2291-2295, Jun. 2009.
- [66] S. Verdú, "Spectral efficiency in the wideband regime", *IEEE Transactions on Information Theory*, vol. 48, no.6, pp.1319-1343, Jun. 2002.
- [67] T. M. Cover and J. A. Thomas, "Elements of Information Theory", *Wiley-Interscience*, Second Edition, 2006.
- [68] C. H. Bennett, G. Brassard, C. Crepeau and U. M. Maurer, "Generalized privacy amplification", *IEEE Transaction on Information Theory*, vol. 41, no. 6, pp. 1915-1923, Nov. 1995.
- [69] G. Cohen and G. Zemor, "Syndrome-coding for the wiretap channel revisited", *Proceedings of 2006 IEEE Information Theory Workshop*, Chengdu, China, pp. 33-36, Oct. 2006.

References

- [70] Y. Cassuto and Z. Bandic “Low-complexity wire-tap codes with security and error-correction guarantees”, *IEEE Information Theory Workshop*, Dublin, pp. 1-5, Aug. 2010.
- [71] H. Chabanne, Cohen and A. Patey, “Secure network coding and non-malleable codes: Protection against linear tampering”, *IEEE International Symposium on Information Theory Proceedings*, pp. 2546-2550, 2012.
- [72] Y. Luo, A. J. H. Vinck and K. Chen, “Some new characters on the wire-tap channel of type II”, *IEEE Transactions on Information Theory* 2005, vol. 51, no. 3, pp. 1222-1229, Mar. 2005.
- [73] W. T. Song, J. Choi and J. Ha , “Perfect secrecy over binary erasure wiretap channel of type II”, *IEEE Transactions on Information Forensics and Security*, vol. 7, pp. 1414-1418, Aug. 2012.
- [74] A. Sanaei and M. Ardakani, “LDPC code design considerations for non-uniform channels”, *IEEE Transactions on Communications*, vol. 58, pp. 101-109, Jan. 2010.
- [75] J. Li, J. Yuan, R. Malaney, M. H. Azmi, M. Xiao, “Network coded LDPC code design for a multi-source relaying system”, *IEEE Transactions on Wireless Communications*, vol. 10, pp. 1538-1551, May 2011.
- [76] K. Zhang, “Best Equivocation Rate codes”, 2013, online, Available: http://www.it.pt/auto_temp_web_page_preview.asp?id=1219.
- [77] F. J. MacWilliams and N.J.A. Sloane, “The theory of error-correcting codes”, *Bell laboratories*, 1981.
- [78] I. G. Bouyukliev and E. Jacobsson, “Results on binary linear codes with minimum distance 8 and 10”, *IEEE Transactions on Information Theory*, vol. 57, no. 9, Sep. 2011.
- [79] K. Dontas and K. De Jong, “Discovery of maximal distance codes using genetic algorithms”, *2nd International IEEE Conference on Tools for Artificial Intelligence*, Herndon, VA, pp. 805-811, 1990.
- [80] C. Cotta, “Scatter search and memetic approaches to the error correcting code problem”, *Evolutionary Computation in Combinatorial Optimization*, J. Gottlieb, G. Raidl (eds.), *Lecture Notes in Computer Science*, Springer-Verlag Berlin, vol. 3004, pp. 51-60, 2004.

References

- [81] J. Miller and J. K. Wolf, "High code rate error correction code design for partial response systems", *IEEE Transactions on Magnetics*, vol. 37, no. 2, pp. 704-707, Mar. 2001.
- [82] E. Alba and S. Khuri, "Applying evolutionary algorithms to combinatorial optimization problems", *International Conference on Computational Science*, vol. 2074, Part II, Springer-Verlag, Berlin, Heidelberg, pp. 689-700, 2001.
- [83] E. Alba and S. Khuri, "Sequential and distributed evolutionary algorithms for combinatorial optimization problems", *Recent Advances in Intelligent Paradigms and Application*, Physica-Verlag, Berlin, Heidelberg, pp. 211-233, 2003.
- [84] W. W. Peterson, "Error-correcting Codes(first edition)", *MIT Press*, 1961.
- [85] R. G. Gallager, "Information theory and reliable communication", *John Wiley and Sons*, 1968.
- [86] R. G. Gallager, "The random coding bound is tight for the average code", *IEEE Transactions on Information Theory*, vol. IT-19, pp. 244-246, Mar. 1973.
- [87] S. J. MacMullan and O. M. Collins, "A comparison of known codes, random codes, and the best codes", *IEEE Transactions on Information Theory*, vol. 44, pp. 3009-3022, Nov. 1998.
- [88] A. Barg and G. D. Forney, "Random codes: minimum distances and error exponents", *IEEE Transactions on Informormation Theory*, vol. 48, no.9, pp. 2568-2573, Sep. 2002.
- [89] Y. Domb, R. Zamir and M. Feder, "The random coding bound is tight for the average linear code", *2012 IEEE 27th Convention of Electrical and Electronics Engineers in Israel*, Eilat, Israel, pp. 1-5, Nov. 2012.
- [90] Christian Walck, "Hand-book on statistical distributions for experimentalists", *Internal report in University of Stockholm*, Sep. 2007.
- [91] A. Justel, D. Peña and R. Zamar, "A multivariate Kolmogorov-Smirnov test of goodness of fit", *Statistics and Probability Letters*, vol. 35, pp. 251-259, 1997.
- [92] "Hash function", http://en.wikipedia.org/wiki/Hash_function.
- [93] S. Kullback, "Information theory and statistics", *John Wiley and Sons, NY*, 1959.
- [94] R. J. McEliece, "A public-key cryptosystem based on algebraic coding theory", *DSN Progress Report*, pp. 42-44, 1978.

References

- [95] E. R. Berlekamp, R. J. McEliece, and H. C. A. Van Tilborg, "On the inherent intractability of certain coding problems", *IEEE Transactions on Information Theory*, vol. 24, no. 5, pp. 384-386, 1978.
- [96] R. L. Rivest, A. Shamir, and L. M. Adleman, "A method for obtaining digital signatures and public-key cryptosystems", *Communications of ACM*, vol. 21, no. 2, pp. 120-126, 1978.
- [97] H. M. Sun, "Enhancing the security of the McEliece public-key cryptosystem", *Journal of Information Science and Engineering*, vol. 16, no. 6, pp. 799-812, 2000.
- [98] M. Barbier, P. S. L. M. Barreto, "Key Reduction of McEliece's Cryptosystem Using List Decoding", *2011 IEEE International Symposium on Information Theory Proceedings*, St. Petersburg, pp. 2681-2685, July 2011.
- [99] D. Bernstein, "List decoding for binary Goppa codes", 2008, <http://cr.yp.to/codes/goppalist-20081107.pdf>.
- [100] P. Loidreau, N. Sendrier, "Weak keys in the McEliece public-key cryptosystem", *IEEE Transaction on Information Theory*, vol. 47, no. 3, pp. 1207-1211, Aug. 2002.
- [101] J. A. Ryan, "Excluding some weak keys in the McEliece cryptosystem", *AFRICON 2007*, Windhoek, pp. 1-5, Sep. 2007.
- [102] P. Lee and E. Brickell, "An observation on the security of McEliece's public-key cryptosystem", *Advances in Cryptology-EUROCRYPTO'88*, New York: Springer-Verlag, pp. 275-280, 1988.
- [103] J. Stern, "A method for finding codewords of small weight", *Coding Theory and Applications, Lecture Notes in Computer Science*, vol. 388, pp. 106-113, 1989.
- [104] T. A. Berson, "Failure of the McEliece public-key cryptosystem under message-resend and related-message attack", *Advanced in Cryptology-CRYPTO'97*, vol. 1294, pp. 213-220, 1997.
- [105] D. J. Bernstein, T. Lange, C. Peters, "Attacking and defending the McEliece cryptosystem". *Proc. 2nd International Workshop on Post-Quantum Cryptography*, vol. 5299, pp. 31-46, Aug. 2008.
- [106] M. K. Shooshtari, M. Ahamadian, and A. Payandeh, "Improving the security of McEliece-like public key cryptosystem based on LDPC codes", *International Conference on Advanced Communication Technology*, vol. 2, pp. 1050-1053, 2009.

References

- [107] N. Döttling, R. Dowsley, J. Müller-Quade, A. C. A. Nascimento, “A CCA2 secure variant of the McEliece cryptosystem”, *IEEE Transactions on Information Theory*, vol. 58, no. 10, pp. 6672-6680, Oct. 2012.
- [108] M. Baldi, M. Bianchi, F. Chiaraluce, “Security and complexity of the McEliece cryptosystem based on quasi-cyclic low-density parity-check codes”, *IET on Information Security*, vol. 7, no. 3, pp. 212-220, Sep. 2013.
- [109] B. Biswas and N. Sendrier, “McEliece cryptosystem in real life: theory and practice”, *J. Buchmann and J. Ding, editors, PQCrypto*, Springer-Verlag, pp.47-62, 2008.
- [110] C. M. Adams and H. Meijer, “Security-related comments regarding McEliece’s public-key cryptosystem”, *IEEE Transaction on Information Theory*, vol. 35, no.2, pp.454-455, 1989.
- [111] M. R. Bloch and J. N. Laneman, “Secrecy from Resolvability”, <http://www3.nd.edu/~jnl/pubs/it2011.pdf>.

Appendix A

Proof of the MAP Decoder for the Syndrome Coding Scheme

We give the proof of that the syndrome decoder is the MAP decoder for the syndrome coding scheme. Alice wishes to convey a sequence of independent and uniformly distributed m -bit binary messages to the legitimate receiver, $M(1), \dots, M(b)$, in which b is the block length of the messages. The channel output is $Y_E(1), \dots, Y_E(b)$, in which $Y_E(i) = V_T(i) + e(i)$ and $e(i)$ is the error vector introduced by the binary symmetric channel. $Y_E(i)$ maps into a received syndrome $S_{Eve}(i) = Y_E(i) \times \mathbf{H}^T = M(i) + S_e(i)$.

The MAP decoder selects the sequence of the estimated messages as follows:

$$\hat{M}(1), \dots, \hat{M}(b) = \arg \max_{M(1), \dots, M(b)} P_{M(1), \dots, M(b) | Y_E(1), \dots, Y_E(b)}(M(1), \dots, M(b) | Y_E(1), \dots, Y_E(b)) \quad (\text{A.1})$$

$$= \arg \max_{M(1), \dots, M(b)} \frac{P_{M(1), \dots, M(b), Y_E(1), \dots, Y_E(b)}(M(1), \dots, M(b), Y_E(1), \dots, Y_E(b))}{P_{Y_E(1), \dots, Y_E(b)}(Y_E(1), \dots, Y_E(b))} \quad (\text{A.2})$$

$$= \arg \max_{M(1), \dots, M(b)} \prod_{i=1}^b P_{M(i) | M(i-1), \dots, M(1), Y_E(1), \dots, Y_E(b)}(M(i) | M(i-1), \dots, M(1), Y_E(1), \dots, Y_E(b)) \quad (\text{A.3})$$

which is based on the Bayes' theorem.

Since the syndrome coding scheme is memoryless, $M(i)$ at time i only depend on the

current $Y_E(i)$, (A.3) can be expressed as follows:

$$\begin{aligned} & \arg \max_{M(1), \dots, M(b)} \prod_{i=1}^b P_{M(i)|Y_E(i)}(M(i)|Y_E(i)) \\ &= \arg \max_{M(1), \dots, M(b)} \prod_{i=1}^b \frac{P_{M(i), Y_E(i)}(M(i), Y_E(i))}{P_{Y_E(i)}(Y_E(i))} \end{aligned} \quad (\text{A.4})$$

$$= \arg \max_{M(1), \dots, M(b)} \prod_{i=1}^b \frac{P_{Y_E(i)|M(i)}(Y_E(i)|M(i)) \cdot P_{M(i)}(M(i))}{P_{Y_E(i)}(Y_E(i))} \quad (\text{A.5})$$

$$= \arg \max_{M(1), \dots, M(b)} \prod_{i=1}^b P_{Y_E(i)|M(i)}(Y_E(i)|M(i)) \quad (\text{A.6})$$

(A.4) and (A.5) are based on the Bayes' theorem and (A.5) is due to the fact that the messages $M(i)$ and the channel output vectors $Y_E(i)$ are both uniform distributed. We now analyse the individual terms in (A.6).

$$\begin{aligned} & P_{Y_E(i)|M(i)}(Y_E(i)|M(i)) \\ &= \sum_{C_T(i)} P_{C_T(i)|M(i)}(C_T(i)|M(i)) \cdot P_{Y_E(i)|C_T(i), M(i)}(Y_E(i)|C_T(i), M(i)) \end{aligned} \quad (\text{A.7})$$

$$= \sum_{C_T(i)} \frac{1}{2^k} \cdot P_{Y_E(i)|C_T(i), M(i)}(Y_E(i)|C_T(i), M(i)) \quad (\text{A.8})$$

$$= \frac{1}{2^k} \cdot \sum_{C_T(i)} P_{e(i)}(Y_E(i) - C_T(i) - E_T(i)) \quad (\text{A.9})$$

$$= \frac{1}{2^k} \cdot P_{S_{e(i)}}(S_{Eve}(i) - M(i)) \quad (\text{A.10})$$

$$= P_{S_{Eve}(i)|M(i)}(S_{Eve}(i)|M(i)) \quad (\text{A.11})$$

(A.8) is due to the fact that $C_T(i)$ is independent of $M(i)$ and there are 2^k possible $C_T(i)$ with equal probability of $\frac{1}{2^k}$. (A.10) is based on:

$$P_{S_{Eve}(i)}(S_{Eve}(i)) = \sum_{Y_E(i): S_{Eve}(i)=Y_E(i) \times H^T} P_{Y_E}(Y_E(i)) \quad (\text{A.12})$$

$$= \sum_{C_T(i)} P_{e'(i)+C_T(i)}(e'(i) + C_T(i)). \quad (\text{A.13})$$

in which $e'(i)$ is the coset leader in the error-syndrom table and is corresponding to $S_{Eve}(i)$.

Finally, we have:

$$\hat{M}(1), \dots, \hat{M}(b) = \arg \max_{M(1), \dots, M(b)} \prod_{i=1}^b P_{S_{Eve}(i)|M(i)}(S_{Eve}(i)|M(i)) \quad (\text{A.14})$$

which means $\hat{M}(i) = S_{Eve}(i)$ is MAP decoder, if and only if

$$p(S_0) > p(S_j), 1 \leq j \leq 2^m - 1 \quad (\text{A.15})$$

in which, $p(S_0)$ is the probability of the zero syndrome, $S_0 = [0]$, in which $[0]$ is a zero vector, and $p(S_j)$ is the probability of the j -th non-zero syndrome, S_j . In this case, we only have to consider the probability of each syndrome.

Let us now consider the probability distribution of the syndromes, $p(S_e(i))$, for the traditional syndrome coding scheme first. There are 2^n possible ways in which error patterns, $e(i)$, occur in each transmitted n -bit vector. In BSC, these error patterns occur with probability:

$$p(e(i)) = \alpha^{w(i)} \cdot (1 - \alpha)^{n-w(i)} \quad (\text{A.16})$$

where $w(i)$ is the weight of $e(i)$. Each error pattern results in one of the 2^m syndromes being produced.

$$S_e(i) = e(i) \times \mathbf{H}^T \quad (\text{A.17})$$

As the code is linear there are 2^k error patterns that produce the same syndrome and the probability of each syndrome due to all possible error patterns is given by

$$p(S_j) = \sum_{e(i)=0}^{2^n-1} p(e(i)) \delta(S_e(i) - S_j) \quad (\text{A.18})$$

where $\delta()$ is the Kronecker Dirac function and $0 \leq j \leq 2^m - 1$.

Based on the analysis above, if the probability of the zero syndrome, $p(S_0)$, is higher than $p(S_j)$, $\forall S_j \neq [0]$, the maximum a posteriori probability (MAP) decoder selects $\hat{M}(i) = S_{Eve}(i)$. Since any error pattern that is a codeword has a syndrome equal to zero, the probability of the zero syndrome can be calculated as follows:

$$p(S_0) = W_C(1 - \alpha, \alpha) = \sum_{i=0}^n A_i (1 - \alpha)^{n-i} \alpha^i \quad (\text{A.19})$$

where $W_C(x, y)$ is the weight enumerator of the code C and A_i denotes the number of codewords of weight i in C . (A.19) shows that the probability of the zero syndrome can be determined from the weight enumerator of the code C .

MacWilliam's theorem [77] gives a way to determine the weight enumerator of the code C from the weight enumerator of its dual code $C^\perp$, which is as follows: if C is an binary linear code with a dual code $C^\perp$ then:

$$W_C(x, y) = \frac{1}{|C^\perp|} W_{C^\perp}(x + y, x - y) \quad (\text{A.20})$$

$$= \frac{1}{|C^\perp|} \sum_{i=0}^n A'_i (x + y)^{n-i} (x - y)^i \quad (\text{A.21})$$

References

where A'_i denotes the number of codewords of weight i in $C^\perp$ and $|C^\perp|$ is the number of the codewords in $C^\perp$.

There are some families of codes where the weight enumerator of the dual code is known, such as: Hamming codes, extended Hamming codes, single parity check codes, double error correcting BCH codes, orthogonal codes, etc. Now we give some examples to show the probability of the zero syndrome for each code.

The (n,m) simplex code is the dual code of the $(n,k=n-m)$ Hamming code, with the weight enumerator:

$$W_{C_1}(x, y) = x^n + n \cdot x^{(n-1)/2} \cdot y^{(n+1)/2} \quad (\text{A.22})$$

so the weight enumerator of the (n,k) Hamming code is:

$$W_{C_1^\perp}(x, y) = \frac{1}{n+1} \cdot W_{C_1}(x+y, x-y) \quad (\text{A.23})$$

$$= \frac{1}{n+1} \cdot ((x+y)^n + n \cdot (x+y)^{(n-1)/2} \cdot (x-y)^{(n+1)/2}) \quad (\text{A.24})$$

The probability of the zero syndrome is:

$$p(S_0) = W_{C_1^\perp}(1-\alpha, \alpha) = W_{C_1}(1, 1-2\alpha) = \frac{1}{n+1} \cdot (1 + n \cdot (1-2\alpha)^{(n+1)/2}) \quad (\text{A.25})$$

and the probability of non-zero syndrome is [77]:

$$p(S_j \neq 0) = (1 - p(S_0))/n \quad (\text{A.26})$$

We can have:

$$p(S_0) - p(S_j \neq 0) = (1 - 2\alpha)^{(n+1)/2} = (1 - 2\alpha)^{2^{m-1}} \geq 0 \quad (\text{A.27})$$

(A.27) shows that the probability of the zero syndrome is the greatest probability of all $p(S_j)$ for the Hamming code for all values of α , which means the decoder $\hat{M}(i) = S_{Eve}(i)$ is a MAP decoder for all α .

Next we consider the single parity check codes. It is known that the dual code of the single parity check code is the repetition code, which has a weight distribution as follows:

$$W_{C_2}(x, y) = x^n + y^n \quad (\text{A.28})$$

so the weight distribution of the single parity check code is:

$$\begin{aligned} W_{C_2^\perp}(x, y) &= \frac{1}{2} \cdot W_{C_2}(x+y, x-y) \\ &= ((x+y)^n + (x-y)^n)/2 \end{aligned} \quad (\text{A.29})$$

For the single parity check code, there are only two possible syndromes: one zero syndrome and one non-zero syndrome. From (A.29), we have the probability of the zero syndrome is:

$$p(S_0) = W_{C2^\perp}(1 - \alpha, \alpha) = W_{C2}(1, 1 - 2\alpha) = \frac{1}{2} \cdot (1 + (1 - 2\alpha)^n) \quad (\text{A.30})$$

and the probability of the non-zero syndrome is:

$$p(S_1 \neq 0) = 1 - p(S_0) \quad (\text{A.31})$$

so

$$p(S_0) - p(S_1 \neq 0) = 2p(S_0) - 1 = (1 - 2\alpha)^n \quad (\text{A.32})$$

which means that when $0 < \alpha < 1/2$, the zero syndrome has the highest probability, i.e. the syndrome decoder is the MAP decoder for the single parity check codes.

The examples above all satisfy the MAP decoder condition: (A.15), which means the syndrome decoder is MAP decoder for these codes.

Appendix B

Best Equivocation Codes Table

Table B.1: Best Equivocation Code Table (BEC)

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
4	7	0.476876	4	3	1 2 4 8 7 11 13
4	8	0.532338	4	2	1 2 4 8 7 11 13 14
4	9	0.572531	3	2	1 2 4 8 6 7 11 13 14
4	10	0.611427 (0.605572)	3	2	1 2 4 8 3 6 7 11 13 14
4	11	0.649014 (0.646301)	3	2	1 2 4 8 3 6 7 11 13 14 15
4	12	0.684835 (0.682751)	3	2	1 2 4 8 3 6 7 10 11 13 14 15
4	13	0.715174	3	2	1 2 4 8 3 5 6 7 10 11 13 14 15
4	14	0.744343	3	2	1 2 4 8 3 5 6 7 9 10 11 13 14 15
4	15	0.770677	3	1	1 2 4 8 3 5 6 7 9 10 11 12 13 14 15
4	16	0.788482 (0.453017)	2	1	1 2 4 8 3 4 5 6 7 9 10 11 12 13 14 15
4	17	0.805304 (0.455276)	2	1	1 2 4 8 3 4 5 6 7 9 10 11 12 13 14 14 15
4	18	0.821182 (0.457098)	2	1	1 2 4 8 3 5 5 6 6 7 9 10 10 11 12 13 14 15
4	19	0.836156 (0.458570)	2	1	1 2 4 8 3 5 5 6 6 7 8 9 10 10 11 12 13 14 15
4	20	0.850259 (0.459759)	2	1	1 2 4 8 2 3 5 5 6 6 7 9 10 10 11 11 12 13 14 15
4	21	0.863396 (0.460720)	2	1	1 2 4 8 1 2 3 5 5 6 6 7 9 10 11 12 12 13 14 15 15
4	22	0.875721 (0.461497)	2	1	1 2 4 8 1 2 3 5 5 6 6 7 8 9 10 11 12 12 13 14 15 15
4	23	0.887202 (0.462126)	2	1	1 2 4 8 1 2 3 5 5 6 6 7 8 9 10 11 11 12 12 13 14 15 15
4	24	0.897025 (0.462634)	2	1	1 2 4 8 1 2 3 3 5 5 6 6 7 8 9 10 11 11 12 12 13 14 15 15
4	25	0.906251 (0.463046)	2	1	1 2 4 8 1 2 3 3 4 5 5 6 6 7 8 9 10 11 11 12 12 13 14 15 15
4	26	0.914909 (0.463379)	2	1	1 2 4 8 1 2 3 3 4 5 5 6 6 7 8 9 9 10 11 11 12 12 13 14 15 15
4	27	0.923001 (0.463649)	2	1	1 2 4 8 1 2 3 3 4 5 5 6 6 7 8 9 9 10 11 11 12 12 13 14 14 15 15
4	28	0.930227 (0.463867)	2	1	1 2 4 8 1 2 3 3 4 5 5 6 6 7 7 8 9 9 10 11 11 12 12 13 14 14 15 15
4	29	0.936994 (0.464044)	2	1	1 2 4 8 1 2 3 3 4 5 5 6 6 7 7 8 9 9 10 10 11 11 12 12 13 14 14 15 15
4	30	0.943175 (0.464188)	2	1	1 2 4 8 1 2 3 3 4 5 5 6 6 7 7 8 9 9 10 10 11 11 12 12 13 13 14 14 15 15
4	31	0.947950 (0.464304)	2	1	1 2 4 8 1 1 2 3 3 4 5 5 6 6 7 7 8 9 9 10 10 11 11 12 12 13 13 14 14 15 15
4	32	0.952399 (0.464397)	2	1	1 2 4 8 1 1 2 3 3 4 5 5 6 6 6 7 7 8 9 9 10 10 11 11 12 12 13 13 14 14 15 15
4	33	0.956541 (0.464474)	2	1	1 2 4 8 1 2 3 3 4 5 5 6 6 7 7 7 8 8 9 9 10 10 11 11 12 12 13 13 14 14 15 15
4	34	0.960396 (0.464535)	2	1	1 2 4 8 1 2 2 3 3 4 5 5 6 6 7 7 7 8 8 9 9 10 10 11 11 12 12 13 13 14 14 15 15
4	35	0.963979 (0.464585)	2	1	1 2 4 8 1 2 2 3 3 4 4 5 5 6 6 7 7 7 8 8 9 9 10 10 11 11 12 12 13 13 14 14 15 15
4	36	0.967288 (0.464625)	2	1	1 2 4 8 1 1 2 3 3 3 4 4 5 5 6 6 6 7 7 8 8 9 9 10 10 11 11 12 12 13 13 13 14 14 15 15
4	37	0.970356 (0.464658)	2	1	1 2 4 8 1 2 2 3 3 3 4 4 5 5 5 6 6 7 7 8 9 9 10 10 10 11 11 11 12 12 12 13 13 13 14 14 15 15
4	38	0.973188 (0.464685)	2	1	1 2 4 8 1 2 2 3 3 3 4 4 5 5 5 6 6 7 7 8 9 9 10 10 10 11 11 11 12 12 12 13 13 13 14 14 15 15
4	39	0.975661 (0.464706)	2	1	1 2 4 8 1 1 2 2 3 3 3 4 4 4 5 5 5 6 6 7 7 8 9 9 10 10 10 11 11 11 12 12 12 13 13 13 14 14 15 15
4	40	0.977953 (0.464724)	2	1	1 2 4 8 1 1 2 2 3 3 3 4 4 4 5 5 5 6 6 7 7 8 9 9 10 10 10 11 11 11 12 12 12 13 13 13 14 14 15 15
5	6	0.343073	6	3	1 2 4 8 16 31
5	7	0.394934	4	3	1 2 4 8 16 15 23
5	8	0.444665 (0.438780)	4	3	1 2 4 8 16 11 23 25
5	9	0.491836 (0.483150)	4	3	1 2 4 8 16 11 13 23 25
5	10	0.536631 (0.530165)	4	2	1 2 4 8 16 11 13 23 25 30
5	11	0.572938 (0.572274)	4	3	1 2 4 8 16 7 11 21 26 28 31
5	12	0.610896 (0.610359)	4	3	1 2 4 8 16 7 11 19 21 26 28 31
5	13	0.646292	4	3	1 2 4 8 16 7 11 19 21 22 26 28 31
5	14	0.679074	4	3	1 2 4 8 16 7 11 19 21 22 25 26 28 31
5	15	0.709297	4	3	1 2 4 8 16 7 11 13 19 21 22 25 26 28 31
5	16	0.737076	4	2	1 2 4 8 16 7 11 13 14 19 21 22 25 26 28 31
5	17	0.756153	3	2	1 2 4 8 16 7 11 13 14 15 19 21 22 25 26 28 31

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
5	18	0.774453	3	2	1 2 4 8 16 7 11 13 14 15 17 19 21 22 25 26 28 31
5	19	0.791984	3	2	1 2 4 8 16 3 7 11 13 14 15 17 19 21 22 25 26 28 31
5	20	0.808759 (0.808622)	3	2	1 2 4 8 16 3 5 7 11 13 14 15 17 19 21 22 25 26 28 31
5	21	0.824777 (0.824666)	3	2	1 2 4 8 16 3 5 7 11 13 14 15 17 19 21 22 24 25 26 28 31
5	22	0.839755	3	2	1 2 4 8 16 3 5 6 11 12 14 15 17 19 20 22 23 25 26 27 29 30
5	23	0.853968	3	2	1 2 4 8 16 3 5 6 9 11 12 14 15 17 19 20 22 23 25 26 27 29 30
5	24	0.867285	3	2	1 2 4 8 16 3 5 6 9 11 12 14 15 17 19 20 22 23 25 26 27 28 29 30
5	25	0.878170	3	2	1 2 4 8 16 3 5 6 7 9 11 12 14 15 17 19 20 22 23 25 26 27 28 29 30
5	26	0.888523	3	2	1 2 4 8 16 3 5 6 7 9 10 11 12 14 15 17 19 20 22 23 25 26 27 28 29 30
5	27	0.898361	3	2	1 2 4 8 16 3 5 6 7 9 11 12 13 14 15 17 18 19 20 22 23 24 25 26 27 29 30
5	28	0.907646	3	2	1 2 4 8 16 3 5 6 7 9 11 12 13 14 15 17 18 19 20 22 23 24 25 26 27 28 29 30
5	29	0.915808	3	2	1 2 4 8 16 3 5 6 7 9 10 11 12 13 14 15 17 18 19 20 21 22 23 24 25 26 27 28 29
5	30	0.923543	3	2	1 2 4 8 16 3 5 6 7 9 10 11 12 13 14 15 17 18 19 20 21 22 23 24 25 26 27 28 29 30
5	31	0.930596	3	1	1 2 4 8 16 3 5 6 7 9 10 11 12 13 14 15 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
5	32	0.935751 (0.428722)	2	1	1 2 4 8 16 3 5 6 7 9 10 11 12 13 14 15 17 18 19 20 21 22 23 24 25 26 27 27 28 29 30 31
5	33	0.940592 (0.428797)	2	1	1 2 4 8 16 1 3 5 6 7 9 10 11 12 13 14 15 17 18 19 20 21 22 23 24 25 26 27 27 28 29 30 31
5	34	0.945135 (0.428858)	2	1	1 2 4 8 16 1 3 5 6 7 9 10 11 12 13 14 15 17 18 19 20 21 22 23 24 25 26 27 27 28 29 30 31
5	35	0.949395 (0.428908)	2	1	1 2 4 8 16 1 3 5 6 7 9 10 11 12 13 14 15 17 18 19 20 21 22 23 24 24 25 26 27 27 28 28 29 30 31
5	36	0.953386 (0.428947)	2	1	1 2 4 8 16 1 3 5 6 7 9 10 11 12 12 13 14 15 17 18 19 20 21 22 23 24 24 25 26 27 27 28 28 29 30 31
5	37	0.957123 (0.428980)	2	1	1 2 4 8 16 3 4 5 6 7 9 10 10 11 12 13 14 15 15 17 17 18 19 20 21 22 23 24 25 25 26 27 28 29 30 31
5	38	0.960606 (0.429006)	2	1	1 2 4 8 16 1 2 3 5 6 6 7 9 10 11 12 13 14 15 17 18 19 19 20 21 22 22 23 24 25 26 27 28 29 29 30 30 31
5	39	0.963855 (0.429027)	2	1	1 2 4 8 16 1 2 3 5 6 6 7 9 10 11 12 13 14 15 15 17 18 19 19 20 21 22 22 23 24 25 26 27 28 29 29 30 30 31
5	40	0.966882 (0.429044)	2	1	1 2 4 8 16 1 2 3 5 6 6 7 9 9 10 11 12 13 14 15 17 18 19 19 20 21 22 22 23 24 25 25 26 27 28 29 29 30 30 31
5	41	0.969700 (0.429058)	2	1	1 2 4 8 16 1 2 3 5 6 6 7 9 9 10 11 12 12 13 14 15 17 18 19 19 20 21 22 22 23 24 25 25 26 27 28 29 29 30 30 31
5	42	0.972274 (0.429069)	2	1	1 2 4 8 16 2 3 4 5 6 7 7 9 9 10 10 11 12 12 13 14 15 15 17 17 18 19 20 20 21 22 23 24 25 25 26 26 27 28 29 30 31
5	43	0.974685 (0.429079)	2	1	1 2 4 8 16 1 2 3 4 5 6 7 7 9 9 10 10 11 12 12 13 14 15 15 17 17 18 19 20 20 21 22 23 24 25 25 26 26 27 28 29 30 31
5	44	0.976917 (0.429086)	2	1	1 2 4 8 16 1 2 3 4 5 6 7 7 9 9 10 10 11 12 12 13 14 15 15 17 17 18 19 20 20 21 22 23 24 25 25 26 26 27 28 28 29 30 31 31
5	45	0.978978 (0.429092)	2	1	1 2 4 8 16 1 2 3 4 5 6 7 7 9 9 10 10 11 12 12 13 14 15 15 17 17 18 19 20 20 21 22 23 24 25 25 26 26 27 28 28 29 30 31 31
5	46	0.980878 (0.429097)	2	1	1 2 4 8 16 1 2 3 4 5 6 7 7 9 9 10 10 11 12 12 13 14 15 15 17 17 18 18 19 20 20 21 22 23 24 25 25 26 26 27 28 28 29 30 31 31
5	47	0.982627 (0.429101)	2	1	1 2 4 8 16 1 2 3 4 5 6 7 7 9 9 10 10 11 12 12 13 14 15 15 17 17 18 18 19 20 20 21 22 23 24 25 25 26 26 27 28 28 29 30 31 31
5	48	0.984097 (0.429104)	2	1	1 2 4 8 16 1 2 3 4 5 6 7 7 9 9 10 10 11 12 12 13 14 15 15 17 17 18 18 19 20 20 21 22 23 23 24 25 25 26 26 27 28 28 29 30 31 31
5	49	0.985463 (0.429107)	2	1	1 2 4 8 16 1 2 3 3 4 5 5 6 7 7 9 9 10 10 11 12 12 13 14 15 15 17 17 18 18 19 20 20 21 22 23 23 24 25 25 26 26 27 28 28 29 30 31 31
5	50	0.986732 (0.429109)	2	1	1 2 4 8 16 1 2 3 3 4 5 5 6 7 7 9 9 10 10 11 12 12 13 14 15 15 16 17 17 18 18 19 20 20 21 22 23 23 24 25 25 26 26 27 28 28 29 30 31 31
6	7	0.333936	7	3	1 2 4 8 16 32 63
6	8	0.379215 (0.379214)	5	3	1 2 4 8 16 32 27 39
6	9	0.422161 (0.413383)	4	3	1 2 4 8 16 32 27 39 41
6	10	0.464022 (0.454054)	4	3	1 2 4 8 16 32 27 39 41 62
6	11	0.503445 (0.490940)	4	3	1 2 4 8 16 32 13 27 39 41 62
6	12	0.541617 (0.524628)	4	3	1 2 4 8 16 32 13 27 39 41 53 62
6	13	0.573884 (0.556813)	4	3	1 2 4 8 16 32 7 11 29 35 45 54 56
6	14	0.605820 (0.586309)	4	3	1 2 4 8 16 32 11 13 23 25 30 35 37 47
6	15	0.636888 (0.613628)	4	3	1 2 4 8 16 32 11 13 23 30 35 37 49 56 63
6	16	0.665063 (0.638814)	4	3	1 2 4 8 16 32 11 13 23 25 30 35 37 49 56 63
6	17	0.691791 (0.661963)	4	2	1 2 4 8 16 32 7 11 19 29 38 41 44 50 55 56 62
6	18	0.716331 (0.696711)	4	3	1 2 4 8 16 32 11 13 23 25 30 35 37 42 49 54 56 63
6	19	0.739626 (0.725720)	4	3	1 2 4 8 16 32 11 13 23 25 30 35 37 41 42 44 49 56 63
6	20	0.761475 (0.750666)	4	2	1 2 4 8 16 32 11 13 23 25 30 35 37 41 42 44 49 54 56 63
6	21	0.778522 (0.772942)	3	2	1 2 4 8 16 32 3 11 13 23 25 30 35 37 41 42 44 49 54 56 63
6	22	0.794847 (0.792641)	4	3	1 2 4 8 16 32 7 11 21 22 25 26 28 31 35 37 38 42 47 50 52 59
6	23	0.811890 (0.810283)	4	3	1 2 4 8 16 32 13 21 22 28 35 37 38 41 42 44 47 49 52 55 56 59 62
6	24	0.827784 (0.826129)	4	3	1 2 4 8 16 32 7 11 19 21 22 25 26 28 31 35 37 38 42 47 50 52 59 62
6	25	0.842420 (0.841639)	4	3	1 2 4 8 16 32 7 11 19 21 22 25 26 28 31 35 37 38 42 47 50 52 59 62
6	26	0.855866 (0.855572)	4	3	1 2 4 8 16 32 7 11 19 21 22 25 26 28 31 35 37 38 41 42 47 50 52 56 59 62
6	27	0.868283 (0.868197)	4	3	1 2 4 8 16 32 7 11 13 19 21 22 25 26 28 31 35 37 38 41 42 47 50 52 56 59 62
6	28	0.879719 (0.879644)	4	3	1 2 4 8 16 32 7 11 13 14 19 21 22 25 26 28 31 35 37 38 41 42 47 50 52 56 59 62
6	29	0.890231	4	3	1 2 4 8 16 32 7 11 13 14 19 21 22 25 26 28 31 35 37 38 41 42 47 50 52 56 59 61 62
6	30	0.899877	4	3	1 2 4 8 16 32 7 11 13 14 19 21 22 25 26 28 31 35 37 38 41 42 44 47 50 52 56 59 61 62
6	31	0.908721	4	3	1 2 4 8 16 32 7 11 13 14 19 21 22 25 26 28 31 35 37 38 41 42 44 47 49 50 52 56 59 61 62
6	32	0.916820	4	2	1 2 4 8 16 32 7 11 13 14 19 21 22 25 26 28 31 35 37 38 41 42 44 47 49 50 52 55 56 59 61 62
6	33	0.922524	3	2	1 2 4 8 16 32 7 11 13 14 19 21 22 25 26 28 31 35 37 38 41 42 44 46 47 49 50 52 55 56 59 61 62
6	34	0.927939	3	2	1 2 4 8 16 32 3 7 11 13 14 19 21 22 25 26 28 31 35 37 38 41 42 44 46 47 49 50 52 55 56 59 61 62
6	35	0.933073	3	2	1 2 4 8 16 32 3 5 7 11 13 14 19 21 22 25 26 28 31 35 37 38 41 42 44 46 47 49 50 52 55 56 59 61 62
6	36	0.937936 (0.937911)	3	2	1 2 4 8 16 32 3 7 9 11 12 13 14 19 21 22 25 26 28 31 35 37 38 41 42 44 47 48 49 50 52 55 56 59 61 62
6	37	0.942538 (0.942516)	3	2	1 2 4 8 16 32 3 7 9 11 12 13 14 19 21 22 23 25 26 28 31 35 37 38 41 42 44 47 48 49 50 52 55 56 59 61 62
6	38	0.946890 (0.946829)	3	2	1 2 4 8 16 32 3 7 9 11 12 13 14 19 21 22 23 25 26 28 31 33 35 37 38 41 42 44 47 48 49 50 52 55 56 59 61 62
6	39	0.950978 (0.950871)	3	2	1 2 4 8 16 32 3 7 9 11 12 13 14 17 19 21 22 23 25 26 28 31 35 37 38 40 41 42 44 47 48 49 50 52 55 56 59 61 62
6	40	0.954822 (0.954645)	3	2	1 2 4 8 16 32 3 6 7 9 11 12 13 14 17 19 21 22 25 26 28 31 35 37 38 41 42 43 44 47 48 49 50 52 53 55 56 59 61 62
6	41	0.958432 (0.958325)	3	2	1 2 4 8 16 32 7 11 13 14 19 21 22 23 25 26 28 29 31 33 34 35 36 37 38 40 41 42 43 44 46 47 49 50 52 55 56 58 59 61 62
6	42	0.961820 (0.961734)	3	2	1 2 4 8 16 32 7 11 13 14 19 21 22 23 25 26 28 29 31 33 34 35 36 37 38 40 41 42 43 44 46 47 48 49 50 52 55 56 58 59 61 62
6	43	0.964909 (0.964897)	3	2	1 2 4 8 16 32 3 6 7 9 11 12 13 14 19 21 22 23 25 26 28 29 31 33 35 36 37 38 41 42 44 47 48 49 50 52 55 56 58 59 61 62 63
6	44	0.967838 (0.967827)	3	2	1 2 4 8 16 32 3 6 7 9 11 12 13 14 18 19 21 22 23 24 25 26 28 29 31 34 35 37 38 39 40 41 42 44 47 49 50 52 54 55 56 59 61 62

Continued on next page

	n	R_e	d	R_c	Packed integer form of the parity check matrix
6	45	0.970564	3	2	1 2 4 8 16 32 3 6 7 9 11 12 13 14 18 19 21 22 23 24 25 26 28 29 31 34 35 37 38 39 40 41 42 44 45 47 49 50 52 54 55 56 59 61 62
6	46	0.973095	3	2	1 2 4 8 16 32 3 6 7 9 11 12 13 14 18 19 21 22 23 24 25 26 28 29 31 34 35 37 38 39 40 41 42 44 45 47 49 50 51 52 54 55 56 59 61 62
6	47	0.975440	3	2	1 2 4 8 16 32 3 6 7 9 11 12 13 14 18 19 21 22 23 24 25 26 28 29 31 34 35 37 38 39 40 41 42 44 45 47 49 50 51 52 54 55 56 57 59 61 62
6	48	0.977609	3	2	1 2 4 8 16 32 3 6 7 9 11 12 13 14 18 19 21 22 23 24 25 26 28 29 31 34 35 37 38 39 40 41 42 44 45 47 49 50 51 52 54 55 56 57 59 60 61 62
6	49	0.979380	3	2	1 2 4 8 16 32 3 5 6 7 9 11 12 13 14 18 19 21 22 23 24 25 26 28 29 31 34 35 37 38 39 40 41 42 44 45 47 49 50 51 52 54 55 56 57 59 60 61 62
6	50	0.981041	3	2	1 2 4 8 16 32 3 5 6 7 9 11 12 13 14 17 18 19 21 22 23 24 25 26 28 29 31 34 35 37 38 39 40 41 42 44 45 47 49 50 51 52 54 55 56 57 59 60 61 62
6	51	0.982599 (0.982598)	3	2	1 2 4 8 16 32 3 5 6 7 9 11 12 13 14 17 18 19 21 22 23 24 25 26 28 29 31 33 34 35 37 38 39 40 41 42 44 45 47 49 50 51 52 54 55 56 57 59 60 61 62
6	52	0.984057 (0.984052)	3	2	1 2 4 8 16 32 3 5 6 7 9 11 12 13 14 17 18 19 21 22 23 24 25 26 28 29 31 33 34 35 37 38 39 40 41 42 43 44 45 47 49 50 51 52 54 55 56 57 59 60 61 62
6	53	0.985421 (0.985418)	3	2	1 2 4 8 16 32 3 5 6 7 9 11 12 13 14 17 18 19 21 22 23 24 25 26 28 29 30 31 33 34 35 37 38 39 40 41 42 43 44 45 47 49 50 51 52 54 55 56 57 59 60 61 62
6	54	0.986687	3	2	1 2 4 8 16 32 3 5 7 10 11 12 13 14 17 18 19 20 21 22 23 24 25 26 27 28 29 31 33 34 35 36 37 38 39 40 41 42 43 44 46 47 48 49 50 52 54 55 56 57 59 61 62 63
6	55	0.987867	3	2	1 2 4 8 16 32 3 5 7 10 11 12 13 14 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 33 34 35 36 37 38 39 40 41 42 43 44 46 47 48 49 50 52 54 55 56 57 59 61 62 63
6	56	0.988962	3	2	1 2 4 8 16 32 3 5 7 10 11 12 13 14 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 52 54 55 56 57 59 61 62 63
6	57	0.989911	3	2	1 2 4 8 16 32 3 5 6 7 10 11 12 13 14 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 52 54 55 56 57 59 61 62 63
6	58	0.990797	3	2	1 2 4 8 16 32 3 5 6 7 9 10 11 12 13 14 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 52 54 55 56 57 59 61 62 63
6	59	0.991622	3	2	1 2 4 8 16 32 3 5 6 7 9 10 11 12 13 14 15 17 18 19 20 21 22 23 24 25 26 27 28 29 31 33 34 35 36 37 38 39 40 41 42 43 44 46 47 48 49 50 52 53 54 55 56 57 59 60 61 62 63
6	60	0.992390	3	2	1 2 4 8 16 32 3 5 6 7 9 10 11 12 13 14 15 17 18 19 20 21 22 23 24 25 26 27 28 29 31 33 34 35 36 37 38 39 40 41 42 43 44 46 47 48 49 50 52 53 54 55 56 57 58 59 60 61 62 63
7	9	0.367001	6	4	1 2 4 8 16 32 64 115 124
7	10	0.405261 (0.404808)	5	4	1 2 4 8 16 32 64 63 115 124
7	11	0.443011	5	3	1 2 4 8 16 32 64 63 85 115 124
7	12	0.478897 (0.461720)	4	3	1 2 4 8 16 32 64 26 63 85 115 124
7	13	0.512617 (0.491069)	4	3	1 2 4 8 16 32 64 63 79 85 86 115 124
7	14	0.545891 (0.518182)	4	3	1 2 4 8 16 32 64 7 53 62 83 93 107 112
7	15	0.575360 (0.543464)	4	3	1 2 4 8 16 32 64 7 25 53 62 83 93 107 112
7	16	0.604499 (0.566880)	4	3	1 2 4 8 16 32 64 7 27 53 58 83 93 108 112 126
7	17	0.631806 (0.588469)	4	3	1 2 4 8 16 32 64 63 74 79 83 84 93 97 103 118 121
7	18	0.658043 (0.618943)	4	3	1 2 4 8 16 32 64 7 46 53 59 75 86

131

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
7	46	0.961848 (0.960586)	3	2	1 2 4 8 16 32 64 3 7 11 13 20 22 24 26 30 34 38 40 44 46 49 51 53 57 63 70 74 76 83 85 87 89 91 93 99 101 103 105 107 109 112 118 120 122 124
7	47	0.964899 (0.963848)	3	2	1 2 4 8 16 32 64 3 7 11 13 20 22 24 26 30 34 38 40 44 46 49 51 53 57 63 70 72 74 76 83 85 87 89 91 93 99 101 103 105 107 109 112 116 118 122 124
7	48	0.967706 (0.966766)	3	2	1 2 4 8 16 32 64 3 7 11 13 20 22 24 26 30 34 38 40 44 46 51 53 55 57 63 70 72 74 76 83 85 87 89 91 93 99 101 103 105 107 109 112 114 116 118 122 124
7	49	0.970311 (0.969876)	3	2	1 2 4 8 16 32 64 3 5 7 11 13 20 22 24 26 30 34 38 40 44 46 51 53 55 57 63 70 72 74 76 83 85 87 89 91 93 99 101 103 105 107 109 112 114 116 118 122 124
7	50	0.972711 (0.972635)	3	2	1 2 4 8 16 32 64 3 7 11 13 15 18 20 24 26 30 34 38 40 44 46 51 53 57 59 63 70 72 74 76 78 83 85 87 89 91 93 99 101 103 105 107 109 112 114 116 118 122 124
7	51	0.974942 (0.975095)	3	2	1 2 4 8 16 32 64 3 7 11 13 15 18 20 24 26 30 34 38 40 44 46 51 53 57 59 61 63 66 70 74 76 78 83 85 87 89 91 93 99 101 103 105 107 109 112 114 116 118 122 124
7	52	0.976991 (0.977298)	3	2	1 2 4 8 16 32 64 3 7 11 13 15 18 20 24 26 30 34 38 40 44 46 51 53 57 59 61 63 66 70 74 76 78 81 83 85 87 89 91 93 99 101 103 105 107 109 112 114 116 118 122 124
7	53	0.979291	4	3	1 2 4 8 16 32 64 11 13 14 19 21 22 25 26 28 35 37 38 41 42 44 49 50 52 56 61 62 67 69 70 73 74 76 81 82 84 88 93 94 97 98 100 104 109 110 112 115 117 118 121 122 124
7	54	0.981087	4	3	1 2 4 8 16 32 64 11 13 14 19 21 22 25 26 28 35 37 38 41 42 44 49 50 52 56 61 62 67 69 70 73 74 76 81 82 84 88 93 94 97 98 100 104 107 109 110 112 115 117 118 121 122 124
7	55	0.982713	4	3	1 2 4 8 16 32 64 11 13 14 19 21 22 25 26 28 35 37 38 41 42 44 49 50 52 56 61 62 67 69 70 73 74 76 81 82 84 88 91 93 94 97 98 100 104 107 109 110 112 115 117 118 121 122 124
7	56	0.984186	4	3	1 2 4 8 16 32 64 11 13 14 19 21 22 25 26 28 35 37 38 41 42 44 49 50 52 56 59 61 62 67 69 70 73 74 76 81 82 84 88 91 93 94 97 98 100 104 107 109 110 112 115 117 118 121 122 124
7	57	0.985569	4	3	1 2 4 8 16 32 64 7 11 13 14 19 21 22 25 26 28 35 37 38 41 42 44 49 50 52 56 59 61 62 67 69 70 73 74 76 81 82 84 88 91 93 94 97 98 100 104 107 109 110 112 115 117 118 121 122 124
7	58	0.986824	4	3	1 2 4 8 16 32 64 7 11 13 14 19 21 22 25 26 28 35 37 38 41 42 44 49 50 52 56 59 61 62 67 69 70 73 74 76 81 82 84 88 91 93 94 97 98 100 103 104 107 109 110 112 115 117 118 121 122 124
7	59	0.987968	4	3	1 2 4 8 16 32 64 7 11 13 14 19 21 22 25 26 28 35 37 38 41 42 44 49 50 52 56 59 61 62 67 69 70 73 74 76 81 82 84 87 88 91 93 94 97 98 100 103 104 107 109 110 112 115 117 118 121 122 124
8	10	0.357352	6	4	1 2 4 8 16 32 64 128 143 243
8	11	0.391739	6	5	1 2 4 8 16 32 64 128 127 155 248
8	12	0.425845	6	4	1 2 4 8 16 32 64 128 127 155 181 248
8	13	0.458840 (0.457283)	5	4	1 2 4 8 16 32 64 128 86 127 155 181 248
8	14	0.489845 (0.489458)	5	4	1 2 4 8 16 32 64 128 91 109 143 181 221 243
8	15	0.520366 (0.520159)	5	3	1 2 4 8 16 32 64 128 57 91 109 143 181 221 243
8	16	0.549739	5	4	1 2 4 8 16 32 64 128 61 86 101 143 156 166 236 243
8	17	0.578286	5	3	1 2 4 8 16 32 64 128 61 86 91 101 143 156 166 236 243
8	18	0.604479 (0.559173)	4	3	1 2 4 8 16 32 64 128 75 86 108 113 127 155 158 181 199 248
8	19	0.630402 (0.582118)	4	3	1 2 4 8 16 32 64 128 75 86 108 113 127 155 158 162 181 199 248
8	20	0.655143 (0.599850)	4	3	1 2 4 8 16 32 64 128 75 86 108 113 127 155 158 162 173 181 199 248
8	21	0.675828 (0.619509)	4	3	1 2 4 8 16 32 64 128 29 30 47 86 104 115 131 169 186 206 217 226 229
8	22	0.696622 (0.634745)	4	3	1 2 4 8 16 32 64 128 19 47 75 86 99 104 125 131 159 180 185 186 206 229
8	23	0.716261 (0.651307)	4	3	1 2 4 8 16 32 64 128 30 57 101 110 118 143 152 170 181 201 210 213 230 243 254
8	24	0.734482 (0.666080)	4	3	1 2 4 8 16 32 64 128 71 78 83 89 92 101 105 114 127 158 173 179 203 213 230 248
8	25	0.752321 (0.679738)	4	3	1 2 4 8 16 32 64 128 46 61 86 89 98 101 105 126 143 156 163 166 183 185 236 243 244
8	26	0.768677 (0.692050)	4	3	1 2 4 8 16 32 64 128 23 28 78 85 91 99 102 109 121 143 153 171 178 181 199 221 243 248
8	27	0.784433 (0.703163)	4	3	1 2 4 8 16 32 64 128 13 23 28 78 85 91 99 102 109 121 143 153 171 178 181 199 221 243 248
8	28	0.799369 (0.713425)	4	3	1 2 4 8 16 32 64 128 13 23 28 49 78 85 91 99 102 109 121 143 153 171 178 181 199 221 243 248
8	29	0.813109 (0.722811)	4	3	1 2 4 8 16 32 64 128 26 46 61 86 89 98 101 105 126 143 156 166 171 183 185 190 193 236 243 244 250
8	30	0.825986 (0.731388)	4	3	1 2 4 8 16 32 64 128 26 46 61 86 89 98 101 105 113 126 143 156 166 171 183 185 190 193 236 243 244 250
8	31	0.838087 (0.739272)	4	3	1 2 4 8 16 32 64 128 22 42 45 71 75 101 118 120 143 149 154 156 167 169 172 178 187 197 198 216 225 243 254
8	32	0.849540 (0.746508)	4	3	1 2 4 8 16 32 64 128 23 26 49 52 61 62 67 77 86 101 110 133 143 153 156 166 170 183 215 236 243 245 248 255
8	33	0.860464 (0.753139)	4	3	1 2 4 8 16 32 64 128 23 26 49 52 61 62 67 77 86 101 110 133 143 153 156 166 170 183 215 222 236 243 245 248 255
8	34	0.870588 (0.776137)	4	3	1 2 4 8 16 32 64 128 23 26 49 52 61 62 67 77 86 92 101 110 133 143 153 156 166 170 183 215 222 236 243 245 248 255
8	35	0.880151 (0.793798)	4	3	1 2 4 8 16 32 64 128 23 26 49 52 61 62 67 77 86 92 101 110 123 133 143 153 156 166 170 183 208 215 236 243 245 248 255
8	36	0.889144 (0.799446)	4	3	1 2 4 8 16 32 64 128 23 26 49 52 61 62 67 77 86 92 101 110 123 133 143 153 156 166 170 183 208 215 222 236 243 245 248 255
8	37	0.896910 (0.813402)	4	3	1 2 4 8 16 32 64 128 23 26 49 52 61 62 67 77 86 92 101 110 123 133 143 153 156 166 170 183 208 215 219 222 236 243 245 248 255
8	38	0.904458 (0.824846)	4	3	1 2 4 8 16 32 64 128 26 35 46 61 79 83 86 89 98 101 105 126 131 143 156 166 171 183 185 193 198 200 203 205 223 231 236 243 244 250
8	39	0.911681 (0.834407)	4	3	1 2 4 8 16 32 64 128 13 23 28 41 42 47 49 59 85 91 99 102 109 112 121 126 143 150 153 171 172 178 181 196 199 221 222 225 234 243 248
8	40	0.918460 (0.842483)	4	2	1 2 4 8 16 32 64 128 13 23 28 41 42 47 49 59 78 85 91 99 102 109 112 121 126 143 150 153 171 172 178 181 196 199 221 222 225 234 243 248
8	41	0.923989 (0.849528)	3	2	1 2 4 8 16 32 64 128 13 23 28 33 41 42 47 49 59 78 85 91 99 102 109 112 121 126 143 150 153 171 172 178 181 196 199 221 222 225 234 243 248
8	42	0.929227 (0.855588)	3	2	1 2 4 8 16 32 64 128 13 23 28 41 42 47 49 59 78 85 91 99 102 109 112 121 126 143 150 153 167 171 172 178 181 196 199 207 221 222 225 234 243 248
8	43	0.934190 (0.860851)	3	2	1 2 4 8 16 32 64 128 23 27 39 42 52 57 63 73 78 83 86 93 97 101 120 123 138 143 147 149 156 163 166 169 175 184 205 208 223 225 226 236 243 253 254
8	44	0.938899 (0.865444)	3	2	1 2 4 8 16 32 64 128 22 27 29 46 60 70 73 87 88 101 113 114 127 129 139 140 146 158 166 173 179 181 185 186 195 206 207 209 218 221 231 233 234 236 246 248
8	45	0.943328 (0.869516)	3	2	1 2 4 8 16 32 64 128 13 23 28 41 42 47 49 59 78 85 90 91 99 102 104 109 121 126 131 143 145 150 153 171 172 178 181 192 198 199 221 222 225 234 243 248 255

Continued on next page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
8	46	0.947510 (0.873114)	3	2	1 2 4 8 16 32 64 128 13 23 28 41 42 47 49 59 78 85 90 91 99 102 104 109 121 126 131 143 145 150 153 171 172 178 181 183 192 198 199 221 222 225 234 243 248 255
8	47	0.951417 (0.876312)	3	2	1 2 4 8 16 32 64 128 6 13 19 23 28 39 41 42 47 49 69 78 80 85 90 91 99 102 109 121 123 126 143 150 153 171 172 178 181 196 199 221 222 225 230 234 243 248 255
8	48	0.955038 (0.879166)	3	2	1 2 4 8 16 32 64 128 6 13 19 23 28 39 41 42 47 49 69 78 79 80 85 90 91 99 102 109 121 123 126 143 150 153 171 172 178 181 196 199 221 222 225 230 234 243 248 255
8	49	0.958418 (0.882106)	3	2	1 2 4 8 16 32 64 128 13 23 28 31 41 42 47 49 59 78 85 90 91 92 99 102 104 109 121 126 131 143 145 150 153 171 172 178 181 183 192 198 199 221 222 225 234 243 244 248 255
8	50	0.961533 (0.884714)	3	2	1 2 4 8 16 32 64 128 6 13 19 23 28 39 41 42 47 49 50 69 78 79 80 85 90 91 99 100 102 109 121 123 126 143 150 153 171 172 178 181 196 199 221 222 225 230 234 243 248 255
8	51	0.964406 (0.887044)	3	2	1 2 4 8 16 32 64 128 13 23 28 31 41 42 47 49 59 78 85 90 91 92 99 102 104 109 121 126 131 133 143 145 150 153 171 172 178 181 183 192 198 199 212 221 222 225 228 234 243 248 255
8	52	0.967091 (0.889008)	3	2	1 2 4 8 16 32 64 128 13 19 23 28 31 41 42 47 49 59 78 80 85 90 91 92 99 102 104 109 121 126 136 143 145 150 153 154 171 172 174 178 181 199 217 221 222 225 230 234 237 243 248 255
8	53	0.969588 (0.890942)	3	2	1 2 4 8 16 32 64 128 13 19 23 28 31 41 42 47 49 59 78 80 85 90 91 92 99 102 104 109 121 126 136 143 145 150 153 154 171 172 174 178 181 196 199 217 221 222 225 230 234 237 243 248 255
8	54	0.971913 (0.892678)	3	2	1 2 4 8 16 32 64 128 13 19 23 28 31 41 42 47 49 59 62 78 85 90 91 92 99 102 104 109 121 125 126 136 143 145 150 153 154 171 172 174 178 181 196 199 217 221 222 225 230 234 237 243 248 255
8	55	0.974079 (0.894251)	3	2	1 2 4 8 16 32 64 128 13 23 28 31 41 42 47 49 59 78 85 90 91 92 99 102 104 109 112 121 126 131 133 138 143 145 150 153 167 171 172 178 181 186 192 198 199 201 212 221 222 225 228 234 243 248 255
8	56	0.976055 (0.895673)	3	2	1 2 4 8 16 32 64 128 13 19 23 28 31 41 42 47 49 59 62 69 78 80 85 90 91 92 99 102 104 109 121 125 126 136 143 145 150 153 154 171 172 174 178 181 196 199 217 221 222 225 230 234 237 243 248 255
8	57	0.977899 (0.896990)	3	2	1 2 4 8 16 32 64 128 6 13 19 23 28 37 41 42 47 49 52 59 69 75 78 85 90 91 99 102 104 109 119 121 126 131 143 145 150 153 167 171 172 174 178 181 186 192 198 199 221 222 225 228 234 237 243 248 255
8	58	0.979611 (0.898186)	3	2	1 2 4 8 16 32 64 128 6 10 13 19 23 28 37 41 42 47 49 52 59 69 75 78 85 90 91 99 102 104 109 119 121 126 131 143 145 150 153 167 171 172 174 178 181 186 192 198 199 221 222 225 228 234 237 243 248 255
8	59	0.981199 (0.899277)	3	2	1 2 4 8 16 32 64 128 6 10 13 19 23 28 37 41 42 47 49 52 59 69 75 78 85 90 91 99 102 104 109 119 121 126 131 143 145 150 153 167 171 172 174 178 181 186 192 198 199 201 221 222 225 228 234 237 243 248 255
8	60	0.982663 (0.900271)	3	2	1 2 4 8 16 32 64 128 6 10 13 19 23 28 37 41 42 47 49 52 59 65 75 78 85 90 91 99 102 104 109 119 121 126 131 143 145 148 150 153 167 171 172 174 178 181 186 192 198 199 201 221 222 225 228 234 237 243 248 255
8	61	0.984019 (0.901188)	3	2	1 2 4 8 16 32 64 128 6 10 13 19 23 28 37 41 42 47 49 52 59 69 75 78 85 90 91 99 102 104 109 118 119 121 126 131 143 145 148 150 153 167 171 172 174 178 181 186 192 198 199 201 221 222 225 228 234 237 243 248 255
8	62	0.985268 (0.902022)	3	2	1 2 4 8 16 32 64 128 6 13 19 23 28 31 37 41 42 47 49 52 59 75 78 85 90 91 99 102 104 109 118 119 121 126 131 138 143 145 150 153 167 171 172 174 178 181

133

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
9	35	0.849072 (0.731867)	4	3	1 2 4 8 16 32 64 128 256 28 69 70 105 119 149 175 195 205 222 228 240 251 278 290 309 315 335 360 391 394 413 428 468 503 510
9	36	0.858952 (0.737420)	4	3	1 2 4 8 16 32 64 128 256 28 69 70 105 119 175 195 205 222 228 240 251 278 290 309 315 335 349 360 391 394 413 417 428 468 503 510
9	37	0.868369 (0.750055)	4	3	1 2 4 8 16 32 64 128 256 28 69 70 105 110 119 175 195 205 222 228 240 251 278 290 309 315 335 346 360 391 394 413 422 428 468 503 510
9	38	0.877168 (0.754846)	3	3	1 2 4 8 16 32 64 128 256 28 69 70 87 105 110 119 175 195 205 222 228 240 251 278 290 309 315 335 346 360 391 394 413 422 428 468 503 510
9	39	0.885393 (0.765018)	4	3	1 2 4 8 16 32 64 128 256 35 124 143 145 156 170 183 202 217 229 230 240 254 303 309 326 376 395 403 404 423 430 441 448 451 460 469 482 507 509
9	40	0.893000 (0.773516)	4	3	1 2 4 8 16 32 64 128 256 28 57 69 70 105 114 119 175 195 209 222 228 240 251 273 278 280 290 309 315 335 360 391 394 413 417 428 442 468 482 503
9	41	0.900208 (0.780789)	4	3	1 2 4 8 16 32 64 128 256 31 45 60 100 113 122 143 153 156 181 191 194 204 214 218 281 291 302 331 332 344 370 383 394 397 403 418 427 440 478 485 488
9	42	0.906909 (0.787001)	4	3	1 2 4 8 16 32 64 128 256 28 57 62 69 70 105 114 119 175 195 209 222 228 237 240 251 273 278 290 309 315 335 360 391 394 413 417 428 442 468 474 482 503
9	43	0.913307 (0.792343)	4	3	1 2 4 8 16 32 64 128 256 62 69 70 88 105 114 119 124 175 179 195 217 222 228 240 251 278 281 290 309 315 335 349 360 391 394 413 417 428 432 442 468 503 510
9	44	0.919294 (0.797022)	4	3	1 2 4 8 16 32 64 128 256 62 69 70 88 105 114 119 124 175 179 195 202 217 222 228 240 251 278 281 290 309 315 335 349 360 391 394 413 417 428 432 442 468 503 510
9	45	0.924899 (0.801120)	4	3	1 2 4 8 16 32 64 128 256 62 69 70 88 105 114 119 124 149 175 179 195 202 217 222 228 240 251 278 281 290 309 315 335 349 360 391 394 413 417 428 432 442 468 503 510
9	46	0.930177 (0.804727)	4	3	1 2 4 8 16 32 64 128 256 62 69 70 88 105 114 119 124 149 175 179 195 202 217 222 228 240 251 278 281 290 309 315 335 349 360 391 394 413 417 428 432 442 468 503 510
9	47	0.935082 (0.807922)	4	3	1 2 4 8 16 32 64 128 256 62 69 70 88 105 114 119 124 149 175 179 195 217 222 228 240 251 278 281 290 309 315 316 335 347 349 360 374 391 394 413 417 428 432 442 468 503 510
9	48	0.939657 (0.810765)	3	3	1 2 4 8 16 32 64 128 256 62 69 70 88 105 114 119 124 149 175 179 189 195 217 222 228 240 251 278 281 290 309 315 316 335 347 349 360 374 391 394 413 417 428 432 442 468 503 510
9	49	0.943951 (0.813604)	3	3	1 2 4 8 16 32 64 128 256 62 69 70 88 105 114 119 124 149 175 179 189 195 217 222 228 240 251 278 281 290 309 315 316 335 347 349 360 374 391 394 411 413 417 428 432 442 468 503 510
9	50	0.947965 (0.816125)	3	3	1 2 4 8 16 32 64 128 256 62 69 70 81 88 105 114 119 124 149 175 179 189 195 217 222 228 240 251 278 281 290 309 315 316 335 347 349 360 374 391 394 411 413 417 428 432 442 468 503 510
9	51	0.951720 (0.818369)	3	3	1 2 4 8 16 32 64 128 256 62 69 70 81 88 105 114 119 124 149 175 179 189 195 217 222 228 240 251 278 281 290 309 315 316 335 343 347 349 360 374 391 394 400 411 413 417 428 442 468 503 510
9	52	0.955226 (0.820306)	3	3	1 2 4 8 16 32 64 128 256 62 69 70 81 88 105 114 119 124 149 175 179 189 195 217 222 228 240 251 278 281 290 309 315 316 335 343 347 349 360 374 391 394 400 411 413 417 428 442 468 497 503 510
9	53	0.958441 (0.822166)	3	3	1 2 4 8 16 32 64 128 256 62 69 70 81 88 105 114 119 124 149 175 179 180 189 195 217 222 228 240 251 278 281 290 309 315 316 335 343 347 349 360 374 391 394 400 411 413 417 428 442 468 497 503 510
9	54	0.961434 (0.823770)	3	2	1 2 4 8 16 32 64 128 256 19 62 69 70 81 88 105 114 119 124 149 175 179 189 195 217 222 228 240 251 278 281 290 309 315 316 335 343 347 349 360 374 391 394 400 411 412 413 417 428 442 468 499 503 510
9	55	0.964240 (0.825313)	3	2	1 2 4 8 16 32 64 128 256 19 62 69 70 81 88 105 114 119 124 149 175 179 189 195 217 222 228 240 251 278 281 290 309 315 316 335 343 347 349 360 374 391 394 411 412 413 417 428 432 442 449 468 499 503 510
9	56	0.966862 (0.826710)	3	2	1 2 4 8 16 32 64 128 256 19 62 69 70 81 88 105 114 119 124 149 175 179 189 195 217 222 223 228 240 251 278 281 290 309 315 316 335 343 347 349 360 374 391 394 411 412 413 417 428 432 442 449 468 499 503 510
9	57	0.969280 (0.827992)	3	2	1 2 4 8 16 32 64 128 256 19 62 69 70 81 88 105 114 119 124 149 175 179 189 195 217 222 223 228 240 251 278 281 290 309 315 316 335 343 347 349 360 374 391 394 411 412 413 417 428 432 442 449 468 495 499 503 510
9	58	0.971542 (0.829155)	3	2	1 2 4 8 16 32 64 128 256 19 62 69 70 81 88 90 105 114 119 124 149 175 179 180 189 195 217 222 228 240 251 278 281 290 309 315 316 335 343 347 349 360 374 391 394 400 411 412 413 417 428 432 442 465 468 499 503 510
9	59	0.973628 (0.830210)	3	2	1 2 4 8 16 32 64 128 256 15 19 62 69 70 81 88 105 114 119 124 149 175 179 180 189 195 217 222 223 228 230 240 251 278 281 290 309 315 316 335 343 347 349 360 374 391 394 411 412 413 417 428 432 442 449 465 468 499 503 510
9	60	0.975573 (0.831180)	3	2	1 2 4 8 16 32 64 128 256 15 19 62 69 70 81 88 105 114 119 124 149 175 179 180 189 195 217 222 223 228 230 240 251 278 281 290 292 309 315 316 335 343 347 349 360 374 391 394 411 412 413 417 428 432 442 449 465 468 499 503 510
9	61	0.977370 (0.832064)	3	2	1 2 4 8 16 32 64 128 256 15 19 62 69 70 81 88 105 114 119 124 149 175 179 180 189 195 217 222 223 228 230 240 251 278 281 290 292 309 315 316 335 343 347 349 360 374 391 394 411 412 413 417 428 432 442 449 465 468 499 503 510
9	62	0.979048 (0.832874)	3	2	1 2 4 8 16 32 64 128 256 19 31 54 62 69 70 81 88 105 114 119 124 149 175 179 180 189 195 214 217 222 223 228 230 240 251 272 278 281 290 309 315 316 335 343 347 349 360 374 391 394 411 412 413 417 428 432 442 465 468 499 503 510
9	63	0.980612 (0.833619)	3	2	1 2 4 8 16 32 64 128 256 19 31 54 62 69 70 81 88 105 114 119 124 149 175 179 180 189 195 214 217 222 223 228 230 240 251 272 278 281 290 309 315 316 335 343 347 349 360 372 374 391 394 411 412 413 417 428 432 442 465 468 499 503 510
9	64	0.982063 (0.834301)	3	2	1 2 4 8 16 32 64 128 256 19 31 54 62 69 70 81 88 105 114 119 124 149 175 179 180 184 189 195 214 217 222 223 228 230 240 251 272 278 281 290 309 315 316 335 343 347 349 360 372 374 391 394 411 412 413 417 428 432 442 465 468 499 503 510
9	65	0.983412 (0.834924)	3	2	1 2 4 8 16 32 64 128 256 19 31 46 54 62 69 70 81 88 105 114 119 124 149 175 179 180 184 189 195 214 217 222 223 228 230 240 251 272 278 281 290 309 315 316 335 343 347 349 360 372 374 391 394 411 412 413 417 428 432 442 465 468 499 503 510
9	66	0.984671 (0.853678)	3	2	1 2 4 8 16 32 64 128 256 19 31 46 54 62 69 70 81 88 105 114 119 124 149 175 179 180 184 189 195 214 217 222 223 228 230 240 251 272 278 281 290 309 315 316 335 343 347 349 360 372 374 391 394 411 412 413 417 428 432 442 465 468 499 503 510
9	67	0.985834 (0.867233)	3	2	1 2 4 8 16 32 64 128 256 19 31 46 54 62 69 70 81 88 105 114 119 124 149 175 179 180 184 189 195 214 217 222 223 228 230 240 251 272 278 281 290 309 315 316 335 343 347 349 360 372 374 391 394 411 412 413 417 428 432 442 465 466 468 499 503 505 510
9	68	0.986911 (0.867759)	3	2	1 2 4 8 16 32 64 128 256 19 31 46 54 62 69 70 81 88 105 114 119 124 149 175 178 179 180 184 189 195 214 217 222 223 228 230 240 251 272 278 281 290 309 315 316 335 343 347 349 360 372 374 391 394 411 412 413 417 428 432 442 465 466 468 499 503 505 510
9	69	0.987905 (0.878014)	3	2	1 2 4 8 16 32 64 128 256 19 31 46 54 62 69 70 81 88 105 114 119 124 149 175 178 179 180 184 189 195 214 217 222 223 228 230 240 251 272 278 281 290 309 315 316 335 343 347 349 360 372 374 391 394 411 412 413 417 428 432 442 451 465 466 468 499 503 505 510

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
9	70	0.988829 (0.885978)	3	2	1 2 4 8 16 32 64 128 256 19 31 46 54 62 69 70 81 88 93 105 114 119 124 149 175 178 179 180 184 189 195 214 217 222 223 228 230 240 251 272 278 281 290 309 315 316 335 343 347 349 360 372 374 391 394 411 412 413 417 428 432 442 451 465 466 468 499 503 505 510
9	71	0.989686 (0.892257)	3	2	1 2 4 8 16 32 64 128 256 19 31 46 54 62 69 70 81 88 93 105 114 119 124 149 175 178 179 180 184 189 195 214 217 222 223 228 230 240 251 272 278 281 290 309 315 316 335 343 347 349 360 372 374 391 394 411 412 413 417 423 428 432 442 451 465 466 468 499 503 505 510
9	72	0.990478 (0.897255)	3	2	1 2 4 8 16 32 64 128 256 19 31 46 54 62 69 70 81 88 93 105 114 119 124 149 175 178 179 180 184 189 195 214 217 222 223 228 230 240 251 272 278 281 290 301 309 315 316 335 343 347 349 360 372 374 391 394 411 412 413 417 423 428 432 442 451 465 466 468 499 503 505 510
10	11	0.315031	11	5	1 2 4 8 16 32 64 128 256 512 1023
10	12	0.343532	8	6	1 2 4 8 16 32 64 128 256 512 191 967
10	13	0.371750 (0.371749)	7	6	1 2 4 8 16 32 64 128 256 512 127 399 691
10	14	0.399868	7	6	1 2 4 8 16 32 64 128 256 512 127 399 691 853
10	15	0.427695	7	5	1 2 4 8 16 32 64 128 256 512 127 399 691 853 1001
10	16	0.454332 (0.451968)	6	5	1 2 4 8 16 32 64 128 256 512 127 214 399 691 853 1001
10	17	0.480743 (0.477734)	6	5	1 2 4 8 16 32 64 128 256 512 55 253 366 399 589 723 794
10	18	0.506738 (0.502858)	6	5	1 2 4 8 16 32 64 128 256 512 55 253 364 399 470 681 723 948
10	19	0.532171 (0.530239)	6	5	1 2 4 8 16 32 64 128 256 512 127 229 313 399 691 807 853 1001 1022
10	20	0.556948 (0.554614)	6	5	1 2 4 8 16 32 64 128 256 512 127 229 313 399 467 691 807 853 1001 1022
10	21	0.581009 (0.578242)	6	5	1 2 4 8 16 32 64 128 256 512 127 229 313 399 467 587 691 807 853 1001 1022
10	22	0.604330 (0.601140)	6	3	1 2 4 8 16 32 64 128 256 512 127 229 313 399 467 587 669 691 807 853 1001 1022
10	23	0.624110 (0.623249)	4	3	1 2 4 8 16 32 64 128 256 512 127 229 313 399 467 498 587 669 691 807 853 1001 1022
10	24	0.644584	6	4	1 2 4 8 16 32 64 128 256 512 199 379 406 431 472 575 611 733 758 787 820 862 877 961
10	25	0.663877 (0.661321)	4	4	1 2 4 8 16 32 64 128 256 512 191 203 230 409 426 468 527 613 661 787 820 838 872 954 994
10	26	0.682877 (0.680647)	4	4	1 2 4 8 16 32 64 128 256 512 191 203 301 409 426 462 468 527 592 613 661 755 787 820 838 994
10	27	0.700864 (0.698895)	4	3	1 2 4 8 16 32 64 128 256 512 60 191 203 230 409 426 462 468 527 592 613 661 787 820 838 872 994
10	28	0.717944 (0.716439)	5	4	1 2 4 8 16 32 64 128 256 512 191 203 230 252 409 426 468 527 596 613 652 661 679 787 820 838 872 994
10	29	0.734264 (0.733170)	5	3	1 2 4 8 16 32 64 128 256 512 191 203 230 278 409 426 435 462 468 527 613 661 787 820 838 872 879 922 994
10	30	0.749871 (0.749126)	4	3	1 2 4 8 16 32 64 128 256 512 191 203 230 278 317 409 426 435 462 468 527 613 661 787 820 838 872 879 922 994
10	31	0.764705 (0.764514)	4	3	1 2 4 8 16 32 64 128 256 512 87 165 191 203 301 350 409 426 468 527 613 661 675 732 755 787 820 879 983 994 1001
10	32	0.779025	5	4	1 2 4 8 16 32 64 128 256 512 77 191 232 341 382 411 495 505 607 629 643 664 682 763 764 815 822 919 971 977 997 1010
10	33	0.792695	5	3	1 2 4 8 16 32 64 128 256 512 77 191 232 341 382 411 495 505 607 629 643 664 682 763 764 815 822 860 919 971 977 997 1010
10	34	0.804736 (0.667461)	4	3	1 2 4 8 16 32 64 128 256 512 53 87 191 203 229 301 326 344 409 426 468 527 613 661 675 700 716 787 820 879 889 951 990 994
10	35	0.816623 (0.681773)	4	3	1 2 4 8 16 32 64 128 256 512 53 87 191 203 229 261 301 326 344 409 426 468 527 613 661 675 700 716 787 820 879 889 951 990 994
10	36	0.827909 (0.687317)	4	3	1 2 4 8 16 32 64 128 256 512 87 191 203 229 301 409 426 467 468 501 527 596 613 622 637 661 675 716 755 787 811 820 858 879 969 994
10	37	0.838597 (0.698936)	4	3	1 2 4 8 16 32 64 128 256 512 87 191 203 229 301 409 426 435 467 468 501 527 596 613 622 637 661 675 716 755 787 811 820 858 879 969 994
10	38	0.848560 (0.703688)	4	3	1 2 4 8 16 32 64 128 256 512 87 191 203 229 301 409 426 435 467 468 501 527 596 613 622 637 661 675 716 755 787 811 820 858 879 969 994 1008
10	39	0.857997 (0.708003)	4	3	1 2 4 8 16 32 64 128 256 512 87 191 203 229 301 409 426 435 467 468 495 498 501 527 596 613 622 637 661 675 716 755 787 811 820 858 879 969 994
10	40	0.866868 (0.717155)	4	3	1 2 4 8 16 32 64 128 256 512 87 191 203 229 301 409 426 467 468 498 501 527 585 594 596 613 622 637 661 675 716 755 787 811 820 858 879 881 969 994
10	41	0.875287 (0.724856)	4	3	1 2 4 8 16 32 64 128 256 512 30 87 191 203 229 301 344 409 426 435 467 468 501 527 613 622 637 661 675 716 735 755 787 820 858 879 889 897 969 994 1008
10	42	0.883216 (0.731373)	4	3	1 2 4 8 16 32 64 128 256 512 29 87 191 203 221 229 301 344 379 409 426 435 467 468 495 501 527 596 613 637 661 675 716 755 787 820 858 879 897 969 994 1008
10	43	0.890740 (0.736905)	4	3	1 2 4 8 16 32 64 128 256 512 29 87 191 203 221 229 301 344 350 379 409 426 435 467 468 495 501 527 596 613 637 661 675 716 755 787 820 858 879 897 969 994 1008
10	44	0.897785 (0.741748)	4	3	1 2 4 8 16 32 64 128 256 512 29 87 191 203 221 229 301 344 350 379 409 426 435 467 468 495 501 527 596 613 637 661 675 716 755 787 820 858 879 897 969 994 1008
10	45	0.904383 (0.745960)	4	3	1 2 4 8 16 32 64 128 256 512 29 87 109 191 203 221 229 301 344 350 379 409 426 435 467 468 495 501 527 539 596 613 637 661 675 716 755 787 820 858 879 897 969 994 1008
10	46	0.910602 (0.749651)	4	3	1 2 4 8 16 32 64 128 256 512 29 30 87 109 191 203 221 229 301 344 379 409 426 435 467 468 495 501 527 596 613 637 646 661 675 716 755 767 787 820 858 879 897 969 994 1008
10	47	0.916443 (0.752892)	4	3	1 2 4 8 16 32 64 128 256 512 28 56 87 191 203 229 283 301 344 350 371 409 426 435 467 468 501 527 613 622 637 661 675 716 755 766 787 820 858 879 889 897 902 911 969 994 1008
10	48	0.921950 (0.755770)	4	3	1 2 4 8 16 32 64 128 256 512 29 30 87 191 203 221 229 267 301 344 350 355 379 409 426 435 467 468 495 501 527 596 613 637 646 661 675 716 755 766 787 820 858 879 897 969 994 1008
10	49	0.927105 (0.758562)	4	3	1 2 4 8 16 32 64 128 256 512 29 30 87 191 203 221 229 267 301 344 350 355 379 400 409 426 435 467 468 495 501 527 596 613 637 646 661 675 716 755 760 787 820 858 879 897 969 994 1008
10	50	0.931938 (0.761043)	4	3	1 2 4 8 16 32 64 128 256 512 29 30 87 191 203 221 229 267 301 344 350 355 379 400 409 426 435 467 468 495 501 527 596 613 637 646 661 675 681 716 755 760 787 820 858 879 897 969 994 1008
10	51	0.936507 (0.763236)	4	3	1 2 4 8 16 32 64 128 256 512 29 87 94 105 191 203 221 229 267 301 344 355 379 409 426 435 467 468 495 501 527 552 596 613 618 637 646 661 675 716 755 787 820 858 879 897 964 969 970 994 1008
10	52	0.940780 (0.765173)	4	3	1 2 4 8 16 32 64 128 256 512 29 87 94 105 191 203 221 229 267 301 344 355 379 409 426 435 467 468 495 501 527 539 552 596 613 618 637 646 661 675 716 755 787 820 858 879 897 964 969 970 994 1008
10	53	0.944789 (0.766996)	4	3	1 2 4 8 16 32 64 128 256 512 29 30 82 87 191 203 221 229 267 301 309 344 350 355 379 409 426 435 467 468 495 501 527 539 596 613 637 646 661 675 716 755 787 793 820 858 879 897 969 970 989 994 1008
10	54	0.948545 (0.768591)	4	3	1 2 4 8 16 32 64 128 256 512 29 30 82 87 164 191 203 221 229 267 301 309 344 350 355 379 409 426 435 467 468 495 501 527 539 596 613 637 646 661 675 716 755 787 793 820 858 879 897 969 970 989 994 1008

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
10	55	0.952060 (0.770032)	4	3	1 2 4 8 16 32 64 128 256 512 29 30 82 87 191 203 221 229 267 301 309 344 350 355 379 409 426 435 467 468 495 501 527 539 596 613 630 637 646 661 675 716 755 757 787 793 820 858 879 897 969 970 989 994 1008
10	56	0.955362 (0.771420)	4	3	1 2 4 8 16 32 64 128 256 512 29 30 82 87 164 191 203 221 229 267 282 301 309 344 350 355 379 409 426 435 467 468 495 501 527 539 596 613 637 646 661 675 716 751 755 787 793 820 858 879 897 969 970 989 994 1008
10	57	0.958447 (0.772688)	4	3	1 2 4 8 16 32 64 128 256 512 29 30 82 87 164 191 203 221 229 267 282 301 309 344 350 355 379 409 426 435 467 468 495 501 527 539 560 596 613 637 646 661 675 716 751 755 787 793 820 858 879 897 969 970 989 994 1008
10	58	0.961344 (0.773839)	4	3	1 2 4 8 16 32 64 128 256 512 29 30 82 87 164 191 203 221 229 267 282 301 309 344 350 355 379 409 426 435 467 468 495 501 527 539 560 596 613 637 646 661 675 716 734 751 755 787 793 820 858 879 897 969 970 989 994 1008
10	59	0.964053 (0.774875)	4	3	1 2 4 8 16 32 64 128 256 512 29 30 82 87 191 203 221 229 267 282 301 309 344 350 355 379 399 409 426 435 467 468 495 501 527 539 560 596 613 630 637 646 661 675 716 734 751 755 787 793 820 858 879 897 969 970 989 994 1008
10	60	0.966579 (0.775831)	4	3	1 2 4 8 16 32 64 128 256 512 29 30 82 87 164 191 203 221 229 267 282 301 309 344 350 355 379 399 409 426 435 467 468 495 501 527 539 560 596 613 630 637 646 661 675 716 734 751 755 787 793 820 858 879 897 969 970 989 994 1008
10	61	0.968938 (0.776703)	4	3	1 2 4 8 16 32 64 128 256 512 29 30 82 87 164 191 203 221 229 267 282 301 309 344 350 355 379 399 409 426 435 467 468 495 501 527 539 560 596 613 630 637 646 661 675 716 734 751 755 787 793 820 858 879 897 954 969 970 989 994 1008
10	62	0.971129 (0.777500)	4	3	1 2 4 8 16 32 64 128 256 512 29 30 82 87 164 191 203 221 229 267 282 301 309 344 350 355 379 399 409 426 435 450 467 468 495 501 527 539 560 596 613 630 637 646 661 675 716 734 751 755 787 793 820 858 879 897 954 969 970 989 994 1008
10	63	0.973181 (0.778231)	4	3	1 2 4 8 16 32 64 128 256 512 29 30 82 87 105 120 164 191 203 221 229 267 282 301 309 344 350 355 379 399 409 426 435 467 468 495 501 527 539 551 560 596 613 630 637 646 661 675 716 734 755 757 787 793 820 858 879 897 969 970 989 994 1008
10	64	0.975098 (0.778901)	4	3	1 2 4 8 16 32 64 128 256 512 29 30 81 82 87 174 185 191 203 221 229 267 282 301 309 344 350 355 379 399 400 409 426 435 467 468 492 495 501 527 539 596 613 630 637 646 661 675 716 734 751 755 787 793 820 858 879 897 954 969 970 989 994 1008
10	65	0.976887 (0.779510)	4	3	1 2 4 8 16 32 64 128 256 512 29 30 81 82 87 174 185 191 203 221 229 267 282 301 309 344 350 355 379 399 400 409 426 435 467 468 492 495 501 527 539 596 613 630 637 646 661 675 716 734 751 755 787 793 820 858 872 879 897 954 969 970 989 994 1008
10	66	0.978559 (0.796407)	4	3	1 2 4 8 16 32 64 128 256 512 29 30 82 87 164 174 191 203 221 229 267 282 301 309 312 344 350 355 379 399 409 426 435 450 467 468 492 495 501 527 539 560 596 601 613 630 637 646 661 675 716 734 751 755 787 793 820 858 879 897 954 969 970 989 994 1008
10	67	0.980117 (0.808624)	4	3	1 2 4 8 16 32 64 128 256 512 29 30 81 82 87 164 174 191 203 221 229 267 282 301 309 312 344 350 355 379 399 409 426 435 450 467 468 492 495 501 527 539 560 596 601 613 630 637 646 661 675 716 734 751 755 787 793 820 858 879 897 954 969 970 989 994 1008
10	68	0.981558 (0.809147)	4	3	1 2 4 8 16 32 64 128 256 512 29 30 81 82 87 164 174 191 203 221 229 267 282 301 309 312 344 350 355 379 399 409 426 435 450 467 468 492 495 501 527 539 560 596 601 613 630 637 646 653 661 675 716 734 751 755 787 793 820 858 879 897 954 969 970 989 994 1008
10	69	0.982900 (0.818402)	4	3	1 2 4 8 16 32 64 128 256 512 29 30 81 82 87 174 185 191 203 221 229 267 282 301 309 312 344 350 355 368 379 392 399 409 426 435 467 468 492 495 501 527 539 560 596 601 613 630 637 646 661 675 677 716 734 751 755 787 793 820 858 879 897 954 969 970 989 994 1008
10	70	0.984157 (0.818852)	4	3	1 2 4 8 16 32 64 128 256 512 29 30 81 82 87 174 185 191 203 221 229 267 282 301 309 312 344 350 355 368 379 392 399 409 426 435 467 468 492 495 501 527 539 560 596 601 613 625 630 637 646 661 675 677 716 734 751 755 787 793 820 858 879 897 954 969 970 989 994 1008
10	71	0.985319 (0.826020)	4	3	1 2 4 8 16 32 64 128 256 512 29 30 81 82 87 155 174 185 191 203 221 229 267 282 301 309 312 344 350 355 368 379 392 399 409 426 435 467 468 492 495 501 527 539 560 596 601 613 625 630 637 646 661 675 677 716 734 751 755 787 793 820 858 879 897 954 969 970 989 994 1008
10	72	0.986403 (0.831670)	3	3	1 2 4 8 16 32 64 128 256 512 29 30 54 81 82 87 174 191 203 221 229 267 282 301 309 312 344 350 355 368 379 399 400 409 426 435 467 468 492 495 501 527 539 560 596 613 625 630 632 637 646 653 661 675 695 716 734 751 755 761 787 793 820 858 879 897 954 969 970 989 994 1008
10	73	0.987410	3	3	1 2 4 8 16 32 64 128 256 512 29 30 52 81 82 87 174 185 191 198 203 221 229 267 282 301 309 312 344 350 355 368 379 392 399 409 426 435 467 468 492 495 501 527 539 560 596 601 613 625 630 637 646 661 675 677 716 734 751 755 787 793 820 858 879 897 919 954 969 970 989 994 1008
10	74	0.988344 (0.839794)	3	3	1 2 4 8 16 32 64 128 256 512 29 30 39 81 82 87 155 174 185 191 198 203 221 229 267 282 301 309 312 344 350 355 368 379 392 399 409 426 435 467 468 492 495 501 527 539 560 596 601 613 625 630 637 646 653 661 675 677 716 734 751 755 787 793 820 858 879 897 954 969 970 989 994 1008
10	75	0.989213 (0.842715)	3	3	1 2 4 8 16 32 64 128 256 512 29 30 39 81 82 87 155 174 185 191 198 203 221 229 267 282 301 309 312 344 350 355 368 379 392 399 409 426 435 467 468 492 495 501 527 539 560 596 601 613 625 630 637 646 653 661 675 677 716 734 737 751 755 787 793 820 858 879 897 954 969 970 989 994 1008
10	76	0.990022 (0.845086)	3	3	1 2 4 8 16 32 64 128 256 512 29 30 52 81 82 87 160 174 185 191 198 203 221 229 267 282 301 309 312 344 350 355 368 379 392 399 409 426 435 467 468 492 495 501 527 539 560 563 596 601 607 613 625 630 637 646 661 675 677 716 734 751 755 787 793 820 858 879 897 919 954 969 970 989 994 1008
10	77	0.990771 (0.847029)	3	3	1 2 4 8 16 32 64 128 256 512 29 30 52 81 82 87 155 160 174 185 191 198 203 221 229 267 282 301 309 312 344 350 355 368 379 392 399 409 426 435 467 468 492 495 501 527 539 560 563 596 601 607 613 625 630 637 646 661 675 677 716 734 751 755 787 793 820 858 879 897 919 954 969 970 989 994 1008
11	12	0.312421	12	6	1 2 4 8 16 32 64 128 256 512 1024 2047
11	13	0.338382	8	6	1 2 4 8 16 32 64 128 256 512 1024 415 1786
11	14	0.364203 (0.364202)	8	7	1 2 4 8 16 32 64 128 256 512 1024 415 875 1710
11	15	0.389947	8	7	1 2 4 8 16 32 64 128 256 512 1024 415 875 1143 1710
11	16	0.415516 (0.415515)	8	6	1 2 4 8 16 32 64 128 256 512 1024 415 875 1404 1710 1753
11	17	0.440378 (0.440234)	7	6	1 2 4 8 16 32 64 128 256 512 1024 415 573 875 1404 1710 1753
11	18	0.465075 (0.465038)	7	6	1 2 4 8 16 32 64 128 256 512 1024 415 497 822 875 1143 1710 2002
11	19	0.489476	7	6	1 2 4 8 16 32 64 128 256 512 1024 415 497 822 875 972 1143 1710 2002
11	20	0.513477 (0.513476)	7	6	1 2 4 8 16 32 64 128 256 512 1024 415 486 811 1079 1227 1402 1710 1863 1938
11	21	0.537014	7	6	1 2 4 8 16 32 64 128 256 512 1024 415 486 606 811 1079 1227 1710 1863 1938 2045
11	22	0.560019	7	6	1 2 4 8 16 32 64 128 256 512 1024 415 486 606 755 811 1079 1227 1710 1863 1938 2045
11	23	0.582470	7	3	1 2 4 8 16 32 64 128 256 512 1024 415 486 606 755 811 1079 1227 1402 1710 1863 1938 2045
11	24	0.601747 (0.592625)	4	3	1 2 4 8 16 32 64 128 256 512 1024 415 486 606 755 770 811 1079 1227 1402 1710 1863 1938 2045
11	25	0.620752 (0.617354)	4	4	1 2 4 8 16 32 64 128 256 512 1024 378 415 486 606 755 811 1079 1094 1180 1227 1710 1863 1938 2045
11	26	0.639913 (0.637823)	5	4	1 2 4 8 16 32 64 128 256 512 1024 177 415 486 647 811 964 1079 1227 1370 1710 1828 1863 1929 1938 2044

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
11	27	0.658573 (0.657212)	5	4	1 2 4 8 16 32 64 128 256 512 1024 415 486 507 647 811 964 1079 1172 1227 1370 1710 1828 1863 1929 1938 2044
11	28	0.676537 (0.675604)	5	4	1 2 4 8 16 32 64 128 256 512 1024 177 415 486 647 811 964 1079 1227 1370 1484 1597 1710 1828 1863 1929 1938 2044
11	29	0.693783 (0.693320)	5	4	1 2 4 8 16 32 64 128 256 512 1024 415 486 647 691 811 964 1079 1172 1227 1370 1484 1597 1710 1828 1863 1929 1938 2044
11	30	0.710315 (0.710222)	5	4	1 2 4 8 16 32 64 128 256 512 1024 415 486 647 691 811 964 1079 1172 1227 1370 1484 1509 1597 1710 1828 1863 1929 1938 2044
11	31	0.726506	6	5	1 2 4 8 16 32 64 128 256 512 1024 55 241 382 677 682 764 875 1010 1214 1239 1364 1437 1528 1585 1630 1753 1838 1942 1994 2020
11	32	0.742073	6	5	1 2 4 8 16 32 64 128 256 512 1024 55 241 382 493 677 682 764 875 1010 1214 1239 1364 1437 1528 1585 1630 1753 1838 1942 1994 2020
11	33	0.757027	6	5	1 2 4 8 16 32 64 128 256 512 1024 55 241 382 493 677 682 764 875 959 1010 1214 1239 1364 1437 1528 1585 1630 1753 1838 1942 1994 2020
11	34	0.77128	6	4	1 2 4 8 16 32 64 128 256 512 1024 55 241 382 493 677 682 764 875 959 1010 1214 1239 1287 1364 1437 1528 1585 1630 1753 1838 1942 1994 2020
11	35	0.782449 (0.775713)	5	4	1 2 4 8 16 32 64 128 256 512 1024 87 370 415 486 647 745 759 811 936 1079 1130 1227 1238 1264 1370 1466 1546 1668 1710 1828 1863 1929 1938 1980
11	36	0.794834 (0.790051)	5	4	1 2 4 8 16 32 64 128 256 512 1024 47 87 255 415 486 550 647 798 811 890 1011 1079 1227 1245 1370 1407 1466 1484 1546 1634 1710 1828 1863 1929 1938
11	37	0.806696 (0.803255)	4	4	1 2 4 8 16 32 64 128 256 512 1024 87 178 370 415 486 647 745 759 811 936 1079 1130 1227 1238 1264 1370 1466 1546 1668 1710 1749 1828 1863 1929 1938 1980
11	38	0.817966 (0.815439)	4	4	1 2 4 8 16 32 64 128 256 512 1024 47 87 255 317 415 486 550 647 693 798 811 890 1011 1079 1227 1245 1370 1407 1466 1484 1546 1634 1710 1828 1863 1929 1938
11	39	0.828624 (0.826902)	4	4	1 2 4 8 16 32 64 128 256 512 1024 47 87 255 317 415 486 550 647 693 798 811 890 1011 1079 1227 1245 1370 1407 1466 1484 1546 1634 1710 1788 1828 1863 1929 1938
11	40	0.838712 (0.837628)	4	4	1 2 4 8 16 32 64 128 256 512 1024 47 87 255 317 415 486 550 647 693 798 811 890 1011 1079 1227 1245 1366 1370 1407 1466 1484 1546 1634 1710 1788 1828 1863 1929 1938
11	41	0.848307 (0.847670)	4	3	1 2 4 8 16 32 64 128 256 512 1024 47 87 255 317 415 486 550 647 693 798 811 868 890 1011 1079 1227 1245 1366 1370 1407 1466 1484 1546 1634 1710 1788 1828 1863 1929 1938
11	42	0.857387 (0.857106)	4	3	1 2 4 8 16 32 64 128 256 512 1024 95 170 337 376 415 456 486 647 734 754 811 842 887 964 977 1079 1082 1227 1239 1370 1420 1507 1546 1596 1710 1828 1863 1929 1938 1975 2041
11	43	0.866054	5	4	1 2 4 8 16 32 64 128 256 512 1024 47 157 278 366 478 542 556 599 646 695 732 762 821 867 886 911 956 1084 1109 1112 1206 1292 1447 1464 1482 1524 1618 1772 1781 1873 1912 1950
11	44	0.874444	5	4	1 2 4 8 16 32 64 128 256 512 1024 47 157 278 366 478 542 556 599 646 695 732 762 821 867 886 911 956 1084 1109 1112 1121 1206 1292 1447 1464 1482 1524 1618 1772 1781 1873 1912 1950
11	45	0.882366	5	4	1 2 4 8 16 32 64 128 256 512 1024 47 157 278 366 478 542 556 599 646 695 732 762 821 867 886 911 956 1084 1109 1112 1121 1206 1292 1447 1464 1482 1524 1573 1618 1772 1781 1873 1912 1950
11	46	0.889848	5	4	1 2 4 8 16 32 64 128 256 512 1024 47 157 278 366 478 542 556 599 646 695 732 762 821 867 886 911 956 1084 1109 1112 1121 1206 1292 1447 1464 1482 1524 1565 1573 1618 1772 1781 1873 1912 1950
11	47	0.896905	5	3	1 2 4 8 16 32 64 128 256 512 1024 47 157 278 366 478 542 556 599 646 695 732 762 821 867 886 911 956 1084 1109 1112 1121 1206 1292 1447 1464 1482 1524 1565 1573 1618 1772 1781 1873 1912 1925 1950
11	48	0.902358 (0.710488)	4	3	1 2 4 8 16 32 64 128 256 512 1024 82 95 103 137 170 340 376 415 416 456 486 579 647 734 754 811 829 887 964 977 1079 1082 1227 1239 1302 1370 1420 1505 1546 1596 1710 1828 1863 1929 1938 1975 2041
11	49	0.908427 (0.713286)	4	3	1 2 4 8 16 32 64 128 256 512 1024 82 95 103 137 170 340 376 415 416 456 486 579 647 734 754 811 829 887 964 977 1079 1082 1227 1239 1302 1370 1420 1505 1546 1596 1710 1828 1863 1900 1929 1938 1975 2041
11	50	0.914199 (0.715762)	4	3	1 2 4 8 16 32 64 128 256 512 1024 82 95 103 137 170 340 376 415 416 456 486 579 647 734 754 811 829 887 964 977 1079 1082 1227 1239 1302 1370 1373 1420 1505 1546 1596 1710 1828 1863 1900 1929 1938 1975 2041
11	51	0.919611 (0.717951)	4	3	1 2 4 8 16 32 64 128 256 512 1024 82 95 103 137 170 340 376 415 416 456 486 579 589 647 734 754 811 829 887 964 977 1079 1082 1227 1239 1302 1370 1373 1420 1505 1546 1596 1710 1828 1863 1900 1929 1938 1975 2041
11	52	0.924695 (0.719886)	4	3	1 2 4 8 16 32 64 128 256 512 1024 82 95 103 137 170 340 376 415 416 456 486 579 589 647 734 754 811 829 887 964 977 1079 1082 1227 1239 1302 1337 1370 1373 1420 1505 1546 1596 1710 1828 1863 1900 1929 1938 1975 2041
11	53	0.929476 (0.721705)	4	3	1 2 4 8 16 32 64 128 256 512 1024 82 95 103 137 170 340 376 415 416 456 486 579 589 647 734 754 811 829 842 887 964 977 1079 1082 1227 1239 1302 1337 1370 1373 1420 1505 1546 1596 1710 1828 1863 1900 1929 1938 1975 2041
11	54	0.933982 (0.723311)	4	3	1 2 4 8 16 32 64 128 256 512 1024 103 137 170 181 340 357 376 415 416 456 486 529 568 579 647 734 754 811 829 887 940 964 977 1079 1082 1227 1239 1332 1339 1358 1370 1420 1546 1596 1710 1828 1863 1877 1889 1929 1938 1975 2041
11	55	0.938224 (0.724755)	4	3	1 2 4 8 16 32 64 128 256 512 1024 103 118 137 170 181 340 376 415 416 456 486 529 568 579 647 734 754 811 829 887 940 964 977 1079 1082 1227 1239 1262 1332 1339 1370 1420 1503 1546 1596 1710 1828 1863 1877 1889 1929 1938 1975 2041
11	56	0.942226 (0.726067)	4	3	1 2 4 8 16 32 64 128 256 512 1024 103 118 137 170 181 340 376 415 416 456 486 529 568 579 647 734 754 811 829 887 940 964 977 1079 1082 1227 1239 1262 1332 1339 1370 1420 1503 1546 1596 1710 1728 1828 1863 1877 1889 1929 1938 1975 2041
11	57	0.945978 (0.727336)	4	3	1 2 4 8 16 32 64 128 256 512 1024 103 137 170 181 340 357 376 415 416 456 475 486 529 568 579 589 647 734 754 811 829 887 940 964 977 1018 1079 1082 1227 1239 1332 1339 1358 1370 1420 1546 1596 1710 1828 1863 1877 1889 1929 1938 1975 2041
11	58	0.949509 (0.728485)	4	3	1 2 4 8 16 32 64 128 256 512 1024 57 103 118 137 170 181 340 376 415 416 456 486 529 568 579 585 647 734 754 811 829 887 940 964 977 1079 1082 1105 1227 1239 1332 1339 1370 1420 1546 1596 1623 1710 1753 1828 1863 1877 1889 1929 1938 1975 2041
11	59	0.952823 (0.729528)	4	3	1 2 4 8 16 32 64 128 256 512 1024 57 103 118 137 170 181 340 376 415 416 456 486 529 568 579 585 647 734 754 811 829 887 940 964 977 1079 1082 1105 1227 1239 1332 1339 1370 1420 1546 1596 1623 1710 1753 1828 1863 1877 1889 1929 1938 1975 1978 2041
11	60	0.955931 (0.730463)	4	3	1 2 4 8 16 32 64 128 256 512 1024 47 103 137 170 181 340 357 376 415 416 456 475 486 529 568 579 589 647 734 754 811 829 887 940 964 977 1018 1079 1082 1227 1239 1287 1332 1339 1358 1370 1420 1546 1596 1604 1710 1828 1863 1877 1889 1929 1938 1975 2041
11	61	0.958863 (0.731332)	4	3	1 2 4 8 16 32 64 128 256 512 1024 103 137 170 181 340 357 376 415 416 456 475 486 529 568 579 589 610 647 734 754 811 829 887 940 964 977 1018 1079 1082 1227 1239 1332 1339 1358 1370 1420 1546 1596 1604 1710 1714 1828 1863 1877 1889 1925 1929 1938 1975 2041
11	62	0.961617 (0.732129)	4	3	1 2 4 8 16 32 64 128 256 512 1024 47 103 137 170 181 267 340 357 376 415 416 456 475 486 529 568 579 589 647 734 754 811 829 887 940 964 977 1018 1079 1082 1227 1239 1332 1339 1358 1370 1394 1420 1546 1596 1604 1710 1747 1828 1863 1877 1889 1929 1938 1975 2041

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
11	63	0.964190 (0.732861)	4	3	1 2 4 8 16 32 64 128 256 512 1024 47 103 137 170 181 267 340 357 376 415 416 456 475 486 529 568 579 589 647 734 754 811 829 887 940 964 977 1018 1079 1082 1227 1239 1332 1339 1358 1370 1394 1420 1546 1596 1604 1710 1747 1828 1863 1877 1889 1929 1938 1968 1975 2041
11	64	0.966606 (0.733520)	4	3	1 2 4 8 16 32 64 128 256 512 1024 47 103 137 170 181 267 337 340 357 376 415 416 456 475 486 529 568 579 589 647 734 754 811 829 887 940 964 977 1018 1079 1082 1227 1239 1332 1339 1358 1370 1394 1420 1546 1596 1604 1710 1747 1828 1863 1877 1889 1929 1938 1968 1975 2041
11	65	0.968862 (0.734121)	4	3	1 2 4 8 16 32 64 128 256 512 1024 103 137 170 181 337 340 357 376 415 416 456 475 486 529 568 579 589 610 647 734 754 811 829 887 940 964 977 1018 1079 1082 1227 1239 1287 1332 1339 1358 1370 1420 1546 1596 1604 1615 1643 1710 1714 1828 1863 1877 1889 1925 1929 1938 1975 2041
11	66	0.970987 (0.749506)	4	3	1 2 4 8 16 32 64 128 256 512 1024 47 103 137 170 181 267 340 357 376 415 416 456 475 486 529 568 579 589 647 734 754 811 829 887 940 964 977 1018 1079 1082 1115 1227 1239 1332 1339 1353 1358 1370 1394 1420 1546 1596 1604 1661 1710 1747 1749 1828 1863 1877 1889 1929 1938 1975 2041
11	67	0.972972 (0.760622)	4	3	1 2 4 8 16 32 64 128 256 512 1024 47 103 137 170 181 246 337 340 357 376 415 416 456 475 486 529 568 579 589 647 734 754 811 829 841 887 940 964 977 1018 1079 1082 1227 1239 1287 1297 1332 1339 1358 1370 1373 1420 1546 1596 1604 1710 1828 1838 1863 1877 1889 1925 1929 1938 1975 2041
11	68	0.974832 (0.761144)	4	3	1 2 4 8 16 32 64 128 256 512 1024 47 103 137 170 181 246 267 340 357 376 415 416 456 475 486 529 568 579 589 647 734 754 800 811 829 887 940 964 977 1018 1079 1082 1115 1227 1239 1332 1339 1353 1358 1370 1394 1420 1546 1596 1604 1661 1710 1747 1749 1828 1863 1877 1889 1929 1938 1975 2041
11	69	0.976573 (0.769587)	4	3	1 2 4 8 16 32 64 128 256 512 1024 47 103 137 170 181 246 267 340 357 376 415 416 456 475 486 529 568 579 589 647 734 754 800 811 829 887 940 964 977 1018 1079 1082 1115 1227 1239 1320 1332 1339 1353 1358 1370 1394 1420 1546 1596 1604 1661 1710 1747 1749 1828 1863 1877 1889 1929 1938 1975 2041
11	70	0.978196 (0.770032)	4	3	1 2 4 8 16 32 64 128 256 512 1024 47 103 137 170 181 246 267 340 357 376 415 416 456 475 486 529 568 579 589 647 734 754 800 811 829 887 940 964 977 1018 1079 1082 1115 1142 1227 1239 1320 1332 1339 1353 1358 1370 1394 1420 1546 1596 1604 1661 1710 1747 1749 1828 1863 1877 1889 1929 1938 1975 2041
11	71	0.979715 (0.770448)	4	3	1 2 4 8 16 32 64 128 256 512 1024 47 103 137 170 181 246 253 267 340 357 376 415 416 456 475 486 529 568 579 589 647 734 754 811 829 887 940 964 977 1018 1079 1082 1115 1227 1239 1320 1332 1339 1353 1358 1370 1394 1404 1420 1546 1556 1596 1604 1661 1710 1747 1749 1828 1863 1877 1889 1929 1938 1975 2041
11	72	0.981130 (0.776960)	4	3	1 2 4 8 16 32 64 128 256 512 1024 47 103 137 170 181 246 253 267 340 357 376 415 416 456 475 486 529 568 579 589 647 734 754 811 829 887 940 964 977 1018 1079 1082 1115 1142 1227 1239 1309 1320 1332 1339 1353 1358 1370 1394 1420 1546 1556 1596 1604 1661 1710 1747 1749 1828 1863 1877 1889 1929 1938 1975 2041
11	73	0.982457 (0.782101)	4	3	1 2 4 8 16 32 64 128 256 512 1024 47 103 137 170 181 246 253 267 340 357 376 415 416 456 475 486 529 568 579 589 608 647 734 754 811 829 887 940 964 977 1018 1079 1082 1115 1227 1239 1309 1320 1332 1339 1353 1358 1370 1394 1420 1453 1546 1556 1596 1604 1661 1710 1747 1749 1828 1863 1877 1889 1929 1938 1975 2041
11	74	0.983696 (0.786203)	4	3	1 2 4 8 16 32 64 128 256 512 1024 47 103 137 170 181 246 253 267 340 357 376 415 416 456 475 486 529 568 579 589 608 647 734 754 811 829 887 940 964 977 1018 1079 1082 1115 1227 1239 1309 1320 1332 1339 1353 1358 1370 1394 1420 1533 1546 1556 1596 1604 1661 1710 1747 1749 1828 1863 1877 1889 1925 1929 1938 1975 2041
11	75	0.984852 (0.789478)	4	3	1 2 4 8 16 32 64 128 256 512 1024 47 103 137 170 181 246 253 267 337 340 357 376 415 416 456 475 486 529 568 579 589 608 647 734 754 811 829 887 940 964 977 1018 1079 1082 1115 1227 1239 1309 1320 1332 1339 1353 1358 1370 1394 1420 1533 1546 1556 1596 1604 1661 1710 1747 1749 1828 1863 1877 1889 1925 1929 1938 1975 2041
11	76	0.985929 (0.792140)	4	3	1 2 4 8 16 32 64 128 256 512 1024 47 103 137 170 181 246 253 267 337 340 357 376 415 416 456 475 486 529 568 579 589 608 647 734 754 811 829 887 940 964 977 1018 1079 1082 1115 1227 1239 1309 1320 1332 1339 1353 1358 1370 1394 1404 1420 1533 1546 1556 1596 1604 1661 1710 1747 1749 1828 1863 1877 1889 1925 1929 1938 1975 2041
11	77	0.986936 (0.794298)	4	3	1 2 4 8 16 32 64 128 256 512 1024 47 103 137 170 181 246 253 267 337 340 357 376 415 416 456 475 486 529 568 579 589 608 647 667 734 754 811 829 887 940 964 977 1018 1079 1082 1115 1227 1239 1309 1320 1332 1339 1347 1353 1358 1370 1394 1420 1453 1546 1556 1596 1604 1661 1710 1747 1749 1828 1863 1877 1889 1925 1929 1938 1975 2041
11	78	0.987877 (0.796060)	4	3	1 2 4 8 16 32 64 128 256 512 1024 47 103 137 170 181 246 253 267 337 340 357 376 415 416 456 475 486 529 568 579 589 608 647 667 734 754 811 829 887 940 964 977 1018 1079 1082 1115 1227 1239 1279 1309 1320 1332 1339 1347 1353 1358 1370 1394 1420 1453 1546 1556 1596 1604 1661 1710 1747 1749 1828 1863 1877 1889 1925 1929 1938 1975 2041
11	79	0.988754 (0.797503)	4	3	1 2 4 8 16 32 64 128 256 512 1024 47 103 137 170 181 246 253 267 337 340 357 376 415 416 456 475 486 529 568 579 589 608 647 667 734 754 811 829 887 940 964 977 1018 1079 1082 1115 1227 1239 1279 1309 1320 1332 1339 1347 1353 1358 1370 1394 1420 1453 1546 1556 1596 1604 1661 1710 1747 1749 1822 1828 1863 1877 1889 1925 1929 1938 1975 2041
12	13	0.310250	13	6	1 2 4 8 16 32 64 128 256 512 1024 2048 1023
12	14	0.334077 (0.334074)	9	6	1 2 4 8 16 32 64 128 256 512 1024 2048 951 3423
12	15	0.357807 (0.357727)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 951 1757 2159
12	16	0.381489 (0.381319)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 951 1487 2796 3191
12	17	0.405042 (0.404862)	8	6	1 2 4 8 16 32 64 128 256 512 1024 2048 415 743 1513 2414 3891
12	18	0.428189 (0.428187)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 415 1845 3119 3414 3731 3812
12	19	0.451301 (0.451295)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 415 878 1259 1629 1976 3731 3812
12	20	0.474157 (0.474151)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 415 878 1259 1629 1976 3399 3731 3812
12	21	0.496721 (0.496716)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 415 1507 1845 2653 2985 3119 3414 3731 3812
12	22	0.518970	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 415 878 1845 2298 2653 2985 3119 3414 3731 3812
12	23	0.540878	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 415 878 1259 1629 1976 2546 2869 3017 3399 3731 3812
12	24	0.562346 (0.562347)	8	4	1 2 4 8 16 32 64 128 256 512 1024 2048 415 878 1259 1629 1976 2546 2869 3017 3134 3399 3731 3812
12	25	0.581271 (0.571447)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 415 878 1259 1629 1976 1983 2546 2869 3017 3134 3399 3731 3812
12	26	0.599934 (0.590553)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 415 729 743 1205 1919 1961 2004 2361 2420 2602 2893 3119 3697 4035
12	27	0.618538 (0.608983)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 214 415 814 1206 1677 1845 1992 2243 2355 2964 2985 3119 3414 3731 3812
12	28	0.636665 (0.626345)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 214 415 638 814 1116 1206 1677 1845 2243 2355 2964 2985 3119 3414 3731 3812
12	29	0.654282 (0.643048)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 214 415 460 814 1206 1677 1803 1845 2030 2243 2355 2964 2985 3119 3414 3731 3812
12	30	0.671260 (0.659303)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 214 415 814 1116 1206 1677 1845 2243 2355 2964 2985 3119 3414 3425 3482 3731 3812 3819
12	31	0.687599 (0.674851)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 214 415 814 1116 1206 1677 1845 2243 2355 2964 2985 3119 3414 3425 3482 3731 3812 3819 3896
12	32	0.703326 (0.689785)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 205 214 415 638 814 1116 1206 1677 1845 2150 2243 2355 2964 2985 3119 3414 3731 3812 3978 4007

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
12	33	0.718548 (0.704115)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 214 415 638 733 793 814 1116 1206 1223 1677 1845 2243 2355 2964 2985 3119 3414 3731 3812 3819 4007
12	34	0.733099 (0.717829)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 214 415 638 733 793 814 1116 1206 1223 1677 1845 2243 2355 2964 2985 3119 3414 3731 3812 3819 3904 4007
12	35	0.747104 (0.743343)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 214 415 638 733 793 814 998 1116 1206 1223 1257 1677 1845 2243 2355 2964 2985 3119 3414 3731 3812 3819 4007
12	36	0.760427 (0.758015)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 214 415 638 733 793 814 1004 1116 1206 1223 1677 1845 1930 2243 2355 2964 2985 3119 3414 3731 3812 3819 3904 4007
12	37	0.773184 (0.771753)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 214 415 638 733 793 814 883 998 1116 1206 1223 1257 1390 1677 1845 2243 2355 2964 2985 3119 3414 3731 3812 3819 4007
12	38	0.785441 (0.784715)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 205 214 415 495 638 754 814 1116 1206 1539 1677 1845 2164 2243 2355 2894 2964 2985 3119 3414 3583 3731 3812 3819 3978 4007
12	39	0.797169 (0.796947)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 43 214 415 638 754 814 1116 1206 1603 1677 1845 2164 2243 2355 2894 2920 2964 2985 3119 3322 3400 3414 3731 3812 3819 3978 4007
12	40	0.808547	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 93 478 542 762 886 956 1084 1159 1389 1524 1618 1772 1912 1999 2168 2217 2882 2893 2994 3048 3170 3236 3286 3382 3523 3544 3824 3847
12	41	0.819566	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 93 478 542 762 886 956 1084 1159 1389 1524 1618 1772 1821 1912 1999 2168 2217 2882 2893 2994 3048 3170 3236 3286 3382 3523 3544 3824 3847
12	42	0.830043	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 93 313 431 478 542 762 886 956 1084 1159 1389 1524 1618 1772 1821 1912 1999 2168 2217 2882 2893 2994 3048 3170 3236 3286 3382 3523 3544 3824 3847
12	43	0.839985	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 93 313 431 478 542 762 886 956 1084 1159 1389 1524 1618 1772 1821 1912 1999 2168 2217 2882 2893 2994 3048 3170 3236 3286 3382 3523 3544 3824 3847 4011
12	44	0.849432	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 93 313 431 478 542 762 886 956 1084 1159 1389 1524 1618 1772 1821 1912 1999 2168 2217 2882 2893 2994 3048 3170 3236 3286 3382 3523 3544 3824 3847 4011
12	45	0.858419	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 93 313 431 478 542 762 886 956 1084 1159 1389 1524 1618 1772 1821 1912 1999 2168 2217 2882 2893 2994 3048 3170 3236 3286 3382 3523 3544 3824 3847 4011
12	46	0.86694	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 93 313 431 478 542 762 886 956 1084 1159 1367 1389 1524 1618 1772 1821 1912 1999 2168 2217 2882 2893 2994 3048 3145 3170 3236 3286 3382 3523 3544 3824 3847 4011
12	47	0.875018	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 93 313 431 478 542 762 886 956 1084 1159 1367 1389 1524 1618 1639 1772 1821 1912 1999 2168 2217 2882 2893 2994 3048 3145 3170 3236 3286 3382 3523 3544 3824 3847 4011
12	48	0.882682	6	4	1 2 4 8 16 32 64 128 256 512 1024 2048 93 313 431 478 542 762 886 956 1084 1159 1367 1389 1524 1618 1639 1772 1821 1912 1999 2168 2217 2414 2882 2893 2994 3048 3145 3170 3236 3286 3382 3523 3544 3824 3847 4011
12	49	0.887425 (0.885946)	4	4	1 2 4 8 16 32 64 128 256 512 1024 2048 108 214 225 327 340 415 460 752 777 814 978 998 1059 1116 1206 1361 1450 1643 1677 1845 2243 2355 2456 2554 2583 2964 2985 3119 3414 3467 3668 3731 3802 3812 3819 3889 3950
12	50	0.894111 (0.892828)	4	4	1 2 4 8 16 32 64 128 256 512 1024 2048 108 214 225 327 415 460 777 814 862 939 978 998 1059 1116 1206 1450 1562 1663 1677 1845 2243 2355 2456 2554 2951 2964 2985 3119 3414 3467 3731 3802 3812 3819 3907 3928 3950 4049
12	51	0.900428 (0.899396)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 108 214 225 327 415 460 777 814 862 939 978 998 1059 1116 1206 1450 1606 1663 1677 1845 2243 2355 2456 2554 2951 2964 2985 3119 3414 3467 3582 3731 3802 3812 3819 3907 3928 3950 4049
12	52	0.906410 (0.905576)	4	4	1 2 4 8 16 32 64 128 256 512 1024 2048 108 214 225 327 415 460 533 777 814 978 998 1059 1116 1206 1405 1458 1663 1677 1740 1845 1871 2190 2243 2355 2424 2456 2554 2876 2915 2964 2985 3119 3414 3467 3731 3802 3812 3819 3888 3950
12	53	0.912066 (0.911415)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 108 214 225 327 415 460 777 814 862 939 978 998 1059 1116 1206 1450 1562 1663 1677 1845 2173 2243 2334 2355 2456 2554 2666 2951 2964 2985 3119 3140 3414 3467 3731 3802 3812 3819 3928 3950 4049
12	54	0.917417 (0.916928)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 108 214 225 327 415 460 465 777 814 862 939 978 998 1059 1116 1206 1450 1562 1663 1677 1845 2173 2243 2334 2355 2456 2554 2666 2951 2964 2985 3119 3140 3414 3467 3731 3802 3812 3819 3928 3950 4049
12	55	0.922450 (0.922140)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 108 214 225 327 415 460 465 550 777 814 862 939 978 998 1059 1116 1206 1450 1562 1663 1677 1845 2173 2243 2334 2355 2456 2554 2666 2951 2964 2985 3119 3140 3414 3467 3731 3802 3812 3819 3928 3950 4049
12	56	0.927198 (0.927044)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 108 214 225 327 415 460 465 550 777 814 862 939 978 998 1059 1116 1176 1206 1450 1562 1663 1677 1845 2173 2243 2334 2355 2456 2554 2666 2951 2964 2985 3119 3140 3414 3467 3731 3802 3812 3819 3928 3950 4049
12	57	0.931678 (0.931665)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 108 214 225 327 415 460 465 550 777 814 862 939 978 998 1059 1116 1176 1206 1450 1562 1663 1677 1839 1845 2173 2243 2334 2355 2456 2554 2666 2951 2964 2985 3119 3140 3414 3467 3731 3802 3812 3819 3928 3950 4049
12	58	0.936020	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 309 345 549 618 657 687 690 717 963 981 1037 1098 1207 1236 1314 1374 1380 1395 1409 1434 1495 1653 1897 1926 1962 2074 2196 2327 2373 2414 2472 2549 2628 2639 2748 2760 2790 2809 2818 2868 2990 3306 3721 3794 3852 3924
12	59	0.940112	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 309 345 549 618 657 687 690 717 963 981 1037 1098 1207 1236 1314 1374 1380 1395 1409 1434 1495 1653 1897 1926 1962 2074 2196 2327 2373 2414 2472 2549 2628 2639 2748 2760 2790 2809 2818 2868 2990 3306 3721 3794 3852 3877 3924
12	60	0.943961	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 309 345 549 618 657 687 690 717 963 981 1037 1098 1207 1236 1314 1374 1380 1395 1409 1434 1495 1653 1897 1926 1962 2074 2196 2327 2373 2414 2472 2549 2628 2639 2748 2760 2790 2809 2818 2868 2990 3132 3306 3721 3794 3852 3877 3924
12	61	0.947587	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 309 345 549 618 657 687 690 717 963 981 1037 1098 1207 1236 1314 1374 1380 1395 1409 1434 1495 1566 1653 1897 1926 1962 2074 2196 2327 2373 2414 2472 2549 2628 2639 2748 2760 2790 2809 2818 2868 2990 3132 3306 3721 3794 3852 3877 3924
12	62	0.950995	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 309 345 549 618 657 687 690 717 783 963 981 1037 1098 1207 1236 1314 1374 1380 1395 1409 1434 1495 1566 1653 1897 1926 1962 2074 2196 2327 2373 2414 2472 2549 2628 2639 2748 2760 2790 2809 2818 2868 2990 3132 3306 3721 3794 3852 3877 3924
12	63	0.954205	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 309 345 549 618 657 687 690 717 783 963 981 1037 1098 1207 1236 1314 1374 1380 1395 1409 1434 1495 1566 1653 1897 1926 1962 2074 2196 2327 2373 2414 2472 2549 2601 2628 2639 2748 2760 2790 2809 2818 2868 2990 3132 3306 3721 3794 3852 3877 3924
12	64	0.957220	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 309 345 549 618 657 687 690 717 783 963 981 1037 1098 1207 1236 1314 1374 1380 1395 1409 1434 1495 1566 1653 1897 1926 1962 2074 2196 2327 2373 2414 2472 2549 2601 2628 2639 2748 2760 2790 2809 2818 2868 2990 3132 3306 3721 3770 3794 3852 3877 3924

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
12	65	0.960062	5	3	1 2 4 8 16 32 64 128 256 512 1024 2048 309 345 549 618 657 687 690 717 783 963 981 1037 1098 1207 1236 1314 1374 1380 1395 1409 1434 1495 1566 1653 1885 1897 1926 1962 2074 2196 2327 2373 2414 2472 2549 2601 2628 2639 2748 2760 2790 2809 2818 2868 2990 3132 3306 3721 3770 3794 3852 3877 3924
12	66	0.962073 (0.710369)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 108 214 225 327 415 460 465 550 624 777 814 862 939 978 998 1059 1116 1206 1220 1450 1531 1562 1608 1633 1663 1677 1845 2173 2243 2264 2334 2340 2355 2456 2554 2561 2666 2951 2964 2985 3037 3119 3140 3365 3414 3467 3731 3802 3812 3819 3928 3950 4049 4063
12	67	0.964528 (0.720591)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 108 214 225 327 415 460 465 550 624 777 814 862 939 978 998 1059 1116 1206 1220 1376 1450 1531 1562 1608 1633 1663 1677 1845 2173 2243 2264 2334 2355 2456 2554 2561 2666 2951 2964 2985 3095 3119 3140 3365 3414 3467 3731 3802 3812 3819 3928 3940 3950 4049 4063
12	68	0.966833 (0.721112)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 108 214 225 293 327 415 460 465 550 624 777 814 862 939 978 998 1059 1116 1206 1220 1376 1450 1531 1562 1608 1633 1663 1677 1845 2173 2243 2264 2334 2355 2456 2554 2561 2666 2951 2964 2985 3095 3119 3140 3365 3414 3467 3731 3802 3812 3819 3928 3940 3950 4049 4063
12	69	0.969001 (0.728867)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 108 214 225 327 415 460 465 550 624 777 814 862 939 978 998 1059 1116 1206 1220 1450 1531 1562 1593 1608 1633 1663 1677 1845 2173 2243 2264 2334 2340 2355 2456 2554 2561 2666 2791 2951 2964 2985 3037 3119 3140 3365 3414 3467 3651 3731 3802 3812 3819 3928 3950 4049 4063
12	70	0.971041 (0.729323)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 85 108 214 225 327 415 460 465 550 624 777 814 862 939 978 998 1055 1059 1116 1206 1220 1450 1531 1562 1608 1633 1663 1677 1746 1845 2173 2243 2264 2334 2340 2355 2456 2554 2561 2666 2951 2964 2985 3037 3119 3140 3365 3414 3467 3651 3731 3802 3812 3819 3928 3950 4049 4063
12	71	0.972963 (0.729743)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 85 108 214 225 327 374 415 460 465 550 624 777 814 862 939 978 998 1055 1059 1116 1206 1220 1450 1481 1531 1562 1608 1633 1663 1677 1746 1845 2173 2243 2264 2334 2340 2355 2456 2554 2561 2666 2951 2964 2985 3037 3119 3140 3365 3414 3467 3731 3802 3812 3819 3928 3950 4049 4063
12	72	0.974773 (0.730105)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 85 108 207 214 225 327 374 415 460 465 550 624 777 814 862 939 978 998 1055 1059 1116 1206 1220 1450 1481 1531 1562 1608 1633 1663 1677 1845 2173 2243 2264 2334 2340 2355 2456 2554 2561 2666 2678 2951 2964 2985 3037 3119 3140 3365 3414 3467 3731 3802 3812 3819 3928 3950 4049 4063
12	73	0.976457 (0.736082)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 85 108 207 214 225 327 374 415 460 465 550 624 777 814 862 939 978 998 1055 1059 1116 1206 1220 1450 1481 1531 1562 1608 1633 1663 1677 1746 1845 2173 2243 2264 2334 2340 2355 2456 2554 2561 2666 2678 2951 2964 2985 3037 3119 3140 3365 3414 3467 3731 3802 3812 3819 3928 3950 4049 4063
12	74	0.978034 (0.740795)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 85 108 214 225 327 374 415 460 465 550 624 777 814 862 939 978 998 1055 1059 1116 1206 1220 1450 1481 1531 1562 1608 1633 1663 1677 1746 1845 1990 2173 2243 2264 2334 2340 2355 2456 2554 2561 2666 2678 2951 2964 2985 3037 3119 3140 3365 3414 3467 3731 3802 3812 3819 3928 3950 4049 4063 4083
12	75	0.979512 (0.744544)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 85 108 214 225 327 374 415 460 465 550 624 744 777 814 862 939 978 998 1055 1059 1116 1206 1220 1450 1481 1531 1562 1608 1633 1663 1677 1746 1845 1990 2173 2243 2264 2334 2340 2355 2456 2554 2561 2666 2678 2951 2964 2985 3037 3119 3140 3365 3414 3467 3731 3802 3812 3819 3928 3950 4049 4063 4083
12	76	0.980897 (0.747572)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 85 108 214 225 327 374 415 460 465 550 624 744 777 814 862 939 978 998 1055 1059 1116 1206 1220 1450 1481 1531 1562 1608 1633 1663 1677 1746 1845 1990 2173 2243 2264 2334 2340 2355 2456 2554 2561 2666 2678 2951 2964 2985 3037 3119 3140 3365 3414 3467 3731 3802 3812 3819 3928 3940 3950 4049 4063 4083
12	77	0.982193 (0.750003)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 85 108 214 225 327 374 415 460 465 550 624 744 777 814 862 939 978 998 1055 1059 1116 1206 1220 1450 1481 1531 1562 1608 1633 1663 1677 1746 1845 1990 2173 2243 2264 2334 2340 2355 2456 2554 2561 2666 2678 2951 2964 2985 3037 3119 3140 3365 3414 3467 3651 3731 3802 3812 3819 3928 3940 3950 4049 4063 4083
12	78	0.983409 (0.751978)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 85 108 214 225 327 374 415 460 465 550 624 744 777 814 862 939 978 998 1055 1059 1116 1206 1220 1450 1481 1531 1562 1608 1633 1663 1677 1746 1845 1990 2173 2243 2264 2334 2340 2355 2456 2554 2561 2666 2678 2951 2964 2985 3037 3119 3140 3280 3365 3414 3467 3651 3731 3802 3812 3819 3928 3940 3950 4049 4063 4083
12	79	0.984547 (0.753588)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 85 108 214 225 327 374 415 460 465 550 624 744 777 814 862 939 978 998 1055 1059 1116 1206 1220 1450 1481 1531 1562 1608 1633 1663 1677 1746 1845 1951 1990 2173 2243 2264 2334 2340 2355 2456 2554 2561 2666 2678 2951 2964 2985 3037 3119 3140 3280 3365 3414 3467 3651 3731 3802 3812 3819 3928 3940 3950 4049 4063 4083
12	80	0.985609 (0.754912)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 85 108 214 225 297 327 374 415 460 465 550 624 777 789 814 862 939 978 998 1055 1059 1116 1206 1220 1450 1481 1531 1562 1608 1633 1663 1677 1746 1845 1953 1990 2173 2243 2264 2334 2340 2355 2456 2554 2561 2666 2678 2951 2964 2985 3037 3042 3119 3140 3318 3365 3414 3418 3467 3731 3802 3812 3819 3928 3940 3950 4049 4063
12	81	0.986605 (0.756012)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 85 108 214 225 327 374 415 460 465 550 624 744 777 814 862 939 978 998 1055 1059 1116 1206 1220 1450 1481 1531 1550 1562 1608 1633 1663 1677 1746 1845 1891 1990 2173 2243 2264 2334 2340 2355 2456 2554 2561 2666 2671 2678 2951 2964 2985 3037 3119 3140 3280 3365 3414 3467 3651 3731 3802 3812 3819 3928 3940 3950 4049 4063 4083
12	82	0.987536 (0.756908)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 85 108 214 225 327 374 415 460 465 550 624 744 777 814 862 939 978 998 1055 1059 1116 1206 1220 1450 1481 1531 1550 1562 1608 1633 1663 1677 1746 1845 1891 1990 2173 2243 2264 2334 2340 2355 2395 2456 2554 2561 2666 2671 2678 2951 2964 2985 3037 3119 3140 3280 3365 3414 3467 3651 3731 3802 3812 3819 3928 3940 3950 4049 4063 4083
12	83	0.988407 (0.757659)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 85 108 214 225 327 374 415 460 465 550 624 744 777 814 862 939 978 998 1055 1059 1116 1206 1220 1286 1450 1481 1531 1550 1562 1608 1633 1663 1677 1746 1845 1891 1990 2173 2243 2264 2334 2340 2355 2395 2456 2554 2561 2666 2671 2678 2951 2964 2985 3037 3119 3140 3280 3365 3414 3467 3651 3731 3802 3812 3819 3928 3940 3950 4049 4063 4083
12	84	0.989219 (0.758284)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 85 108 214 225 327 374 415 460 465 550 624 744 777 814 862 939 978 998 1055 1059 1116 1206 1220 1286 1450 1481 1531 1550 1562 1608 1633 1663 1677 1746 1845 1891 1990 2173 2243 2264 2334 2340 2355 2395 2456 2554 2561 2666 2671 2678 2915 2951 2964 2985 3037 3119 3140 3280 3365 3414 3467 3651 3731 3802 3812 3819 3928 3940 3950 4049 4063 4083
12	85	0.989976 (0.758800)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 85 108 214 225 289 327 374 415 460 465 550 624 744 777 814 862 939 954 978 998 1055 1059 1116 1206 1220 1286 1450 1481 1531 1550 1562 1608 1633 1663 1677 1746 1845 1891 1990 2173 2243 2264 2334 2340 2355 2395 2456 2554 2561 2666 2671 2678 2951 2964 2985 3037 3119 3140 3280 3365 3414 3467 3651 3731 3802 3812 3819 3928 3940 3950 4049 4063 4083
12	86	0.990684 (0.759244)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 85 108 214 225 289 327 374 415 460 465 550 624 744 777 814 862 939 954 978 998 1055 1059 1116 1206 1220 1262 1286 1450 1481 1531 1550 1562 1608 1633 1663 1677 1746 1845 1891 1990 2173 2243 2264 2334 2340 2355 2395 2456 2554 2561 2666 2671 2678 2951 2964 2985 3037 3119 3140 3280 3365 3414 3467 3651 3731 3802 3812 3819 3928 3940 3950 4049 4063 4083

Continued on next page

Table B.1 – Continued from previous page					
m	n	R_e	d	R_c	Packed integer form of the parity check matrix
12	87	0.991350 (0.759610)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 85 108 214 225 289 327 374 415 460 465 550 624 744 777 814 862 939 954 978 998 1055 1059 1116 1206 1220 1262 1286 1450 1481 1531 1550 1562 1608 1633 1663 1677 1715 1746 1845 1891 1990 2173 2243 2264 2334 2340 2355 2395 2456 2554 2561 2666 2671 2678 2951 2964 2985 3037 3119 3140 3280 3365 3414 3467 3651 3731 3802 3812 3819 3928 3940 3950 4049 4063 4083
12	88	0.991966 (0.759915)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 11 85 108 214 225 289 327 374 415 460 465 550 624 744 777 814 862 939 954 978 998 1055 1059 1116 1206 1220 1262 1286 1450 1481 1531 1550 1562 1608 1633 1663 1677 1715 1746 1845 1891 1990 2173 2243 2264 2334 2340 2355 2395 2456 2554 2561 2666 2671 2678 2951 2964 2985 3037 3119 3140 3280 3365 3414 3467 3651 3731 3802 3812 3819 3928 3940 3950 4049 4063 4083
12	89	0.992541	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 11 85 108 214 225 289 327 374 415 460 465 550 624 744 777 814 862 939 954 978 998 1055 1059 1116 1206 1220 1262 1286 1450 1481 1531 1550 1562 1608 1633 1663 1677 1715 1746 1845 1891 1990 2173 2243 2264 2334 2340 2355 2395 2456 2554 2561 2666 2671 2678 2951 2964 2985 3037 3119 3140 3280 3365 3414 3467 3519 3651 3731 3802 3812 3819 3928 3940 3950 4049 4063 4083
13	14	0.308427	14	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8191
13	15	0.330462 (0.330438)	10	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 3679 4605
13	16	0.352405 (0.352234)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 1999 2111 4988
13	17	0.374282 (0.374069)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 463 5949 7153 8006
13	18	0.396103 (0.395755)	8	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 463 5949 7153 7578 8006
13	19	0.417703 (0.417284)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 1999 3770 4988 6317 6411 7257
13	20	0.439206 (0.438609)	8	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 3437 5617 5958 6490 6965 7827 8104
13	21	0.460392 (0.459705)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 1147 1999 2731 4972 5816 6751 8033 8182
13	22	0.481383 (0.480539)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 463 1898 2995 3996 5073 5949 6895 7672 8006
13	23	0.502146 (0.501078)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 463 1898 2995 3429 3996 5073 5949 6895 7672 8006
13	24	0.522597 (0.521287)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 463 1898 2995 3429 3996 5073 5949 6427 6895 7672 8006
13	25	0.542738 (0.541129)	7	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 463 1898 2995 3429 3996 4518 5073 5949 6427 6895 7672 8006
13	26	0.562112 (0.561407)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 914 3437 4347 4493 5617 5806 5958 6490 6774 6965 7247 7827 8104
13	27	0.580624 (0.580619)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 914 3437 4347 4493 4721 5617 5806 5958 6490 6774 6965 7247 7827 8104
13	28	0.598803 (0.584146)	5	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 1147 1999 2556 2731 3523 3865 4507 4972 5413 5816 6751 7372 7466 8033 8182
13	29	0.616471 (0.600094)	5	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 1147 1999 2556 2731 2853 3523 3865 4507 4972 5413 5816 6751 7372 7466 8033 8182
13	30	0.633640 (0.615804)	5	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 1147 1557 1999 2409 2731 3523 3581 3865 4507 4972 5183 5816 6751 7372 7466 8033 8182
13	31	0.650363 (0.630747)	5	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 463 496 1118 1898 2405 2995 3299 3996 4473 5073 5396 5576 5949 6895 7435 7672 7716 8006
13	32	0.666596 (0.645068)	5	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 248 463 667 1118 1898 2405 2995 3299 3996 4150 5073 5201 5576 5949 6895 7435 7672 7716 8006
13	33	0.682319 (0.658811)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 463 1118 1898 2405 2995 3299 3996 4473 5073 5114 5396 5576 5949 6297 6895 7435 7672 7716 7870 8006
13	34	0.697426 (0.672017)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 463 496 1118 1293 1898 2405 2995 3299 3996 4473 5073 5396 5576 5949 6054 6482 6895 7435 7672 7716 8006
13	35	0.712023 (0.695359)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 463 1118 1898 2405 2995 3299 3996 4239 4473 5073 5114 5396 5576 5844 5949 6297 6895 7435 7672 7716 7870 8006
13	36	0.726170 (0.709329)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 463 858 1118 1898 2405 2995 3299 3996 4473 5073 5114 5396 5550 5576 5949 6297 6895 7044 7435 7672 7716 7870 8006
13	37	0.739738 (0.722512)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 463 1118 1898 2405 2995 3299 3655 3996 4473 5073 5114 5396 5550 5576 5876 5949 6297 6895 7044 7435 7672 7716 7870 8006
13	38	0.752845 (0.734897)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 463 1118 1898 2405 2995 3299 3996 4239 4473 4720 5073 5114 5396 5576 5844 5949 6019 6297 6895 7044 7435 7672 7716 7870 8006
13	39	0.765296 (0.746688)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 463 1118 1898 2405 2995 3299 3996 4239 4473 4720 5073 5114 5396 5576 5844 5949 6019 6297 6895 7022 7044 7435 7672 7716 7870 8006
13	40	0.777290 (0.757860)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 413 463 1118 1898 2405 2995 3299 3996 4210 4239 4414 4473 4897 5073 5114 5396 5576 5876 5949 6297 6895 7044 7435 7672 7716 7870 8006
13	41	0.788779 (0.768509)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 413 463 1118 1460 1898 2405 2995 3299 3996 4210 4239 4414 4473 4897 5073 5114 5396 5576 5876 5949 6297 6895 7044 7435 7672 7716 7870 8006
13	42	0.799808 (0.778566)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 413 463 1118 1460 1898 2405 2995 3299 3996 4210 4239 4414 4473 4897 5073 5114 5396 5576 5876 5949 6297 6895 7044 7435 7602 7672 7716 7870 8006
13	43	0.810366 (0.788222)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 413 463 1118 1460 1898 2405 2900 2995 3299 3996 4210 4239 4414 4473 4897 5073 5114 5396 5576 5876 5949 6297 6895 7044 7435 7602 7672 7716 7870 8006
13	44	0.820450 (0.797404)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 413 463 1118 1460 1898 2405 2900 2995 3299 3996 4210 4239 4414 4473 4897 5073 5114 5396 5576 5876 5949 5996 6297 6895 7044 7435 7602 7672 7716 7870 8006
13	45	0.830094 (0.806131)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 110 413 463 1118 1460 1898 2405 2900 2995 3165 3299 3996 4210 4239 4414 4473 4897 5073 5114 5396 5576 5876 5949 5996 6297 6895 7044 7435 7672 7716 7870 8006
13	46	0.839276 (0.814416)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 110 413 463 1118 1460 1898 2405 2900 2995 3165 3299 3996 4210 4239 4414 4473 4897 5073 5114 5396 5576 5876 5949 5996 6297 6895 7044 7252 7435 7672 7716 7870 8006
13	47	0.848016 (0.822280)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 110 413 463 1118 1460 1801 1898 2405 2900 2995 3299 3996 4210 4239 4414 4473 4897 5073 5114 5396 5576 5876 5949 5996 6297 6895 7044 7252 7435 7602 7672 7716 7870 8006
13	48	0.856378 (0.829739)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 233 413 463 1118 1460 1898 1936 2405 2690 2900 2995 3299 3996 4210 4239 4414 4473 4897 5030 5073 5114 5396 5576 5876 5949 6297 6895 7044 7435 7516 7602 7672 7716 7870 8006
13	49	0.864354 (0.864222)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 110 233 413 463 1118 1460 1898 2405 2900 2995 3165 3299 3856 3885 3996 4210 4239 4414 4473 4897 5073 5114 5396 5576 5876 5949 5996 6297 6895 7044 7435 7611 7672 7716 7870 8006
13	50	0.871969	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 309 345 618 687 690 1207 1236 1374 1380 1653 1897 2373 2414 2472 2639 2748 2760 3306 3794 4315 4746 4807 4828 4887 4944 5278 5496 5520 6225 6612 7263 7273 7411 7588 7848 8007 8137
13	51	0.879353	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 309 345 618 687 690 1207 1236 1374 1380 1653 1897 2373 2414 2472 2639 2748 2760 3306 3794 3924 4315 4746 4807 4828 4887 4944 5278 5496 5520 6225 6612 7263 7273 7411 7588 7848 8007 8137

Continued on next page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
13	52	0.886341	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 309 345 618 687 690 1207 1236 1374 1380 1653 1897 1962 2373 2414 2472 2639 2748 2760 3306 3794 3924 4315 4746 4807 4828 4887 4944 5278 5496 5520 6225 6612 7263 7273 7411 7588 7848 8007 8137
13	53	0.892969	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 309 345 618 687 690 981 1207 1236 1374 1380 1653 1897 1962 2373 2414 2472 2639 2748 2760 3306 3794 3924 4315 4746 4807 4828 4887 4944 5278 5496 5520 6225 6612 7263 7273 7411 7588 7848 8007 8137
13	54	0.899263	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 309 345 618 687 690 981 1207 1236 1374 1380 1653 1897 1962 2373 2414 2472 2639 2748 2760 3306 3794 3924 4315 4746 4807 4828 4887 4944 5278 5496 5520 6225 6612 7263 7273 7411 7449 7588 7848 8007 8137
13	55	0.905225	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 309 345 618 687 690 981 1207 1236 1374 1380 1653 1897 1962 2373 2414 2472 2639 2748 2760 3306 3794 3924 4315 4735 4746 4807 4828 4887 4944 5278 5496 5520 6225 6612 7263 7273 7411 7449 7588 7848 8007 8137
13	56	0.910876	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 309 345 618 687 690 981 1207 1236 1374 1380 1653 1897 1962 2373 2414 2472 2639 2748 2760 3306 3794 3924 4315 4735 4746 4807 4828 4887 4944 5278 5496 5520 5580 6225 6612 7263 7273 7411 7449 7588 7848 8007 8137
13	57	0.916237	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 309 345 618 687 690 981 1207 1236 1374 1380 1653 1897 1962 2373 2414 2472 2639 2748 2760 2790 3306 3794 3924 4315 4735 4746 4807 4828 4887 4944 5278 5496 5520 5580 6225 6612 7263 7273 7411 7449 7588 7848 8007 8137
13	58	0.921314	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 309 345 618 687 690 981 1207 1236 1374 1380 1395 1653 1897 1962 2373 2414 2472 2639 2748 2760 2790 3306 3794 3924 4315 4735 4746 4807 4828 4887 4944 5278 5496 5520 5580 6225 6612 7263 7273 7411 7449 7588 7848 8007 8137
13	59	0.926114	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 309 345 618 687 690 981 1207 1236 1374 1380 1395 1653 1897 1962 2373 2414 2472 2639 2748 2760 2790 3306 3794 3924 4315 4735 4746 4807 4828 4887 4944 5278 5496 5520 5580 6225 6612 7263 7273 7411 7449 7588 7754 7848 8007 8137
13	60	0.930657	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 309 345 618 687 690 981 1207 1236 1374 1380 1395 1653 1897 1962 2373 2414 2472 2639 2748 2760 2790 3306 3794 3877 3924 4315 4735 4746 4807 4828 4887 4944 5278 5496 5520 5580 6225 6612 7263 7273 7411 7449 7588 7754 7848 8007 8137
13	61	0.934951	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 309 345 618 687 690 981 1207 1236 1374 1380 1395 1653 1897 1962 2373 2414 2472 2639 2748 2760 2790 3306 3794 3877 3924 4315 4735 4746 4807 4828 4887 4944 5278 5496 5520 5580 6225 6612 7009 7263 7273 7411 7449 7588 7754 7848 8007 8137
13	62	0.939008	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 309 345 618 687 690 981 1207 1236 1374 1380 1395 1653 1897 1962 2373 2414 2472 2639 2748 2760 2790 3306 3794 3877 3924 4315 4419 4735 4746 4807 4828 4887 4944 5278 5496 5520 5580 6225 6612 7009 7263 7273 7411 7449 7588 7754 7848 8007 8137
13	63	0.942838	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 309 345 618 687 690 981 1207 1236 1374 1380 1395 1653 1897 1962 2373 2414 2472 2639 2748 2760 2790 3306 3794 3877 3924 4315 4419 4735 4746 4807 4828 4887 4944 5202 5278 5496 5520 5580 6225 6612 7009 7263 7273 7411 7449 7588 7754 7848 8007 8137
13	64	0.946455	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 309 345 618 687 690 981 1207 1236 1374 1380 1395 1653 1897 1962 2373 2414 2472 2601 2639 2748 2760 2790 3306 3794 3877 3924 4315 4419 4735 4746 4807 4828 4887 4944 5202 5278 5496 5520 5580 6225 6612 7009 7263 7273 7411

142

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
13	77	0.976770	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 79 171 248 542 892 986 1087 1126 1265 1391 1425 1722 1744 1879 1928 2083 2118 2167 2485 2518 2528 2855 2905 3233 3413 3618 3756 3799 4036 4141 4327 4447 4467 4528 4545 4678 4853 4882 4904 5005 5092 5103 5353 5383 5400 5722 5731 5774 6079 6172 6443 6638 6816 6979 7017 7165 7360 7389 7411 7668 7705 7799 7936 8050
13	78	0.978291	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 79 171 248 542 892 986 1087 1126 1265 1391 1425 1722 1744 1879 1928 2083 2118 2167 2485 2518 2528 2855 2905 2962 3233 3413 3618 3756 3799 4036 4141 4327 4447 4467 4528 4545 4678 4853 4882 4904 5005 5092 5103 5353 5383 5400 5722 5731 5774 6079 6172 6443 6638 6816 6979 7017 7165 7360 7389 7411 7668 7705 7799 7936 8050
13	79	0.979724	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 79 171 248 542 892 986 1087 1126 1265 1391 1425 1722 1744 1879 1928 2083 2118 2167 2485 2518 2528 2855 2905 2962 3233 3413 3618 3756 3799 4036 4141 4327 4447 4467 4528 4545 4678 4853 4882 4904 5005 5092 5103 5234 5353 5383 5400 5722 5731 5774 6079 6172 6443 6638 6816 6979 7017 7165 7360 7389 7411 7668 7705 7799 7936 8050
13	80	0.981068	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 79 171 248 542 892 940 986 1087 1126 1265 1391 1425 1722 1744 1879 1928 2083 2118 2167 2485 2518 2528 2855 2905 2962 3233 3413 3618 3756 3799 4036 4141 4327 4447 4467 4528 4545 4678 4853 4882 4904 5005 5092 5103 5234 5353 5383 5400 5722 5731 5774 6079 6172 6443 6638 6816 6979 7017 7165 7360 7389 7411 7668 7705 7799 7936 8050
13	81	0.982336	5	3	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 79 171 248 542 892 940 986 1087 1126 1265 1391 1425 1722 1744 1879 1928 2083 2118 2167 2485 2518 2528 2855 2905 2962 3233 3413 3618 3756 3799 4036 4141 4327 4447 4467 4528 4545 4678 4853 4882 4904 5005 5092 5103 5234 5353 5383 5400 5722 5731 5774 6079 6172 6443 6638 6816 6844 6979 7017 7165 7360 7389 7411 7668 7705 7799 7936 8050
13	82	0.983231 (0.719887)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 61 110 233 413 463 652 923 1118 1306 1346 1460 1715 1841 1898 1936 2099 2354 2387 2405 2462 2577 2681 2721 2794 2893 2900 2995 3258 3299 3649 3878 3996 4210 4239 4414 4429 4458 4473 4713 4738 4878 4897 5032 5073 5114 5396 5418 5576 5642 5876 5895 5949 5996 6297 6357 6630 6722 6895 7044 7335 7435 7540 7602 7672 7716 7870 7933 8006 8140
13	83	0.984344 (0.720720)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 61 110 233 413 463 652 923 1118 1306 1346 1460 1715 1841 1898 1936 2099 2354 2387 2405 2462 2577 2681 2721 2794 2893 2900 2995 3258 3299 3613 3649 3878 3996 4210 4239 4414 4429 4458 4473 4713 4738 4878 4897 5032 5073 5114 5396 5418 5576 5642 5876 5895 5949 5996 6297 6357 6630 6722 6895 7044 7335 7435 7540 7602 7672 7716 7870 7933 8006 8140
13	84	0.985391 (0.721411)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 61 110 233 413 463 652 923 1118 1306 1346 1460 1715 1841 1898 1936 2099 2387 2405 2577 2681 2721 2794 2893 2900 2995 3096 3258 3299 3613 3649 3996 4015 4210 4239 4414 4429 4458 4473 4713 4738 4878 4897 5032 5073 5114 5396 5418 5576 5642 5876 5895 5949 5996 6168 6297 6357 6630 6722 6895 7044 7335 7435 7540 7602 7672 7716 7870 7933 8006 8140
13	85	0.986369 (0.721987)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 61 110 233 413 463 652 923 1118 1306 1346 1460 1715 1841 1898 1936 2099 2387 2405 2548 2577 2681 2721 2794 2842 2893 2900 2995 3096 3258 3299 3613 3649 3996 4015 4210 4239 4414 4429 4458 4473 4713 4738 4878 4897 5032 5073 5114 5396 5418 5576 5642 5876 5895 5949 5996 6168 6297 6357 6630 6722 6895 7044 7335 7435 7540 7602 7672 7716 7870 7933 8006 8140
13	86	0.987287 (0.722468)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 61 110 233 413 463 652 923 1118 1306 1346 1460 1715 1841 1898 1936 2099 2387 2405 2548 2577 2681 2721 2794 2893 2900 2995 3096 3258 3299 3613 3649 3996 4015 4210 4239 4414 4429 4458 4473 4713 4738 4878 4897 5032 5073 5114 5396 5418 5576 5642 5876 5895 5949 5996 6168 6297 6357 6630 6722 6895 7044 7170 7335 7435 7540 7602 7672 7688 7716 7870 7933 8006 8140
13	87	0.988146 (0.722871)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 61 110 233 413 463 652 923 1118 1306 1346 1460 1715 1841 1898 1936 2099 2387 2405 2548 2577 2681 2721 2794 2893 2900 2995 3096 3258 3299 3613 3649 3996 4015 4210 4239 4414 4429 4458 4473 4713 4738 4878 4897 5032 5058 5073 5114 5396 5418 5576 5642 5876 5895 5949 5996 6168 6297 6357 6630 6722 6895 7044 7170 7335 7435 7540 7602 7672 7688 7716 7870 7933 8006 8140
13	88	0.988954 (0.723210)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 61 110 233 413 463 652 923 1031 1118 1306 1346 1460 1715 1841 1898 1936 2099 2354 2387 2405 2462 2577 2681 2721 2794 2893 2900 2995 3258 3299 3613 3649 3878 3996 4210 4239 4414 4429 4458 4473 4713 4738 4875 4878 4897 5032 5073 5114 5396 5418 5576 5642 5876 5895 5949 5996 6297 6357 6630 6722 6895 7044 7152 7170 7335 7435 7540 7602 7672 7716 7870 7933 8006 8008 8140
13	89	0.989706 (0.723500)	4	3	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 61 110 233 413 463 652 923 1031 1118 1306 1346 1460 1715 1841 1898 1936 2086 2099 2354 2387 2405 2462 2577 2681 2721 2794 2893 2900 2995 3258 3299 3613 3649 3878 3996 4210 4239 4414 4429 4458 4473 4713 4738 4875 4878 4897 5032 5073 5114 5161 5396 5418 5576 5642 5876 5895 5949 5996 6297 6357 6630 6722 6895 7044 7152 7335 7435 7540 7602 7672 7716 7870 7933 8006 8008 8140
14	15	0.306854	15	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16383
14	16	0.3273	10	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 15423 16352
14	17	0.347706	9	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 14543 15417 16352
14	18	0.368074 (0.367806)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 5619 7967 9087 11439
14	19	0.388399 (0.387943)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 3870 4066 4975 10967 16129
14	20	0.408610 (0.407914)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 3902 5611 10739 13150 15494 16129
14	21	0.428674 (0.427737)	8	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 3870 4975 5366 8890 9781 10957 16129
14	22	0.448606 (0.447326)	8	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 3902 5487 7882 10621 13150 14535 15036 16129
14	23	0.468312 (0.466672)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 3747 3870 4975 5366 6619 9595 12985 15046 16129
14	24	0.487855 (0.485744)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 3747 3870 4975 5366 6555 9595 10220 12985 14990 16129
14	25	0.507165 (0.504509)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 3561 3870 4975 5306 6877 9523 11956 13266 14563 14712 16129
14	26	0.526228 (0.525901)	8	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 3747 3870 4975 5366 6619 9595 10148 11421 12985 14708 15046 16129
14	27	0.544936 (0.544622)	7	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 3747 3870 4975 5366 6555 9595 10212 11477 13041 14409 14652 14990 16129
14	28	0.563108 (0.563046)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 3747 3870 4975 5366 6555 7010 9595 10212 11477 13041 14409 14652 14990 16129
14	29	0.580770 (0.579950)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 2541 3902 5106 5291 7797 8603 9575 9871 11736 13150 13906 14535 15036 15798 16129
14	30	0.597995 (0.597384)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 894 2239 3372 3834 4367 5609 6694 9125 9771 10194 10611 11463 12310 14923 15260 16129
14	31	0.614551 (0.614415)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 2541 3902 5106 5291 7797 8603 9575 9871 11200 11736 13150 13326 13906 14535 15036 15798 16129

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
14	32	0.630893 (0.606169)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 1903 2553 4807 4978 6833 7557 7706 8435 9814 10732 11031 11428 11837 12510 13759 13762 15640 16352
14	33	0.646817 (0.619510)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 1532 1903 3046 4807 4978 6833 7557 8435 9814 10732 11160 11837 13759 13762 14325 14841 15111 15640 16352
14	34	0.662334 (0.632246)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 1903 3046 4807 4978 5174 5337 5717 6833 7557 8435 9814 10732 11837 12399 13759 13762 14325 14841 15640 16352
14	35	0.677334 (0.653696)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 1903 4807 4978 5174 6833 7557 7706 8435 9814 10732 10864 11031 11428 11837 12347 12510 13759 13762 15498 15640 16352
14	36	0.691924 (0.667090)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 1903 3046 4807 4978 5240 5717 6833 7557 8435 9814 10732 11010 11837 12399 12913 13759 13762 14325 14841 15640 16027 16352
14	37	0.706017 (0.679792)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 1903 2956 3990 4807 4978 6833 7557 7706 8435 9814 10709 10732 11031 11428 11618 11837 12638 13759 13762 15309 15640 16302 16352
14	38	0.719665 (0.691726)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 1262 1903 4807 4978 6833 7557 7706 8435 9707 9814 10709 10732 11031 11428 11837 12347 12638 12683 12904 13759 13762 15640 16302 16352
14	39	0.732800 (0.703057)	5	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 1903 3093 4137 4807 4978 6435 6833 7289 7557 7706 8316 8435 9550 9814 9841 10732 10991 11190 11837 12315 13759 13762 14590 15640 16352
14	40	0.745474 (0.713865)	5	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 1903 3093 4137 4807 4978 5845 6435 6833 7289 7557 7706 8316 8435 9550 9814 9841 10732 10991 11190 11837 12315 13759 13762 14590 15640 16352
14	41	0.757694 (0.724278)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 1262 1903 4807 4978 5760 6833 7557 7706 8435 9351 9707 9814 10709 10732 11031 11428 11837 12347 12638 12683 12904 13759 13762 15539 15640 16302 16352
14	42	0.769466 (0.734080)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 1262 1903 4807 4881 4978 5760 6833 6975 7557 7706 8388 8435 9707 9814 10709 10732 11031 11428 11837 12347 12638 12683 12872 13759 13762 15640 16302 16352
14	43	0.780769 (0.743396)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 1262 1903 4807 4881 4978 5760 6833 6975 7557 7706 8435 9707 9814 10709 10732 11031 11428 11837 12347 12638 12683 12872 13759 13762 14177 14626 15640 16302 16352
14	44	0.791649 (0.752371)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 757 1262 1903 4807 4881 4978 5760 6833 6975 7557 7706 8435 9707 9814 10709 10732 11031 11428 11837 12347 12638 12683 12872 13759 13762 14177 15539 15640 16302 16352
14	45	0.802094 (0.760914)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 1306 1903 2681 3018 3508 3727 4807 4978 5286 6833 7557 7706 8435 9492 9814 10158 10732 10769 10991 11355 11837 12664 13759 13762 13880 14358 15411 15498 15640 15910 16352
14	46	0.812104 (0.769011)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 841 1306 1903 2681 3018 3508 3727 4807 4978 5286 6833 7557 7706 8435 9492 9814 10158 10732 10769 10991 11355 11837 12664 13759 13762 13880 14358 15411 15498 15640 15910 16352
14	47	0.821699 (0.776704)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 1306 1903 2606 3018 3727 4065 4807 4978 5286 6833 7557 7600 7706 8435 9035 9492 9814 10732 10769 10991 11355 11837 12664 13759 13762 13880 14533 14562 15411 15498 15640 15910 16352
14	48	0.830905 (0.784001)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 1903 3093 3574 4807 4978 6435 6833 6849 7289 7557 7570 7706 8316 8435 8999 9448 9550 9589 9814 9841 10689 10732 10991 11190 11837 11904 12315 13002 13759 13762 14590 15640 15648 16352
14	49	0.839684 (0.815367)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 1903 3093 3574 4807 4978 6213 6435 6833 6849 7289 7557 7570 7706 8316 8435 8999 9448 9550 9814 9841 10689 10732 10991 11190 11837 11904 12315 13002 13759 13762 14590 15259 15640 15648 16352
14	50	0.848094 (0.822894)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 659 1306 1903 2017 2420 2681 2774 3018 3727 3848 4807 4978 5286 5755 6833 7557 7706 8435 9814 10158 10732 10769 10991 11355 11837 12664 13759 13762 13880 15043 15411 15498 15640 15910 16068 16352
14	51	0.856126 (0.830118)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 1306 1903 2606 3018 3727 4065 4807 4978 5286 6833 7557 7600 7706 8435 9035 9492 9814 10732 10769 10991 11355 11837 12664 13073 13237 13705 13759 13762 13880 14533 14562 15411 15498 15640 15910 16151 16352
14	52	0.863790 (0.836948)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 1306 1903 2606 3018 3727 4065 4807 4978 5286 6833 7557 7600 7706 8435 9035 9492 9814 10732 10769 10991 11170 11355 11837 12664 13073 13237 13705 13759 13762 13880 14533 14562 15411 15498 15640 15910 16151 16352
14	53	0.871112 (0.843439)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 1306 1903 2606 3018 3727 4065 4807 4978 5286 6833 7557 7600 7706 8435 9035 9492 9814 10732 10769 10991 11170 11355 11837 12664 13073 13237 13705 13759 13762 13880 14533 14562 14958 15411 15498 15640 15910 16151 16352
14	54	0.878092 (0.849595)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 1306 1903 2606 3018 3727 4065 4807 4978 5286 6833 7557 7600 7706 8435 9035 9492 9814 9836 10732 10769 10991 11170 11355 11837 12664 13073 13237 13705 13759 13762 13880 14533 14562 14958 15411 15498 15640 15910 16151 16352
14	55	0.884720 (0.855433)	4	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 1306 1903 2606 3018 3727 4065 4807 4978 5286 6833 7557 7600 7706 7914 8435 9035 9492 9814 9836 10732 10769 10991 11170 11355 11837 12664 13073 13237 13705 13759 13762 13880 14533 14562 14958 15411 15498 15640 15910 16151 16352
14	56	0.891061 (0.860979)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 659 1306 1903 2017 2420 2623 2681 2774 3009 3018 3592 3727 4807 4978 5102 5286 5495 5755 6833 7003 7557 7706 8435 9492 9814 10158 10732 10769 10991 11355 11837 12664 13759 13762 13880 15043 15411 15498 15640 15910 16068 16352
14	57	0.897085 (0.866239)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 659 1306 1903 2017 2420 2623 2681 2774 3009 3018 3357 3592 3727 4807 4978 5102 5286 5495 5755 6833 7003 7557 7706 8435 9492 9814 10158 10732 10769 10991 11355 11837 12664 13759 13762 13880 15043 15411 15498 15640 15910 16068 16352
14	58	0.902823 (0.871221)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 463 659 1306 1903 2017 2420 2623 2681 2774 3009 3018 3372 3462 3727 3848 4807 4978 5102 5286 5495 5755 6833 7557 7706 8435 9814 10158 10732 10769 10991 11355 11837 12664 12788 13759 13762 13880 15043 15411 15498 15640 15910 16068 16352
14	59	0.908281 (0.875936)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 659 1183 1306 1903 2017 2420 2623 2681 2774 3018 3153 3357 3727 4807 4978 5102 5286 5495 5755 6833 7003 7557 7706 8435 8792 9319 9814 10158 10732 10769 10991 11355 11837 12664 13759 13762 13880 14155 15043 15411 15498 15640 15910 16068 16352
14	60	0.913471 (0.880408)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 659 920 1183 1306 1903 2017 2420 2623 2681 2774 3018 3153 3357 3727 4807 4978 5102 5286 5495 5755 6833 7003 7557 7706 8435 8792 9319 9814 10158 10732 10769 10991 11355 11837 12664 13759 13762 13880 14155 15043 15411 15498 15640 15910 16068 16352
14	61	0.918401 (0.884630)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 659 920 1183 1306 1903 2017 2420 2623 2681 2774 3018 3153 3357 3727 4807 4978 5102 5286 5495 5635 5755 6833 7003 7557 7706 8435 8792 9319 9814 10158 10732 10769 10991 11355 11837 12664 13759 13762 13880 14155 15043 15411 15498 15640 15910 16068 16352

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
14	62	0.923071 (0.888622)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 659 920 1183 1306 1660 1903 2017 2420 2623 2681 2774 3018 3153 3357 3727 4807 4978 5102 5286 5495 5635 5755 6833 7003 7557 7706 8435 8792 9319 9814 10158 10732 10769 10991 11355 11837 12664 13759 13762 13880 14155 15043 15411 15498 15640 15910 16068 16352
14	63	0.927496 (0.892391)	4	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 659 1183 1306 1903 2017 2407 2420 2623 2681 2774 3018 3153 3357 3727 4807 4978 5102 5286 5495 5755 6833 7003 7557 7706 8435 8692 8792 9319 9814 10158 10732 10769 10991 11355 11837 12664 12717 13759 13762 13880 14155 15043 15246 15411 15498 15640 15910 16068 16352
14	64	0.931695 (0.895949)	4	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 659 1183 1306 1903 2017 2407 2420 2623 2681 2774 3018 3153 3357 3727 4807 4978 5102 5244 5286 5495 5635 5755 6833 7003 7557 7706 8435 8692 8792 9319 9814 10158 10732 10769 10991 11355 11837 12664 13759 13762 13880 14155 15043 15246 15411 15498 15640 15910 16068 16352
14	65	0.935664 (0.899309)	4	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 659 920 1306 1903 2017 2420 2623 2681 2774 3018 3153 3559 3727 3762 4807 4978 5102 5286 5495 5755 6833 7003 7294 7481 7557 7706 8435 9073 9319 9814 10158 10732 10769 10991 11355 11837 12664 13115 13759 13762 13880 14155 14641 15043 15411 15469 15498 15640 15910 16068 16352
14	66	0.939444 (0.902452)	4	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 659 920 1306 1903 2017 2420 2623 2681 2774 3018 3153 3462 3559 3727 3762 4807 4978 5102 5286 5495 5755 6833 7003 7294 7481 7557 7706 8435 9073 9319 9814 10158 10732 10769 10991 11355 11837 12664 13115 13759 13762 13880 14155 14641 15043 15411 15469 15498 15640 15910 16068 16352
14	67	0.943009 (0.905436)	4	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 659 920 1306 1903 2017 2420 2623 2681 2774 3018 3153 3462 3559 3727 3762 4807 4978 5102 5286 5495 5755 6833 7003 7294 7481 7557 7706 8435 9073 9319 9814 10158 10732 10769 10991 11355 11378 11837 12664 13115 13759 13762 13880 14155 14641 15043 15411 15469 15498 15640 15910 16068 16352
14	68	0.946382 (0.918521)	4	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 659 920 1306 1903 2017 2420 2623 2681 2774 3018 3085 3153 3462 3727 3762 4807 4978 5102 5286 5495 5755 6833 7003 7294 7557 7706 8435 8579 9073 9319 9814 10158 10732 10769 10991 11355 11378 11837 12664 12785 13115 13759 13762 13880 14155 14641 15043 15411 15469 15498 15640 15910 16068 16352
14	69	0.949577 (0.928520)	4	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 659 920 1306 1903 2017 2420 2623 2681 2774 3018 3085 3153 3462 3727 3762 4807 4978 5102 5286 5495 5755 6833 7003 7294 7481 7557 7706 8435 8579 9073 9319 9814 10158 10732 10769 10991 11355 11378 11837 12664 12785 13115 13759 13762 13880 14155 14641 15043 15411 15469 15498 15640 15910 16068 16352
14	70	0.952589 (0.936545)	4	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 659 920 1306 1903 2017 2420 2623 2681 2774 3018 3153 3462 3559 3727 3762 4807 4978 5102 5286 5495 5755 6833 7003 7294 7481 7557 7706 8435 9073 9319 9452 9814 9970 10158 10732 10769 10991 11355 11837 12664 12679 12785 13115 13759 13762 13880 14155 14641 15043 15411 15469 15498 15640 15910 16068 16352
14	71	0.955444 (0.943171)	4	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 659 920 1306 1903 2017 2420 2623 2681 2774 3018 3153 3462 3559 3727 3762 4807 4978 5102 5286 5495 5755 6283 6833 7003 7294 7481 7557 7706 8435 9073 9319 9814 9964 10158 10732 10769 10991 11355 11378 11837 12664 12785 13115 13365 13759 13762 13880 14155 14641 15043 15411 15469 15498 15640 15910 16068 16352
14	72	0.958137 (0.948742)	4	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 659 677 920 1306 1903 2017 2420 2623 2681 2774 3018 3085 3153 3462 3727 3762 4807 4978 5102 5286 5495 5755 6833 6997 7003 7294 7481 7557 7706 8435 8517 8579 9073 9319 9814 10158 10732 10769 10991 11355 11837 12664 12785 13115 13759 13762 13880 14155 14508 14641 15043 15411 15469 15498 15640 15910 16068 16352
14	73	0.960679 (0.953497)	4	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 659 920 1306 1903 2017 2420 2623 2681 2774 3018 3153 3462 3559 3727 3762 4382 4807 4978 5102 5286 5495 5755 6283 6833 7003 7294 7481 7557 7706 8435 9073 9319 9814 9964 10158 10732 10769 10991 11355 11378 11837 12612 12664 12785 13115 13365 13759 13762 13880 14155 14641 15043 15411 15469 15498 15640 15910 16068 16352
14	74	0.963088 (0.957610)	4	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 659 920 1306 1903 2017 2420 2623 2681 2774 3018 3153 3462 3559 3727 3762 4382 4807 4978 5102 5286 5495 5755 6283 6833 7003 7294 7481 7557 7706 8435 9073 9319 9814 9964 10158 10732 10769 10991 11340 11355 11378 11837 12612 12664 12785 13115 13365 13759 13762 13880 14155 14641 15043 15411 15469 15498 15640 15910 16068 16352
14	75	0.965351 (0.961211)	4	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 342 659 920 1306 1903 2017 2420 2623 2681 2774 3018 3153 3462 3559 3727 3762 4807 4978 5102 5286 5373 5495 5755 6833 6997 7003 7294 7481 7557 7706 8435 9073 9319 9452 9814 9970 10158 10732 10769 10991 11355 11837 12664 12679 12785 12842 13115 13300 13759 13762 13880 14155 14641 15043 15411 15469 15498 15640 15910 16068 16352
16	17	0.304297	17	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65535
16	18	0.322194	12	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 39167 59183
16	19	0.340076 (0.340064)	10	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 31160 32263 53375
16	20	0.357940 (0.357933)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 7021 15923 35315 59183
16	21	0.37579	10	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 7021 15923 35315 54485 59183
16	22	0.393584 (0.393564)	9	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 7628 22649 32767 36383 51527 65296
16	23	0.411324	9	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 27349 32143 33695 40115 46953 50936 53375
16	24	0.428980 (0.427210)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 10555 25340 31842 32143 33695 40115 42821 53375
16	25	0.446543 (0.444138)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 17852 32733 33695 40148 42341 52050 53375 63088 63879
16	26	0.463983 (0.462450)	8	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 13993 17852 32733 33695 40148 42341 47763 52050 53375 59559
16	27	0.481223 (0.479306)	8	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 5096 7518 14525 21942 28654 30429 36383 45466 48489 60005 65296
16	28	0.498311 (0.495947)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 10555 19995 25340 31842 32143 33695 40115 42821 46930 47914 53375
16	29	0.515225 (0.513434)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 7223 12140 13993 14606 17852 32733 33695 40148 42341 52050 53375 59559 64388
16	30	0.531919 (0.529906)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 10555 14862 19995 25340 27881 31842 32143 33695 40115 42821 46930 53375 53724
16	31	0.548369 (0.546069)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 10555 15669 18838 19995 25340 31842 32143 33695 36504 40115 42821 46930 53375 53724
16	32	0.564527 (0.561907)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 10555 14854 15669 19995 22934 25340 31842 32143 33695 36504 40115 42821 46930 53375 53724
16	33	0.580390 (0.577473)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 10555 15669 16614 19995 25340 27538 31842 32143 33695 34891 37345 40115 42821 46930 53375 53724

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
16	34	0.595954 (0.594112)	6	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 10555 14854 15669 19995 25340 27362 31842 32143 33695 35022 36504 39400 40115 42821 46930 53375 53724
16	35	0.611194 (0.610017)	7	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 10555 15669 17638 19995 25340 27538 31842 32143 33695 34891 36632 37345 39557 40115 42821 46930 53375 53724
16	36	0.626140 (0.625527)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 10555 14854 15669 19995 25340 31842 32143 33695 36504 37614 40115 40234 42821 46930 48340 53375 53724 54977 58409
16	37	0.640703 (0.640500)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 10555 12523 15669 18238 19995 25340 27538 31842 32143 33695 34891 36632 37345 40115 42821 46930 53375 53724 58085 61352
16	38	0.654932 (0.655010)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 10555 14854 15669 19995 22267 24019 25340 31842 32143 33695 36504 37614 40115 40234 42821 46930 48098 48340 53375 53724 54977
16	39	0.668817 (0.667947)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 10555 14854 15669 19995 25340 26519 27710 31842 32143 32710 33695 35022 36504 40115 42821 46930 48340 53375 53724 55283 59916 62499
16	40	0.682338 (0.681506)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 7164 10555 14581 14854 15669 19995 25340 25477 27710 31842 32143 33695 35022 36504 40115 42821 46930 48340 53375 53724 55283 59916 62499
16	41	0.695484 (0.694995)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 10555 10858 14854 15669 19995 21878 25340 30369 31152 31842 32143 33695 36504 40115 42821 46930 48104 48340 53375 53724 57249 59324 60102 61303
16	42	0.708267 (0.707933)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 10555 10858 14854 15669 19995 21878 25340 30369 31152 31842 32143 33695 35651 36504 40115 42821 46930 48104 48340 53375 53724 57249 59324 60102 61303
16	43	0.720675 (0.720370)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 10555 11028 14854 15669 19995 22612 23204 25340 30782 31842 32143 33695 36504 37525 37614 40115 40234 42821 46930 48340 53375 53724 58409 60639 62217 64083
16	44	0.732719 (0.732673)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 4769 10555 14854 15669 19995 22612 23204 25340 30782 31842 32143 33695 36504 37614 40115 40234 42821 46930 48340 52673 53375 53724 57180 58409 60639 62217 64083
16	45	0.74449	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 6715 8547 12466 16631 21349 21565 28585 32767 33383 33963 35123 35759 37069 38414 38624 39153 41301 43542 43872 44044 49561 51818 52250 52640 61468 61542 61888 64190 65024
16	46	0.755715	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 6715 8547 12466 16631 21349 21565 27348 28585 32767 33383 33963 35123 35759 37069 38414 38624 39153 41301 43542 43872 44044 49561 51818 52250 52640 61468 61542 61888 64190 65024
16	47	0.766637	7	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 6715 8547 12466 14040 16631 21349 21565 27348 28585 32767 33383 33963 35123 35759 37069 38414 38624 39153 41301 43542 43872 44044 49561 51818 52250 52640 61468 61542 61888 64190 65024
16	48	0.777092 (0.708660)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 2174 10555 12221 12999 14586 14926 19995 23953 25340 27319 27881 28822 31842 32143 32309 33695 35300 40115 41850 42821 43146 46587 46930 50224 52924 53375 53724 55136 57952 61511 62572
16	49	0.787260 (0.735027)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 2174 10555 12221 12999 14586 14926 19995 23953 25340 27319 27881 28822 31842 32143 32309 33695 35300 40115 41850 42821 43146 46587 46930 50224 52924 53375 53724 55136 57952 61511 62572 64008
16	50	0.797071 (0.742252)	5	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 6623 10555 11116 12221 12999 14586 14926 19995 23953 25340 27319 27881 28822 31261 31842 32143 33695 35300 40115 41850 42821 43146 46587 46930 50224 50393 52924 53375 53724 55136 57952 61511 62572
16	51	0.806515 (0.749193)	5	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 2174 6623 10555 11116 12221 12999 14586 14926 19995 23953 25340 27319 27881 28822 31261 31842 32143 33695 35300 40115 41850 42821 43146 46587 46930 50224 50393 52924 53375 53724 55136 57952 61511 62572
16	52	0.815609 (0.755784)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 7546 9685 10555 12221 14586 14926 19995 23953 25340 26531 27319 27881 28822 31261 31842 32143 33695 35300 40115 41850 42821 43146 46587 46930 50224 52924 53375 53724 55136 57952 58519 58832 61511 62572 64948
16	53	0.824372 (0.762079)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 7546 9685 10555 12221 14586 14926 19995 23953 25340 26531 27319 27881 28822 31261 31842 32143 33695 34136 35300 40115 41850 42821 43146 46587 46930 50224 52924 53375 53724 55136 57952 58519 58832 61511 62572 64948
16	54	0.832780 (0.768060)	5	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 7546 9685 10555 12221 14586 14926 19995 23953 25340 26531 27319 27881 28822 31261 31842 32143 33695 34136 35300 40115 41850 42821 43124 43146 46587 46930 50224 52924 53375 53724 55136 57952 58519 58832 61511 62572 64948
16	55	0.840862 (0.773722)	5	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 5528 7546 9685 10555 12221 14586 14926 19995 23953 25340 26531 27319 27881 28822 31261 31842 32143 33695 34136 35300 40115 41850 42821 43124 43146 46587 46930 50224 52924 53375 53724 55136 57952 58519 58832 61511 62572 64948
16	56	0.848615 (0.779108)	5	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 2174 6623 10555 11116 11896 12221 12999 14586 14926 19995 20788 23508 23953 25340 27319 27881 28822 31261 31842 32143 33695 35079 35300 40115 41850 42821 43146 46587 46930 48617 50224 50393 52924 53375 53724 55136 57952 61511 62572
16	57	0.856056 (0.784238)	5	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 2174 6623 10555 11116 11896 12221 12999 14586 14926 19995 20788 23508 23953 25340 27319 27881 28822 31261 31842 32143 33695 35079 35300 40115 41850 42821 43146 43545 46587 46930 48617 50224 50393 52924 53375 53724 55136 57952 61511 62572
16	58	0.863177 (0.789107)	5	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 2174 6623 10555 11116 11896 12221 12999 14586 14926 19995 20788 23508 23953 25340 27319 27881 28822 31261 31842 32143 33695 35079 35300 40115 41850 42821 43146 43545 46587 46930 48617 48950 50224 50393 52924 53375 53724 55136 57952 61511 62572
16	59	0.870002 (0.793726)	5	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 2174 6623 10555 11116 11896 12221 12999 14586 14926 19995 20788 23508 23953 25340 27319 27881 28822 30718 31261 31842 32143 33695 35079 35300 40115 41850 42821 43146 46587 46930 48617 48950 50224 50393 51094 52924 53375 53724 55136 57952 61511 62572
16	60	0.876552 (0.798105)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 2174 6623 9685 10555 11116 11896 12221 12999 14586 14926 19995 20228 20788 23508 23953 25340 27319 27881 28822 31261 31842 32143 33695 35079 35300 40115 41850 42821 43146 46587 46930 48617 48950 50224 50393 52924 53375 53724 54946 55136 57952 61511 62572
16	61	0.882825 (0.802240)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 2174 6623 9685 10416 10555 11116 11896 12221 12999 14586 14926 19995 20228 20788 23508 23953 25340 27319 27881 28822 31261 31842 32143 33695 35079 35300 40115 41850 42821 43146 46587 46930 48617 48950 50224 50393 52924 53375 53724 55136 57952 61511 62572 65419

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
16	62	0.888812 (0.806151)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 2174 6623 9685 10416 10555 11116 11896 12221 12999 14403 14586 14926 19995 20228 20788 23508 23953 25340 27319 27881 28822 31261 31842 32143 33695 35079 35300 40115 41850 42821 43146 46587 46930 48617 48950 50224 50393 52924 53375 53724 55136 57952 61511 62572 65419
16	63	0.894530 (0.809853)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 2174 6623 9685 10416 10555 11116 11896 12221 12999 14586 14926 19995 20228 20788 23508 23953 25340 27319 27881 28822 31261 31842 32143 33695 35079 35300 35659 36455 40115 41850 42821 43146 46587 46930 48617 48950 50224 50393 52924 53375 53724 55136 57952 61511 62572 65419
16	64	0.900000 (0.813344)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 2026 2174 6623 9685 10416 10555 11116 11896 12221 12999 14403 14586 14926 19995 20228 20788 23508 23688 23953 25340 27319 27881 28822 30545 31261 31842 32143 33695 35079 35300 40115 41850 42821 43146 46587 46930 48617 48950 50224 50393 52924 53375 53724 55136 57952 61511 62572 65419
18	19	0.302308	19	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262143
18	20	0.318218	13	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 260159 262080
18	21	0.334124	12	10	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 175323 212764 249824
18	22	0.350022	11	10	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 4094 62449 92759 235063
18	23	0.365916	11	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 4094 62449 92759 174891 235063
18	24	0.381784 (0.381725)	10	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 30106 40171 85575 155446 260159 262080
18	25	0.397608 (0.397594)	9	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 30106 40171 85575 91061 155446 260159 262080
18	26	0.413414 (0.413400)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 79291 93349 99305 140135 145868 190155 213789 262142
18	27	0.429176 (0.429164)	10	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 79291 93349 99305 140135 145868 184739 190155 213789 262142
18	28	0.444881	10	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 64047 69235 79291 93349 99305 112420 140135 190155 213789 262142
18	29	0.460493 (0.460429)	9	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 22043 64047 69235 79291 93349 99305 140135 190155 213789 261632 262142
18	30	0.475983 (0.475968)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19192 64047 69235 79291 87898 93349 99305 140135 190155 213789 261632 262142
18	31	0.491408	9	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 59335 107885 144815 158090 168791 174764 175314 181937 218452 235928 254432 261555 261632
18	32	0.506618 (0.502811)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19192 64047 69235 79291 87898 93024 93349 99305 111365 140135 190155 213789 261632 262142
18	33	0.521730 (0.517164)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19192 64047 69235 79291 87898 93024 93349 99305 111365 140135 145904 190155 213789 261632 262142
18	34	0.536683 (0.532203)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19192 64047 69235 79291 87898 93024 93349 99305 111365 140135 145904 158924 190155 213789 261632 262142
18	35	0.551457 (0.546738)	8	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 26037 30317 59335 107885 136906 144815 158090 168791 174764 175314 181937 218452 229684 235928 254432 261555 261632
18	36	0.566001 (0.560943)	8	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 5241 19192 64047 68271 69235 79291 87898 93024 93349 99305 111365 140135 145904 158924 190155 213789 261632 262142
18	37	0.579878 (0.574730)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 30519 39506 41588 59339 61038 79012 118678 122076 150903 158024 178583 181223 196319 237356 238775 244152 257225 260479
18	38	0.594094 (0.588175)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 29102 30519 39506 41588 59339 61038 79012 118678 122076 150903 158024 178583 181223 196319 237356 238775 244152 257225 260479
18	39	0.608064 (0.604083)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 29102 30519 39506 41588 59339 61038 79012 118678 122076 150903 158024 178583 181223 196319 237356 238775 241868 244152 257225 260479
18	40	0.621783 (0.617242)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 29102 30519 39506 41588 59339 61038 79012 118678 122076 150903 152269 158024 178583 181223 196319 215462 237356 238775 241868 244152 257225 260479
18	41	0.635241 (0.630427)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 29102 30519 39506 41588 59339 61038 79012 118678 122076 150903 152269 158024 178583 181223 196319 215462 237356 238775 241868 244152 257225 260479
18	42	0.648429 (0.643255)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 29102 30519 39506 41588 59339 61038 79012 112538 118678 122076 150903 152269 158024 178583 181223 196319 215462 237356 238775 241868 244152 257225 260479
18	43	0.661359 (0.655653)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 29102 30519 39506 41588 59339 61038 79012 112538 118678 122076 150903 152269 158024 164308 178583 181223 196319 215462 237356 238775 241868 244152 257225 260479
18	44	0.673971 (0.667699)	8	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 30519 39506 50961 59339 61038 75021 76704 79012 118678 119094 122076 138179 150903 158024 178583 181223 189005 196319 225212 237356 238775 244152 250526 257225 261666
18	45	0.685972 (0.679663)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 30519 39506 59339 61038 72205 79012 95317 103043 118678 122076 144410 150903 158024 178583 181223 188483 190634 191031 192269 196319 206086 218733 237356 238775 244152 257225
18	46	0.698099 (0.691306)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 30519 39506 59339 61038 72205 79012 95317 103043 118678 122076 144410 150903 158024 158897 178583 181223 188483 190634 191031 192269 196319 206086 218733 237356 238775 244152 257225
18	47	0.709905 (0.702531)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 30519 39506 59339 61038 72205 79012 82667 95317 103043 118678 122076 144410 150903 158024 158897 178583 181223 188483 190634 191031 192269 196319 206086 218733 237356 238775 244152 257225
18	48	0.721366 (0.713411)	8	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 30519 39506 59339 61038 72205 79012 82667 92837 95317 103043 118678 122076 144410 150903 158024 158897 178583 181223 188483 190634 191031 192269 196319 206086 218733 237356 238775 244152 257225
18	49	0.732555 (0.732434)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 30519 39506 59339 61038 72205 79012 95317 103043 118678 122076 144410 150903 158024 178583 181223 185674 188483 190634 191031 192269 196319 201183 201258 206086 218733 220070 237356 238775 244152 257225

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
18	50	0.743450 (0.743348)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 30519 39506 59339 61038 72205 79012 95317 103043 118678 122076 144410 144885 150903 158024 178583 181223 185674 188483 190634 191031 192269 196319 201183 201258 206086 218733 220070 237356 238775 244152 257225
18	51	0.754039 (0.753950)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 30519 39506 59339 61038 72205 79012 95317 100629 103043 118678 122076 144410 144885 150903 158024 178583 181223 185674 188483 190634 191031 192269 196319 201183 201258 206086 218733 220070 237356 238775 244152 257225
18	52	0.764315 (0.764241)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 30519 39506 59339 61038 72205 79012 95317 100629 103043 118678 122076 144410 144885 150903 158024 169165 178583 181223 185674 188483 190634 191031 192269 196319 201183 201258 206086 218733 220070 237356 238775 244152 257225
18	53	0.774287 (0.774229)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 30519 39506 59339 61038 72205 73191 79012 95317 100629 103043 118678 122076 144410 144885 150903 158024 169165 178583 181223 185674 188483 190634 191031 192269 196319 201183 201258 206086 218733 220070 237356 238775 244152 257225
18	54	0.783958 (0.783916)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 30519 39506 59339 61038 72205 73191 79012 95317 100629 103043 118678 122076 144410 144885 150903 158024 169165 178583 181223 185674 188483 190634 191031 192269 196319 201183 201258 206086 212569 218733 220070 237356 238775 244152 257225
18	55	0.793329 (0.793301)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 30519 39506 59339 61038 72205 73191 79012 95317 100629 103043 118678 122076 144410 144885 150903 158024 169165 178583 181223 185674 188483 190634 191031 192269 196319 201183 201258 206086 212569 215493 218733 220070 237356 238775 244152 257225
18	56	0.802401 (0.802378)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 30519 39506 59339 61038 72205 73191 79012 95317 100629 103043 118678 122076 144410 144885 150903 158024 169165 178583 181223 185674 188483 190634 191031 192269 196319 201183 201258 206086 212569 215493 218547 218733 220070 237356 238775 244152 257225
18	57	0.811179 (0.811160)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 30519 39506 59339 61038 72205 73191 79012 95317 100629 103043 118678 122076 137155 144410 144885 150903 158024 169165 178583 181223 185674 188483 190634 191031 192269 196319 201183 201258 206086 212569 215493 218547 218733 220070 237356 238775 244152 257225
18	58	0.819664 (0.819648)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 30519 39506 59339 61038 72205 73191 79012 95317 100629 103043 118678 122076 137155 144410 144885 146382 150903 158024 169165 178583 181223 185674 188483 190634 191031 192269 196319 201183 201258 206086 212569 215493 218547 218733 220070 237356 238775 244152 257225
18	59	0.827862 (0.827853)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 30519 39506 59339 61038 72205 73191 79012 92837 95317 100629 103043 118678 122076 137155 144410 144885 146382 150903 158024 169165 178583 181223 185674 188483 190634 191031 192269 196319 201183 201258 206086 212569 215493 218547 218733 220070 237356 238775 244152 257225
18	60	0.835774 (0.835770)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 30519 39506 59339 61038 72205 73191 79012 92837 95317 100629 103043 110035 118678 122076 137155 144410 144885 146382 150903 158024 169165 178583 181223 185674 188483 190634 191031 192269 196319 201183 201258 206086 212569 215493 218547 218733 220070 237356 238775 244152 257225
18	61	0.843406 (0.843405)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 30519 39506 59339 61038 72205 73191 79012 92837 95317 100629 103043 110035 118678 122076 137155 144410 144885 146382 150903 158024 165334 169165 178583 181223 185674 188483 190634 191031 192269 196319 201183 201258 206086 212569 215493 218547 218733 220070 237356 238775 244152 257225
18	62	0.850762	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 30519 39506 59339 61038 72205 73191 79012 82667 92837 95317 100629 103043 110035 118678 122076 137155 144410 144885 146382 150903 158024 165334 169165 178583 181223 185674 188483 190634 191031 192269 196319 201183 201258 206086 212569 215493 218547 218733 220070 237356 238775 244152 257225
18	63	0.857847	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 30519 39506 59339 61038 72205 73191 79012 82667 92837 95317 100629 103043 110035 118678 122076 137155 144410 144885 146382 150903 158024 165334 169165 178583 181223 185674 188483 190634 191031 192269 196319 201183 201258 206086 207486 212569 215493 218547 218733 220070 237356 238775 244152 257225
18	64	0.864664	8	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 30519 39506 59339 61038 72205 73191 79012 82667 92837 95317 100629 103043 103743 110035 118678 122076 137155 144410 144885 146382 150903 158024 165334 169165 178583 181223 185674 188483 190634 191031 192269 196319 201183 201258 206086 207486 212569 215493 218547 218733 220070 237356 238775 244152 257225
18	65	0.871169 (0.869426)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 30519 39506 59339 61038 72205 73191 79012 82667 92837 95317 100629 103043 103743 110035 118678 122076 135910 137155 144410 144885 146382 150903 158024 165334 169165 178583 181223 185674 188483 190634 191031 192269 196319 201183 201258 206086 207486 212569 215493 218547 218733 220070 237356 238775 244152 257225
18	66	0.877457 (0.875531)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 19753 30519 39506 59339 61038 67955 72205 73191 79012 82667 92837 95317 100629 103043 103743 110035 118678 122076 135910 137155 144410 144885 146382 150903 158024 165334 169165 178583 181223 185674 188483 190634 191031 192269 196319 201183 201258 206086 207486 212569 215493 218547 218733 220070 237356 238775 244152 257225
18	67	0.883195 (0.881422)	7	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 3015 19753 30519 39506 59339 61038 67955 72205 73191 79012 82667 92837 95317 100629 103043 103743 110035 118678 122076 135910 137155 144410 144885 146382 150903 158024 165334 169165 178583 181223 185674 188483 190634 191031 192269 196319 201183 201258 206086 207486 212569 215493 218547 218733 220070 237356 238775 244152 257225
18	68	0.887086	7	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 11955 13936 32282 32999 38800 39250 42924 47840 55393 56951 58478 60921 69584 81270 92272 93719 98994 101001 102495 107633 112602 135116 148158 154496 159264 161706 168209 175488 176485 180456 187783 193685 194276 195402 203311 206292 207540 207680 208910 210560 225118 230979 235516 238390 247552 252600 254562 254936 258694 260096
18	69	0.892673 (0.776207)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 11955 13936 32282 32999 38800 39250 42924 47840 55393 56951 58478 60921 69584 81270 92272 93719 98994 101001 102495 107633 112602 135116 148158 154496 159264 161706 168209 175488 176485 180456 187783 193685 194276 195402 203311 206292 207540 207680 208910 210560 225118 230979 235516 238390 247552 252600 254562 254936 256876 258694 260096

Continued on next page

		n	R_e	d	R_c	Packed integer form of the parity check matrix
18	70		0.897962 (0.782792)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 11955 13936 32282 32999 38800 39250 42924 47840 55393 56951 58478 60921 69584 81270 92272 93719 98994 101001 102495 107633 112602 118747 135116 148158 154496 159264 161706 168209 175488 176485 180456 187783 193685 194276 195402 203311 206292 207540 207680 208910 210560 225118 230979 235516 238390 247552 252600 254562 254936 256876 258694 260096
18	71		0.903035 (0.788320)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 11955 13936 32282 32999 38800 39250 42924 47840 55393 56951 58478 60921 69584 75429 81270 92272 93719 98994 101001 102495 107633 112602 118747 135116 148158 154496 159264 161706 168209 175488 176485 180456 187783 193685 194276 195402 203311 206292 207540 207680 208910 210560 225118 230979 235516 238390 247552 252600 254562 254936 256876 258694 260096
18	72		0.907878 (0.793025)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 11955 13936 32239 32282 32999 38800 39250 42924 47840 55393 56951 58478 60921 69584 75429 81270 92272 93719 98994 101001 102495 107633 112602 118747 135116 148158 154496 159264 161706 168209 175488 176485 180456 187783 193685 194276 195402 203311 206292 207540 207680 208910 210560 225118 230979 235516 238390 247552 252600 254562 254936 256876 258694 260096
18	73		0.912521 (0.797088)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 11955 13936 32239 32282 32999 35245 38800 39250 42924 47840 55393 56951 58478 60921 69584 75429 81270 92272 93719 98994 101001 102495 107633 112602 118747 135116 148158 154496 159264 161706 168209 175488 176485 180456 187783 193685 194276 195402 203311 206292 207540 207680 208910 210560 225118 230979 235516 238390 247552 252600 254562 254936 256876 258694 260096
18	74		0.916946 (0.800659)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 11955 13936 19001 32239 32282 32999 35245 38800 39250 42924 47840 55393 56951 58478 60921 69584 75429 81270 92272 93719 98994 101001 102495 107633 112602 118747 135116 148158 154496 159264 161706 168209 175488 176485 180456 187783 193685 194276 195402 203311 206292 207540 207680 208910 210560 225118 230979 235516 238390 247552 252600 254562 254936 256876 258694 260096
18	75		0.921191 (0.803836)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 11955 13936 19001 32239 32282 32999 35245 38800 39250 42924 47840 55393 56951 58478 60921 69584 75429 81270 92272 93719 98994 101001 102495 107633 112602 118747 135116 148158 154496 159264 161706 168209 175488 176485 180456 187783 193685 194276 195402 203311 206292 207540 207680 208910 210560 225118 230979 233933 235516 238390 247552 252600 254562 254936 256876 258694 260096
18	76		0.925238 (0.805872)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 11955 13936 19001 32239 32282 32999 35245 38800 39250 42924 47840 55393 56951 58478 60921 69584 75429 81270 92272 93719 98994 101001 102495 107633 112602 118747 135116 148158 154496 159264 161706 168209 175488 176485 180456 187783 193685 194276 195402 203311 206292 207540 207680 208910 210560 225118 230979 233933 235516 238390 245691 247552 252600 254562 254936 256876 258694 260096
18	77		0.929118 (0.808611)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 11955 13936 19001 31091 32239 32282 32999 35245 38800 39250 42924 47840 55393 56951 58478 60921 69584 75429 81270 92272 93719 98994 101001 102495 107633 112602 118747 135116 148158 154496 159264 161706 168209 175488 176485 180456 187783 193685 194276 195402 203311 206292 207540 207680 208910 210560 225118 230979 233933 235516 238390 245691 247552 252600 254562 254936 256876 258694 260096
18						

149

	m	n	R_e	d	R_c	Packed integer form of the parity check matrix
18	86		0.956758 (0.858690)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 11955 13936 19001 31091 31388 32239 32282 32999 35245 38800 39250 42924 47840 55393 56951 58478 60921 69584 72521 75429 79581 81270 87381 92272 93719 98994 101001 102495 107633 112602 115645 118747 135116 148158 150091 154496 159264 161706 168209 175488 176485 180456 180639 187783 193685 194276 195402 203311 206292 207540 207680 208910 210560 225118 230979 233933 235516 238390 245691 247552 251691 252600 254562 254936 255875 256876 258694 260096
18	87		0.959131 (0.860082)	6	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 11955 13936 19001 31091 31388 32239 32282 32999 35245 38800 39250 42924 47840 55393 56951 58478 60921 69584 72521 75429 79581 81270 87381 92272 93719 98994 101001 102495 107633 112602 115645 118747 135116 148158 150091 154496 159264 161706 168209 175488 176485 180456 180639 187783 193685 194276 195402 203311 206292 207540 207680 208910 210560 225118 230979 233933 235516 238390 245274 245691 247552 251691 252600 254562 254936 255875 256876 258694 260096
18	88		0.961409 (0.861395)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 11955 13936 19001 31091 31388 32239 32282 32999 35245 38800 39250 42924 47840 55393 56951 58478 60921 69584 72521 75429 79581 81270 87381 92272 93719 98994 101001 102495 107633 112602 115645 118747 135116 148158 150091 154496 159264 161706 168209 175488 176485 180456 180639 187783 193685 194276 195402 203311 206292 207540 207680 208910 210560 225118 230979 233933 235516 238390 245274 245691 247552 251691 252600 254562 254936 255875 256876 258694 260096
18	89		0.963530 (0.862635)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 11955 13936 19001 19184 31091 31388 32239 32282 32999 35245 38800 39250 42924 47840 55393 56951 58478 60921 69584 72521 75429 79581 81270 87381 92272 93719 98994 101001 102495 107633 112602 115645 118747 135116 148158 150091 154496 159264 161706 168209 175488 176485 180456 180639 187783 193685 194276 195402 203311 206292 207540 207680 208910 210560 225118 230979 233933 235516 238390 245274 245691 247552 251691 252600 254562 254936 255875 256876 258694 260096
18	90		0.965539 (0.863803)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 9648 11955 13936 19001 19184 31091 31388 32239 32282 32999 35245 38800 39250 42924 47840 55393 56951 58478 60921 69584 72521 75429 79581 81270 87381 92272 93719 98994 101001 102495 107633 112602 115645 118747 135116 148158 150091 154496 159264 161706 168209 175488 176485 180456 180639 187783 193685 194276 195402 203311 206292 207540 207680 208910 210560 225118 230979 233933 235516 238390 245274 245691 247552 251691 252600 254562 254936 255875 256876 258694 260096
18	91		0.967452 (0.864903)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 9648 11955 13936 19001 19184 31091 31388 32239 32282 32999 35245 38800 39250 42924 47840 55393 56951 58478 60921 69584 72521 75429 79581 81270 87381 92272 93719 98994 101001 102495 107633 112602 115645 118747 135116 148158 150091 154496 159264 161706 168209 175488 176485 180456 180639 187783 193685 194276 195402 203311 206292 207540 207680 208910 210560 225118 230979 233046 233933 235516 238390 245274 245691 247552 251691 252600 254562 254936 255875 256876 258694 260096
18	92		0.969269 (0.865937)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 9648 11955 13936 19001 19184 31091 31388 32239 32282 32999 35245 38800 39250 42924 47840 55393 56951 58478 60921 69584 72521 75429 79581 81270 87381 92272 93719 98994 101001 102495 107633 112602 115645 118747 135116 148158 150091 154496 159264 161706 168209 175488 176485 177484 180456 180639 187783 193685 194276 195402 203311 206292

150

	m	n	R_e	d	R_c	Packed integer form of the parity check matrix
18	99		0.979569 (0.871604)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 9648 11955 13936 19001 19184 19422 31091 31388 32239 32282 32999 35245 38800 39250 42924 47840 55393 56951 58478 60921 69325 69584 72521 75429 77985 81270 87381 92272 93398 93719 98994 101001 102495 107633 112602 115645 118747 135116 148158 150091 154496 159264 161706 168209 175488 176485 177484 180456 180639 187783 193685 194276 195402 198375 203311 206292 207540 207680 208910 210560 225118 230979 232719 233046 233933 235516 238390 245274 245691 247552 250822 251691 251919 252600 254562 254936 255875 256876 258694 260096
18	100		0.980749 (0.872227)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 9648 11955 13936 15892 19001 19184 19422 31091 31388 32239 32282 32999 35245 38800 39250 42924 47840 55393 56951 58478 60921 69325 69584 72521 75429 77985 79581 81270 87381 92272 93398 93719 98994 101001 102495 107633 112602 115645 118747 135116 148158 150091 154496 159264 161706 168209 175488 176485 177484 180456 180639 187783 193685 194276 195402 198375 203311 206292 207540 207680 208910 210560 225118 230979 232719 233046 233933 235516 238390 245274 245691 247552 250822 251691 251919 252600 254562 254936 255875 256876 258694 260096
18	101		0.981870 (0.872811)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 9648 11955 13936 15892 19001 19184 19422 31091 31388 32239 32282 32999 35245 38800 39250 42924 47840 55393 56951 58478 60921 69325 69584 72521 75429 77985 79581 81270 87381 89247 92272 93398 93719 98994 101001 102495 107633 112602 115645 118747 135116 148158 150091 154496 159264 161706 168209 175488 176485 177484 180456 180639 187783 193685 194276 195402 198375 203311 206292 207540 207680 208910 210560 225118 230979 232719 233046 233933 235516 238390 245274 245691 247552 250822 251691 251919 252600 254562 254936 255875 256876 258694 260096
18	102		0.982930 (0.873360)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 9648 11955 13936 15892 19001 19184 19422 31091 31388 32239 32282 32999 35245 38800 39250 42924 47840 55393 56951 58478 60921 69325 69584 72521 75429 77985 79581 81270 87381 89247 92272 93398 93719 98994 101001 102495 107633 112602 115645 118747 135116 148158 150091 154496 159264 161706 168209 175488 176485 177484 180456 180639 183339 187783 193685 194276 195402 198375 203311 206292 207540 207680 208910 210560 225118 230979 232719 233046 233933 235516 238390 245274 245691 247552 250822 251691 251919 252600 254562 254936 255875 256876 258694 260096
18	103		0.983935 (0.873874)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 9648 11955 13936 15892 19001 19184 19422 31091 31388 32239 32282 32999 35245 38800 39250 42924 47840 55393 56951 58478 60921 69325 69584 72521 75429 77985 79581 81270 87381 89247 92272 93398 93719 98994 101001 102495 107633 112602 115645 118747 135116 148158 150091 154496 159264 161706 168209 175488 176485 177484 180456 180639 183339 187783 193685 194276 195402 198375 203311 206292 207540 207680 208910 210560 211336 225118 230979 232719 233046 233933 235516 238390 245274 245691 247552 250822 251691 251919 252600 254562 254936 255875 256876 258694 260096
18	104		0.984888 (0.874356)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 9648 11955 13936 15892 19001 19184 19422 31091 31388 32239 32282 32999 35245 38800 39250 42924 47840 55393 56951 58478 60921 68349 69325 69584 72521 75429 77985 79581 81270 87381 89247 92272 93398 93719 98994 101001 102495 107633 112602 115645 118747 135116 148158 150091 154496 159264 161706 168209 175488 176485 177484 180456 180639 183339 187783 193685 194276 195402 198375 203311 206292 207540 207680 208910 210560 211336 225118 230979 232719 233046 233933 235516 238390 245274 245691 247552 250822 251

151

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
18	111	0.990219 (0.876985)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 9648 11955 12586 13936 15892 19001 19184 19422 23688 31091 31388 32239 32282 32999 35245 38800 39250 42924 47840 55393 56951 58478 60921 68349 69325 69584 72521 74683 75429 77985 79581 81270 82026 87381 89073 89247 92272 93398 93719 98994 101001 102495 107633 112602 115645 118747 135116 143558 148158 150091 154496 159264 161706 168209 175488 176485 177484 180456 180639 183339 187783 193685 194276 195402 198375 203311 206292 207540 207680 208910 210560 211336 220683 225118 230979 232719 233046 233933 235516 238390 245274 245691 247552 250108 250822 251691 251919 252600 254562 254936 255875 256876 258694 260096
18	112	0.990820 (0.877270)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 9648 11955 12586 13936 15892 19001 19184 19422 23688 31091 31388 32239 32282 32999 35245 38800 39250 42924 47840 55393 56951 58478 60921 68349 69325 69584 72521 74683 75429 77985 79581 81270 82026 87381 89073 89247 92272 93398 93719 98994 101001 102495 107633 112602 115645 118747 135116 143558 148158 150091 154496 159264 161706 168209 175488 176485 177484 180456 180639 183339 187783 193685 194276 195402 198375 203311 206292 207540 207680 208910 210560 211336 220683 225118 230979 232719 233046 233933 235516 238390 245274 245691 247552 250108 250822 251691 251919 252600 254562 254936 255875 256876 258694 260096
18	113	0.991387 (0.877535)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 9648 11955 12586 13936 15892 19001 19184 19422 23688 31091 31388 32239 32282 32999 35245 38800 39250 42924 47840 55393 56951 58478 60921 68349 69325 69584 72521 74683 75429 77985 79581 81270 82026 87381 89073 89247 92272 93398 93719 98994 101001 102495 107633 112602 115645 118747 127837 135116 143558 148158 150091 154496 159264 161706 168209 175488 176485 177484 180456 180639 183339 187783 193685 194276 195402 198375 203311 206292 207540 207680 208910 210560 211336 220683 225118 230979 232719 233046 233933 235516 238390 245274 245691 247552 250108 250822 251691 251919 252600 254562 254936 255875 256876 258694 260096
18	114	0.991921 (0.877783)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 9648 11955 12586 13936 15892 19001 19184 19422 23688 31091 31388 32239 32282 32999 35245 38800 39250 42924 47840 55393 56951 58478 60921 68349 69325 69584 72521 74683 75429 77985 79581 81270 82026 87381 89073 89247 92272 93398 93719 98994 101001 102495 107633 112602 115645 118747 127837 135116 143558 148158 150091 154496 159264 161706 168209 175488 176485 177484 179031 180456 180639 183339 187783 193685 194276 195402 198375 203311 206292 207540 207680 208910 210560 211336 220683 225118 230979 232719 233046 233933 235516 238390 245274 245691 247552 250108 250822 251691 251919 252600 254562 254936 255875 256876 258694 260096
18	115	0.992425 (0.878015)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 1688 9648 11955 12586 13936 15892 19001 19184 19422 23688 31091 31388 32239 32282 32999 35245 38800 39250 42924 47840 55393 56951 58478 60921 68349 69325 69584 72521 74683 75429 77985 79581 81270 82026 87381 89073 89247 92272 93398 93719 98994 101001 102495 107633 112602 115645 118747 127837 135116 143558 148158 150091 154496 159264 161706 168209 175488 176485 177484 179031 180456 180639 183339 187783 193685 194276 195402 198375 203311 206292 207540 207680 208910 210560 211336 220683 225118 230979 232719 233046 233933 235516 238390 245274 245691 247552 250108 250822 251691 251919 252600 254562 254936 255875 256876 258694 260096
18	116	0.992899 (0.878231)	5	4	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 1688 9648 11147 11955 12586 13936 15892 19001 19184 19422 23688 31091 31388 32239 32282 32999 35245 38800 39250 42924 47840 55393 5

152

Table B.1 – Continued from previous page					
m	n	R_e	d	R_c	Packed integer form of the parity check matrix
20	31	0.443545	11	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 57017 114034 228068 351011 355469 456136 531415 702022 710938 912272 965549
20	32	0.457486 (0.457354)	10	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 64370 211396 687398 699056 701256 758828 790396 873808 943712 957151 1017728 1046528
20	33	0.471525 (0.471496)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 64370 211396 687398 699056 701256 758828 790396 873808 893355 943712 1017728 1037278 1046528
20	34	0.485578 (0.485552)	10	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 64370 211396 687398 699056 701256 758828 786075 790396 873808 893355 943712 1017728 1037278 1046528
20	35	0.499517 (0.499508)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 64370 211396 311065 665077 687398 699056 701256 758828 787615 790396 873808 943712 1017728 1037278 1046528
20	36	0.513376	10	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 64370 101579 211396 311065 665077 687398 699056 701256 758828 787615 790396 873808 943712 1017728 1037278 1046528
20	37	0.527089 (0.526980)	9	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 64370 101579 151347 211396 311065 665077 687398 699056 701256 758828 787615 790396 873808 943712 1017728 1037278 1046528
20	38	0.540648 (0.540587)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 64370 101579 151347 211396 311065 339951 665077 687398 699056 701256 758828 787615 790396 873808 943712 1017728 1037278 1046528
20	39	0.554082 (0.554072)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 64370 101579 151347 211396 256437 311065 339951 665077 687398 699056 701256 758828 787615 790396 873808 943712 1017728 1037278 1046528
20	40	0.567425	9	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 145521 291042 359811 390047 396789 401775 504867 539951 582084 587657 597745 612863 719622 780094 793578 803550 1002433 1009734 1023226 1046953
20	41	0.580632	9	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 145521 291042 359811 390047 396789 401775 504867 511613 539951 582084 587657 597745 612863 719622 780094 793578 803550 1002433 1009734 1023226 1046953
20	42	0.586702 (0.584546)	8	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 80161 85153 194017 330735 371905 399681 524272 534128 543408 593104 614624 617984 696672 701952 763025 792976 836000 842240 884833 983488 990208 1040384
20	43	0.600364 (0.596201)	8	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 72986 80161 85153 194017 330735 371905 399681 524272 534128 543408 593104 614624 617984 696672 701952 763025 792976 836000 842240 884833 983488 990208 1040384
20	44	0.613904 (0.607596)	8	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 72986 80161 85153 90684 194017 330735 371905 399681 524272 534128 543408 593104 614624 617984 696672 701952 763025 792976 836000 842240 884833 983488 990208 1040384
20	45	0.627148 (0.618860)	8	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 52551 72986 80161 85153 90684 194017 330735 371905 399681 524272 534128 543408 593104 614624 617984 696672 701952 763025 792976 836000 842240 884833 983488 990208 1040384
20	46	0.639841 (0.629811)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 52551 72986 80161 85153 90684 147145 194017 330735 371905 399681 524272 534128 543408 593104 614624 617984 696672 701952 763025 792976 836000 842240 884833 983488 990208 1040384
20	47	0.652297 (0.640441)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 52551 72986 80161 85153 90684 147145 194017 254550 330735 371905 399681 524272 534128 543408 593104 614624 617984 696672 701952 763025 792976 836000 842240 884833 983488 990208 1040384
20	48	0.664516 (0.650861)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 52551 72986 80161 85153 90684 147145 194017 254550 330735 350811 371905 399681 524272 534128 543408 593104 614624 617984 696672 701952 763025 792976 836000 842240 884833 983488 990208 1040384
20	49	0.676464 (0.667475)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 52551 72986 80161 85153 90684 147145 194017 254550 330735 350811 371905 399681 524272 534128 543408 593104 614624 617984 696672 701952 740228 763025 792976 836000 842240 884833 983488 990208 1040384
20	50	0.687996 (0.677823)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 52551 72986 80161 85153 90684 147145 194017 254550 330735 332396 350811 371905 399681 524272 534128 543408 593104 614624 617984 696672 701952 740228 763025 792976 836000 842240 884833 983488 990208 1040384
20	51	0.699294 (0.687880)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 52551 72986 80161 85153 90684 147145 194017 254550 330735 332396 350811 371905 399681 496370 524272 534128 543408 593104 614624 617984 696672 701952 740228 763025 792976 836000 842240 884833 983488 990208 1040384
20	52	0.709874 (0.697691)	8	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 52551 72986 80161 85153 90684 147145 194017 254550 330735 332396 350811 371905 399681 496370 524272 534128 543408 593104 614624 617984 660816 696672 701952 740228 763025 792976 836000 842240 884833 983488 990208 1040384
20	53	0.712939 (0.707227)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 55520 203200 297636 305552 307528 376927 396676 398210 440804 534368 543332 543920 567618 593220 614692 617984 641744 673064 680048 701952 714952 740328 780192 792772 801448 842240 912488 922096 950664 959312 990208 1040384 1047856
20	54	0.724879 (0.716481)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 55520 203200 297636 305552 307528 376927 396676 398210 440804 534368 543332 543920 567618 591421 593220 614692 617984 641744 673064 680048 701952 714952 740328 780192 792772 801448 842240 912488 922096 950664 959312 990208 1040384 1047856
20	55	0.736433 (0.725461)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 55520 203200 297636 305552 307528 376927 396676 398210 440804 534368 543332 543920 567618 591421 593220 614692 617984 641744 673064 680048 701952 714952 740328 780192 792772 801448 842240 893635 912488 922096 950664 959312 990208 1040384 1047856
20	56	0.747166 (0.734172)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 55520 78819 203200 297636 305552 307528 376927 396676 398210 440804 534368 543332 543920 567618 591421 593220 614692 617984 641744 673064 680048 701952 714952 740328 780192 792772 801448 842240 893635 912488 922096 950664 959312 990208 1040384 1047856
20	57	0.757613 (0.742611)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 55520 78819 203200 297636 305552 307528 376927 396676 398210 440804 534368 543332 543920 567618 591421 593220 614692 617984 641744 673064 680048 701952 714952 740328 780192 792772 801448 842240 893635 912488 922096 950664 959312 990208 1022717 1040384 1047856

Continued on next page

	m	n	R_e	d	R_c	Packed integer form of the parity check matrix
20	58		0.767638 (0.750779)	8	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 55520 78819 96126 203200 297636 305552 307528 376927 396676 398210 440804 534368 543332 543920 567618 591421 593220 614692 617984 641744 673064 680048 701952 714952 740328 780192 792772 801448 842240 893635 912488 922096 950664 959312 990208 1022717 1040384 1047856
20	59		0.771625 (0.758689)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 55520 136801 173218 203200 297636 305552 307528 355711 396676 398210 440804 522338 534368 543332 543920 567618 593220 614692 617984 641744 642530 673064 680048 701952 714952 740328 780192 792772 801448 805154 842240 912488 922096 942786 950664 959312 990208 1040384 1047856
20	60		0.782156 (0.766333)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 55520 136801 173218 203200 297636 305552 307528 355711 396676 398210 440804 522338 534368 535711 543332 543920 567618 593220 614692 617984 641744 642530 673064 680048 701952 714952 740328 780192 792772 801448 805154 842240 912488 922096 942786 950664 959312 990208 1040384 1047856
20	61		0.791853 (0.773713)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 55520 136801 173218 203200 297636 305552 307528 355711 396676 398210 440804 522338 534368 535711 543332 543920 567618 593220 614692 617984 641744 642530 673064 680048 701952 714952 740328 780192 792772 801448 805154 842240 891361 912488 922096 942786 950664 959312 990208 1040384 1047856
20	62		0.801140 (0.780835)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 55520 136801 173218 203200 297636 305552 307528 355711 396676 398210 440804 522338 534368 535711 543332 543920 567618 593220 611903 614692 617984 641744 642530 673064 680048 701952 714952 740328 780192 792772 801448 805154 842240 891361 912488 922096 942786 950664 959312 990208 1040384 1047856
20	63		0.809840 (0.787705)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 55520 136801 173218 203200 297636 305552 307528 355711 396676 398210 440804 522338 534368 535711 543332 543920 567618 593220 611903 614692 617984 641744 642530 673064 680048 701952 714952 740328 780192 792772 801448 805154 842240 891361 912488 922096 942786 950664 959312 987743 990208 1040384 1047856
20	64		0.818257 (0.794326)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 55520 136801 173218 203200 297636 305552 307528 355711 396676 398210 440804 516577 522338 534368 535711 543332 543920 567618 593220 611903 614692 617984 641744 642530 673064 680048 701952 714952 740328 780192 792772 801448 805154 842240 891361 912488 922096 942786 950664 959312 987743 990208 1040384 1047856
20	65		0.826193 (0.813722)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 55520 136801 173218 203200 297636 305552 307528 355711 396676 398210 440804 516577 522338 534368 535711 543332 543920 567618 593220 611903 614692 617984 641744 642530 673064 680048 682591 701952 714952 740328 780192 792772 801448 805154 842240 891361 912488 922096 942786 950664 959312 987743 990208 1040384 1047856
20	66		0.833864 (0.820751)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 55520 64321 136801 173218 203200 297636 305552 307528 355711 396676 398210 440804 516577 522338 534368 535711 543332 543920 567618 593220 611903 614692 617984 641744 642530 673064 680048 682591 701952 714952 740328 780192 792772 801448 805154 842240 891361 912488 922096 942786 950664 959312 987743 990208 1040384 1047856
20	67		0.841164 (0.827426)			

154

Table B.1 – Continued from previous page						
m	n	R_e	d	R_c	Packed integer form of the parity check matrix	
20	75	0.889486 (0.888476)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 55520 123326 127489 166782 173218 203200 264350 275934 297636 305552 307528 313547 379905 396676 398210 412503 440321 440804 522338 534368 543332 543920 567618 593220 614692 617984 641744 642530 673064 673854 680048 688047 701952 713470 714952 740328 759763 780192 792772 801448 805154 816129 833822 842240 912488 922096 942786 950664 959312 989821 990208 1000542 1001433 1040384 1047856	
20	76	0.894866 (0.893708)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 55520 123326 127489 166782 173218 203200 264350 275934 297636 305552 307528 313547 379905 396676 398210 412503 440321 440804 522338 534368 543332 543920 567618 593220 614692 617984 641744 642530 673064 673854 680048 688047 701952 713470 714952 740328 755461 759763 780192 792772 801448 805154 816129 833822 842240 912488 922096 942786 950664 959312 989821 990208 1000542 1001433 1040384 1047856	
20	77	0.900739 (0.898675)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 41772 53076 70933 85654 95837 102185 121984 143743 159055 160966 162356 177792 186954 191057 197822 208965 259410 268572 272682 296292 308553 313067 320896 366336 371328 405616 442424 445952 450779 457021 475870 491392 532261 539043 560237 602893 616832 625083 632370 646431 661559 680024 692328 692902 701447 716691 741146 746583 749617 756858 782531 799616 814830 923790 985856 996729 1044736	
20	78	0.904864 (0.903400)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 55520 123326 127489 166782 173218 203200 264350 275934 297636 305552 307528 313547 350579 379905 396676 398210 412503 440321 440804 522338 534368 543332 543920 567618 593220 614692 617984 641744 642530 673064 673854 680048 688047 701952 713470 714952 740328 755461 759763 780192 792772 801448 805154 816129 833822 842240 912488 922096 942786 947607 950664 959312 989821 990208 1000542 1001433 1040384 1047856	
20	79	0.909516 (0.907905)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 55520 123326 127489 166782 173218 203200 264350 275934 297636 305552 307528 313547 350579 379905 396676 398210 412503 440321 440804 522338 534368 543332 543920 567618 593220 614692 617984 641744 642530 673064 673854 680048 688047 701952 713470 714952 740328 755461 759763 780192 780729 792772 801448 805154 816129 833822 842240 912488 922096 942786 947607 950664 959312 989821 990208 1000542 1001433 1040384 1047856	
20	80	0.913972 (0.912525)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 55520 123326 127489 166782 173218 203200 264350 275934 297636 305552 307528 313547 350579 379905 396676 398210 412503 440321 440804 469899 522338 534368 543332 543920 567618 593220 614692 617984 641744 642530 673064 673854 680048 688047 701952 713470 714952 740328 755461 759763 780192 780729 792772 801448 805154 816129 833822 842240 912488 922096 942786 947607 950664 959312 989821 990208 1000542 1001433 1040384 1047856	
20	81	0.918193 (0.916885)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 55520 123326 127489 166782 173218 203200 264350 275934 297636 305552 307528 313547 350579 379905 396676 398210 412503 440321 440804 469899 522338 534368 543332 543920 567618 593220 614692 617984 641744 642530 673064 673854 680048 688047 701952 713470 714952 740328 755461 759763 780192 780729 792772 801448 805154 816129 833822 842240 912488 922096 939237 942786 947607 950664 959312 989821 990208 1000542 1001433 1040384 1047856	
20	82	0.922233 (0.921153)	8	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 55520 123326 126319 127489 166782 173218 203200 264350 275934 297636 305552 307528 313547 350579 379905 396676 398210 412503 440321 440804 469899 522338 534368 543332 543920 567618 593220 614692 617984 641744 642530 673064 673854 680048 688047 701952 713470 714952 740328 755461 759763 780192 780729 792772 801448 805154 816129 828569 833822 834425 842240 912488 922096 942786 945495 950664 959312 990208 1000542 1040384 1047856	
20	83	0.925325 (0.925115)	8	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 51289 55520 123326 127489 155137 166782 173218 176697 203200 252659 264350 275934 289821 297636 305552 307528 344499 379905 396676 398210 440321 440804 468481 522338 534368 543332 543920 567618 593220 614692 617984 641744 642530 673064 673854 676825 680048 701952 713470 714952 740328 756225 780192 780729 792772 801448 805154 816129 828569 833822 834425 842240 912488 922096 923929 942786 945495 950664 959312 990208 1000542 1040384 1047856	
20	84	0.929111 (0.928930)	8	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 51289 55520 123326 127489 155137 166782 173218 176697 203200 225555 264350 275934 289821 297636 305552 307528 374775 379905 396676 398210 440321 440804 468481 522338 534368 543332 543920 567618 593220 614692 615101 617984 641744 642530 673064 673854 676825 680048 701952 713470 714952 740328 756225 780192 780729 792772 801448 805154 816129 828569 833822 834425 842240 895607 912488 922096 923929 942786 950664 959312 990208 1000542 1040384 1047856	
20	85	0.932639 (0.932545)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 51289 55520 123326 127489 155137 166782 173218 176697 203200 225555 264350 275934 289821 296247 297636 305552 307528 379905 396676 398210 440321 440804 468481 522338 534368 543332 543920 567618 593220 614692 615101 617984 641744 642530 673064 673854 676825 680048 701952 713470 714952 740328 748691 756225 780192 780729 792772 801448 805154 816129 828569 833822 834425 842240 857171 912488 922096 923929 942786 950664 959312 990208 1000542 1040384 1047856	
20	86	0.936110 (0.935997)	8	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 28211 51289 55520 123326 127489 155137 166782 173218 176697 203200 225555 264350 275934 289821 296247 297636 305552 307528 379905 396676 398210 440321 440804 468481 522338 534368 543332 543920 567618 593220 614692 615101 617984 641744 642530 673064 673854 676825 680048 701952 713470 714952 740328 748691 756225 780192 780729 792772 801448 805154 816129 828569 833822 834425 842240 857171 912488 922096 923929 942786 950664 959312 990208 1000542 1040384 1047856	
20	87	0.939388 (0.939387)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 28211 51289 55520 123326 127489 155137 166782 173218 176697 203200 225555 264350 275934 289821 297636 305552 307528 379905 396676 398210 440321 440804 468481 471543 522338 534368 543332 543920 567618 593220 614692 615101 617984 641744 642530 673064 673854 676825 680048 701952 713470 714952 740328 748691 756225 780192 780729 792772 801448 805154 816129 828569 833822 834425 842240 857171 864211 912488 922096 923929 942786 950664 959312 990208 1000542 1040384 1047856	
20	88	0.942590 (0.942554)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 28211 51289 55520 123326 127489 155137 166782 173218 176697 203200 225555 264350 275934 289821 297636 305552 307528 379905 396676 398210 440321 440804 468481 471543 522338 534368 543332 543920 567618 593220 614692 615101 617984 641744 642530 673064 673854 676825 680048 685879 701952 713470 714952 740328 748691 756225 780192 780729 792772 801448 805154 816129 828569 833822 834425 842240 857171 864211 912488 922096 923929 942786 950664 959312 990208 1000542 1040384 1047856	

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
20	89	0.94558	8	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 28211 51289 55520 109239 123326 127489 155137 166782 173218 176697 203200 225555 264350 275934 289821 297636 305552 307528 379905 396676 398210 440321 440804 468481 471543 522338 534368 543332 543920 567618 593220 614692 615101 617984 641744 642530 673064 673854 676825 680048 685879 701952 713470 714952 740328 748691 756225 780192 780729 792772 801448 805154 816129 828569 833822 834425 842240 857171 864211 912488 922096 923929 942786 950664 959312 990208 1000542 1040384 1047856
20	90	0.948187 (0.948116)	7	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 48281 55520 84409 103958 123326 151863 157501 166782 173218 190977 203200 226305 241203 264350 275934 278017 282638 297636 305552 307528 328793 368275 379905 396676 398210 440804 496129 513422 522338 534368 543332 543920 565529 567618 593220 600659 603571 614692 617984 636929 641744 642530 673064 673854 680048 693305 701952 713470 714952 740328 748573 780192 792772 799993 801448 805154 833822 836883 842240 912488 922096 935425 942786 950664 959312 966515 990208 1000542 1036153 1040384 1047856
20	91	0.950860 (0.950790)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 48281 55520 84409 123326 151863 157501 166782 173218 190977 203200 226305 241203 264350 275934 278017 282638 297636 305552 307528 328793 368275 379905 396676 398210 440804 496129 513422 522338 534368 543332 543920 565529 567618 593220 600659 603571 614692 617984 636929 641744 642530 673064 673854 680048 693305 701952 713470 714952 740328 748573 780192 792772 799993 801448 805154 833822 836883 842240 912488 922096 935425 942786 950664 959312 966515 990208 1000542 1024180 1036153 1040384 1047856
20	92	0.953647 (0.953361)	7	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 41772 49797 53076 70933 85654 95837 102185 121984 143743 159055 160966 162356 177792 186954 191057 197822 208965 259410 268572 272682 296292 308553 313067 320896 366336 371328 405616 439999 442424 445952 450779 457021 475870 491392 532261 539043 560237 602893 616832 625083 632370 646431 661559 680024 692328 692902 701447 716691 741146 746583 749617 756858 782531 789109 799616 801916 802585 810240 811874 814830 850234 892811 915559 923790 964172 985856 988915 996729 1022198 1026657 1028399 1044736
20	93	0.955866 (0.955808)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 48281 55520 84409 123326 151863 157501 166782 173218 174344 190977 203200 226305 241203 264350 275934 278017 282638 297636 305552 307528 328793 368275 379905 396676 398210 440804 496129 513422 522338 534368 543332 543920 565529 567618 593220 600659 603571 614692 617984 636929 641744 642530 673064 673854 680048 693305 701952 713470 714952 740328 748573 780192 792772 799993 801448 805154 833822 836883 842240 888066 912488 922096 935425 942786 950664 959312 966515 990208 1000542 1024180 1036153 1040384 1047856
20	94	0.958185 (0.958178)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 48281 55520 84409 123326 151863 157501 166782 173218 174344 190977 203200 226305 241203 264350 275934 278017 282638 297636 305552 307528 328793 368275 379905 396676 398210 440804 496129 513422 522338 534368 543332 543920 565529 567618 593220 600659 603571 614692 617984 636929 641744 642530 673064 673854 680048 693305 701952 713470 714952 740328 748573 777572 780192 792772 799993 801448 805154 833822 836883 842240 888066 912488 922096 935425 942786 950664 959312 966515 990208 1000542 1024180 1036153 1040384 1047856
20	95	0.960392	7	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 48281 55520 84409 123326 151863 157501 166782 173218 174344 190977 203200 226305 241203 264350 275934 278017 282638 297636 305552 307528 328793 368275 379905 396676 398210 440804 496129 513422 522338 534368 543332 543920 565529 567618 580698 593220 600659 603571 614692 617984 636929 641744 642530 673064 673854 680048 693305 701952 713470 714952 740328 748573 777572 780192 792772 799993 801448 805154 833822 836883 842240 888066 912488 922096 935425 942786 950664 959312 966515 990208 1000542 1024180 1036153 1040384 1047856
20	96	0.962543 (0.809674)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 48281 55520 84409 123326 151863 157501 166782 173218 174344 190977 203200 226305 241203 264350 275934 278017 282638 297636 305552 307528 328793 368275 374213 379905 396676 398210 440804 496129 513422 522338 534368 543332 543920 565529 567618 580698 593220 600659 603571 614692 617984 636929 641744 642530 673064 673854 680048 693305 701952 713470 714952 740328 748573 777572 780192 792772 799993 801448 805154 833822 836883 842240 888066 912488 922096 935425 942786 950664 959312 966515 990208 1000542 1024180 1036153 1040384 1047856
20	97	0.964564 (0.810443)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 48281 55520 84409 103958 123326 151863 157501 166782 173218 174344 190977 203200 226305 241203 264350 275934 278017 282638 297636 305552 307528 328793 368275 374213 379905 396676 398210 440804 496129 513422 522338 534368 543332 543920 565529 567618 580698 593220 600659 603571 614692 617984 636929 641744 642530 673064 673854 680048 693305 701952 713470 714952 740328 748573 777572 780192 792772 799993 801448 805154 833822 836883 842240 888066 912488 922096 935425 942786 950664 959312 966515 990208 1000542 1024180 1036153 1040384 1047856
20	98	0.966484 (0.811166)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 48281 55520 84409 90333 103958 123326 151863 157501 166782 173218 174344 190977 203200 226305 241203 264350 275934 278017 282638 297636 305552 307528 328793 368275 374213 379905 396676 398210 440804 496129 513422 522338 534368 543332 543920 565529 567618 580698 593220 600659 603571 614692 617984 636929 641744 642530 673064 673854 680048 693305 701952 713470 714952 740328 748573 777572 780192 792772 799993 801448 805154 833822 836883 842240 888066 912488 922096 935425 942786 950664 959312 966515 990208 1000542 1024180 1036153 1040384 1047856
20	99	0.968288 (0.811845)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 48281 55520 84409 90333 103958 123326 151863 157501 166782 173218 174344 190977 203200 226305 241203 264350 275934 278017 282638 297636 305552 307528 328793 368275 374213 379905 396676 398210 440804 496129 513422 522338 534368 543332 543920 565529 567618 580698 593220 600659 603571 614692 617984 636929 641744 642530 673064 673854 680048 693305 701952 713470 714952 740328 748573 777572 780192 792772 799993 801448 805154 833822 836883 842240 888066 912488 922096 935425 942786 950664 959312 966515 990208 1000542 1024180 1036153 1040384 1047856
20	100	0.970014 (0.812481)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 37295 48281 55520 84409 90333 103958 123326 151863 157501 166782 173218 174344 190977 203200 226305 241203 264350 275934 278017 282638 297636 305552 307528 328793 368275 374213 379905 396676 398210 440804 496129 513422 522338 534368 543332 543920 565529 567618 580698 593220 600659 603571 614692 617984 636929 641744 642530 673064 673854 680048 693305 701952 713470 714952 740328 748573 777572 780192 792772 799993 801448 805154 833822 836883 842240 859115 888066 912488 922096 935425 942786 950664 959312 966515 990208 1000542 1024180 1036153 1040384 1047856

Continued on next page

	m	n	R_e	d	R_c	Packed integer form of the parity check matrix
20	101		0.971652 (0.813078)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 37295 48281 55520 84409 90333 103958 123326 151863 157501 166782 173218 174344 190977 203200 226305 241203 264350 275934 278017 282638 297636 305552 307528 328793 355045 368275 374213 379905 396676 398210 440804 496129 513422 522338 534368 543332 543920 565529 567618 580698 593220 600659 603571 614692 617984 636929 641744 642530 673064 673854 680048 693305 701952 713470 714952 740328 748573 777572 780192 792772 799993 801448 805154 833822 836883 842240 859115 888066 912488 922096 935425 942786 950664 959312 966515 990208 1000542 1024180 1036153 1040384 1047856
20	102		0.973197 (0.813640)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 37295 48281 55520 84409 90333 103958 107439 123326 151863 157501 166782 173218 174344 190977 203200 226305 241203 264350 275934 278017 282638 297636 305552 307528 328793 355045 368275 374213 379905 396676 398210 440804 496129 513422 522338 534368 543332 543920 565529 567618 580698 593220 600659 603571 614692 617984 636929 641744 642530 673064 673854 680048 693305 701952 713470 714952 740328 748573 777572 780192 792772 799993 801448 805154 833822 836883 842240 859115 888066 912488 922096 935425 942786 950664 959312 966515 990208 1000542 1024180 1036153 1040384 1047856
20	103		0.974663 (0.814166)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 37295 48281 55520 84409 90333 103958 107439 123326 151863 157501 166782 173218 174344 190977 203200 226305 241203 264350 275934 277460 278017 282638 297636 305552 307528 328793 355045 368275 374213 379905 396676 398210 440804 496129 513422 522338 534368 543332 543920 565529 567618 580698 593220 600659 603571 614692 617984 636929 641744 642530 673064 673854 680048 693305 701952 713470 714952 740328 748573 777572 780192 792772 799993 801448 805154 833822 836883 842240 859115 888066 912488 922096 935425 942786 950664 959312 966515 990208 1000542 1024180 1036153 1040384 1047856
20	104		0.976057 (0.814659)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 37295 48281 55520 84409 90333 103958 107439 123326 151863 157501 166782 173218 174344 190977 203200 226305 241203 264350 275934 277460 278017 282638 297636 305552 307528 328793 355045 368275 374213 379905 396676 398210 440804 496129 513422 522338 534368 543332 543920 565529 567618 580698 593220 600659 603571 614692 617984 636929 641744 642530 673064 673854 680048 693305 701952 713470 714952 740328 748573 777572 780192 792772 799993 801448 805154 833822 836883 842240 859115 888066 912488 922096 935425 942786 950664 959312 966515 990208 994054 1000542 1024180 1036153 1040384 1047856
20	105		0.977375 (0.815122)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 37295 48281 55520 84409 90333 103958 107439 123326 151863 157501 166782 173218 174344 190977 195255 203200 226305 241203 264350 275934 277460 278017 282638 297636 305552 307528 328793 355045 368275 374213 379905 396676 398210 440804 496129 513422 522338 534368 543332 543920 565529 567618 580698 593220 600659 603571 614692 617984 636929 641744 642530 673064 673854 680048 693305 701952 713470 714952 740328 748573 777572 780192 792772 799993 801448 805154 833822 836883 842240 859115 888066 912488 922096 935425 942786 950664 959312 966515 990208 994054 1000542 1024180 1036153 1040384 1047856
20	106		0.978626 (0.815558)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 37295 48281 55520 84409 90333 103958 107439 123326 151863 157501 166782 173218 174344 190977 195255 203200 226305 241203 261676 264350 275934 277460 278017 282638 297636 305552 307528 328793 355045 368275 374213 379905 396676 398210 440804 496

157

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
20	112	0.984902 (0.817654)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 37295 48281 55520 83934 84409 90333 103958 107439 123326 151863 157501 166782 173218 174344 190977 195255 203200 226305 241203 261676 264350 275934 277460 278017 282638 297636 305552 307528 325040 328793 355045 368275 374213 379905 396676 398210 440804 496129 513422 522338 534368 537195 543332 543920 565529 567618 580698 593220 600659 603571 614692 617984 636929 641744 642530 673064 673854 680048 693163 693305 699891 701952 713470 714952 740328 748573 777572 780192 792772 799993 801448 805154 833822 836883 842240 859115 888066 912488 922096 933126 935425 942786 950664 959312 966515 990208 994054 1000542 1024180 1036153 1040384 1047856
20	113	0.985764 (0.817928)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 37295 48281 55520 83934 84409 90333 103958 107439 123326 151863 157501 166782 173218 174344 190977 195255 203200 226305 241203 261676 264350 275934 277460 278017 282638 297636 305552 307528 325040 328793 355045 368275 374213 379905 396676 398210 440804 496129 513422 522338 534368 537195 543332 543920 565529 567618 580698 593220 600659 603571 614692 617984 636929 641744 642530 673064 673854 680048 693163 693305 699891 701952 713470 714952 740328 748573 777572 780192 792772 799993 801448 805154 833822 836883 842240 859115 888066 912488 922096 933126 935425 942786 950664 959312 966515 978891 990208 994054 1000542 1024180 1036153 1040384 1047856
20	114	0.986582 (0.818184)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 37295 48281 55520 83934 84409 90333 103958 107439 123326 151863 157501 166782 173218 174344 190977 195255 203200 226305 237127 241203 261676 264350 275934 277460 278017 282638 297636 305552 307528 325040 328793 355045 368275 374213 379905 396676 398210 440804 496129 513422 522338 534368 537195 543332 543920 565529 567618 580698 593220 600659 603571 614692 617984 636929 641744 642530 673064 673854 680048 693163 693305 699891 701952 713470 714952 740328 748573 777572 780192 792772 799993 801448 805154 833822 836883 842240 859115 888066 912488 922096 933126 935425 942786 950664 959312 966515 978891 990208 994054 1000542 1024180 1036153 1040384 1047856
20	115	0.987356 (0.818423)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 37295 48281 55520 83934 84409 90333 103958 107439 123326 151863 157501 166782 173218 174344 190977 195255 203200 212443 226305 237127 241203 261676 264350 275934 277460 278017 282638 297636 305552 307528 325040 328793 355045 368275 374213 379905 396676 398210 440804 496129 513422 522338 534368 537195 543332 543920 565529 567618 580698 593220 600659 603571 614692 617984 636929 641744 642530 673064 673854 680048 693163 693305 699891 701952 713470 714952 740328 748573 777572 780192 792772 799993 801448 805154 833822 836883 842240 859115 888066 912488 922096 933126 935425 942786 950664 959312 966515 978891 990208 994054 1000542 1024180 1036153 1040384 1047856
20	116	0.988089 (0.818646)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 37295 48281 55520 83934 84409 90333 103958 107439 123326 151863 157501 166782 173218 174344 190977 195255 203200 212443 226305 237127 241203 261676 264350 275934 277460 278017 282638 297636 305552 307528 325040 328793 355045 368275 374213 379905 396676 398210 440804 496129 513422 522338 534368 537195 543332 543920 565529 567618 580698 593220 600659 603571 614692 617984 636929 641744 642530 673064 673854 680048 693163 693305 699891 701952 713470 714952 740328 748573 777572 780192 792772 799993 801448 805154 833822 836883 842240 854988 859115 888066 912488 922096 933126 935425 942786 950664 959312 966515 978891 990208 994054 1000542 1024180 1036153 1040384 1047856
20	117	0.988782 (0.818854)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 37295 48281 55520 83934 84409 90333 103958 107439 123326 151863 157501 166782 173218 174344 190977 195255 203200 212443 226305 237127 241203 261676 264350 275934 277460 278017 282638 297636 305552 307528 325040 328793 355045 368275 374213 379905 396676 398210 440804 496129 513422 522338 534368 537195 543332 543920 565529 567618 580698 593220 600659 603571 614692 617984 636929 641744 642530 673064 673854 680048 693163 693305 699891 701952 713470 714952 740328 748573 777572 780192 792772 799993 801448 805154 831293 833822 836883 842240 854988 859115 888066 912488 922096 933126 935425 942786 950664 959312 966515 978891 990208 994054 1000542 1024180 1036153 1040384 1047856
20	118	0.989438 (0.819048)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 37295 48281 55520 83934 84409 90333 102953 103958 107439 123326 151863 157501 166782 173218 174344 190977 195255 203200 212443 226305 237127 241203 261676 264350 275934 277460 278017 282638 297636 305552 307528 325040 328793 355045 368275 374213 379905 396676 398210 440804 496129 513422 522338 534368 537195 543332 543920 565529 567618 580698 593220 600659 603571 614692 617984 636929 641744 642530 673064 673854 680048 693163 693305 699891 701952 713470 714952 740328 748573 777572 780192 792772 799993 801448 805154 831293 833822 836883 842240 854988 859115 888066 912488 922096 933126 935425 942786 950664 959312 966515 978891 990208 994054 1000542 1024180 1036153 1040384 1047856
20	119	0.990058 (0.819230)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 7340 37295 48281 55520 83934 84409 90333 102953 103958 107439 123326 151863 157501 166782 173218 174344 190977 195255 203200 212443 226305 237127 241203 261676 264350 275934 277460 278017 282638 297636 305552 307528 325040 328793 355045 368275 374213 379905 396676 398210 440804 496129 513422 522338 534368 537195 543332 543920 565529 567618 580698 593220 600659 603571 614692 617984 636929 641744 642530 673064 673854 680048 693163 693305 699891 701952 713470 714952 740328 748573 777572 780192 792772 799993 801448 805154 831293 833822 836883 842240 854988 859115 888066 912488 922096 933126 935425 942786 950664 959312 966515 978891 990208 994054 1000542 1024180 1036153 1040384 1047856
20	120	0.990644 (0.819399)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 7340 37295 48281 55520 83934 84409 90333 102953 103958 107439 123326 151863 157501 166782 173218 174344 190977 195255 203200 212443 226305 237127 241203 261676 264350 275934 277460 278017 282638 297636 305552 307528 325040 328793 355045 368275 374213 379905 396676 398210 440804 496129 513422 522338 534368 537195 543332 543920 565529 567618 580698 593220 600659 603571 614692 617984 636929 641744 642530 673064 673854 680048 683111 693163 693305 699891 701952 713470 714952 740328 748573 777572 780192 792772 799993 801448 805154 831293 833822 836883 842240 854988 859115 888066 912488 922096 933126 935425 942786 950664 959312 966515 978891 990208 994054 1000542 1024180 1036153 1040384 1047856
22	23	0.299415	23	11	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194303
22	24	0.312433	16	12	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 1295359 4194176
22	25	0.32545	13	12	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 228495 1295359 4194176
22	26	0.338466	12	12	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 228495 351078 1295359 4194176
22	27	0.351481	12	11	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 228495 351078 1295359 3872309 4194176
22	28	0.364494 (0.364479)	12	10	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 228495 351078 1295359 3076700 3872309 4194176

Continued on next page

Table B.1 – Continued from previous page					
m	n	R_e	d	R_c	Packed integer form of the parity check matrix
22	29	0.377500 (0.377481)	12	10	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 279519 622158 1294335 2428023 3058921 3295866 3725615
22	30	0.390492 (0.390474)	11	10	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 228495 351078 1295359 1855865 2461933 3076700 3872309 4194176
22	31	0.403481 (0.403455)	12	10	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 881570 1291684 1372223 1420632 2545320 3628639 3729200 4130752 4192256
22	32	0.416451 (0.416420)	11	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 228495 351078 1295359 1855865 2433367 2461933 3076700 3153383 3872309 4194176
22	33	0.429405 (0.429404)	12	10	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 55727 252505 881570 1291684 1420632 2185061 2545320 3710278 3729200 4130752 4192256
22	34	0.44234	12	10	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 252505 881570 1291684 1346735 1420632 2131931 2185061 2545320 3710278 3729200 4130752 4192256
22	35	0.455220 (0.455195)	11	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 252505 881570 1291684 1346735 1420632 1457079 2131931 2185061 2545320 3710278 3729200 4130752 4192256
22	36	0.468056	11	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 735983 903908 958451 1010728 1213821 1301366 1360334 1450558 1885739 2675341 3108055 3356126 3514459 4183942
22	37	0.480834 (0.480485)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 735983 903908 958451 1010728 1213821 1301366 1360334 1450558 1885739 2675341 2789483 3108055 3356126 3514459 4183942
22	38	0.493567 (0.493156)	10	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 735983 903908 958451 1010728 1213821 1301366 1360334 1450558 1885739 1962217 2675341 2789483 3108055 3356126 3514459 4183942
22	39	0.506224 (0.505754)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 145286 373036 928356 1013475 1141835 1280665 2026950 2169795 2283670 2561330 2833145 3306068 3342653 3537234 3913496 3935337 4053900
22	40	0.518846 (0.518269)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 145286 373036 928356 1013475 1141835 1280665 1684051 2026950 2169795 2283670 2561330 2833145 3306068 3342653 3537234 3913496 3935337 4053900
22	41	0.531389 (0.530693)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 145286 373036 928356 1013475 1141835 1280665 1684051 2026950 2169795 2283670 2561330 2833145 3306068 3342653 3537234 3865769 3913496 3935337 4053900
22	42	0.543845 (0.543015)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 145286 373036 928356 1013475 1141835 1280665 1684051 2026950 2169795 2283670 2561330 2833145 3306068 3342653 3537234 3632689 3865769 3913496 3935337 4053900
22	43	0.556209	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 145286 373036 928356 1013475 1141835 1280665 1684051 1875093 2026950 2169795 2283670 2561330 2833145 3306068 3342653 3537234 3632689 3865769 3913496 3935337 4053900
22	44	0.568471	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 145286 373036 928356 1013475 1141835 1280665 1684051 1875093 2026950 2169795 2283670 2561330 2833145 3306068 3342653 3537234 3632689 3750186 3865769 3913496 3935337 4053900
22	45	0.580621	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 145286 373036 928356 1013475 1141835 1280665 1684051 1875093 2026950 2169795 2283670 2541900 2561330 2833145 3306068 3342653 3537234 3632689 3750186 3865769 3913496 3935337 4053900
22	46	0.59265	10	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 145286 373036 928356 1013475 1141835 1280665 1684051 1875093 2026950 2169795 2283670 2541900 2561330 2833145 3306068 3342653 3368102 3537234 3632689 3750186 3865769 3913496 3935337 4053900
22	47	0.603302	9	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 258194 440310 467888 720744 772748 809440 861870 939402 1273560 1384062 1476826 1651276 1723962 2244144 2445624 2460606 2667854 2741746 3022827 3160166 3170218 3314354 3502750 3703764 3964990
22	48	0.615328	9	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 258194 440310 467888 720744 772748 809440 861870 939402 1273560 1384062 1476826 1651276 1723962 2244144 2445624 2460606 2667854 2741746 2979165 3022827 3160166 3170218 3314354 3502750 3703764 3964990
22	49	0.627066	9	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 232227 258194 440310 467888 720744 772748 809440 861870 910217 939402 1273560 1384062 1476826 1651276 1723962 2244144 2445624 2460606 2630865 2667854 2741746 3160166 3170218 3314354 3502750 3703764 3964990
22	50	0.638550 (0.623228)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 232227 258194 440310 467888 720744 772748 809440 861870 910217 939402 1273560 1384062 1473071 1476826 1651276 1723962 2244144 2445624 2460606 2630865 2667854 2741746 3160166 3170218 3314354 3502750 3703764 3964990
22	51	0.649813 (0.632849)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 232227 258194 296475 440310 467888 720744 772748 809440 861870 910217 939402 1273560 1384062 1473071 1476826 1651276 1723962 2244144 2445624 2460606 2630865 2667854 2741746 3160166 3170218 3314354 3502750 3703764 3964990
22	52	0.660865 (0.642242)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 232227 258194 296475 384511 440310 467888 720744 772748 809440 861870 910217 939402 1273560 1384062 1473071 1476826 1651276 1723962 2244144 2445624 2460606 2630865 2667854 2741746 3160166 3170218 3314354 3502750 3703764 3964990
22	53	0.671712 (0.651390)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 232227 258194 296475 384511 440310 467888 720744 772748 809440 861870 910217 939402 1273560 1384062 1473071 1476826 1651276 1723962 1891133 2244144 2445624 2460606 2630865 2667854 2741746 3160166 3170218 3314354 3502750 3703764 3964990
22	54	0.682358 (0.660306)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 232227 258194 296475 384511 440310 467888 720744 772748 809440 861870 910217 939402 959077 1273560 1384062 1473071 1476826 1651276 1723962 1891133 2244144 2445624 2460606 2630865 2667854 2741746 3160166 3170218 3314354 3502750 3703764 3964990
22	55	0.692801 (0.668967)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 232227 258194 296475 384511 440310 467888 720744 772748 809440 861870 910217 939402 959077 1273560 1384062 1473071 1476826 1651276 1723962 1891133 2244144 2445624 2460606 2630865 2667854 2741746 3160166 3170218 3301029 3314354 3502750 3703764 3964990

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
22	56	0.703049 (0.677377)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 232227 258194 296475 384511 440310 467888 720744 772748 809440 861870 910217 939402 959077 1273560 1384062 1473071 1476826 1651276 1723962 1891133 2244144 2445624 2460606 2630865 2667854 2741746 2870491 3160166 3170218 3301029 3314354 3502750 3703764 3964990
22	57	0.713078 (0.685540)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 232227 258194 296475 384511 440310 467888 720744 772748 809440 835545 861870 910217 939402 959077 1273560 1384062 1473071 1476826 1651276 1723962 1891133 2244144 2445624 2460606 2630865 2667854 2741746 2870491 3160166 3170218 3301029 3314354 3502750 3703764 3964990
22	58	0.722891 (0.693455)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 232227 258194 296475 384511 440310 467888 720744 772748 809440 835545 861870 910217 939402 959077 1157201 1273560 1384062 1473071 1476826 1651276 1723962 1891133 2244144 2445624 2460606 2630865 2667854 2741746 2870491 3160166 3170218 3301029 3314354 3502750 3703764 3964990
22	59	0.732488 (0.701135)	8	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 232227 258194 296475 384511 440310 467888 720744 772748 809440 835545 861870 910217 939402 959077 1157201 1273560 1384062 1473071 1476826 1651276 1723962 1891133 2244144 2445624 2460606 2630865 2667854 2741746 2870491 3024591 3160166 3170218 3301029 3314354 3502750 3703764 3964990
22	60	0.740955 (0.708565)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 111264 120844 135402 309690 330532 338475 567303 682894 714276 923298 982920 1351071 1363702 1376188 1419610 1551602 1884223 2073026 2289013 2314096 2376601 2501526 2687586 2703451 2758616 2779212 2825279 3047974 3082337 3167196 3212496 3367783 3595701 3675563 3727842 3753502 3954669 4013090
22	61	0.750226 (0.715750)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 111264 120844 135402 309690 330532 338475 449842 567303 682894 714276 923298 982920 1351071 1363702 1376188 1419610 1551602 1884223 2073026 2289013 2314096 2376601 2501526 2687586 2703451 2758616 2779212 2825279 3047974 3082337 3167196 3212496 3367783 3595701 3675563 3727842 3753502 3954669 4013090
22	62	0.759260 (0.722696)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 111264 120844 135402 158960 309690 330532 338475 449842 567303 682894 714276 923298 982920 1351071 1363702 1376188 1419610 1551602 1884223 2073026 2289013 2314096 2376601 2501526 2687586 2703451 2758616 2779212 2825279 3047974 3082337 3167196 3212496 3367783 3595701 3675563 3727842 3753502 3954669 4013090
22	63	0.768078 (0.729407)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 111264 120844 135402 158960 309690 330532 338475 449842 567303 682894 714276 923298 982920 1351071 1363702 1376188 1419610 1551602 1884223 2073026 2289013 2314096 2376601 2501526 2687586 2703451 2758616 2779212 2825279 3047974 3082337 3167196 3212496 3367783 3595701 3675563 3727842 3753502 3954669 4006192 4013090
22	64	0.776669 (0.735885)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 111264 120844 135402 158960 309690 330532 338475 449842 567303 682894 714276 923298 982920 1351071 1363702 1376188 1419610 1551602 1884223 2073026 2289013 2314096 2376601 2501526 2687586 2703451 2758616 2779212 2825279 2977421 3047974 3082337 3167196 3212496 3367783 3595701 3675563 3727842 3753502 3954669 4006192 4013090
22	65	0.785026 (0.752731)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 111264 120844 135402 158960 309690 330532 338475 449842 567303 682894 714276 923298 982920 1349323 1351071 1363702 1376188 1419610 1551602 1884223 2073026 2289013 2314096 2376601 2501526 2687586 2703451 2758616 2779212 2825279 2977421 3047974 3082337 3167196 3212496 3367783 3595701 3675563 3727842 3753502 3954669 4006192 4013090
22	66	0.793184 (0.759631)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 111264 120844 135402 158960 309690 330532 338475 449842 567303 615152 682894 714276 923298 982920 1349323 1351071 1363702 1376188 1419610 1551602 1884223 2073026 2289013 2314096 2376601 2501526 2687586 2703451 2758616 2779212 2825279 2977421 3047974 3082337 3167196 3212496 3367783 3595701 3675563 3727842 3753502 3954669 4006192 4013090
22	67	0.801098 (0.766177)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 111264 120844 135402 158960 309690 330532 338475 449842 567303 615152 682894 714276 923298 982920 1349323 1351071 1363702 1376188 1419610 1551602 1884223 2073026 2289013 2314096 2376601 2501526 2535860 2687586 2703451 2758616 2779212 2825279 2977421 3047974 3082337 3167196 3212496 3367783 3595701 3675563 3727842 3753502 3954669 4006192 4013090
22	68	0.808780 (0.772415)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 111264 120844 135402 158960 309690 330532 338475 449842 567303 615152 682894 683049 714276 923298 982920 1349323 1351071 1363702 1376188 1419610 1551602 1884223 2073026 2289013 2314096 2376601 2501526 2535860 2687586 2703451 2758616 2779212 2825279 2977421 3047974 3082337 3167196 3212496 3367783 3595701 3675563 3727842 3753502 3954669 4006192 4013090
22	69	0.816243 (0.778305)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 111264 120844 135402 158960 309690 330532 338475 449842 567303 615152 682894 683049 714276 923298 982920 1349323 1351071 1363702 1376188 1419610 1551602 1884223 2073026 2289013 2314096 2376601 2501526 2535860 2687586 2703451 2758616 2779212 2825279 2977421 3047974 3082337 3167196 3212496 3367783 3595701 3675563 3727842 3746293 3753502 3954669 4006192 4013090
22	70	0.823495 (0.794018)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 111264 120844 135402 158960 309690 330532 338475 449842 567303 615152 682894 683049 714276 923298 982920 1349323 1351071 1363702 1376188 1419610 1551602 1884223 2073026 2289013 2314096 2339310 2376601 2501526 2535860 2687586 2703451 2758616 2779212 2825279 2977421 3047974 3082337 3167196 3212496 3367783 3595701 3675563 3727842 3746293 3753502 3954669 4006192 4013090
22	71	0.830520 (0.800688)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 111264 120844 135402 158960 309690 330532 338475 449842 567303 615152 682894 683049 714276 923298 982920 1267239 1349323 1351071 1363702 1376188 1419610 1551602 1884223 2073026 2289013 2314096 2339310 2376601 2501526 2535860 2687586 2703451 2758616 2779212 2825279 2977421 3047974 3082337 3167196 3212496 3367783 3595701 3675563 3727842 3746293 3753502 3954669 4006192 4013090
22	72	0.837327 (0.806350)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 111264 120844 135402 158960 309690 330532 338475 449842 567303 615152 682894 683049 714276 923298 982920 1267239 1349323 1351071 1363702 1376188 1401888 1419610 1551602 1884223 2073026 2289013 2314096 2339310 2376601 2501526 2535860 2687586 2703451 2758616 2779212 2825279 2977421 3047974 3082337 3167196 3212496 3367783 3595701 3675563 3727842 3746293 3753502 3954669 4006192 4013090

Continued on next page

	m	n	R_e	d	R_c	Packed integer form of the parity check matrix
	22	73	0.843926 (0.812535)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 111264 120844 135402 158960 309690 330532 338475 449842 567303 615152 682894 683049 714276 923298 982920 1267239 1349323 1351071 1363702 1376188 1401888 1419610 1551602 1884223 2073026 2289013 2314096 2339310 2376601 2501526 2535860 2687586 2703451 2758616 2779212 2825279 2977421 3047974 3082337 3163304 3167196 3212496 3367783 3395701 3675563 3727842 3746293 3753502 3954669 4006192 4013090
	22	74	0.850313 (0.818326)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 111264 120844 135402 158960 309690 330532 338475 449842 567303 615152 682894 683049 714276 923298 982920 1267239 1349323 1351071 1363702 1376188 1401888 1419610 1551602 1884223 2036993 2073026 2289013 2314096 2339310 2376601 2501526 2535860 2687586 2703451 2758616 2779212 2825279 2977421 3047974 3082337 3163304 3167196 3212496 3367783 3395701 3675563 3727842 3746293 3753502 3954669 4006192 4013090
	22	75	0.856491 (0.823752)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 111264 120844 135402 158960 309690 330532 338475 449842 567303 615152 682894 683049 714276 923298 982920 1267239 1349323 1351071 1363702 1376188 1401888 1419610 1551602 1884223 2036993 2073026 2289013 2314096 2339310 2376601 2501526 2535860 2687586 2703451 2758616 2779212 2825279 2977421 3047974 3082337 3163304 3167196 3212496 3367783 3405303 3595701 3675563 3727842 3746293 3753502 3954669 4006192 4013090
	22	76	0.862470 (0.828648)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 111264 120844 135402 158960 309690 330532 338475 449842 567303 615152 682894 683049 714276 923298 982920 1267239 1349323 1351071 1363702 1376188 1401888 1419610 1551602 1884223 2036993 2073026 2289013 2314096 2339310 2376601 2501526 2535860 2687586 2703451 2758616 2779212 2825279 2977421 3047974 3082337 3163304 3167196 3212496 3367783 3391777 3405303 3595701 3675563 3727842 3746293 3753502 3954669 4006192 4013090
	22	77	0.868247 (0.833626)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 111264 120844 135402 158960 309690 330532 338475 449842 567303 615152 682894 683049 714276 923298 982920 1267239 1349323 1351071 1363702 1376188 1401888 1419610 1551602 1644838 1884223 2036993 2073026 2289013 2314096 2339310 2376601 2501526 2535860 2687586 2703451 2758616 2779212 2825279 2977421 3047974 3082337 3163304 3167196 3212496 3367783 3391777 3405303 3595701 3675563 3727842 3746293 3753502 3954669 4006192 4013090
	22	78	0.873827 (0.838406)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 111264 120844 135402 158960 309690 330532 338475 449842 567303 615152 682894 683049 714276 923298 982920 1267239 1349323 1351071 1363702 1376188 1401888 1419610 1551602 1644838 1884223 2036993 2073026 2289013 2314096 2339310 2376601 2501526 2535860 2687586 2703451 2758616 2779212 2825279 2977421 3047974 3082337 3163304 3167196 3212496 3367783 3391777 3405303 3575932 3595701 3675563 3727842 3746293 3753502 3954669 4006192 4013090
	22	79	0.879213 (0.842805)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 54309 111264 120844 135402 158960 309690 330532 338475 449842 567303 615152 682894 683049 714276 923298 982920 1267239 1349323 1351071 1363702 1376188 1401888 1419610 1551602 1644838 1884223 2036993 2073026 2289013 2314096 2339310 2376601 2501526 2535860 2687586 2703451 2758616 2779212 2825279 2977421 3047974 3082337 3163304 3167196 3212496 3367783 3391777 3405303 3575932 3595701 3675563 3727842 3746293 3753502 3

161

	m	n	R_e	d	R_c	Packed integer form of the parity check matrix
	22	99	0.953224 (0.953198)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 1419135064 56625 111264 120844 135402 156066 193837 197083 297854 309690 330532 338475 430909 437603 472518 567303 682894 714276 869995 905402 923298 982920 1018059 1149582 1351071 1376188 1380394 1380845 1419610 1487155 1551602 1884223 1964241 2073026 2141889 2289013 2314096 2350121 2357441 2376601 2422209 2501526 2638285 2645293 2683664 2687586 2703451 2758616 2779212 2803680 2825279 2893319 3047974 3082337 3148814 3167196 3172106 3212496 3222837 3244862 3347185 3367783 3595701 3671892 3675563 3727842 3733852 3753502 3954669 3966529 3984378 3994343 4013090 4134793 4137089 4153396
	22	100	0.955558 (0.955526)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 1419135064 56625 111264 120844 135402 156066 193837 197083 297854 309690 330532 338475 430909 437603 472518 567303 682894 714276 869995 905402 923298 982920 1018059 1149582 1351071 1376188 1380394 1380845 1419610 1487155 1551602 1788591 1884223 1964241 2073026 2141889 2289013 2314096 2350121 2357441 2376601 2422209 2501526 2638285 2645293 2683664 2687586 2703451 2758616 2779212 2803680 2825279 2893319 3047974 3082337 3148814 3167196 3172106 3212496 3222837 3244862 3347185 3367783 3595701 3671892 3675563 3727842 3733852 3753502 3954669 3966529 3984378 3994343 4013090 4134793 4137089 4153396
	22	101	0.957789 (0.957745)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 1419135064 56625 111264 120844 135402 156066 193837 197083 297854 309690 330532 338475 430909 437603 472518 567303 682894 714276 869995 905402 923298 982920 1018059 1149582 1351071 1376188 1380394 1380845 1419610 1487155 1551602 1788591 1884223 1964241 2073026 2112114 2141889 2289013 2314096 2350121 2357441 2376601 2422209 2501526 2638285 2645293 2683664 2687586 2703451 2758616 2779212 2803680 2825279 2893319 3047974 3082337 3148814 3167196 3172106 3212496 3222837 3244862 3347185 3367783 3595701 3671892 3675563 3727842 3733852 3753502 3954669 3966529 3984378 3994343 4013090 4134793 4137089 4153396
	22	102	0.959921 (0.959897)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 1419135064 56625 111264 120844 135402 156066 193837 197083 297854 309690 330532 338475 430909 437603 472518 567303 682894 714276 869995 905402 923298 982920 1018059 1149582 1351071 1376188 1380394 1380845 1419610 1487155 1551602 1788591 1884223 1964241 2073026 2112114 2141889 2289013 2314096 2350121 2357441 2376601 2403317 2422209 2501526 2638285 2645293 2683664 2687586 2703451 2758616 2779212 2803680 2825279 2893319 3047974 3082337 3148814 3167196 3172106 3212496 3222837 3244862 3347185 3367783 3595701 3671892 3675563 3727842 3733852 3753502 3954669 3966529 3984378 3994343 4013090 4134793 4137089 4153396
	22	103	0.961959 (0.961934)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 1419135064 56625 111264 120844 135402 156066 193837 197083 297854 309690 330532 338475 430909 437603 472518 567303 682894 714276 869995 905402 923298 982920 1018059 1149582 1351071 1376188 1380394 1380845 1419610 1487155 1551602 1788591 1884223 1964241 2073026 2112114 2141889 2289013 2314096 2350121 2357441 2376601 2403317 2422209 2501526 2638285 2645293 2683664 2687586 2703451 2758616 2779212 2803680 2825279 2893319 3047974 3082337 3148814 3167196 3172106 3212496 3222837 3229251 3244862 3347185 3367783 3595701 3671892 3675563 3727842 3733852 3753502 3954669 3966529 3984378 3994343 4013090 4134793 4137089 4153396
	22	104	0.963904 (0.963880)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 1419135064 56625 111264 1208

163

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
22	109	0.972371 (0.972356)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 14191 35064 56625 111264 120844 135402 156066 193837 197083 297854 309690 321677 330532 338475 430909 437603 472518 567303 682894 714276 720942 869995 905402 923298 982920 1018059 1144286 1149582 1351071 1376188 1380394 1380845 1400615 1419610 1487155 1551602 1788591 1884223 1964241 2073026 2112114 2141889 2289013 2314096 2350121 2357441 2376601 2403317 2422209 2501526 2638285 2645293 2683664 2687586 2703451 2758616 2779212 2803680 2825279 2893319 2937230 3047974 3082337 3148814 3167196 3172106 3212496 3222837 3229251 3244862 3347185 3354297 3367783 3578876 3595701 3671892 3675563 3727842 3733852 3753502 3954669 3966529 3984378 3994343 4013090 4134793 4137089 4153396
22	110	0.973835 (0.973822)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 14191 35064 56625 111264 120844 135402 156066 193837 197083 297854 309690 321677 330532 338475 430909 437603 472518 567303 682894 714276 720942 869995 905402 923298 982920 1018059 1144286 1149582 1351071 1376188 1380394 1380845 1400615 1419610 1487155 1551602 1788591 1884223 1964241 2073026 2112114 2141889 2289013 2314096 2350121 2357441 2376601 2403317 2422209 2501526 2638285 2645293 2683664 2687586 2703451 2758616 2779212 2803680 2825279 2893319 2937230 3047974 3082337 3148814 3167196 3172106 3212496 3222837 3229251 3244862 3347185 3354297 3367783 3578876 3595701 3671892 3675563 3727842 3733852 3753502 3954669 3966529 3984378 3994343 4013090 4134793 4137089 4153396
22	111	0.975228 (0.975219)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 14191 35064 56625 111264 120844 135402 156066 193837 197083 297854 309690 321677 330532 338475 430909 437603 472518 567303 682894 714276 720942 869995 905402 923298 982920 1018059 1144286 1149582 1351071 1376188 1380394 1380845 1400615 1419610 1487155 1551602 1788591 1884223 1964241 2073026 2112114 2141889 2289013 2314096 2350121 2357441 2376601 2403317 2422209 2501526 2638285 2645293 2683664 2687586 2703451 2758616 2779212 2803680 2825279 2893319 2937230 3047974 3082337 3148814 3167196 3172106 3212496 3222837 3229251 3244862 3347185 3354297 3367783 3578876 3595701 3671892 3675563 3727842 3733852 3753502 3954669 3966529 3984378 3994343 4013090 4113367 4134793 4137089 4153396
22	112	0.976555 (0.976548)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 14191 35064 56625 111264 120844 135402 156066 193837 197083 297854 309690 321677 330532 338475 430909 437603 472518 567303 682894 714276 720942 869995 905402 923298 982920 1018059 1144286 1149582 1351071 1376188 1380394 1380845 1400615 1419610 1487155 1551602 1788591 1884223 1964241 1999207 2073026 2112114 2141889 2289013 2314096 2350121 2357441 2376601 2403317 2422209 2501526 2638285 2645293 2683664 2687586 2703451 2758616 2779212 2803680 2825279 2893319 2937230 3047974 3082337 3148814 3167196 3172106 3212496 3222837 3229251 3244862 3347185 3354297 3367783 3578876 3595701 3671892 3675563 3727842 3733852 3753502 3954669 3966529 3984378 3994343 4013090 4113367 4134793 4137089 4153396
22	113	0.977817 (0.977813)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 14191 35064 56625 111264 120844 135402 156066 193837 197083 297854 309690 321677 330532 338475 430909 437603 472518 567303 682894 714276 720942 869995 905402 923298 982920 1018059 1144286 1149582 1351071 1376188 1380394 1380845 1400615 1419610 1487155 1551602 1788591 1884223 1964241 1999207 2073026 2112114 2141889 2289013 2314096 2350121 2357441 2376601 2403317 2422209 2501526 2638285 2645293 2683664 2687586 2703451 2758616 2779212 2803680 2825279 2893319 2937230 3047974 3082337 3148814 3167196 3172106 3

164

	m	n	R_e	d	R_c	Packed integer form of the parity check matrix
22	118		0.983257 (0.983254)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 41919 35064 56625 111264 120844 135402 156066 193837 197083 297854 309690 321677 330532 338475 430909 437603 472518 530206 567303 682894 714276 720942 869995 905402 923298 982920 1018059 1144286 1149582 1351071 1376188 1380394 1380845 1400615 1419610 1487155 1551602 1567285 1681021 1734457 1788591 1884223 1964241 1999207 2073026 2112114 2141889 2289013 2314096 2350121 2357441 2376601 2403317 2422209 2501526 2638285 2645293 2683664 2687586 2703451 2758616 2779212 2803680 2825279 2893319 2937230 3047974 3082337 3148814 3167196 3172106 3212496 3222837 3229251 3244862 3315690 3347185 3354297 3367783 3578876 3595701 3671892 3675563 3727842 3731531 3733852 3753502 3954669 3966529 3984378 3994343 4013090 4113367 4134793 4137089 4153396
22	119		0.984188 (0.984185)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 41919 35064 56625 111264 120844 135402 156066 193837 197083 297854 309690 321677 330532 338475 430909 437603 472518 530206 567303 682894 714276 720942 869995 905402 923298 982920 1018059 1144286 1149582 1351071 1376188 1380394 1380845 1400615 1419610 1487155 1551602 1567285 1681021 1734457 1788591 1884223 1964241 1999207 2039135 2073026 2112114 2141889 2289013 2314096 2350121 2357441 2376601 2403317 2422209 2501526 2638285 2645293 2683664 2687586 2703451 2758616 2779212 2803680 2825279 2893319 2937230 3047974 3082337 3148814 3167196 3172106 3212496 3222837 3229251 3244862 3315690 3347185 3354297 3367783 3578876 3595701 3671892 3675563 3727842 3731531 3733852 3753502 3954669 3966529 3984378 3994343 4013090 4113367 4134793 4137089 4153396
22	120		0.985071 (0.985068)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 41919 35064 56625 111264 120844 135402 156066 193837 197083 297854 309690 321677 330532 331454 338475 430909 437603 472518 530206 567303 682894 714276 720942 869995 905402 923298 982920 1018059 1144286 1149582 1351071 1376188 1380394 1380845 1400615 1419610 1487155 1551602 1567285 1681021 1734457 1788591 1884223 1964241 1999207 2039135 2073026 2112114 2141889 2289013 2314096 2350121 2357441 2376601 2403317 2422209 2501526 2638285 2645293 2683664 2687586 2703451 2758616 2779212 2803680 2825279 2893319 2937230 3047974 3082337 3148814 3167196 3172106 3212496 3222837 3229251 3244862 3315690 3347185 3354297 3367783 3578876 3595701 3671892 3675563 3727842 3731531 3733852 3753502 3954669 3966529 3984378 3994343 4013090 4113367 4134793 4137089 4153396
22	121		0.985908 (0.985906)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 41919 35064 56625 111264 120844 135402 156066 193837 197083 297854 309690 321677 330532 331454 338475 430909 437603 472518 530206 567303 682894 714276 720942 869995 905402 923298 982920 1018059 1144286 1149582 1351071 1376188 1380394 1380845 1400615 1419610 1487155 1551602 1567285 1638098 1681021 1734457 1788591 1884223 1964241 1999207 2039135 2073026 2112114 2141889 2289013 2314096 2350121 2357441 2376601 2403317 2422209 2501526 2638285 2645293 2683664 2687586 2703451 2758616 2779212 2803680 2825279 2893319 2937230 3047974 3082337 3148814 3167196 3172106 3212496 3222837 3229251 3244862 3315690 3347185 3354297 3367783 3578876 3595701 3671892 3675563 3727842 3731531 3733852 3753502 3954669 3966529 3984378 3994343 4013090 4113367 4134793 4137089 4153396
22	122		0.986703 (0.986701)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 41919 35064 56625 111264 120844 135402 156066 193837 197083 297854 309690 321677 330532 331454 338475 430909 437603 472518 530206 567303 682894 714276 720942 869995 905402 923298 982920 1018059 1144286 1

165

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
22	127	0.990094	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 14191 35064 56625 111264 120844 135402 156066 193837 197083 297854 309690 321677 330532 331454 338475 430909 437603 472518 530206 567303 682894 714276 720942 825500 869995 905402 923298 982920 1018059 1144286 1149582 1351071 1376188 1380394 1380845 1400615 1419610 1487155 1551602 1567285 1638098 1681021 1694007 1734457 1788591 1884223 1964241 1999207 2029487 2039135 2073026 2112114 2141889 2289013 2314096 2350121 2357441 2376601 2389911 2403317 2422209 2501526 2638285 2645293 2683664 2687586 2703451 2758616 2779212 2803680 2825279 2893319 2937230 3047974 3082337 3148814 3167196 3172106 3212496 3222837 3229251 3244862 3315690 3347185 3354297 3367783 3578876 3586605 3595701 3671892 3675563 3727842 3731531 3733852 3753502 3770633 3954669 3966529 3984378 3994343 4013090 4113367 4134793 4137089 4153396
22	128	0.990669	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 14191 35064 56625 111264 120844 135402 156066 193837 197083 297854 309690 321677 330532 331454 338475 430909 437603 472518 530206 567303 682894 714276 720942 825500 869995 905402 923298 982920 1018059 1144286 1149582 1351071 1376188 1380394 1380845 1400615 1419610 1487155 1551602 1567285 1638098 1681021 1694007 1734457 1788591 1884223 1964241 1999207 2029487 2039135 2073026 2112114 2141889 2289013 2314096 2350121 2357441 2376601 2389911 2403317 2422209 2501526 2638285 2645293 2683664 2687586 2703451 2758616 2779212 2803680 2825279 2893319 2937230 3047974 3082337 3148814 3167196 3172106 3212496 3222837 3229251 3244862 3315690 3347185 3354297 3367783 3578876 3586605 3595701 3671892 3675563 3727842 3731531 3733852 3753502 3770633 3906240 3954669 3966529 3984378 3994343 4013090 4113367 4134793 4137089 4153396
22	129	0.991212	8	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 14191 35064 50219 56625 111264 120844 135402 156066 193837 197083 297854 309690 321677 330532 331454 338475 430909 437603 472518 530206 567303 682894 714276 720942 825500 869995 905402 923298 982920 1018059 1144286 1149582 1351071 1376188 1380394 1380845 1400615 1419610 1487155 1551602 1567285 1638098 1681021 1694007 1734457 1788591 1884223 1964241 1999207 2029487 2039135 2073026 2112114 2141889 2289013 2314096 2350121 2357441 2376601 2389911 2403317 2422209 2501526 2638285 2645293 2683664 2687586 2703451 2758616 2779212 2803680 2825279 2893319 2937230 3047974 3082337 3148814 3167196 3172106 3212496 3222837 3229251 3244862 3315690 3347185 3354297 3367783 3578876 3586605 3595701 3671892 3675563 3727842 3731531 3733852 3753502 3770633 3906240 3954669 3966529 3984378 3994343 4013090 4113367 4134793 4137089 4153396
22	130	0.991714 (0.973505)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 14191 35064 50219 56625 111264 120844 135402 156066 193837 197083 297854 309690 321677 330532 331454 338475 430909 437603 472518 530206 567303 682894 714276 720942 825500 869995 905402 923298 982920 1018059 1144286 1149582 1306705 1351071 1376188 1380394 1380845 1400615 1419610 1487155 1551602 1567285 1638098 1681021 1694007 1734457 1788591 1884223 1964241 1999207 2029487 2039135 2073026 2112114 2141889 2289013 2314096 2350121 2357441 2376601 2389911 2403317 2422209 2501526 2638285 2645293 2683664 2687586 2703451 2758616 2779212 2803680 2825279 2893319 2937230 3047974 3082337 3148814 3167196 3172106 3212496 3222837 3229251 3244862 3315690 3347185 3354297 3367783 3578876 3586605 3595701 3671892 3675563 3727842 3731531 3733852 3753502 3770633 3906240 3954669 3966529 3984378 3994343 4013090 4113367 4134793 4137089 4153396
22	131	0.992189 (0.977863)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288

166

	m	n	R_e	d	R_c	Packed integer form of the parity check matrix
	22	135	0.993849 (0.988084)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 14191 35064 50219 56625 111264 120844 135402 156066 171662 193837 197083 238995 297854 309690 321677 330532 331454 338475 430909 437603 472518 530206 567303 682894 714276 720942 825500 869995 905402 923298 982920 989422 1018059 1144286 1149582 1306705 1351071 1376188 1380394 1380845 1400615 1419610 1487155 1551602 1567285 1638098 1681021 1694007 1734457 1788591 1884223 1964241 1999207 2029487 2039135 2073026 2112114 2141889 2289013 2314096 2350121 2357441 2376601 2389911 2403317 2422209 2501526 2638285 2645293 2683664 2687586 2703451 2758616 2779212 2803680 2825279 2872471 2893319 2937230 3047974 3082337 3148814 3167196 3172106 3212496 3222837 3229251 3244862 3315690 3347185 3354297 3367783 3578876 3586605 3595701 3671892 3675563 3699681 3727842 3731531 3733852 3753502 3770633 3906240 3954669 3966529 3984378 3994343 4013090 4113367 4134793 4137089 4153396
	22	136	0.994210 (0.989589)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 14191 35064 50219 56625 111264 120844 135402 156066 171662 193837 197083 238995 297854 309690 321677 330532 331454 338475 430909 437603 472518 530206 567303 682894 714276 720942 825500 869995 905402 923298 982920 989422 1018059 1144286 1149582 1306705 1351071 1376188 1380394 1380845 1400615 1419610 1487155 1551602 1567285 1638098 1681021 1694007 1734457 1788591 1884223 1964241 1999207 2029487 2039135 2073026 2112114 2141889 2289013 2314096 2350121 2357441 2376601 2389911 2403317 2422209 2501526 2638285 2645293 2683664 2687586 2703451 2758616 2779212 2803680 2825279 2872471 2893319 2937230 3047974 3082337 3148814 3167196 3172106 3212496 3222837 3229251 3244862 3315690 3347185 3354297 3367783 3578876 3586605 3595701 3671892 3675563 3699681 3727842 3731531 3733852 3753502 3770633 3869821 3906240 3954669 3966529 3984378 3994343 4013090 4113367 4134793 4137089 4153396
	22	137	0.994551 (0.990842)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 14191 35064 50219 56625 111264 120844 135402 156066 171662 193837 197083 238995 297854 309690 321677 330532 331454 338475 430909 437603 472518 530206 567303 682894 714276 720942 825500 869995 905402 923298 982920 989422 1018059 1109081 1144286 1149582 1306705 1351071 1376188 1380394 1380845 1400615 1419610 1487155 1551602 1567285 1638098 1681021 1694007 1734457 1788591 1884223 1964241 1999207 2029487 2039135 2073026 2112114 2141889 2289013 2314096 2350121 2357441 2376601 2389911 2403317 2422209 2501526 2638285 2645293 2683664 2687586 2703451 2758616 2779212 2803680 2825279 2872471 2893319 2937230 3047974 3082337 3148814 3167196 3172106 3212496 3222837 3229251 3244862 3315690 3347185 3354297 3367783 3578876 3586605 3595701 3671892 3675563 3699681 3727842 3731531 3733852 3753502 3770633 3869821 3906240 3954669 3966529 3984378 3994343 4013090 4113367 4134793 4137089 4153396
	22	138	0.994874 (0.991893)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 14191 35064 50219 56625 111264 120844 135402 156066 171662 193837 197083 238995 297854 309690 321677 330532 331454 338475 430909 437603 472518 530206 567303 682894 714276 720942 825500 869995 905402 923298 982920 989422 1018059 1109081 1144286 1149582 1306705 1351071 1376188 1380394 1380845 1400615 1419610 1487155 1551602 1567285 1638098 1681021 1694007 1734457 1788591 1884223 1964241 1999207 2029487 2039135 2073026 2112114 2141889 2289013 2314096 2350121 2357441 2376601 2389911 2403317 2422209 2501526 2638285 2645293 2683664 2687586 2703451 2758616 2779212 2803680 2825279 2872471 2893319 2937230 3047974 3082337 3148814 3167196 3172106 3212496 3222837 3229251 3244862 3315690 3347185 3354297 3367783 3578876 3586605 3595701 3671892 3675563 369968

167

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
22	143	0.996238	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 14191 35064 50219 56625 111264 120844 135402 156066 171662 193837 197083 197990 238995 297854 309690 321677 330532 331454 338475 356185 430909 437603 472518 530206 567303 682894 714276 720942 825500 869995 905402 923298 982920 989422 1018059 1109081 1144286 1149582 1221482 1306705 1351071 1376188 1380394 1380845 1400615 1419610 1487155 1551602 1567285 1638098 1681021 1694007 1734457 1788591 1884223 1964241 1999207 2029487 2039135 2073026 2112114 2141889 2289013 2314096 2350121 2357441 2376601 2389911 2403317 2422209 2501526 2638285 2645293 2683664 2687586 2703451 2758616 2779212 2803680 2825279 2872471 2893319 2937230 2978972 3047974 3082337 3148814 3167196 3172106 3212496 3222837 3229251 3244862 3315690 3347185 3354297 3367783 3578876 3586605 3595701 3671892 3675563 3699681 3727842 3731531 3733852 3742366 3753502 3764120 3770633 3869821 3906240 3954669 3966529 3984378 3994343 4013090 4113367 4134793 4137089 4153396
22	144	0.996466	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 14191 35064 50219 56625 111264 120844 135402 156066 171662 193837 197083 197990 238995 297854 309690 321677 330532 331454 338475 356185 430909 437603 472518 530206 567303 682894 714276 720942 825500 869995 905402 923298 982920 989422 1018059 1109081 1144286 1149582 1221482 1306705 1351071 1376188 1380394 1380845 1400615 1419610 1487155 1551602 1567285 1638098 1681021 1694007 1734457 1788591 1884223 1964241 1999207 2029487 2039135 2073026 2112114 2141889 2289013 2314096 2350121 2357441 2376601 2389911 2403317 2422209 2501526 2638285 2645293 2683664 2687586 2703451 2758616 2779212 2797554 2803680 2825279 2872471 2893319 2937230 2978972 3047974 3082337 3148814 3167196 3172106 3212496 3222837 3229251 3244862 3315690 3347185 3354297 3367783 3578876 3586605 3595701 3671892 3675563 3699681 3727842 3731531 3733852 3742366 3753502 3764120 3770633 3869821 3906240 3954669 3966529 3984378 3994343 4013090 4113367 4134793 4137089 4153396
22	145	0.996682	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 14191 35064 50219 56625 111264 120844 135402 156066 171662 193837 197083 197990 238995 297854 309690 321677 330532 331454 338475 356185 430909 437603 472518 530206 567303 682894 714276 720942 825500 869995 905402 923298 982920 989422 1018059 1109081 1144286 1149582 1221482 1306705 1351071 1376188 1380394 1380845 1400615 1419610 1487155 1551602 1567285 1638098 1681021 1694007 1734457 1788591 1884223 1964241 1999207 2029487 2039135 2073026 2112114 2141889 2289013 2314096 2350121 2357441 2376601 2389911 2403317 2422209 2501526 2638285 2645293 2683664 2687586 2703451 2758616 2779212 2797554 2803680 2825279 2872471 2893319 2937230 2978972 3047974 3082337 3148814 3167196 3172106 3212496 3222837 3229251 3244862 3315690 3347185 3354297 3367783 3540526 3578876 3586605 3595701 3671892 3675563 3699681 3727842 3731531 3733852 3742366 3753502 3764120 3770633 3869821 3906240 3954669 3966529 3984378 3994343 4013090 4113367 4134793 4137089 4153396
24	25	0.29833	25	12	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777215
24	26	0.310263	17	12	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 5170431 16776960
24	27	0.322196	15	13	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 5196015 13582108 15978464
24	28	0.334129 (0.334128)	14	13	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 5208289 11185371 13582108 15978464
24	29	0.346061	14	13	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 1420655 5208289 11185371 13582108 15978464
24	30	0.357993	14	11	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 1420655 3372630 5208289 11185371 13582108 15978464
24	31	0.36992	13	12	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 3947641 7059133 10440899 11184471 13421978 16712188 16776704
24	32	0.381846	13	10	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 3947641 5434162 7059133 10440899 11184471 13421978 16712188 16776704
24	33	0.393762 (0.393717)	11	10	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 777010 3311293 3947641 5170734 10440899 11184471 13421978 16712188 16776704
24	34	0.405672 (0.405610)	11	10	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 777010 918399 3311293 3947641 5170734 10440899 11184471 13421978 16712188 16776704
24	35	0.417568 (0.417519)	12	10	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 914106 1273716 5181645 8129641 9230633 11435775 11722525 13284797 14005927 15303671 16259282
24	36	0.429448 (0.429391)	12	10	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 3930540 5878091 7536957 8129641 9230633 11722525 11884983 13284797 14005927 15073914 15303671 16259282
24	37	0.441302 (0.441244)	12	11	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 7536957 8129641 9230633 11722525 11884983 12910540 13284797 14005927 14054898 14477557 15073914 15303671 16259282
24	38	0.453149	12	11	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 6264096 7536957 8129641 9230633 11722525 11884983 12910540 13284797 14005927 14054898 14477557 15073914 15303671 16259282
24	39	0.464972	12	11	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 6264096 7536957 8129641 9230633 10543947 11722525 11884983 12910540 13284797 14005927 14054898 14477557 15073914 15303671 16259282
24	40	0.476767 (0.476766)	12	10	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 1811121 6264096 7536957 8129641 9230633 10543947 11722525 11884983 12910540 13284797 14005927 14054898 14477557 15073914 15303671 16259282
24	41	0.488527	12	11	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 7027449 7536957 8129641 9230633 11722525 11884983 12141653 13284797 13472408 13662979 14005927 14054898 15000744 15073914 15303671 16215601 16259282
24	42	0.500248	12	11	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 1875093 7027449 7536957 8129641 9230633 11722525 11884983 12141653 13284797 13472408 13662979 14005927 14054898 15000744 15073914 15303671 16215601 16259282
24	43	0.511923	12	11	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 1875093 7027449 7536957 8129641 9230633 11722525 11884983 12141653 13284797 13472408 13662979 14005927 14054898 15000744 15073914 15303671 16215601 16259282

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
24	44	0.523546	12	11	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 1875093 3750186 7027449 7500372 7536957 8129641 9230633 11722525 11884983 12141653 13284797 13472408 13662979 14005927 14054898 15000744 15073914 15303671 16215601 16259282
24	45	0.535111	12	11	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 1684051 1875093 3750186 7027449 7500372 7536957 8129641 9230633 11722525 11884983 12141653 13284797 13472408 13662979 14005927 14054898 15000744 15073914 15303671 16215601 16259282
24	46	0.546612	12	11	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 1684051 1875093 3368102 3750186 7027449 7500372 7536957 8129641 9230633 11722525 11884983 12141653 13284797 13472408 13662979 14005927 14054898 15000744 15073914 15303671 16215601 16259282
24	47	0.55804	12	11	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 1684051 1875093 3368102 3750186 7027449 7500372 7536957 8129641 8558435 9230633 11722525 11884983 12141653 13284797 13472408 13662979 14005927 14054898 15000744 15073914 15303671 16215601 16259282
24	48	0.569391	12	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 1684051 1875093 3368102 3750186 6736204 7027449 7500372 7536957 8129641 8558435 9230633 11722525 11884983 12141653 13284797 13472408 13662979 14005927 14054898 15000744 15073914 15303671 16215601 16259282
24	49	0.579991 (0.578804)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 16283648 15099392 664237 902795 1029920 1625264 3295992 3382336 4977040 5172343 6082584 8170562 8706820 10443340 10641232 10998368 11184896 11220096 11264348 12141248 12601840 12646336 13233960 13980928 15643859 16596448 16615338
24	50	0.591160 (0.589775)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 16283648 15099392 664237 902795 1029920 1625264 3295992 3382336 4977040 5172343 6082584 7887453 8170562 8706820 10443340 10641232 10998368 11184896 11220096 11264348 12141248 12601840 12646336 13233960 13980928 15643859 16596448 16615338
24	51	0.602192 (0.601289)	10	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 16283648 15099392 664237 902795 1029920 1625264 1747206 3295992 3382336 4977040 5172343 6082584 7887453 8170562 8706820 10443340 10641232 10998368 11184896 11220096 11264348 12141248 12601840 12646336 13233960 13980928 15643859 16596448 16615338
24	52	0.613030 (0.612436)	10	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 16283648 15099392 351809 748931 1029920 1625264 3295992 3382336 4883803 4977040 5659518 6082584 6721849 8170562 8706820 10443340 10641232 10998368 11184896 11207590 11220096 11264348 11611319 12141248 12601840 12646336 13233960 13980928 16596448 16615338
24	53	0.623674 (0.623430)	10	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 351809 1029920 1625264 3295992 3382336 4977040 5364503 5659518 6082584 6721849 8170562 8706820 10443340 10641232 10998368 11184896 11207590 11220096 11264348 12141248 12601840 12646336 13233960 13980928 14706795 15099392 16283648 16596448 16615338
24	54	0.634372 (0.634204)	10	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 351809 748931 1029920 1625264 3295992 3382336 4883803 4977040 5659518 6082584 6721849 8170562 8706820 10443340 10641232 10998368 11184896 11207590 11220096 11264348 12141248 12601840 12646336 13233960 13980928 14101295 15099392 16283648 16596448 16615338
24	55	0.64485	10	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 351809 748931 1029920 1625264 3295992 3382336 4883803 4977040 5659518 6082584 6721849 8170562 8706820 9357431 10443340 10641232 10998368 11184896 11207590 11220096 11264348 12141248 12601840 12646336 13233960 13980928 15099392 16283648 16596448 16615338 16744448
24	56	0.655262 (0.655142)	9	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 701023 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 4447108 4642815 4865882 4981695 5087490 6813668 6982303 7894909 7933395 8894748 9354907 9466165 10033669 10290550 10887849 11095914 11204875 11485683 12233544 12889002 13314240 14157669 14795911
24	57	0.665489 (0.665358)	9	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 701023 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 4447108 4642815 4865882 4981695 5087490 6813668 6982303 7894909 7933395 8894748 9354907 9466165 10033669 10290550 10887849 11095914 11204875 11485683 12233544 12889002 13314240 14157669 14795911 15923713
24	58	0.675553 (0.675429)	9	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 701023 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 4447108 4642815 4865882 4981695 5087490 6813668 6982303 7894909 7933395 8894748 9354907 9466165 10033669 10290550 10887849 11095914 11204875 11485683 12233544 12889002 13314240 14157669 14795911 15923713 15972347 16540142
24	59	0.685461 (0.685325)	9	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 701023 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 4447108 4642815 4865882 4981695 5087490 6813668 6982303 7894909 7933395 8894748 9354907 9466165 10033669 10290550 10887849 11095914 11204875 11485683 12233544 12889002 13314240 14157669 14795911 15923713 16540142
24	60	0.695206 (0.695065)	9	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 701023 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 4447108 4642815 4865882 4981695 5087490 6813668 6982303 7894909 7933395 8894748 9354907 9466165 10033669 10290550 10887849 11095914 11204875 11485683 12233544 12889002 13314240 14157669 14795911 15923713 15972347 16540142 16554318
24	61	0.704786 (0.704645)	9	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 701023 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 4447108 4642815 4865882 4981695 5087490 6813668 6982303 7894909 7933395 8894748 9354907 9466165 10033669 10290550 10887849 11095914 11204875 11485683 12233544 12889002 13314240 14157669 14795911 15923713 15972347 16494372 16540142 16554318
24	62	0.714189 (0.714060)	9	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 701023 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 4447108 4642815 4865882 4981695 5087490 6813668 6982303 7894909 7933395 8894748 9354907 9466165 10033669 10290550 10887849 11095914 11204875 11485683 12233544 12889002 13314240 14157669 14795911 15923713 15972347 16494372 16540142 16554318
24	63	0.723408 (0.723289)	9	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 701023 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 4447108 4642815 4865882 4981695 5087490 6813668 6982303 7894909 7933395 8894748 9354907 9466165 10033669 10290550 10887849 11095914 11204875 11485683 12233544 12889002 13314240 13707676 14157669 14795911 15923713 15972347 16494372 16540142 16554318

Continued on next page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
24	64	0.732451 (0.732331)	9	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 701023 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 4447108 4642815 4865882 4981695 5087490 6813668 6982303 7894909 7933395 8894748 9354907 9466165 10033669 10290550 10887849 11095914 11204875 11485683 12233544 12889002 13314240 13707676 14157669 14795911 15150513 15923713 15972347 16494372 16540142 16554318
24	65	0.741288 (0.700697)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 701023 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 4447108 4642815 4865882 4981695 5087490 5176601 6813668 6982303 7894909 7933395 8894748 9354907 9466165 10033669 10290550 10887849 11095914 11204875 11485683 12233544 12889002 13314240 13707676 14157669 14795911 15150513 15923713 15972347 16494372 16540142 16554318
24	66	0.749943 (0.707588)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 701023 935950 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 4447108 4642815 4865882 4981695 5087490 5176601 6813668 6982303 7894909 7933395 8894748 9354907 9466165 10033669 10290550 10887849 11095914 11204875 11485683 12233544 12889002 13314240 13707676 14157669 14795911 15150513 15923713 15972347 16494372 16540142 16554318
24	67	0.758410 (0.714104)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 649314 701023 935950 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 4447108 4642815 4865882 4981695 5087490 5176601 6813668 6982303 7894909 7933395 8894748 9354907 9466165 10033669 10290550 10887849 11095914 11204875 11485683 12233544 12889002 13314240 13707676 14157669 14795911 15150513 15923713 15972347 16494372 16540142 16554318
24	68	0.766689 (0.720404)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 649314 701023 935950 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 4447108 4642815 4865882 4981695 5087490 5176601 6813668 6982303 7894909 7933395 8894748 9354907 9466165 10033669 10290550 10887849 11095914 11204875 11485683 12233544 12889002 13314240 13707676 14157669 14795911 14923311 15150513 15923713 15972347 16494372 16540142 16554318
24	69	0.774775 (0.726276)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 649314 701023 935950 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 4447108 4642815 4865882 4981695 5087490 5176601 6813668 6982303 7894909 7933395 8894748 9354907 9466165 10033669 10290550 10887849 11095914 11204875 11485683 11796941 12233544 12889002 13314240 13707676 14157669 14795911 14923311 15150513 15923713 15972347 16494372 16540142 16554318
24	70	0.782668 (0.740446)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 649314 701023 935950 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 4447108 4642815 4865882 4933529 4981695 5087490 5176601 6813668 6982303 7894909 7933395 8894748 9354907 9466165 10033669 10290550 10887849 11095914 11204875 11485683 11796941 12233544 12889002 13314240 13707676 14157669 14795911 14923311 15150513 15923713 15972347 16494372 16540142 16554318
24	71	0.790372 (0.746689)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 573185 649314 701023 935950 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 4447108 464

170

	m	n	R_e	d	R_c	Packed integer form of the parity check matrix
24	78		0.838948 (0.783622)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 573185 576055 649314 701023 902444 935950 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 4447108 4642815 4865882 4933529 4981695 5087490 5170378 5176601 5211409 5493873 6813668 6982303 7139876 7894909 7933395 8894748 9354907 9466165 10033669 10290550 10320865 10887849 11095914 11204875 11485683 11796941 12233544 12889002 13314240 13707676 14157669 14795911 14923311 15150513 15855350 15923713 15972347 16494372 16540142 16554318
24	79		0.845134 (0.787822)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 573185 576055 649314 701023 902444 935950 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 4447108 4642815 4865882 4933529 4981695 5087490 5170378 5176601 5211409 5493873 6813668 6982303 7139876 7894909 7933395 8894748 9354907 9466165 10033669 10290550 10320865 10887849 11095914 11204875 11485683 11796941 12233544 12889002 13314240 13707676 14157669 14795911 14923311 15150513 15855350 15923713 15972347 16494372 16540142 16554318
24	80		0.851139 (0.792380)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 573185 576055 649314 701023 902444 935950 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 4447108 4642815 4865882 4933529 4981695 5087490 5170378 5176601 5211409 5493873 6813668 6982303 7139876 7894909 7933395 8894748 9354907 9466165 10033669 10290550 10320865 10887849 11095914 11204875 11485683 11796941 12233544 12889002 13314240 13707676 14157669 14737865 14795911 14923311 15150513 15855350 15923713 15972347 16494372 16540142 16554318
24	81		0.856971 (0.796831)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 573185 576055 649314 701023 902444 935950 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 4447108 4642815 4865882 4933529 4981695 5087490 5170378 5176601 5211409 5493873 6813668 6982303 7139876 7894909 7933395 8894748 9354907 9466165 10033669 10290550 10320865 10887849 11095914 11204875 11347996 11485683 11796941 12233544 12889002 13314240 13707676 14157669 14737865 14795911 14923311 15150513 15855350 15923713 15972347 16494372 16540142 16554318
24	82		0.862622 (0.801027)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 573185 576055 649314 701023 902444 935950 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 3857484 4447108 4642815 4865882 4933529 4981695 5087490 5170378 5176601 5211409 5493873 6813668 6982303 7139876 7894909 7933395 8894748 9354907 9466165 10033669 10290550 10320865 10887849 11095914 11204875 11347996 11485683 11796941 12233544 12889002 13314240 13707676 14157669 14737865 14795911 14923311 15150513 15855350 15923713 15972347 16494372 16540142 16554318
24	83		0.868096 (0.804923)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 573185 576055 649314 701023 902444 935950 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 3857484 4447108 4642815 4865882 4933529 4981695 5087490 5170378 5176601 5211409 5493873 6813668 6982303 7139876 7894909 7933395 8894748 9354907 9466165 10033669 10290550 10320865 10887849 11095914 11204875 11347996 11485683 11796941 12233544 12889002 13314240 13707676 14157669 14737865 14795911 14923311 15150513 15341837 15855350 15923713 15972347 16494372 16540142 16554318
24	84		0.873396 (0.808726)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536

171

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
24	90	0.901784 (0.833170)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 573185 576055 605300 649314 701023 902444 935950 972344 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 3857484 4410637 4447108 4642815 4754320 4865882 4933529 4981695 5087490 5170378 5176601 5211409 5493873 6813668 6982303 7139876 7894909 7933395 8894748 9354907 9466165 10033669 10290550 10320865 10887849 11095914 11204875 11347996 11485683 11727110 11796941 12233544 12889002 13314240 13707676 14157669 14737865 14795911 14837286 14923311 15030096 15150513 15341837 15855350 15923713 15972347 16494372 16540142 16554318
24	91	0.905964 (0.839750)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 573185 576055 605300 649314 701023 902444 935950 972344 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 3857484 4410637 4447108 4642815 4754320 4865882 4933529 4981695 5087490 5170378 5176601 5211409 5493873 6813668 6982303 7139876 7894909 7933395 8894748 9354907 9466165 10033669 10290550 10320865 10887849 11095914 11098497 11204875 11347996 11485683 11727110 11796941 12233544 12889002 13314240 13707676 14157669 14737865 14795911 14837286 14923311 15030096 15150513 15341837 15855350 15923713 15972347 16494372 16540142 16554318
24	92	0.910005 (0.845267)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 573185 576055 605300 649314 701023 902444 935950 972344 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 3857484 4410637 4447108 4642815 4754320 4865882 4933529 4981695 5087490 5170378 5176601 5211409 5493873 6813668 6982303 7139876 7894909 7933395 8894748 9354907 9466165 10033669 10290550 10320865 10887849 11095914 11098497 11204875 11347996 11485683 11727110 11796941 12233544 12889002 13314240 13471851 13707676 14157669 14737865 14795911 14837286 14923311 15030096 15150513 15341837 15855350 15923713 15972347 16494372 16540142 16554318
24	93	0.913897 (0.850019)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 573185 576055 605300 649314 701023 902444 935950 972344 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 3857484 4406019 4410637 4447108 4642815 4754320 4865882 4933529 4981695 5087490 5170378 5176601 5211409 5493873 6813668 6982303 7139876 7894909 7933395 8894748 9354907 9466165 10033669 10290550 10320865 10887849 11095914 11098497 11204875 11347996 11485683 11727110 11796941 12233544 12889002 13314240 13471851 13707676 14157669 14737865 14795911 14837286 14923311 15030096 15150513 15341837 15855350 15923713 15972347 16494372 16540142 16554318
24	94	0.917651 (0.854168)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 573185 576055 605300 649314 701023 902444 935950 972344 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 3857484 4406019 4410637 4447108 4642815 4661116 4754320 4865882 4933529 4981695 5087490 5170378 5176601 5211409 5493873 6813668 6982303 7139876 7894909 7933395 8894748 9354907 9466165 10033669 10290550 10320865 10887849 11095914 11098497 11204875 11347996 11485683 11727110 11796941 12233544 12889002 13314240 13471851 13707676 14157669 14737865 14795911 14837286 14923311 15030096 15150513 15341837 15855350 15923713 15972347 16494372 16540142 16554318
24	95	0.921265 (0.857915)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 573185 576055 605300 649314 701023 902444 935950 972344 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 3857484 4406019 4410637 4447108 464

172

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
24	100	0.937405 (0.895386)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 573185 576055 605300 649314 701023 902444 935950 972344 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 3857484 4046019 4292634 4410637 4447108 4642815 4661116 4754320 4865882 4933529 4981695 5087490 5104610 5170378 5176601 5211409 5493873 6813668 6982303 7139876 7894909 7933395 8894748 9008563 9354907 9466165 10033669 10290550 10320865 10887849 11095914 11098497 11204875 11347996 11485683 11727110 11796941 12233544 12889002 13314240 13471851 13563383 13707676 14157669 14737865 14795911 14837286 14923311 15030096 15150513 15341837 15768860 15855350 15923713 15972347 16494372 16540142 16554318 16689280
24	101	0.940268 (0.897624)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 605300 649314 701023 902444 935950 972344 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 3857484 4046019 4292634 4410637 4447108 4642815 4661116 4754320 4865882 4933529 4981695 5087490 5104610 5170378 5176601 5211409 5493873 6813668 6982303 7139876 7894909 7933395 8894748 9008563 9354907 9466165 10033669 10290550 10320865 10887849 11095914 11098497 11204875 11347996 11485683 11727110 11796941 12233544 12889002 13314240 13471851 13563383 13707676 14157669 14737865 14795911 14837286 14923311 15030096 15150513 15341837 15768860 15855350 15923713 15972347 16494372 16540142 16554318 16689280
24	102	0.943024 (0.899765)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 605300 649314 701023 902444 935950 972344 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 3857484 4046019 4292634 4410637 4447108 4642815 4661116 4754320 4865882 4933529 4981695 5087490 5104610 5170378 5176601 5211409 5493873 6813668 6982303 7139876 7663866 7894909 7933395 8894748 9008563 9354907 9466165 10033669 10290550 10320865 10887849 11095914 11098497 11204875 11347996 11485683 11727110 11796941 12233544 12889002 13314240 13471851 13563383 13707676 14157669 14737865 14795911 14837286 14923311 15030096 15150513 15341837 15768860 15855350 15923713 15972347 16494372 16540142 16554318 16689280
24	103	0.945665 (0.901815)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 605300 649314 701023 902444 935950 972344 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 3857484 4046019 4292634 4410637 4447108 4642815 4661116 4754320 4865882 4933529 4981695 5087490 5104610 5170378 5176601 5211409 5493873 6813668 6982303 7139876 7663866 7894909 7933395 8894748 9008563 9354907 9466165 10033669 10290550 10320865 10887849 11095914 11098497 11204875 11266610 11347996 11485683 11727110 11796941 12233544 12889002 13314240 13471851 13563383 13707676 14157669 14737865 14795911 14837286 14923311 15030096 15150513 15341837 15768860 15855350 15923713 15972347 16494372 16540142 16554318 16689280
24	104	0.948199 (0.903769)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 605300 649314 701023 902444 935950 972344 1107341 1216620 1391649 1700311 1849660 2435061 2801845 3274804 3727802 3857484 4046019 4292634 4410637 4447108 4642815 4661116 4754320 4865882 4933529 4981695 5087490 5104610 5170378 5176601 5211409 5493873 6813668 6982303 7139876 7663866 7894909 7933395 8894748 9008563 9354907 9466165 10033669 10290550 10320865 10887849 11095914 11098497 11204875 11266610 11347996 11485683 11727110 11796941 12233544 12889002 13314240 13471851 13563383 13707676 14157669 14737865 14795911 14837286 14923311 15030096 15150513 1534183

173

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
24	109	0.959397 (0.912304)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 605300 649314 701023 902444 935950 972344 1107341 1216620 1391649 1700311 1849660 2435061 2504405 2801845 3274804 3727802 3857484 4046019 4292634 4410637 4447108 4642815 4661116 4754320 4865882 4903110 4933529 4981695 5087490 5104610 5170378 5176601 5211409 5493873 6813668 6982303 7139876 7663866 7894909 7933395 8894748 9008563 9265730 9354907 9466165 10033669 10290550 10320865 10775437 10887849 11095914 11098497 11204875 11266610 11347996 11485683 11727110 11796941 12233544 12889002 13314240 13471851 13563383 13707676 14157669 14737865 14795911 14837286 14923311 15030096 15150513 15341837 15768860 15855350 15923713 15972347 16138559 16194446 16494372 16540142 16554318 16689280
24	110	0.961362 (0.913775)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 605300 649314 701023 902444 935950 972344 1107341 1216620 1391649 1700311 1849660 2435061 2504405 2801845 3274804 3727802 3857484 4046019 4292634 4410637 4447108 4642815 4661116 4754320 4865882 4903110 4933529 4981695 5087490 5104610 5170378 5176601 5211409 5493873 6813668 6982303 7139876 7663866 7894909 7933395 8644652 8894748 9008563 9265730 9354907 9466165 10033669 10290550 10320865 10775437 10887849 11095914 11098497 11204875 11266610 11347996 11485683 11727110 11796941 12233544 12889002 13314240 13471851 13563383 13707676 14157669 14737865 14795911 14837286 14923311 15030096 15150513 15341837 15768860 15855350 15923713 15972347 16138559 16194446 16494372 16540142 16554318 16689280
24	111	0.963243 (0.915196)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 605300 649314 701023 902444 935950 972344 1107341 1216620 1391649 1700311 1849660 2435061 2504405 2801845 3274804 3727802 3857484 4046019 4292634 4410637 4447108 4642815 4661116 4754320 4865882 4903110 4933529 4981695 5087490 5104610 5170378 5176601 5211409 5493873 6813668 6982303 7139876 7663866 7894909 7933395 8498409 8644652 8894748 9008563 9265730 9354907 9466165 10033669 10290550 10320865 10775437 10887849 11095914 11098497 11204875 11266610 11347996 11485683 11727110 11796941 12233544 12889002 13314240 13471851 13563383 13707676 14157669 14737865 14795911 14837286 14923311 15030096 15150513 15341837 15768860 15855350 15923713 15972347 16138559 16194446 16494372 16540142 16554318 16689280
24	112	0.965042 (0.916535)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 605300 649314 701023 902444 935950 972344 1107341 1216620 1391649 1700311 1849660 2435061 2504405 2801845 3274804 3727802 3857484 4046019 4292634 4410637 4447108 4642815 4661116 4754320 4865882 4903110 4933529 4981695 5087490 5104610 5170378 5176601 5211409 5493873 6813668 6982303 7139876 7663866 7894909 7933395 8498409 8644652 8678584 8894748 9008563 9265730 9354907 9466165 10033669 10290550 10320865 10775437 10887849 11095914 11098497 11204875 11266610 11347996 11485683 11727110 11796941 12233544 12889002 13314240 13471851 13563383 13707676 14157669 14737865 14795911 14837286 14923311 15030096 15150513 15341837 15768860 15855350 15923713 15972347 16138559 16194446 16494372 16540142 16554318 16689280
24	113	0.966764 (0.917818)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 605300 649314 701023 902444 935950 972344 1107341 1216620 1391649 1700311 1849660 2435061 2504405 2801845 3274804 3727802 3857484 4046019 4292634 4410637 4447108 4642815 4661116 4697043 4754320 4865882 4903110 4933529 4981695 5087490 5104610 5170378 517660

174

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
24	118	0.974293 (0.923338)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 605300 649314 701023 902444 935950 972344 1107341 1216620 1391649 1681295 1700311 1849660 2435061 2504405 2801845 3274804 3727802 3837096 3857484 4046019 4292634 4410637 4447108 4642815 4661116 4697043 4754320 4865882 4903110 4933529 4981695 5087490 5104610 5170378 5176601 5211409 5493873 6280895 6813668 6982303 7139876 7663866 7894909 7933395 8498409 8644652 8678584 8894748 9008563 9265730 9354907 9466165 10033669 10290550 10320865 10775437 10887849 11095914 11098497 11204875 11266610 11347996 11485683 11727110 11796941 12233544 12889002 13167991 13314240 13331118 13471851 13563383 13707676 14157669 14737865 14795911 14837286 14923311 15030096 15150513 15341837 15768860 15855350 15923713 15972347 16138559 16194446 16494372 16540142 16554318 16689280
24	119	0.975600 (0.924283)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 605300 649314 701023 842347 902444 935950 972344 1107341 1216620 1391649 1681295 1700311 1849660 2435061 2504405 2801845 3274804 3727802 3837096 3857484 4046019 4292634 4410637 4447108 4642815 4661116 4697043 4754320 4865882 4903110 4933529 4981695 5087490 5104610 5170378 5176601 5211409 5493873 6280895 6813668 6982303 7139876 7663866 7894909 7933395 8498409 8644652 8678584 8894748 9008563 9265730 9354907 9466165 10033669 10290550 10320865 10775437 10887849 11095914 11098497 11204875 11266610 11347996 11485683 11727110 11796941 12233544 12889002 13167991 13314240 13331118 13471851 13563383 13707676 14157669 14737865 14795911 14837286 14923311 15030096 15150513 15341837 15768860 15855350 15923713 15972347 16138559 16194446 16494372 16540142 16554318 16689280
24	120	0.976848 (0.925181)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 605300 649314 701023 842347 902444 935950 972344 1107341 1216620 1391649 1681295 1700311 1849660 2435061 2504405 2801845 3274804 3727802 3837096 3857484 4046019 4292634 4410637 4447108 4642815 4661116 4697043 4754320 4865882 4903110 4933529 4981695 5087490 5104610 5170378 5176601 5211409 5493873 6280895 6813668 6982303 7139876 7156527 7663866 7894909 7933395 8498409 8644652 8678584 8894748 9008563 9265730 9354907 9466165 10033669 10290550 10320865 10775437 10887849 11095914 11098497 11204875 11266610 11347996 11485683 11727110 11796941 12233544 12889002 13167991 13314240 13331118 13471851 13563383 13707676 14157669 14737865 14795911 14837286 14923311 15030096 15150513 15341837 15768860 15855350 15923713 15972347 16138559 16194446 16494372 16540142 16554318 16689280
24	121	0.978039 (0.926035)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 605300 649314 701023 842347 902444 935950 972344 1107341 1216620 1391649 1681295 1700311 1849660 2435061 2504405 2801845 3274804 3727802 3837096 3857484 4046019 4292634 4410637 4447108 4642815 4661116 4697043 4754320 4865882 4903110 4933529 4981695 5087490 5104610 5170378 5176601 5211409 5493873 6280895 6813668 6982303 7139876 7156527 7663866 7894909 7933395 8498409 8644652 8678584 8894748 9008563 9265730 9354907 9450080 9466165 10033669 10290550 10320865 10775437 10887849 11095914 11098497 11204875 11266610 11347996 11485683 11727110 11796941 12233544 12889002 13167991 13314240 13331118 13471851 13563383 13707676 14157669 14737865 14795911 14837286 14923311 15030096 15150513 15341837 15768860 15855350 15923713 15972347 16138559 16194446 16494372 16540142 16554318 16689280
24	122	0.979174 (0.926845)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 605300 649314 701023 842347 902444 935950 972344 1107341 1216620 1391649 1681295 1700311 1849660 2435061 2504405 2801845 3274804 3727802 3837096 3857484 4046019 4292634 4410637 4447108 4642815 4661116 4697043 4754320 4865882 4903110 4933529 4981695 5087490 5104610 5170378 5176601 5211409 5493873 6280895 6813668 6982303 7139876 7156527 7663866 7894909 7933395 8498409 8644652 8678584 8894748 9008563 9265730 9354907 9450080 9466165 10033669 10290550 10320865 10775437 10887849 11095914 11098497 11204875 11266610 11347996 11485683 11727110 11796941 12233544 12889002 13167991 13314240 13331118 13471851 13563383 13707676 14157669 14737865 14795911 14837286 14923311 15030096 15150513 15341837 15768860 15855350 15923713 15972347 16138559 16194446 16494372 16540142 16554318 16689280 16759320
24	123	0.980256 (0.927614)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 605300 649314 701023 842347 902444 935950 972344 1107341 1216620 1391649 1681295 1700311 1849660 2435061 2504405 2801845 3274804 3727802 3837096 3857484 4046019 4292634 4410637 4447108 4642815 4661116 4697043 4754320 4865882 4903110 4933529 4981695 5087490 5104610 5170378 5176601 5211409 5493873 6280895 6813668 6982303 7139876 7156527 7663866 7894909 7933395 8498409 8644652 8678584 8894748 9008563 9265730 9354907 9450080 9466165 10033669 10290550 10320865 10775437 10887849 11095914 11098497 11204875 11266610 11347996 11485683 11727110 11796941 12233544 12723309 12889002 13167991 13314240 13331118 13471851 13563383 13707676 14157669 14737865 14795911 14837286 14923311 15030096 15150513 15341837 15768860 15855350 15923713 15972347 16138559 16194446 16494372 16540142 16554318 16689280 16759320
24	124	0.981286 (0.928343)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 605300 649314 701023 842347 902444 935950 972344 1107341 1216620 1391649 1681295 1700311 1849660 2435061 2504405 2801845 3274804 3646867 3727802 3837096 3857484 4046019 4292634 4410637 4447108 4642815 4661116 4697043 4754320 4865882 4903110 4933529 4981695 5087490 5104610 5170378 5176601 5211409 5493873 6280895 6813668 6982303 7139876 7156527 7663866 7894909 7933395 8498409 8644652 8678584 8894748 9008563 9265730 9354907 9450080 9466165 10033669 10290550 10320865 10775437 10887849 11095914 11098497 11204875 11266610 11347996 11485683 11727110 11796941 12233544 12723309 12889002 13167991 13314240 13331118 13471851 13563383 13707676 14157669 14737865 14795911 14837286 14923311 15030096 15150513 15341837 15768860 15855350 15923713 15972347 16138559 16194446 16494372 16540142 16554318 16689280 16759320
24	125	0.982267 (0.929033)	7	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 605300 649314 701023 842347 902444 935950 972344 1107341 1216620 1391649 1681295 1700311 1849660 2435061 2504405 2801845 3274804 3646867 3727802 3837096 3857484 4046019 4292634 4410637 4447108 4642815 4661116 4697043 4754320 4865882 4903110 4933529 4981695 5087490 5104610 5170378 5176601 5211409 5493873 6280895 6813668 6982303 7139876 7156527 7663866 7894909 7933395 8498409 8644652 8678584 8894748 9008563 9265730 9354907 9450080 9466165 10033669 10290550 10320865 10775437 10887849 11095914 11098497 11204875 11266610 11347996 11485683 11727110 11796941 12233544 12723309 12889002 13167991 13314240 13331118 13471851 13563383 13707676 14157669 14737865 14795911 14837286 14923311 15030096 15150513 15341837 15768860 15855350 15923713 15972347 16138559 16194446 16272064 16494372 16540142 16554318 16689280 16759320

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
24	126	0.983201 (0.929688)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 605300 649314 701023 842347 902444 935950 972344 1107341 1216620 1391649 1681295 1700311 1849660 2435061 2504405 2801845 3274804 3646867 3727802 3837096 3857484 4046019 4292634 4410637 4447108 4642815 4661116 4697043 4754320 4865882 4903110 4933529 4981695 5087490 5104610 5170378 5170573 5176601 5211409 5493873 6280895 6813668 6982303 7139876 7156527 7663866 7894909 7933395 8498409 8644652 8678584 8894748 9008563 9265730 9354907 9450080 9466165 10033669 10290550 10320865 10775437 10887849 11095914 11098497 11204875 11266610 11347996 11485683 11727110 11796941 12233544 12723309 12889002 13167991 13314240 13331118 13471851 13563383 13707676 14157669 14737865 14795911 14837286 14923311 15030096 15150513 15341837 15768860 15855350 15923713 15972347 16138559 16194446 16272064 16494372 16540142 16554318 16689280 16759320
24	127	0.984090 (0.930309)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 605300 649314 701023 842347 901542 902444 935950 972344 1107341 1216620 1391649 1681295 1700311 1849660 2435061 2504405 2801845 3274804 3646867 3727802 3837096 3857484 4046019 4292634 4410637 4447108 4642815 4661116 4697043 4754320 4865882 4903110 4933529 4981695 5087490 5104610 5170378 5170573 5176601 5211409 5493873 6280895 6813668 6982303 7139876 7156527 7663866 7894909 7933395 8498409 8644652 8678584 8894748 9008563 9265730 9354907 9450080 9466165 10033669 10290550 10320865 10775437 10887849 11095914 11098497 11204875 11266610 11347996 11485683 11727110 11796941 12233544 12723309 12889002 13167991 13314240 13331118 13471851 13563383 13707676 14157669 14737865 14795911 14837286 14923311 15030096 15150513 15341837 15768860 15855350 15923713 15972347 16138559 16194446 16272064 16494372 16540142 16554318 16689280 16759320
24	128	0.984936 (0.930896)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 576634 605300 649314 701023 842347 901542 902444 935950 972344 1107341 1216620 1391649 1681295 1700311 1849660 2435061 2504405 2801845 3274804 3646867 3727802 3837096 3857484 4046019 4292634 4410637 4447108 4642815 4661116 4697043 4754320 4865882 4903110 4933529 4981695 5087490 5104610 5170378 5170573 5176601 5211409 5493873 6280895 6813668 6982303 7139876 7156527 7663866 7894909 7933395 8498409 8644652 8678584 8894748 9008563 9265730 9354907 9450080 9466165 10033669 10290550 10320865 10775437 10887849 11095914 11098497 11204875 11266610 11347996 11485683 11727110 11796941 12233544 12723309 12889002 13167991 13314240 13331118 13471851 13563383 13707676 14157669 14737865 14795911 14837286 14923311 15030096 15150513 15341837 15768860 15855350 15923713 15972347 16138559 16194446 16272064 16494372 16540142 16554318 16689280 16759320
24	129	0.985741 (0.931452)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 576634 605300 649314 701023 842347 901542 902444 935950 972344 1107341 1216620 1391649 1681295 1700311 1849660 2435061 2504405 2801845 3274804 3646867 3727802 3837096 3857484 4046019 4292634 4410637 4447108 4642815 4661116 4697043 4754320 4865882 4903110 4933529 4981695 5087490 5104610 5170378 5170573 5176601 5211409 5493873 6280895 6813668 6982303 7139876 7156527 7663866 7894909 7933395 8498409 8644652 8678584 8894748 9008563 9265730 9354907 9450080 9466165 10033669 10290550 10320865 10775437 10887849 11095914 11098497 11204875 11266610 11347996 11485683 11727110 11796941 12233544 12723309 12889002 13167991 13314240 13331118 13471851 13563383 13707676 14157669 14736091 14737865 14795911 14837286 14923311 15030096 15150513 15341837 15768860 15855350 15923713 15972347 16138559 16194446 16272064 16494372 16540142 16554318 16689280 16759320
24	130	0.986507 (0.943269)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 576634 605300 649314 701023 842347 901542 902444 935950 972344 1107341 1216620 1391649 1681295 1700311 1849660 2435061 2504405 2801845 3274804 3646867 3727802 3837096 3857484 4046019 4292634 4410637 4447108 4642815 4661116 4697043 4754320 4865882 4903110 4933529 4981695 5087490 5104610 5170378 5170573 5176601 5211409 5493873 6280895 6813668 6982303 7139876 7156527 7663866 7894909 7933395 8498409 8644652 8678584 8894748 9008563 9265730 9354907 9450080 9466165 10033669 10290550 10320865 10775437 10887849 11095914 11098497 11204875 11266610 11346763 11347996 11485683 11727110 11796941 12233544 12723309 12889002 13167991 13314240 13331118 13471851 13563383 13707676 14157669 14736091 14737865 14795911 14837286 14923311 15030096 15150513 15341837 15768860 15855350 15923713 15972347 16138559 16194446 16272064 16494372 16540142 16554318 16689280 16759320
24	131	0.987236 (0.947356)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 576634 605300 649314 701023 842347 901542 902444 935950 972344 1107341 1216620 1391649 1681295 1700311 1849660 2435061 2504405 2801845 3274804 3646867 3727802 3837096 3856259 3857484 4046019 4292634 4410637 4447108 4642815 4661116 4697043 4754320 4865882 4903110 4933529 4981695 5087490 5104610 5170378 5170573 5176601 5211409 5493873 6280895 6813668 6982303 7139876 7156527 7663866 7894909 7933395 8498409 8644652 8678584 8894748 9008563 9265730 9354907 9450080 9466165 10033669 10290550 10320865 10775437 10887849 11095914 11098497 11204875 11266610 11346763 11347996 11485683 11727110 11796941 12233544 12723309 12889002 13167991 13314240 13331118 13471851 13563383 13707676 14157669 14736091 14737865 14795911 14837286 14923311 15030096 15150513 15341837 15768860 15855350 15923713 15972347 16138559 16194446 16272064 16494372 16540142 16554318 16689280 16759320
24	132	0.987928 (0.950596)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 576634 605300 649314 701023 842347 901542 902444 935950 972344 1107341 1216620 1391649 1681295 1700311 1849660 2435061 2504405 2801845 3274804 3646867 3727802 3837096 3856259 3857484 4046019 4292634 4410637 4447108 4642815 4661116 4697043 4754320 4865882 4903110 4933529 4981695 5087490 5104610 5170378 5170573 5176601 5211409 5493873 6280895 6813668 6982303 7139876 7156527 7663866 7894909 7933395 8498409 8644652 8678584 8894748 9008563 9265730 9354907 9450080 9466165 10033669 10290550 10320865 10775437 10887849 11095914 11098497 11204875 11266610 11346763 11347996 11485683 11727110 11796941 12233544 12723309 12889002 13167991 13314240 13331118 13471851 13563383 13707676 14157669 14736091 14737865 14795911 14837286 14923311 15030096 15150513 15340608 15341837 15768860 15855350 15923713 15972347 16138559 16194446 16272064 16494372 16540142 16554318 16689280 16759320

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
24	133	0.988586 (0.953211)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 576634 605300 649314 701023 842347 901542 902444 935950 972344 1107341 1216620 1391649 1681295 1700311 1849660 2435061 2504405 2801845 3274804 3646867 3727802 3837096 3856259 3857484 4046019 4292634 4410043 4410637 4447108 4642815 4661116 4697043 4754320 4865882 4903110 4933529 4981695 5087490 5104610 5170378 5170573 5176601 5211409 5493873 6280895 6813668 6982303 7139876 7156527 7663866 7894909 7933395 8498409 8644652 8678584 8894748 9008563 9265730 9354907 9450080 9466165 10033669 10290550 10320865 10775437 10887849 11095914 11098497 11204875 11266610 11346763 11347996 11485683 11727110 11796941 12233544 12723309 12889002 13167991 13314240 13331118 13471851 13563383 13707676 14157669 14736091 14737865 14795911 14837286 14923311 15030096 15150513 15340608 15341837 15768860 15855350 15923713 15972347 16138559 16194446 16272064 16494372 16540142 16554318 16689280 16759320
24	134	0.989212 (0.955350)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 576634 605300 649314 701023 842347 901542 902444 935950 972344 1107341 1216620 1391649 1681295 1700311 1849660 2435061 2504405 2801845 3274804 3646867 3727802 3837096 3856259 3857484 4046019 4292634 4410043 4410637 4447108 4642815 4661116 4697043 4754320 4865882 4903110 4933529 4981695 5087490 5104610 5170378 5170573 5176601 5211409 5493873 6280895 6813668 6982303 7139876 7156527 7663866 7894909 7933395 8498409 8644652 8678584 8894748 9008563 9265730 9354907 9450080 9466165 10033669 10290550 10320865 10775437 10887849 11095914 11098497 11204875 11266610 11346763 11347996 11485683 11724867 11727110 11796941 12233544 12723309 12889002 13167991 13314240 13331118 13471851 13563383 13707676 14157669 14736091 14737865 14795911 14837286 14923311 15030096 15150513 15340608 15341837 15768860 15855350 15923713 15972347 16138559 16194446 16272064 16494372 16540142 16554318 16689280 16759320
24	135	0.989805 (0.957120)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 576634 605300 649314 701023 842347 901542 902444 935950 972344 1107341 1216620 1391649 1681295 1700311 1849660 2435061 2504405 2801845 3274804 3646867 3727802 3837096 3856259 3857484 4046019 4292634 4410043 4410637 4447108 4642815 4661116 4697043 4754320 4865882 4903110 4933529 4981695 5087490 5104610 5170378 5170573 5176601 5211409 5493873 6280895 6813668 6982303 7139876 7156527 7663866 7894909 7933395 8498409 8644652 8678584 8894748 9008563 9265730 9354907 9450080 9466165 10033669 10290550 10320865 10775437 10887849 11095914 11098497 11204875 11266610 11346763 11347996 11485683 11724867 11727110 11796941 12233544 12723309 12889002 13167991 13314240 13331118 13471851 13563383 13707676 14157669 14736091 14737865 14795911 14833094 14837286 14923311 15030096 15150513 15340608 15341837 15768860 15855350 15923713 15972347 16138559 16194446 16272064 16494372 16540142 16554318 16689280 16759320
24	136	0.990369 (0.958598)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 576634 601560 605300 649314 701023 842347 901542 902444 935950 972344 1107341 1216620 1391649 1681295 1700311 1849660 2435061 2504405 2801845 3274804 3646867 3727802 3837096 3856259 3857484 4046019 4292634 4410043 4410637 4447108 4642815 4661116 4697043 4754320 4865882 4903110 4933529 4981695 5087490 5104610 5170378 5170573 5176601 5211409 5493873 6280895 6813668 6982303 7139876 7156527 7663866 7894909 7933395 8498409 8644652 8678584 8894748 9008563 9265730 9354907 9450080 9466165 10033669 10290550 10320865 10775437 10887849 11095914 11098497 11204875 11266610 11346763 11347996 11485683 11724867 11727110 11796941 12233544 12723309 12889002 13167991 13314240 13331118 13471851 13563383 13707676 14157669 14736091 14737865 14795911 14833094 14837286 14923311 15030096 15150513 15340608 15341837 15768860 15855350 15923713 15972347 16138559 16194446 16272064 16494372 16540142 16554318 16689280 16759320
24	137	0.990905 (0.959012)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 576634 601560 605300 649314 701023 842347 901542 902444 935950 972344 1107341 1216620 1391649 1681295 1700311 1849660 2435061 2504405 2801845 3274804 3646867 3727802 3837096 3856259 3857484 4046019 4292634 4410043 4410637 4447108 4642815 4661116 4697043 4754320 4865882 4903110 4933529 4981695 5087490 5104610 5170378 5170573 5176601 5211409 5493873 6280895 6813668 6982303 7139876 7156527 7663866 7894909 7933395 8498409 8644652 8678584 8894748 9008563 9265730 9354907 9450080 9466165 10033669 10290550 10320865 10775437 10887849 11095914 11098497 11204875 11266610 11346763 11347996 11485683 11724867 11727110 11796941 12233544 12723309 12889002 13167991 13314240 13331118 13471851 13563383 13707676 14157669 14736091 14737865 14795911 14833094 14837286 14923311 15030096 15031033 15150513 15340608 15341837 15768860 15855350 15923713 15972347 16138559 16194446 16272064 16494372 16540142 16554318 16689280 16759320
24	138	0.991412 (0.960238)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 576634 601560 605300 649314 701023 842347 901542 902444 935950 969123 972344 1107341 1216620 1391649 1681295 1700311 1849660 2435061 2504405 2801845 3274804 3646867 3727802 3837096 3856259 3857484 4046019 4292634 4410043 4410637 4447108 4642815 4661116 4697043 4754320 4865882 4903110 4933529 4981695 5087490 5104610 5170378 5170573 5176601 5211409 5493873 6280895 6813668 6982303 7139876 7156527 7663866 7894909 7933395 8498409 8644652 8678584 8894748 9008563 9265730 9354907 9450080 9466165 10033669 10290550 10320865 10775437 10887849 11095914 11098497 11204875 11266610 11346763 11347996 11485683 11724867 11727110 11796941 12233544 12723309 12889002 13167991 13314240 13331118 13471851 13563383 13707676 14157669 14736091 14737865 14795911 14833094 14837286 14923311 15030096 15031033 15150513 15340608 15341837 15768860 15855350 15923713 15972347 16138559 16194446 16272064 16494372 16540142 16554318 16689280 16759320
24	139	0.991894 (0.961277)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 576634 601560 605300 649314 701023 842347 901542 902444 935950 969123 972344 1107341 1216620 1391649 1681295 1700311 1849660 2435061 2504405 2801845 3274804 3646867 3727802 3837096 3856259 3857484 4046019 4292634 4410043 4410637 4447108 4642815 4661116 4697043 4751556 4754320 4865882 4903110 4933529 4981695 5087490 5104610 5170378 5170573 5176601 5211409 5493873 6280895 6813668 6982303 7139876 7156527 7663866 7894909 7933395 8498409 8644652 8678584 8894748 9008563 9265730 9354907 9450080 9466165 10033669 10290550 10320865 10775437 10887849 11095914 11098497 11204875 11266610 11346763 11347996 11485683 11724867 11727110 11796941 12233544 12723309 12889002 13167991 13314240 13331118 13471851 13563383 13707676 14157669 14736091 14737865 14795911 14833094 14837286 14923311 15030096 15031033 15150513 15340608 15341837 15768860 15855350 15923713 15972347 16138559 16194446 16272064 16494372 16540142 16554318 16689280 16759320

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
24	140	0.992350 (0.962166)	6	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 508248 573185 576055 576634 601560 605300 649314 701023 842347 901542 902444 935950 969123 972344 1107341 1216620 1391649 1681295 1700311 1849660 2435061 2504405 2801845 3274804 3646867 3727802 3837096 3856259 3857484 4046019 4292634 4410043 4410637 4447108 4642815 4661116 4697043 4751556 4754320 4865882 4903110 4933529 4981695 5087490 5104610 5170378 5170573 5176601 5211409 5493873 6280895 6813668 6982303 7139876 7156527 7663866 7894909 7933395 8498409 8644652 8678584 8894748 9008563 9265730 9354907 9450080 9466165 10033669 10290550 10320865 10775437 10887849 11091655 11095914 11098497 11204875 11266610 11346763 11347996 11485683 11724867 11727110 11796941 12233544 12723309 12889002 13167991 13314240 13331118 13471851 13563383 13707676 14157669 14736091 14737865 14795911 14833094 14837286 14923311 15030096 15031033 15150513 15340608 15341837 15768860 15855350 15923713 15972347 16138559 16194446 16272064 16494372 16540142 16554318 16689280 16759320
25	26	0.297853	26	13	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554431
25	27	0.309309	18	13	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33489151 33554176
25	28	0.320764	16	14	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 10340479 33424376 33553408
25	29	0.33222	15	14	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 13839963 31581127 33424376 33553408
25	30	0.343675	15	14	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 10908909 13839963 31581127 33424376 33553408
25	31	0.35513	15	11	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 6994294 10908909 13839963 31581127 33424376 33553408
25	32	0.366582	14	13	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 7895282 10419908 19470307 22368942 26843956 30597213 33424376
25	33	0.378033	14	11	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 3845689 7895282 10419908 19470307 22368942 26843956 30597213 33424376
25	34	0.389478 (0.389479)	12	11	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 3845689 7895282 10342507 10419908 19470307 22368942 26843956 30597213 33424376
25	35	0.400918 (0.400862)	12	11	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 2286058 3845689 7895282 10342507 10419908 19470307 22368942 26843956 30597213 33424376
25	36	0.412351 (0.412275)	12	11	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 2286058 3299599 3845689 7895282 10342507 10419908 19470307 22368942 26843956 30597213 33424376
25	37	0.423774 (0.423670)	12	11	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 2286058 3299599 3845689 7895282 10342507 10419908 19470307 22368942 26843956 29130072 30597213 33424376
25	38	0.435185 (0.435103)	12	10	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 2286058 3299599 3845689 5042513 7895282 10342507 10419908 19470307 22368942 26843956 29130072 30597213 33424376
25	39	0.446579 (0.446477)	11	10	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 2286058 3299599 3845689 5042513 7895282 10342507 10419908 10517036 19470307 22368942 26843956 29130072 30597213 33424376
25	40	0.457956 (0.457829)	11	10	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 2286058 3299599 3845689 4285667 5042513 7895282 10342507 10419908 10517036 19470307 22368942 26843956 29130072 30597213 33424376
25	41	0.469307 (0.469151)	11	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 2286058 3299599 3845689 4285667 5042513 7895282 10342507 10419908 10517036 15709933 19470307 22368942 26843956 29130072 30597213 33424376
25	42	0.480621 (0.480442)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 2286058 3299599 3845689 4285667 5042513 7895282 10342507 10369181 10419908 10517036 15709933 19470307 22368942 26843956 29130072 30597213 33424376
25	43	0.491909 (0.491694)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1912776 2286058 3299599 3845689 4285667 5042513 7895282 10342507 10369181 10419908 10517036 15709933 19470307 22368942 26843956 29130072 30597213 33424376
25	44	0.503154 (0.502902)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1168348 1912776 2286058 3299599 3845689 4285667 5042513 7895282 10342507 10369181 10419908 10517036 15709933 19470307 22368942 26843956 29130072 30597213 33424376
25	45	0.514356 (0.514060)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1168348 1912776 2286058 3299599 3845689 4285667 5042513 7895282 10342507 10369181 10419908 10517036 15709933 19470307 22368942 26843956 29130072 30597213 32528240 33424376
25	46	0.525507 (0.525163)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1168348 1912776 2286058 3299599 3845689 4285667 5042513 7895282 10342507 10369181 10419908 10517036 15709933 19470307 22368942 26843956 28700830 29130072 30597213 32528240 33424376
25	47	0.536601 (0.536203)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1168348 1912776 2286058 3299599 3845689 4285667 5042513 7895282 10342507 10369181 10419908 10517036 15709933 19470307 21402133 22368942 26843956 28700830 29130072 30597213 32528240 33424376
25	48	0.547633 (0.547175)	10	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1168348 1912776 2286058 3299599 3845689 4285667 5042513 7895282 10342507 10369181 10419908 10517036 15709933 19045699 19470307 21402133 22368942 26843956 28700830 29130072 30597213 32528240 33424376
25	49	0.558583 (0.558071)	9	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1168348 1912776 2286058 3299599 3845689 4285667 5042513 7895282 10342507 10349577 10369181 10419908 10517036 15709933 19045699 19470307 21402133 22368942 26843956 28700830 29130072 30597213 32528240 33424376

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
25	50	0.569458 (0.567176)	9	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1168348 1812374 1912776 2286058 3299599 3845689 4285667 5042513 7895282 10342507 10349577 10369181 10419908 10517036 15709933 19045699 19470307 21402133 22368942 26843956 28700830 29130072 30597213 32528240 33424376
25	51	0.580253 (0.578285)	9	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1168348 1203837 1812374 1912776 2286058 3299599 3845689 4285667 5042513 7895282 10342507 10349577 10369181 10419908 10517036 15709933 19045699 19470307 21402133 22368942 26843956 28700830 29130072 30597213 32528240 33424376
25	52	0.590962 (0.589061)	9	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1168348 1203837 1812374 1912776 2286058 3299599 3845689 4285667 5042513 7895282 10342507 10349577 10369181 10419908 10517036 15709933 19045699 19470307 21402133 22368942 26843956 28700830 29130072 29552947 30597213 32528240 33424376
25	53	0.601574 (0.599699)	9	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1168348 1203837 1812374 1912776 2286058 3299599 3845689 4285667 5042513 7895282 10342507 10349577 10369181 10419908 10517036 15709933 19045699 19470307 21402133 22368942 22800504 26843956 28700830 29130072 29552947 30597213 32528240 33424376
25	54	0.612081 (0.610144)	9	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1168348 1203837 1812374 1912776 2286058 3299599 3845689 4285667 5042513 7895282 7996354 10342507 10349577 10369181 10419908 10517036 15709933 19045699 19470307 21402133 22368942 22800504 26843956 28700830 29130072 29552947 30597213 32528240 33424376
25	55	0.622481 (0.620453)	9	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1168348 1203837 1812374 1912776 2286058 3299599 3845689 4285667 5042513 7895282 7996354 10342507 10349577 10369181 10419908 10517036 15709933 19045699 19470307 21402133 22368942 22800504 26843956 28700830 29130072 29552947 30597213 31115935 32528240 33424376
25	56	0.632764 (0.632499)	9	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1168348 1203837 1812374 1912776 2286058 3299599 3845689 4285667 5042513 7895282 7996354 10342507 10349577 10369181 10419908 10517036 12807199 15709933 19045699 19470307 21402133 22368942 22800504 26843956 28700830 29130072 29552947 30597213 31115935 32528240 33424376
25	57	0.642934 (0.642669)	9	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1168348 1203837 1812374 1912776 2286058 3299599 3845689 4285667 5042513 7895282 7996354 10342507 10349577 10369181 10419908 10517036 12807199 15709933 19045699 19470307 21402133 22368942 22800504 26041256 26843956 28700830 29130072 29552947 30597213 31115935 32528240 33424376
25	58	0.652970 (0.652719)	9	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1168348 1203837 1812374 1912776 2232798 2286058 3299599 3845689 4285667 5042513 7895282 7996354 10342507 10349577 10369181 10419908 10517036 12807199 15709933 19045699 19470307 21402133 22368942 22800504 26041256 26843956 28700830 29130072 29552947 30597213 31115935 32528240 33424376
25	59	0.662871 (0.662642)	9	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1168348 1203837 1812374 1912776 2232798 2286058 3299599 3845689 4285667 5042513 7895282 7996354 10342507 10349577 10369181 10419908 10517036 12807199 15709933 19045699 19139598 19470307 21402133 22368942 22800504 26041256 26843956 28700830 29130072 29552947 30597213 31115935 32528240 33424376
25	60	0.672634 (0.672426)	9	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1168348 1203837 1812374 1912776 2232798 2286058 3299599 3845689 4285667 5042513 7895282 7996354 10342507 10349577 10369181 10419908 10517036 12807199 15709933 16192222 19045699 19139598 19470307 21402133 22368942 22800504 26041256 26843956 28700830 29130072 29552947 30597213 31115935 32528240 33424376
25	61	0.682243 (0.682070)	9	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1168348 1203837 1812374 1912776 2232798 2286058 3299599 3845689 4285667 5042513 7895282 7996354 10342507 10349577 10369181 10419908 10517036 12807199 15709933 16192222 19045699 19139598 19470307 21402133 21730182 22368942 22800504 26041256 26843956 28700830 29130072 29552947 30597213 31115935 32528240 33424376
25	62	0.691700 (0.691569)	9	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1168348 1203837 1812374 1912776 2232798 2286058 3299599 3845689 4285667 5042513 7895282 7996354 10342507 10349577 10369181 10419908 10517036 12807199 15709933 16192222 19045699 19139598 19470307 21402133 21730182 22368942 22800504 25990493 26041256 26843956 28700830 29130072 29552947 30597213 31115935 32528240 33424376
25	63	0.701002 (0.700921)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1168348 1203837 1812374 1912776 2232798 2286058 3299599 3845689 4285667 5042513 7895282 7996354 10342507 10348782 10349577 10369181 10419908 10517036 12807199 15709933 16192222 19045699 19139598 19470307 21402133 21730182 22368942 22800504 25990493 26041256 26843956 28700830 29130072 29552947 30597213 31115935 32528240 33424376
25	64	0.710155 (0.710116)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1168348 1203837 1807215 1812374 1912776 2232798 2286058 3299599 3845689 4285667 5042513 7895282 7996354 10342507 10348782 10349577 10369181 10419908 10517036 12807199 15709933 16192222 19045699 19139598 19470307 21402133 21730182 22368942 22800504 25990493 26041256 26843956 28700830 29130072 29552947 30597213 31115935 32528240 33424376
25	65	0.719149	10	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 194979 202365 389958 404730 779916 809460 1559832 1618920 2321243 3119664 3237840 3585457 4642486 6239328 6475680 6656631 7170914 7808211 9267443 9284972 12478656 12951360 13313262 13571761 14341828 15616422 17493877 18490889 18534886 18569944 18925745 19502051 22933153 24957312 25902720 26186537 26626524 27143522 28683656 31232844
25	66	0.728004 (0.725270)	9	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 194979 202365 389958 404730 779916 809460 1559832 1618920 2321243 3119664 3237840 3585457 4642486 6239328 6475680 6656631 7170914 7808211 9267443 9284972 10481979 12478656 12951360 13313262 13571761 14341828 15616422 17493877 18490889 18534886 18569944 18925745 19502051 22933153 24957312 25902720 26186537 26626524 27143522 28683656 31232844

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
25	67	0.736701 (0.734594)	9	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 194979 202365 389958 404730 779916 809460 1559832 1618920 2321243 3119664 3237840 3585457 4642486 6239328 6475680 6656631 7170914 7808211 9267443 9284972 10157246 10481979 12478656 12951360 13313262 13571761 14341828 15616422 17493877 18490889 18534886 18569944 18925745 19502051 22933153 24957312 25902720 26186537 26626524 27143522 28683656 31232844
25	68	0.745131 (0.743595)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1162144 1168348 1203837 1807215 1812374 1912776 2232798 2286058 3299599 3845689 4285667 5042513 7729011 7895282 7996354 10342507 10348782 10349577 10369181 10419908 10517036 12807199 15709933 16192222 19045699 19139598 19470307 21402133 21730182 22368942 22704308 22800504 25990493 26041256 26843956 28700830 29130072 29480772 29552947 30597213 31115935 32528240 33424376
25	69	0.753463 (0.752286)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1162144 1168348 1203837 1807215 1812374 1912776 2232798 2286058 3299599 3845689 4285667 5042513 7729011 7895282 7996354 10342507 10348782 10349577 10369181 10419908 10517036 12807199 15709933 16192222 19045699 19139598 19470307 21402133 21730182 22368942 22704308 22800504 25990493 26041256 26843956 28700830 29130072 29480772 29552947 30597213 31115935 32528240 33424376
25	70	0.761615 (0.760669)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1162144 1168348 1203837 1807215 1812374 1912776 2232798 2286058 3299599 3845689 4285667 5042513 7729011 7895282 7996354 8851898 10342507 10348782 10349577 10369181 10419908 10517036 12807199 15709933 16192222 19045699 19139598 19470307 21402133 21730182 22368942 22704308 22800504 25990493 26041256 26843956 28700830 29130072 29480772 29552947 30597213 30688342 31115935 32528240 33424376
25	71	0.769589 (0.768865)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1162144 1168348 1203837 1807215 1812374 1912776 2232798 2286058 3299599 3845689 4285667 5042513 7729011 7895282 7996354 8851898 10342507 10348782 10349577 10369181 10419908 10517036 12807199 15709933 16192222 19045699 19139598 19470307 21402133 21730182 22368942 22704308 22800504 23514363 25990493 26041256 26843956 28700830 29130072 29480772 29552947 30597213 30688342 31115935 32528240 33424376
25	72	0.777386 (0.776838)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1162144 1168348 1203837 1807215 1812374 1912776 2232798 2286058 3299599 3845689 4285667 5042513 7729011 7895282 7996354 8851898 10342507 10348782 10349577 10369181 10419908 10517036 12807199 15709933 16192222 19045699 19139598 19470307 21402133 21730182 22368942 22704308 22800504 23514363 25990493 26041256 26843956 28700830 29130072 29480772 29552947 29848995 30597213 30688342 31115935 32528240 33424376
25	73	0.785010 (0.734238)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1162144 1168348 1203837 1294333 1807215 1812374 1912776 2232798 2286058 3299599 3845689 4285667 5042513 7729011 7895282 7996354 8851898 10342507 10348782 10349577 10369181 10419908 10517036 12807199 15709933 16192222 19045699 19139598 19470307 21402133 21730182 22368942 22704308 22800504 23514363 25990493 26041256 26843956 28700830 29130072 29480772 29552947 29848995 30597213 30688342 31115935 32528240 33424376
25	74	0.792455 (0.739833)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1162144 1168348 1203837 1294333 1807215 1812374 1912776 2232798 2286058 3299599 3845689 4285667 5042513 7729011 7895282 7996354 8851898 10342507 10348782 10349577 10369181 10419908 10517036 12807199 15709933 16192222 19045699 19139598 19470307 21402133 21730182 22368942 22704308 22800504 23514363 25990493 26041256 26843956 28700830 29130072 29480772 29552947 29848995 30355192 30597213 30688342 31115935 32528240 33424376
25	75	0.799721 (0.745091)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1162144 1168348 1203837 1294333 1807215 1812374 1912776 2232798 2286058 3299599 3845689 4285667 5042513 7729011 7895282 7996354 8851898 10342507 10348782 10349577 10369181 10419908 10517036 12807199 15709933 16192222 19045699 19139598 19470307 21402133 21730182 22368942 22704308 22800504 23514363 25990493 26041256 26843956 28700830 29130072 29480772 29552947 29848995 30355192 30597213 30688342 31115935 32528240 33424376
25	76	0.806816 (0.750051)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1162144 1168348 1203837 1294333 1807215 1812374 1912776 2232798 2286058 2337225 3299599 3845689 4285667 5042513 7729011 7895282 7996354 8851898 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12807199 15709933 16192222 19045699 19139598 19470307 21402133 21730182 22368942 22704308 22800504 23514363 25990493 26041256 26843956 28700830 29130072 29480772 29552947 29848995 30355192 30597213 30688342 31115935 32528240 33424376
25	77	0.813726 (0.754888)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1162144 1168348 1203837 1294333 1807215 1812374 1912776 2232798 2286058 2337225 3299599 3845689 4285667 5042513 7729011 7895282 7996354 8851898 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12807199 15709933 16192222 19045699 19139598 19470307 21402133 21730182 22368942 22704308 22800504 23514363 24577428 25990493 26041256 26843956 28700830 29130072 29480772 29552947 29848995 30355192 30597213 30688342 31115935 32528240 33424376
25	78	0.820462 (0.759384)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1162144 1168348 1203837 1294333 1807215 1812374 1912776 2232798 2286058 2337225 3299599 3845689 4285667 5042513 7729011 7895282 7996354 8851898 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12807199 15709933 16192222 19045699 19139598 19470307 21402133 21730182 22368942 22704308 22800504 23514363 24577428 25990493 26041256 26843956 28700830 29130072 29370079 29480772 29552947 29848995 30355192 30597213 30688342 31115935 32528240 33424376
25	79	0.827027 (0.763603)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1162144 1168348 1203837 1294333 1807215 1812374 1912776 2232798 2286058 2337225 3299599 3845689 4285667 5042513 7729011 7895282 7996354 8851898 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12807199 14438887 15709933 16192222 19045699 19139598 19470307 21402133 21730182 22368942 22704308 22800504 23514363 24577428 25990493 26041256 26843956 28700830 29130072 29370079 29480772 29552947 29848995 30355192 30597213 30688342 31115935 32528240 33424376

Continued on next page

	m	n	R_e	d	R_c	Packed integer form of the parity check matrix
25	80		0.833412 (0.768130)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1162144 1168348 1203837 1294333 1807215 1812374 1912776 2232798 2286058 2337225 3299599 3845689 4285667 5042513 7729011 7895282 7996354 8851898 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12807199 14438887 15709933 16150229 16192222 19045699 19139598 19470307 21402133 21730182 22368942 22704308 22800504 23514363 24577428 25990493 26041256 26843956 28700830 29130072 29370079 29480772 29552947 29848995 30355192 30597213 30688342 31115935 32528240 33424376
25	81		0.839621 (0.772284)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1162144 1168348 1203837 1294333 1807215 1812374 1912776 2232798 2286058 2337225 3299599 3845689 4285667 5042513 7729011 7895282 7996354 8851898 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 14438887 15709933 16150229 16192222 19045699 19139598 19470307 21402133 21730182 22368942 22704308 22800504 23514363 24577428 25990493 26041256 26843956 28700830 29130072 29370079 29480772 29552947 29848995 30355192 30597213 30688342 31115935 32528240 33424376
25	82		0.845650 (0.776488)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1162144 1168348 1203837 1294333 1807215 1812374 1912776 2232798 2286058 2337225 3299599 3845689 4285667 5042513 7729011 7895282 7996354 8851898 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 14438887 15709933 16150229 16192222 19045699 19139598 19470307 21402133 21730182 22368942 22704308 22800504 23514363 24577428 25990493 26041256 26435187 26843956 28700830 29130072 29370079 29480772 29552947 29848995 30355192 30597213 30688342 31115935 32528240 33424376
25	83		0.851508 (0.780494)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1162144 1168348 1203837 1294333 1807215 1812374 1912776 2232798 2286058 2337225 3299599 3845689 4285667 5042513 7729011 7895282 7996354 8851898 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 14438887 15709933 16150229 16192222 19045699 19139598 19470307 21402133 21730182 22368942 22704308 22800504 23514363 24577428 25990493 26041256 26435187 26843956 28700830 29130072 29370079 29480772 29552947 29642880 29848995 30355192 30597213 30688342 31115935 32528240 33424376
25	84		0.857193 (0.784245)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1162144 1168348 1203837 1294333 1807215 1812374 1912776 2232798 2286058 2337225 3299599 3845689 4285667 5042513 7729011 7895282 7996354 8851898 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 15709933 16150229 16192222 19045699 19139598 19470307 21402133 21730182 22368942 22704308 22800504 23514363 24577428 25990493 26041256 26435187 26843956 28700830 29130072 29370079 29480772 29552947 29642880 29848995 30355192 30597213 30688342 31115935 32528240 33424376
25	85		0.862712 (0.787887)	8	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1162144 1168348 1203837 1294333 1807215 1812374 1912776 2232798 2286058 2337225 3299599 3845689 4285667 5042513 7729011 7895282 7996354 8851898 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 15709933 16150229 16192222 19045699 19139598 19470307 21402133 21730182 22368942 22704308 22800504 23514363 24577428 25990493 26041256 26435187 26843956 28700830 29130072 29370079 2

181

	m	n	R_e	d	R_c	Packed integer form of the parity check matrix
25	91		0.892390 (0.814880)	7	6	$\begin{bmatrix} 1 & 2 & 4 & 8 & 16 & 32 & 64 & 128 & 256 & 512 & 1024 & 2048 & 4096 & 8192 & 16384 & 32768 & 65536 & 131072 & 262144 & 524288 & 1048576 & 2097152 & 4194304 \\ 8388608 & 16777216 & 1153115 & 1162144 & 1168348 & 1203837 & 1294333 & 1802840 & 1807215 & 1812374 & 1912776 & 2232798 & 2286058 & 2337225 & 3299599 & 3845689 & 4285667 & 5042513 & 7712994 & 7729011 & 7895282 & 7996354 & 8851898 & 10341228 & 10342507 & 10348782 & 10349577 \\ 10369181 & 10419908 & 10517036 & 10746226 & 12700471 & 12807199 & 13161042 & 14438887 & 14537918 & 15709933 & 16150229 & 16192222 & 19045699 & 19139598 & 19470307 & 21402133 & 21730182 & 22368942 & 22692960 & 22704308 & 22800504 & 23514363 & 24577428 & 25990493 & 26041256 & 26435187 & 26843956 & 28700830 & 29130072 & 29370079 & 29474042 & 29480772 & 29552947 & 29642880 & 29848995 & 30355192 \\ 30597213 & 30688342 & 31115935 & 32528240 & 33424376 & \end{bmatrix}$
25	92		0.896799 (0.820255)	7	6	$\begin{bmatrix} 1 & 2 & 4 & 8 & 16 & 32 & 64 & 128 & 256 & 512 & 1024 & 2048 & 4096 & 8192 & 16384 & 32768 & 65536 & 131072 & 262144 & 524288 & 1048576 & 2097152 & 4194304 \\ 8388608 & 16777216 & 1153115 & 1162144 & 1168348 & 1203837 & 1294333 & 1802840 & 1807215 & 1812374 & 1912776 & 2232798 & 2286058 & 2337225 & 3299599 & 3845689 & 4285667 & 5042513 & 7712994 & 7729011 & 7895282 & 7996354 & 8851898 & 10341228 & 10342507 & 10348782 & 10349577 \\ 10369181 & 10419908 & 10517036 & 10746226 & 12700471 & 12807199 & 13161042 & 14438887 & 14537918 & 15709933 & 16150229 & 16192222 & 19045699 & 19139598 & 19470307 & 21402133 & 21730182 & 22368942 & 22692960 & 22704308 & 22800504 & 23514363 & 24577428 & 25990493 & 26041256 & 26435187 & 26843956 & 28700830 & 29130072 & 29370079 & 29474042 & 29480772 & 29552947 & 29642880 & 29848995 & 30355192 \\ 30597213 & 30688342 & 31115935 & 32528240 & 33424376 & \end{bmatrix}$
25	93		0.901058 (0.824905)	7	6	$\begin{bmatrix} 1 & 2 & 4 & 8 & 16 & 32 & 64 & 128 & 256 & 512 & 1024 & 2048 & 4096 & 8192 & 16384 & 32768 & 65536 & 131072 & 262144 & 524288 & 1048576 & 2097152 & 4194304 \\ 8388608 & 16777216 & 1153115 & 1162144 & 1168348 & 1203837 & 1294333 & 1802840 & 1807215 & 1812374 & 1912776 & 2232798 & 2286058 & 2337225 & 3299599 & 3845689 & 4285667 & 5042513 & 7712994 & 7729011 & 7895282 & 7996354 & 8818602 & 8851898 & 10341228 & 10342507 & 10348782 & 10349577 \\ 10369181 & 10419908 & 10517036 & 10746226 & 12700471 & 12807199 & 13161042 & 14438887 & 14537918 & 15709933 & 16150229 & 16192222 & 19045699 & 19139598 & 19470307 & 21402133 & 21730182 & 22368942 & 22692960 & 22704308 & 22800504 & 23514363 & 24577428 & 25990493 & 26041256 & 26435187 & 26843956 & 28700830 & 29130072 & 29370079 & 29474042 & 29480772 & 29552947 & 29642880 & 29848995 & 30355192 \\ 30597213 & 30688342 & 31115935 & 32528240 & 33424376 & \end{bmatrix}$
25	94		0.905179 (0.828987)	7	6	$\begin{bmatrix} 1 & 2 & 4 & 8 & 16 & 32 & 64 & 128 & 256 & 512 & 1024 & 2048 & 4096 & 8192 & 16384 & 32768 & 65536 & 131072 & 262144 & 524288 & 1048576 & 2097152 & 4194304 \\ 8388608 & 16777216 & 1153115 & 1162144 & 1168348 & 1203837 & 1294333 & 1802840 & 1807215 & 1812374 & 1912776 & 2232798 & 2286058 & 2337225 & 3299599 & 3845689 & 4285667 & 5042513 & 7712994 & 7729011 & 7895282 & 7996354 & 8818602 & 8851898 & 10341228 & 10342507 & 10348782 & 10349577 \\ 10369181 & 10419908 & 10517036 & 10746226 & 12700471 & 12807199 & 13161042 & 14438887 & 14537918 & 15709933 & 16150229 & 16192222 & 19045699 & 19139598 & 19470307 & 21402133 & 21730182 & 22368942 & 22692960 & 22704308 & 22800504 & 23514363 & 24577428 & 25990493 & 26041256 & 26435187 & 26843956 & 28700$

182

	m	n	R_e	d	R_c	Packed integer form of the parity check matrix
25	101		0.930241 (0.871017)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 3299599 3845689 4285667 5042513 7712994 7729011 7895282 7996354 8818602 8851898 9502328 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 15709933 16150229 16192222 19045699 19139598 19470307 21402133 21730182 22186199 22368942 22692960 22704308 22800504 23451790 23514363 24577428 25990493 26041256 26435187 26843956 26929259 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 32528240 33424376
25	102		0.933321 (0.873180)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 3299599 3845689 4285667 5042513 7712994 7729011 7895282 7996354 8084794 8818602 8851898 9502328 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 15709933 16150229 16192222 19045699 19139598 19470307 21402133 21730182 22186199 22368942 22692960 22704308 22800504 23451790 23514363 24577428 25990493 26041256 26435187 26843956 26929259 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 32528240 33424376
25	103		0.936282 (0.875225)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 3299599 3845689 4285667 5042513 7712994 7729011 7895282 7996354 8084794 8818602 8851898 9288975 9502328 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 15709933 16150229 16192222 19045699 19139598 19470307 21402133 21730182 22186199 22368942 22692960 22704308 22800504 23451790 23514363 24577428 25990493 26041256 26435187 26843956 26929259 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 32528240 33424376
25	104		0.939129 (0.877193)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 3299599 3845689 4285667 5042513 7712994 7729011 7895282 7996354 8084794 8818602 8851898 9288975 9502328 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 15709933 16150229 16192222 19045699 19139598 19470307 21402133 21730182 22186199 22368942 22692960 22704308 22800504 23451790 23514363 24577428 25990493 26041256 26435187 26843956 26929259 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 32528240 33424376
25	105		0.941869 (0.879070)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 3299599 3845689 4285667 5042513 7712994 7729011 7895282 7996354 8084794 8856224 8818602 8851898 9288975 9502328 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 15709933 16150229 16192222 19045699 19139598

183

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
25	110	0.954049 (0.887267)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 3299599 3845689 4285667 5042513 7712994 7729011 7895282 7996354 8084794 8556224 8818602 8851898 9288975 9502328 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 15709933 16150229 16192222 18018298 19045699 19139598 19470307 21402133 21730182 22186199 22368942 22692960 22704308 22800504 23451790 23514363 24577428 25990493 26041256 26435187 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31491176 32528240 33356143 33424376
25	111	0.956200 (0.888690)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 3299599 3845689 4285667 5042513 7712994 7729011 7895282 7996354 8084794 8556224 8818602 8851898 9288975 9502328 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 15279982 15709933 16150229 16192222 18018298 19045699 19139598 19470307 21402133 21730182 22186199 22368942 22692960 22704308 22800504 23451790 23514363 24577428 25990493 26041256 26435187 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31491176 32528240 33356143 33424376
25	112	0.958260 (0.890045)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 3299599 3845689 4285667 5042513 7712994 7729011 7895282 7996354 8084794 8556224 8818602 8851898 9288975 9502328 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 15279982 15709933 16150229 16192222 18018298 19045699 19139598 19470307 21402133 21730182 22186199 22368942 22450958 22692960 22704308 22800504 23451790 23514363 24577428 25990493 26041256 26435187 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31491176 32528240 33356143 33424376
25	113	0.960237 (0.891330)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 3299599 3845689 4285667 5042513 7712994 7729011 7895282 7996354 8084794 8556224 8818602 8851898 9288975 9502328 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 15279982 15709933 16150229 16192222 18018298 19045699 19139598 19470307 21402133 21730182 22186199 22368942 22450958 22692960 22704308 22800504 23451790 23514363 24577428 25990493 26041256 26435187 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31491176 32528240 33356143 33424376
25	114	0.962133 (0.892559)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 3299599 3845689 4285667 5042513 7712994 7729011 7895282 7996354 8084794 8556224 8818602 8851898 9288975 9502328 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 15279982 15709933 16150229 16192222 18018298 19045699 19139598 19470307 21402133 21730182 22186199 22368942 22450958 22692960 22704308 22800504 23451790 23514363 24577428 25990493 26041256 26435187 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31491176 32225533 32360454 32528240 33356143 33424376
25	115	0.963946 (0.893728)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 3299599 3845689 4285667 5042513 7712994 7729011 7895282 7996354 8084794 8556224 8818602 8851898 9107270 9288975 9502328 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 15279982 15709933 16150229 16192222 18018298 19045699 19139598 19470307 21402133 21730182 22186199 22368942 22450958 22692960 22704308 22800504 23451790 23514363 24577428 25990493 26041256 26435187 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31491176 32225533 32360454 32528240 33356143 33424376
25	116	0.965682 (0.894840)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 3299599 3845689 4285667 5042513 7712994 7729011 7895282 7996354 8084794 8556224 8818602 8851898 9107270 9288975 9502328 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 15279982 15709933 16150229 16192222 18018298 19045699 19139598 19470307 21132854 21402133 21730182 22186199 22368942 22450958 22692960 22704308 22800504 23451790 23514363 24577428 25990493 26041256 26435187 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31491176 32225533 32360454 32528240 33356143 33424376
25	117	0.967347 (0.895897)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 3299599 3845689 4285667 4748357 5042513 7712994 7729011 7895282 7996354 8084794 8556224 8818602 8851898 9107270 9288975 9502328 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 15279982 15709933 16150229 16192222 18018298 19045699 19139598 19470307 21132854 21402133 21730182 22186199 22368942 22450958 22692960 22704308 22800504 23451790 23514363 24577428 25990493 26041256 26435187 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31491176 32225533 32360454 32528240 33356143 33424376

Continued on next page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
25	118	0.968938 (0.896905)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 3299599 3845689 4285667 4748357 5042513 7712994 7729011 7895282 7996354 8084794 8556224 8818602 8851898 9107270 9288975 9502328 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 15279982 15709933 16150229 16192222 18018298 18122333 19045699 19139598 19470307 21132854 21402133 21730182 22186199 22368942 22450958 22692960 22704308 22800504 23451790 23514363 24577428 25990493 26041256 26435187 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31491176 32225533 32360454 32528240 33356143 33424376
25	119	0.970461 (0.897859)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 3299599 3845689 4285667 4748357 5042513 7712994 7729011 7895282 7996354 8084794 8556224 8818602 8851898 9107270 9288975 9502328 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 15279982 15709933 15768802 16150229 16192222 18018298 18122333 19045699 19139598 19470307 21132854 21402133 21730182 22186199 22368942 22450958 22692960 22704308 22800504 23451790 23514363 24577428 25990493 26041256 26435187 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31491176 32225533 32360454 32528240 33356143 33424376
25	120	0.971917 (0.898768)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 3299599 3845689 4285667 4748357 5042513 7712994 7729011 7895282 7996354 8084794 8556224 8818602 8851898 9107270 9288975 9502328 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 15279982 15709933 15768802 16150229 16192222 16241263 18018298 18122333 19045699 19139598 19470307 21132854 21402133 21730182 22186199 22368942 22450958 22692960 22704308 22800504 23451790 23514363 24577428 25990493 26041256 26435187 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31491176 32225533 32360454 32528240 33356143 33424376
25	121	0.973308 (0.899631)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 3299599 3845689 4285667 4748357 5042513 7712994 7729011 7895282 7996354 8084794 8556224 8818602 8851898 9107270 9288975 9502328 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 15279982 15709933 15768802 16150229 16192222 16241263 16421916 18018298 18122333 19045699 19139598 19470307 21132854 21402133 21730182 22186199 22368942 22450958 22692960 22704308 22800504 23451790 23514363 24577428 25990493 26041256 26435187 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31491176 32225533 32360454 32528240 33356143 33424376
25	122	0.974			

185

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
25	126	0.979385 (0.903327)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 112508 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 2468367 3299599 3845689 4285667 4748357 5042513 5894071 6333021 7712994 7729011 7895282 7996354 8084794 8556224 8818602 8851898 9107270 9288975 9502328 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 14732605 15279982 15709933 15768802 16150229 16192222 16241263 16421916 18018298 18122333 19045699 19139598 19470307 21132854 21402133 21730182 22186199 22368942 22450958 22692960 22704308 22800504 23451790 23514363 24577428 25990493 26041256 26435187 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31491176 32225533 32360454 32528240 33356143 33424376
25	127	0.980440 (0.903956)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 112508 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 2468367 3299599 3845689 4285667 4748357 5042513 5894071 6333021 7712994 7729011 7895282 7996354 8084794 8556224 8818602 8851898 9107270 9288975 9502328 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 14732605 15279982 15709933 15768802 16150229 16192222 16241263 16421916 18018298 18122333 19045699 19139598 19470307 21132854 21402133 21730182 22186199 22368942 22450958 22692960 22704308 22800504 23451790 23514363 23932116 24577428 25990493 26041256 26435187 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31491176 32225533 32360454 32528240 33356143 33424376
25	128	0.981445 (0.904552)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 112508 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 2468367 3299599 3845689 4285667 4748357 4889653 5042513 5894071 6333021 7712994 7729011 7895282 7996354 8084794 8556224 8818602 8851898 9107270 9288975 9502328 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 14732605 15279982 15709933 15768802 16150229 16192222 16241263 16421916 18018298 18122333 19045699 19139598 19470307 21132854 21402133 21730182 22186199 22368942 22450958 22692960 22704308 22800504 23451790 23514363 23932116 24577428 25990493 26041256 26435187 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31491176 32225533 32360454 32528240 33356143 33424376
25	129	0.982403 (0.905116)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 112508 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 2468367 3299599 3845689 4285667 4748357 4889653 5042513 5894071 6333021 7712994 7729011 7895282 7996354 8084794 8556224 8818602 8851898 9107270 9288975 9502328 9668085 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 14732605 15279982 15709933 15768802 16150229 16192222 16241263 16421916 18018298 18122333 19045699 19139598 19470307 21132854 21402133 21730182 22186199 22368942 22450958 22692960 22704308 22800504 23451790 23514363 23932116 24577428 25990493 26041256 26435187 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31491176 32225533 32360454 32528240 33356143 33424376
25	130	0.983317 (0.916457)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 112508 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 2468367 3299599 3845689 4285667 4748357 4889653 5042513 5894071 6333021 7712994 7729011 7895282 7996354 8084794 8556224 8818602 8851898 9107270 9288975 9502328 9668085 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 14732605 15279982 15709933 15768802 16150229 16192222 16241263 16421916 18018298 18122333 19045699 19139598 19470307 21132854 21402133 21730182 22186199 22368942 22450958 22692960 22704308 22800504 23208288 23451790 23514363 23932116 24577428 25990493 26041256 26435187 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31491176 32225533 32360454 32528240 33356143 33424376
25	131	0.984188 (0.920398)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 112508 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 2468367 3299599 3845689 4285667 4748357 4889653 5042513 5894071 6333021 7712994 7712994 7729011 7895282 7996354 8084794 8556224 8818602 8851898 9107270 9288975 9502328 9668085 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 14732605 15279982 15709933 15768802 16150229 16192222 16241263 16421916 18018298 18122333 19045699 19139598 19470307 21132854 21402133 21730182 22186199 22368942 22450958 22692960 22704308 22800504 23208288 23451790 23514363 23932116 24577428 25990493 26041256 26435187 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31491176 32225533 32360454 32528240 33356143 33424376
25	132	0.985017 (0.923528)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 112508 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 2468367 3299599 3845689 4285667 4748357 4889653 5042513 5894071 6333021 7712994 7712994 7729011 7895282 7996354 8084794 8556224 8818602 8851898 9107270 9288975 9502328 9668085 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 14732605 15279982 15709933 15768802 16150229 16192222 16241263 16421916 18018298 18122333 19045699 19139598 19470307 21132854 21402133 21730182 22186199 22368942 22450958 22692960 22704308 22800504 23208288 23451790 23514363 23932116 24577428 25990493 26041256 26435187 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31491176 31866787 32225533 32360454 32528240 33356143 33424376

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
25	133	0.985807 (0.926059)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 112508 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 2468367 3299599 3845689 4285667 4748357 4889653 5042513 5894071 6333021 7128143 7712994 7729011 7895282 7996354 8084794 8556224 8818602 8851898 9107270 9288975 9502328 9668085 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 14732605 15279982 15709933 15768802 16150229 16192222 16241263 16421916 18018298 18122333 19045699 19139598 19470307 21132854 21402133 21730182 22186199 22368942 22450958 22692960 22704308 22800504 23208288 23451790 23514363 23932116 24577428 25990493 26041256 26435187 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31489432 31491176 31866787 32225533 32360454 32528240 33356143 33424376
25	134	0.986559 (0.928133)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 112508 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 2468367 3299599 3845689 4285667 4748357 4889653 5042513 5894071 6333021 7128143 7712994 7729011 7895282 7996354 8084794 8556224 8818602 8851898 9107270 9288975 9502328 9668085 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 14732605 15279982 15709933 15768802 16150229 16192222 16241263 16421916 18018298 18122333 18930238 19045699 19139598 19470307 21132854 21402133 21730182 22186199 22368942 22450958 22692960 22704308 22800504 23208288 23451790 23514363 23932116 24577428 25990493 26041256 26435187 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31489432 31491176 31866787 32225533 32360454 32528240 33356143 33424376
25	135	0.987274 (0.929851)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 112508 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 2468367 3299599 3845689 4285667 4748357 4889653 5042513 5894071 6333021 7128143 7712994 7729011 7895282 7996354 8084794 8556224 8818602 8851898 9107270 9288975 9502328 9668085 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 14732605 15279982 15709933 15768802 16150229 16192222 16241263 16421916 17493832 18018298 18122333 18930238 19045699 19139598 19470307 21132854 21402133 21730182 22186199 22368942 22450958 22692960 22704308 22800504 23208288 23451790 23514363 23932116 24577428 25990493 26041256 26435187 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31489432 31491176 31866787 32225533 32360454 32528240 33356143 33424376
25	136	0.987955 (0.931290)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 112508 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 2468367 2634907 3299599 3845689 4285667 4748357 4889653 5042513 5894071 6333021 7128143 7712994 7729011 7895282 7996354 8084794 8556224 8818602 8851898 9107270 9288975 9502328 9668085 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 14732605 15279982 15709933 15768802 16150229 16192222 16241263 16421916 17493832 18018298 18122333 18930238 19045699 19139598 19470307 21132854 21402133 21730182 22186199 22368942 22450958 22692960 22704308 22800504 23208288 23451790 23514363 23932116 24577428 25990493 26041256 26435187 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31489432 31491176 31866787 32225533 32360454 32528240 33356143 33424376
25	137	0.988602 (0.931710)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 112508 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 2468367 2634907 3299599 3845689 4285667 4748357 4889653 5042513 5894071 6333021 7128143 7712994 7729011 7895282 7996354 8084794 8556224 8818602 8851898 9107270 9288975 9502328 9668085 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 13161042 14438887 14537918 14732605 15279982 15709933 15768802 16150229 16192222 16241263 16421916 17493832 18018298 18122333 18930238 19045699 19139598 19470307 21132854 21402133 21730182 22186199 22368942 22450958 22692960 22704308 22800504 23208288 23451790 23514363 23932116 24577428 25990493 26041256 26218590 26435187 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31489432 31491176 31866787 32225533 32360454 32528240 33356143 33424376
25	139	0.989802 (0.933284)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 112508 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 2468367 2634907 3299599 3845689 4285667 4748357 4889653 5042513 5894071 6333021 7128143 7712994 7729011 7895282 7996354 8084794 8556224 8720364 8818602 8851898 9107270 9288975 9502328 9668085 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 12877090 13161042 14438887 14537918 14732605 15279982 15709933 15768802 16150229 16192222 16241263 16421916 17493832 18018298 18122333 18930238 19045699 19139598 19470307 21132854 21402133 21730182 22186199 22368942 22450958 22692960 22704308 22800504 23208288 23451790 23514363 23932116 24577428 25990493 26041256 26218590 26435187 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31489432 31491176 31866787 32225533 32360454 32528240 33356143 33424376
25	140	0.990357 (0.934282)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 112508 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 2468367 2634907 3299599 3845689 4285667 4748357 4889653 5042513 5785320 5894071 6333021 7128143 7712994 7729011 7895282 7996354 8084794 8556224 8720364 8818602 8851898 9107270 9288975 9502328 9668085 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 12877090 13161042 14438887 14537918 14732605 15279982 15709933 15768802 16150229 16192222 16241263 16421916 17493832 18018298 18122333 18930238 19045699 19139598 19470307 21132854 21402133 21730182 22186199 22368942 22450958 22692960 22704308 22800504 23208288 23451790 23514363 23932116 24577428 25990493 26041256 26218590 26435187 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31489432 31491176 31866787 32225533 32360454 32528240 33356143 33424376

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
25	141	0.990885 (0.935134)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 112508 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 2468367 2634907 3299599 3845689 4285667 4748357 4889653 5042513 5785320 5894071 6333021 7128143 7712994 7729011 7895282 7996354 8084794 8556224 8720364 8818602 8851898 9107270 9288975 9502328 9668085 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 12877090 13161042 14438887 14537918 14732605 15279982 15709933 15768802 16150229 16192222 16241263 16421916 17493832 18018298 18122333 18930238 19045699 19139598 19470307 21132854 21402133 21730182 22186199 22368942 22450958 22692960 22704308 22800504 23208288 23451790 23514363 23932116 24577428 25990493 26041256 26218590 26435187 26655657 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31489432 31491176 31866787 32225533 32360454 32528240 33356143 33424376
25	142	0.991386 (0.935869)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 112508 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 2468367 2634907 3299599 3845689 4285667 4748357 4889653 5042513 5785320 5894071 6333021 7128143 7275684 7712994 7729011 7895282 7996354 8084794 8556224 8720364 8818602 8851898 9107270 9288975 9502328 9668085 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 12877090 13161042 14438887 14537918 14732605 15279982 15709933 15768802 16150229 16192222 16241263 16421916 17493832 18018298 18122333 18930238 19045699 19139598 19470307 21132854 21402133 21730182 22186199 22368942 22450958 22692960 22704308 22800504 23208288 23451790 23514363 23932116 24577428 25990493 26041256 26218590 26435187 26655657 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31489432 31491176 31866787 32225533 32360454 32528240 33356143 33424376
25	143	0.991861 (0.936506)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 112508 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 2468367 2634907 3299599 3845689 4285667 4748357 4889653 5042513 5785320 5894071 6333021 7128143 7275684 7712994 7729011 7895282 7996354 8084794 8556224 8720364 8818602 8851898 9107270 9288975 9502328 9668085 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 12877090 13161042 14438887 14537918 14732605 15279982 15709933 15768802 16150229 16192222 16241263 16421916 17103359 17493832 18018298 18122333 18930238 19045699 19139598 19470307 21132854 21402133 21730182 22186199 22368942 22450958 22692960 22704308 22800504 23208288 23451790 23514363 23932116 24577428 25990493 26041256 26218590 26435187 26655657 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31489432 31491176 31866787 32225533 32360454 32528240 33356143 33424376
25	144	0.992312 (0.937063)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 112508 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 2468367 2634907 3299599 3845689 4285667 4748357 4889653 5042513 5785320 5894071 6333021 7128143 7275684 7712994 7729011 7895282 7996354 8084794 8556224 8720364 8818602 8851898 9107270 9288975 9502328 9668085 9936232 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 12700471 12807199 12877090 13161042 14438887 14537918 14732605 15279982 15709933 15768802 16150229 16192222 16241263 16421916 17103359 17493832 18018298 18122333 18930238 19045699 19139598 19470307 21132854 21402133 21730182 22186199 22368942 22450958 22692960 22704308 22800504 23208288 23451790 23514363 23932116 24577428 25990493 26041256 26218590 26435187 26655657 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31489432 31491176 31866787 32225533 32360454 32528240 33356143 33424376
25	145	0.992740 (0.937553)	7	5	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 112508 879021 1153115 1162144 1168348 1200870 1203837 1294333 1802840 1807215 1812374 1912776 1939932 2232798 2286058 2337225 2468367 2634907 3299599 3845689 4285667 4748357 4889653 5042513 5785320 5894071 6333021 7128143 7275684 7712994 7729011 7895282 7996354 8084794 8556224 8720364 8818602 8851898 9107270 9288975 9502328 9668085 9936232 10198212 10341228 10342507 10348782 10349577 10369181 10419908 10517036 10746226 11148849 12700471 12807199 12877090 13161042 14438887 14537918 14732605 15279982 15709933 15768802 16150229 16192222 16241263 16421916 17103359 17493832 18018298 18122333 18930238 19045699 19139598 19470307 21132854 21402133 21730182 22186199 22368942 22450958 22692960 22704308 22800504 23208288 23451790 23514363 23932116 24577428 25990493 26041256 26218590 26435187 26655657 26843956 26929259 27084511 28700830 29130072 29370079 29474042 29480772 29552947 29642880 29668973 29848995 30064109 30355192 30597213 30681457 30688342 31115935 31489432 31491176 31866787 32225533 32360454 32528240 33356143 33424376
26	27	0.297412	27	13	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 67108863
26	28	0.308427	18	13	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 66978047 67108608
26	29	0.319443	16	14	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 20680495 66848752 67106816
26	30	0.330458	16	15	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 21814487 63162254 66848752 67106816
26	31	0.341473	16	15	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 13986491 21814487 63162254 66848752 67106816
26	32	0.352487	16	12	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 4763391 9795415 20683669 36076988 63162254 66848752
26	33	0.363499 (0.363498)	14	13	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 19876123 20027038 40054076 41695044 43567695 49580465 51997156
26	34	0.374511	14	13	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 12999289 19876123 20027038 25998578 40054076 43567695 49580465 51997156
26	35	0.385521	14	12	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 12999289 19876123 20027038 25998578 40054076 43567695 49580465 51997156 67108352

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
26	36	0.396527 (0.396524)	13	11	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 12999289 19876123 20027038 25998578 40054076 43567695 48867014 49580465 51997156 67108352
26	37	0.407528	13	12	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 7236829 11782005 12432118 23581418 27408781 32784947 36004302 38270809 47250131 56582919 63577961
26	38	0.418526	13	10	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 7236829 11782005 12432118 23581418 27408781 32784947 36004302 38270809 44658299 47250131 56582919 63577961
26	39	0.429436 (0.429383)	12	11	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 28109796 30147828 32518564 36922532 41667151 46890100 47539932 53139188 56023708 56219592 60295656 61214684 65037128
26	40	0.440439 (0.440321)	9	10	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 3593725 5170491 7187450 13016450 17835914 25281241 30829808 39521969 47041821 48558718 52038685 52707248 63534374 65931195
26	41	0.451319 (0.451234)	12	11	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 28109796 30147828 32518564 36922532 41669715 46890100 47539932 48566612 53139188 54651916 56023708 56219592 60295656 61214684 65037128
26	42	0.462277 (0.462122)	12	11	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 28109796 30147828 32518564 36922532 41669715 46890100 47539932 48566612 53139188 54651916 56023708 56219592 60295656 61214684 65037128
26	43	0.473218 (0.472978)	12	10	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 28109796 30147828 32518564 33601333 36922532 41669715 46890100 47539932 48566612 53139188 54651916 56023708 56219592 56401527 60295656 61214684 65037128
26	44	0.483958 (0.483797)	12	11	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 28109796 30001488 30147828 32518564 36922532 41680539 46890100 47539932 48566612 53139188 54651916 56023708 56219592 60002976 60295656 61214684 65037128
26	45	0.494861 (0.494575)	12	11	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 15000744 28109796 30001488 30147828 32518564 36922532 41680539 46890100 47539932 48566612 53139188 54651916 56023708 56219592 60002976 60295656 61214684 65037128
26	46	0.505647 (0.505304)	12	10	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 28109796 30001488 30147828 32518564 36922532 41680539 46890100 47539932 48566612 53139188 54651916 56023708 56219592 60002976 60295656 61214684 65037128
26	47	0.516261 (0.515979)	12	11	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 7500372 15000744 28109796 30001488 30147828 32518564 34233740 36922532 42194971 46890100 47539932 48566612 53139188 53889632 54651916 56023708 56219592 60002976 60295656 61214684 65037128
26	48	0.527098 (0.526595)	12	11	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 7500372 15000744 28109796 30001488 30147828 32518564 34233740 36922532 42194971 46890100 47539932 48566612 53139188 53889632 54651916 56023708 56219592 60002976 60295656 61214684 64862406 65037128
26	49	0.537882 (0.537145)	12	11	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 7500372 15000744 28109796 30001488 30147828 32518564 34233740 36922532 42194971 46890100 47539932 48566612 53139188 53889632 54651916 56023708 56219592 58602597 60002976 60295656 61214684 64862406 65037128
26	50	0.548597 (0.547622)	12	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 7500372 15000744 28109796 30001488 30147828 32518564 34233740 36209615 36922532 42194971 46890100 47539932 48566612 53139188 53889632 54651916 56023708 56219592 58602597 60002976 60295656 61214684 64862406 65037128
26	51	0.558521 (0.557677)	11	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 6736204 7500372 13472408 15000744 26944816 28109796 30001488 30147828 32518564 34233740 36922532 41666587 46890100 47539932 48566612 53139188 53889632 54651916 56023708 56219592 60002976 60295656 61214684 64862404 65037128
26	52	0.569166 (0.567357)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 6736204 7500372 13472408 15000744 26944816 28109796 30001488 30147828 32518564 34233740 36922532 41666587 46890100 47539932 48566612 53139188 53889632 54651916 55995609 56023708 56219592 60002976 60295656 61214684 64862404 65037128
26	53	0.579752 (0.577669)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 6736204 7500372 13472408 15000744 26944816 28109796 30001488 30147828 32518564 34233740 35995418 36922532 41666587 46890100 47539932 48566612 53139188 53889632 54651916 55995609 56023708 56219592 60002976 60295656 61214684 64862404 65037128
26	54	0.590245 (0.587806)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 6736204 7500372 13472408 15000744 26944816 28109796 30001488 30147828 32518564 34233740 35995418 36922532 41666587 46890100 47539932 48566612 53139188 53889632 54651916 55995609 56023708 56219592 56510387 60002976 60295656 61214684 64862404 65037128
26	55	0.600574 (0.597819)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 6736204 7500372 13472408 15000744 26944816 28109796 29635563 30001488 30147828 32518564 34233740 35995418 36922532 41666587 46890100 47539932 48566612 53139188 53889632 54651916 55995609 56023708 56219592 56510387 60002976 60295656 61214684 64862404 65037128
26	56	0.610820 (0.609299)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 6736204 7500372 13472408 15000744 26944816 28109796 29635563 30001488 30147828 32518564 34233740 35995418 36922532 41666587 46890100 47539932 48566612 50308326 53139188 53889632 54651916 55995609 56023708 56219592 56510387 60002976 60295656 61214684 64862404 65037128

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
26	57	0.620964 (0.619187)	10	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 6736204 7500372 13472408 15000744 26944816 28109796 29635563 30001488 30147828 32518564 34233740 35995418 36922532 41666587 46890100 47539932 48566612 50380326 53139188 53889632 54651916 55995609 56023708 56219592 56510387 56672377 60002976 60295656 61214684 64862404 65037128
26	58	0.629970 (0.628970)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 5920230 10065590 11840460 14468230 20131180 22007002 23029182 23125594 23680920 23718866 23796474 28386122 28936460 32736986 34093210 36344194 38498346 40262360 41668157 44014004 46058364 46251188 46786566 47361840 47437732 47592948 50507482 54780620 56014239 56772244 57872920 65473972
26	59	0.640070 (0.638638)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 5920230 10065590 11840460 14468230 20131180 22007002 23029182 23125594 23680920 23718866 23796474 28386122 28936460 32736986 34093210 36060197 36344194 38498346 40262360 41668157 44014004 46058364 46251188 46786566 47361840 47437732 47592948 50507482 54780620 56014239 56772244 57872920 65473972
26	60	0.650289 (0.648174)	7	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1890247 2913921 3780494 7560988 8228375 9333731 11578600 12550313 13196818 13740983 14406489 14910031 15121976 20659095 21070098 21316674 26790587 27995338 29927392 33084050 36520625 37417067 42140196 43515314 45651974 48373092 51749307 53041350 59679862 60495998 62056274 65625617 65809466 65949330
26	61	0.659739 (0.657577)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 5920230 10065590 11840460 14468230 20131180 22007002 23029182 23125594 23680920 23718866 23796474 28386122 28936460 32736986 34093210 36060197 36344194 38498346 40262360 41668157 44014004 46058364 46251188 46786566 47361840 47437732 47592948 50507482 54588295 54780620 56014239 56772244 57872920 58293729 65473972
26	62	0.669327 (0.666851)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 54001 5920230 10065590 11840460 14468230 20131180 22007002 23029182 23125594 23680920 23718866 23796474 28386122 28936460 32736986 34093210 36060197 36344194 38498346 40262360 41668157 44014004 46058364 46251188 46786566 47361840 47437732 47592948 50507482 54588295 54780620 56014239 56772244 57872920 58293729 65473972
26	63	0.678759 (0.675987)	10	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 54001 5920230 10065590 11840460 14468230 18382699 20131180 22007002 23029182 23125594 23680920 23718866 23796474 28386122 28936460 32736986 34093210 36060197 36344194 38498346 40262360 41668157 44014004 46058364 46251188 46786566 47361840 47437732 47592948 50507482 54588295 54780620 56014239 56772244 57872920 58293729 65473972
26	64	0.685725 (0.684976)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 5920230 10065590 11840460 14468230 20131180 22007002 23029182 23125594 23680920 23718866 23796474 26831716 27166198 27390310 28386122 28936460 32736986 34093210 36344194 38498346 40262360 41674745 44014004 44636194 45523562 46058364 46251188 46786566 47361840 47437732 47592948 50507482 53663432 54332396 54780620 56772244 57872920 65473972
26	65	0.695512 (0.693819)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 5920230 10065590 11840460 14468230 20131180 22007002 23029182 23125594 23680920 23718866 23796474 26831716 27166198 27390310 28386122 28936460 32736986 34093210 36344194 38498346 40262360 41674745 44014004 44636194 45523562 46058364 46251188 46786566 47361840 47437732 47592948 50507482 53663432 54332396 54780620 56471213 56772244 57872920 65473972
26	66	0.704950 (0.704280)	10	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 5920230 10065590 11840460 14468230 20131180 22007002 23029182 23125594 23680920 23718866 23796474 26831716 27166198 27390310 28386122 28936460 32736986 34093210 36344194 38498346 40262360 41674745 44014004 44636194 45523562 46058364 46251188 46786566 47361840 47437732 47592948 50507482 53663432 54332396 54780620 56471213 56772244 57872920 62582069 65473972
26	67	0.713612 (0.713589)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 5920230 10065590 11840460 13415858 14468230 20131180 20858617 22007002 23029182 23125594 23680920 23718866 23796474 26831716 27166198 27390310 28386122 28936460 32736986 34093210 36344194 38498346 40262360 41848895 44014004 44475543 44636194 45523562 46058364 46251188 46786566 47361840 47437732 47592948 50507482 53663432 54332396 54780620 56772244 57872920 65473972
26	68	0.723086 (0.722614)	6	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 343005 2075880 4151760 5418443 7224008 7506315 8303520 10236408 10836886 14448016 15012630 16852789 20472816 21580137 21673772 23703085 31103675 34093521 34507379 34741143 38781914 39001261 41158762 43160274 43347544 43722410 44718530 45477317 46568817 47406170 50276651 52228105 52810689 52999301 54052647 57749198 60751485 61292691 62207350 62563659 63465543 64018419
26	69	0.731380 (0.731390)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 5920230 10065590 11840460 13415858 14468230 20131180 20858617 21098123 22007002 23029182 23125594 23680920 23718866 23796474 26831716 27166198 27390310 28386122 28936460 32736986 34093210 36344194 38498346 38776405 40262360 41848895 44014004 44475543 44636194 45523562 46058364 46251188 46786566 47361840 47437732 47592948 50507482 53663432 54332396 54780620 56772244 57872920 65473972
26	70	0.740148 (0.739907)	7	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 991772 1983544 3018459 3213607 6218577 6427214 6461437 6631650 7574433 11845948 12922874 15148866 16650269 16948645 19188899 19894431 21019620 23691896 26459924 26641502 30297732 30359117 33245179 33300538 35043181 43554600 46722702 48125686 48468229 49322059 50273205 52090011 52366162 53362860 55498090 58275772 58450099 61636468 61804158 62490851 62662397 64217937 64416832 64530712
26	71	0.748277 (0.748271)	7	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 4101443 5685692 6570187 11163677 11214098 11325862 13411599 14249811 22327354 22428196 22651724 25878594 26577896 26823198 26997833 28237800 28499622 28733117 29101996 30203193 33874701 37576136 39969251 41957393 42655039 44856392 45303448 47467251 49881143 51655593 51757188 53155792 53995666 56475600 56997208 57439042 57466234 58174093 58203992 60406386 60488333 61015049 61240990 65319551 65492005

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
26	72	0.756438 (0.756436)	10	9	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1217711 5490489 5920230 10065590 11840460 13415858 14468230 20131180 20858617 22007002 23029182 23125594 23680920 23718866 23796474 26361011 26831716 27166198 27390310 27845705 28386122 28936460 32736986 34093210 36344194 38498346 40262360 44014004 44475543 44636194 45523562 46058364 46251188 46786566 47361840 47437732 47592948 48485407 50507482 53663432 54332396 54780620 56772244 57872920 64592813 65473972
26	73	0.764293	10	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1217711 5490489 5920230 10065590 11840460 13415858 14468230 20131180 20858617 22007002 23029182 23125594 23680920 23718866 23796474 26361011 26831716 27166198 27390310 27845705 28386122 28936460 32736986 34093210 36344194 38498346 40262360 44014004 44475543 44636194 45523562 46058364 46251188 46786566 47361840 47437732 47592948 48485407 50507482 53663432 54332396 54780620 56772244 57872920 62624622 64592813 65473972
26	74	0.772105 (0.767562)	9	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1217711 5920230 10065590 11840460 13415858 14468230 20131180 20858617 22007002 23029182 23125594 23680920 23718866 23796474 26361011 26831716 27166198 27390310 27845705 28386122 28936460 32736986 34093210 36344194 38498346 40262360 41743267 43800121 44014004 44475543 44636194 45523562 46058364 46251188 46786566 47361840 47437732 47592948 48485407 50507482 53663432 54332396 54780620 56772244 57872920 62624622 64592813 65473972
26	75	0.779727 (0.776372)	9	8	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1217711 5920230 10065590 11840460 13415858 14468230 20131180 20858617 22007002 23029182 23125594 23680920 23718866 23796474 26361011 26831716 27166198 27390310 27845705 28386122 28936460 32736986 34093210 36344194 38498346 40262360 41743267 43800121 44014004 44475543 44636194 45523562 46058364 46251188 46786566 47361840 47437732 47592948 48485407 50507482 53663432 54332396 54780620 56772244 57872920 62624622 64592813 65473972
26	76	0.787107 (0.784656)	9	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1217711 5920230 10065590 11840460 11965872 13415858 14468230 20131180 20858617 22007002 23029182 23125594 23680920 23718866 23796474 26361011 26831716 27166198 27390310 27845705 28386122 28936460 32736986 34093210 36344194 38498346 40262360 41743267 43800121 44014004 44475543 44636194 45523562 46058364 46251188 46786566 47361840 47437732 47592948 48485407 50507482 53663432 54332396 54780620 56772244 57872920 62624622 64592813 65473972
26	77	0.794042 (0.792519)	7	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 827471 1572509 1654942 2348749 2504041 2546558 2864117 3145018 5308487 5728234 5943045 6453729 9478839 12907458 13508415 14202110 14355618 18761390 18957678 22537583 23770841 24880635 26013197 26563294 28064155 28231753 28716467 29065946 31120738 34986490 37260920 37748416 39684257 42876830 44073622 45140131 46017357 46384439 47107875 52897872 53208842 54311310 54879670 56795752 58051178 62613176 63197560 63278422 63851038 64364016 64858609
26	80	0.814528 (0.745574)	6	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 70829 141658 283316 1492620 2132566 4693497 5010707 5647973 6629488 7725897 9126989 9234737 9386994 94076733 11295946 13651790 14531105 14841687 18469474 18773980 19052641 2021273 21268722 22087761 22949362 23695762 27303580 30159562 30236523 31263484 32411375 32756010 34118683 37308511 38584583 41175712 41615788 44094008 44175522 44837124 46249100 47878881 48568346 48772513 49136084 49610380 50640494 55086333 60319124 60473046 60557683 60605256 62526968 65934225
26	81	0.820959 (0.749741)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 11840460 12139956 14534737 17641732 17992524 18692090 20131180 20858618 22562674 23680920 24279912 26280684 26831716 27845706 28552180 28936460 32564036 34048695 34254884 35283464 35986053 36286430 38278580 40262360 41667867 43800122 44014004 44811375 46058364 46251188 47221140 47347900 47361840 47437732 47592948 47926580 48559824 49569716 52561368 53663432 54332396 54780620 55996779 56474461 56772244 57104360 57872920 58140380 58689469 61409218 62624620 64592814 65128072 65473972
26	82	0.827479 (0.753928)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 11840460 12139956 14534737 17641732 17992524 18692090 20131180 20858618 22562674 23680920 24279912 25579531 26280684 26831716 27845706 28552180 28936460 32564036 34048695 34254884 35283464 35985048 35986053 36286430 38278580 40262360 41667867 43800122 44014004 44811375 46058364 46251188 47221140 47347900 47361840 47437732 47592948 47926580 48559824 49569716 52561368 53663432 54332396 54780620 55996779 56474461 56772244 57104360 57872920 58140380 58689469 61409218 62624620 64592814 65128072 65473972
26	83	0.833795 (0.757673)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 11840460 12139956 14534737 17641732 17992524 18692090 20131180 20858618 22460065 22562674 23680920 24279912 25579531 26280684 26831716 27845706 28552180 28936460 32564036 34048695 34254884 35283464 35985048 35986053 36286430 38278580 40262360 41667867 43800122 44014004 44811375 46058364 46251188 47221140 47347900 47361840 47437732 47592948 47926580 48559824 49569716 52561368 53663432 54332396 54780620 55996779 56474461 56772244 57104360 57872920 58140380 58689469 61409218 62624620 64592814 65128072 65473972
26	84	0.839919 (0.761527)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 11840460 12139956 14534737 17641732 17992524 18692090 20131180 20858618 22460065 22562674 23680920 24279912 25579531 26280684 26831716 27845706 28552180 28936460 32564036 34048695 34254884 35283464 35985048 35986053 36286430 38278580 40262360 41667867 43800122 44014004 44811375 46058364 46251188 47221140 47347900 47361840 47437732 47592948 47926580 48559824 49569716 52561368 53663432 54332396 54780620 55996779 56474461 56772244 57104360 57872920 58140380 58689469 61409218 62624620 63555923 64592814 65128072 65473972
26	85	0.845855 (0.765171)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 9395871 11840460 12139956 14534737 17641732 17992524 18692090 20131180 20858618 22460065 22562674 23680920 24279912 25579531 26280684 26831716 27845706 28552180 28936460 32564036 34048695 34254884 35283464 35985048 35986053 36286430 38278580 40262360 41667867 43800122 44014004 44811375 46058364 46251188 47221140 47347900 47361840 47437732 47592948 47926580 48559824 49569716 52561368 53663432 54332396 54780620 55996779 56474461 56772244 57104360 57872920 58140380 58689469 61409218 62624620 63555923 64592814 65128072 65473972

Continued on next page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
26	86	0.851626 (0.768637)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 9395871 11068401 11840460 12139956 14534737 17641731 17992524 18692090 20131180 20858618 22460065 22562674 23680920 24279912 25579531 26280684 26831716 27845706 28552180 28936460 32564036 34048695 34254884 35283464 35985048 35986053 36286430 38278580 40262360 41667867 43800122 44014004 44811375 46058364 46251188 47221140 47347900 47361840 47437732 47592948 47926580 48559824 49569716 52561368 53663432 54332396 54780620 55996779 56474461 56772244 57104360 57872920 58140380 58689469 61409218 62624620 63555923 64592814 65128072 65473972
26	87	0.857214 (0.771984)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 9395871 11068401 11840460 12139956 14534737 14816701 17641732 17992524 18692090 20131180 20858618 22460065 22562674 23680920 24279912 25579531 26280684 26831716 27845706 28552180 28936460 32564036 34048695 34254884 35283464 35985048 35986053 36286430 38278580 40262360 41667867 43800122 44014004 44811375 46058364 46251188 47221140 47347900 47361840 47437732 47592948 47926580 48559824 49569716 52561368 53663432 54332396 54780620 55996779 56474461 56772244 57104360 57872920 58140380 58689469 61409218 62624620 63555923 64592814 65128072 65473972
26	88	0.862634 (0.775131)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 9395871 11068401 11840460 12139956 14534737 14816701 17641732 17992524 18692090 20131180 20858618 22460065 22562674 23680920 24279912 25579531 26280684 26831716 27845706 28552180 28936460 32564036 34048695 34254884 35283464 35985048 35986053 36286430 38278580 40262360 41667867 43800122 44014004 44811375 46058364 46251188 47221140 47347900 47361840 47437732 47592948 47926580 48559824 49569716 52561368 53663432 54332396 54780620 55996779 56474461 56772244 57104360 57872920 58140380 58689469 60935573 61409218 62624620 63555923 64592814 65128072 65473972
26	89	0.867894 (0.778078)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 9395871 11068401 11840460 12139956 14534737 14816701 17641732 17992524 18692090 20131180 20858618 22460065 22562674 23680920 24279912 25579531 26280684 26831716 27845706 28552180 28936460 32564036 34048695 34254884 35283464 35985048 35986053 36286430 38278580 40262360 41667867 43800122 44014004 44811375 46058364 46251188 47221140 47347900 47361840 47437732 47592948 47926580 48559824 49569716 52561368 53663432 54332396 54744803 54780620 55996779 56474461 56772244 57104360 57872920 58140380 58689469 60935573 61409218 62624620 63555923 64592814 65128072 65473972
26	90	0.872994 (0.785614)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 3419307 9395871 11068401 11840460 12139956 14534737 14816701 17641732 17992524 18692090 20131180 20858618 22460065 22562674 23680920 24279912 25579531 26280684 26831716 27845706 28552180 28936460 32564036 34048695 34254884 35283464 35985048 35986053 36286430 38278580 40262360 41667867 43800122 44014004 44811375 46058364 46251188 47221140 47347900 47361840 47437732 47592948 47926580 48559824 49569716 52561368 53663432 54332396 54744803 54780620 55996779 56474461 56772244 57104360 57872920 58140380 58689469 60935573 61409218 62624620 63555923 64592814 65128072 65473972
26	91	0.877934 (0.791829)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 3419307 9395871 11068401 11840460 12139956 14534737 14816701 17641732 17992524 18692090 201311

192

	m	n	R_e	d	R_c	Packed integer form of the parity check matrix
26	96		0.900399 (0.811638)	8	7	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 3419307 4492287 9395871 10093907 11068401 11840460 12139956 14534737 14816701 17641732 17992524 18692090 20131180 20858618 22460065 22562674 23680920 23791323 24279912 25579531 26280684 26831716 27845706 28552180 28936460 32564036 34048695 34254884 35283464 35985048 35986053 36286430 38278580 39088097 40262360 41667867 43800122 44014004 44811375 46058364 46251188 47221140 47347900 47389686 47437732 47592948 47926580 48559824 49569716 52561368 53663432 54332396 54744803 54780620 55996779 56474461 56772244 57104360 57872920 58140380 58689469 60935573 61409218 62624620 63555923 64592814 65128072 65473972
26	97		0.904463 (0.836756)	8	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 3419307 4492287 9395871 10093907 11068401 11840460 12139956 14534737 14816701 17641732 17992524 18692090 20131180 20858618 22460065 22562674 23680920 23791323 24279912 25579531 26280684 26831716 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 36286430 38278580 39088097 40262360 41667867 43800122 44014004 44811375 46058364 46251188 47221140 47347900 47389686 47437732 47592948 47926580 48559824 49569716 52561368 53663432 54332396 54744803 54780620 55996779 56474461 56772244 57104360 57872920 58140380 58689469 60935573 61409218 62624620 63555923 64592814 65128072 65473972
26	98		0.908379 (0.839310)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 3419307 4492287 9395871 10093907 11068401 11840460 12139956 14534737 14816701 17641732 17992524 18692090 20131180 20858618 22460065 22562674 23680920 23791323 24279912 25579531 26280684 26831716 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 36286430 38278580 39088097 40262360 41666801 41667867 43800122 44014004 44811375 46058364 46251188 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 52561368 53663432 54332396 54744803 54780620 55996779 56474461 56772244 57104360 57872920 58140380 58689469 60935573 61409218 62624620 63555923 64592814 65128072 65473972
26	99		0.912167 (0.841767)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 3419307 4492287 9395871 10093907 11068401 11840460 12139956 14534737 14816701 17641732 17992524 18692090 20131180 20858618 22460065 22562674 23680920 23791323 24279912 25579531 26280684 26831716 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 36286430 38278580 39088097 40262360 41666801 41667867 43800122 44014004 44811375 46058364 46251188 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 52561368 53663432 54332396 54744803 54780620 55989894 55996779 56474461 56772244 57104360 57872920 58140380 58689469 60935573 61409218 62624620 63555923 64592814 65128072 65473972
26	100		0.915824 (0.844119)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 3419307 4492287 9395871 10093907 11068401 11840460 12139956 14534737 14816701 17641732 17992524 18692090 20131180 20858618 22460065 22562674 23680920 23791323 24279912 25579531 26280684 26831716 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 41666801 41667867 43800122 44014004 44811375 46058364 46251188 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 52561368 53663432 54332396 54

193

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
26	105	0.932274 (0.854474)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 3419307 4492287 9353857 9395871 10093907 11068401 11840460 12139956 14534737 14816701 17641732 17992524 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23791323 24279912 25575249 25579531 26280684 26831716 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 41666801 41667867 43800122 44014004 44811375 46058364 46251188 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 52561368 53663432 54332396 54744803 54780620 55989894 55996779 56459293 56474461 56772244 57104360 57872920 58140380 58689469 60935573 61409218 62624620 63424582 63555923 64592814 65128072 65473972
26	106	0.935219 (0.856280)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 3419307 4492287 9353857 9395871 10093907 10934306 11068401 11840460 12139956 14534737 14816701 17641732 17992524 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23791323 24279912 25575249 25579531 26280684 26831716 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 41666801 41667867 43800122 44014004 44811375 46058364 46251188 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 52561368 53663432 54332396 54744803 54780620 55989894 55996779 56459293 56474461 56772244 57104360 57872920 58140380 58689469 60935573 61409218 62624620 63424582 63555923 64592814 65128072 65473972
26	107	0.938057 (0.858013)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 3419307 4492287 9353857 9395871 10093907 10934306 11068401 11840460 12139956 14134723 14534737 14816701 17641732 17992524 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23791323 24279912 25575249 25579531 26280684 26831716 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 41666801 41667867 43800122 44014004 44811375 46058364 46251188 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 52561368 53663432 54332396 54744803 54780620 55989894 55996779 56459293 56474461 56772244 57104360 57872920 58140380 58689469 60935573 61409218 62624620 63424582 63555923 64592814 65128072 65473972
26	108	0.940785 (0.859651)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 3419307 4492287 9353857 9395871 10093907 10934306 11068401 11840460 12139956 14134723 14534737 14816701 17641732 17992524 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23791323 24279912 25575249 25579531 26280684 26831716 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 41666801 41667867 43800122 44014004 44811375 46058364 46251188 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 52561368 53663432 54332396 54744803 54780620 55989894 55996779 56459293 56474461 56772244 57104360 57872920 58140380 58689469 60399235 60935573 61409218 62624620 63424582 63555923 64592814 65128072 65473972
26	109	0.943411 (0.861219)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 3419307 4492287 9353857 9395871 10093907 10934306 11068401 11840460 12139956 14134723 14534737 14816701 17641732 17992524 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23791323 24279912 25575249 25579531 26280684 26831716 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 41666801 41667867 43800122 44014004 44811375 46058364 46251188 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 51804442 52561368 53663432 54332396 54744803 54780620 55989894 55996779 56459293 56474461 56772244 57104360 57872920 58140380 58689469 60399235 60935573 61409218 62624620 63424582 63555923 64592814 65128072 65473972
26	110	0.945938 (0.862726)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 3419307 4492287 9353857 9395871 10093907 10934306 11068401 11840460 12139956 14134723 14534737 14816701 17641732 17992524 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23791323 24279912 25575249 25579531 26280684 26831716 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 41666801 41667867 43800122 44014004 44811375 46058364 46251188 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 51804442 52561368 53663432 54332396 54744803 54780620 55989894 55996779 56459293 56474461 56772244 57104360 57872920 58140380 58689469 60399235 60935573 61409218 62624620 63424582 63555923 64592814 65128072 65473972
26	111	0.948369 (0.864160)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 3419307 4492287 9353857 9395871 10093907 10934306 11068401 11840460 12139956 14134723 14534737 14816701 17641732 17992524 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23791323 24279912 25575249 25579531 26280684 26831716 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 41666801 41667867 43800122 44014004 44811375 46058364 46251188 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 51804442 52561368 53663432 54332396 54744803 54780620 55989894 55996779 56459293 56474461 56772244 57104360 57872920 58140380 58689469 60399235 60935573 61409218 62624620 63424582 63555923 64592814 65128072 65473972
26	112	0.950706 (0.865527)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 9353857 9395871 10093907 10934306 11068401 11840460 12139956 14134723 14534737 14816701 17641732 17992524 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23791323 24279912 25575249 25579531 26280684 26831716 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 40620697 41666801 41667867 43800122 44014004 44811375 46058364 46251188 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 51804442 52561368 53663432 54332396 54744803 54780620 55989894 55996779 56459293 56474461 56772244 57104360 57872920 58140380 58689469 60399235 60935573 61409218 62624620 63424582 63555923 64592814 65128072 65473972

Continued on next page

Table B.1 – Continued from previous page						
m	n	R_e	d	R_c	Packed integer form of the parity check matrix	
26	113	0.952949 (0.866821)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 9353857 9395871 10093907 10934306 11068401 11840460 12139956 14134723 14534737 14816701 17641732 17992524 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23791323 24279912 25575249 25579531 26280684 26831716 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 40620697 41666801 41667867 42880205 43800122 44014004 44811375 46058364 46251188 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 51804442 52561368 53663432 54332396 54744803 54780620 55989894 55996779 56459293 56474461 56772244 57104360 57872920 58140380 58689469 60399235 60935573 61409218 62624620 63424582 63555923 64592814 65128072 65473972	
26	114	0.955103 (0.868062)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 9353857 9395871 10093907 10934306 11068401 11840460 12139956 14134723 14534737 14816701 17641732 17992524 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23791323 24279912 25575249 25579531 26280684 26831716 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 40620697 41666801 41667867 42880205 43800122 44014004 44811375 46058364 46251188 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 51804442 52561368 53663432 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57872920 58140380 58689469 60399235 60935573 61409218 62624620 63424582 63555923 64592814 65128072 65473972	
26	115	0.957173 (0.869248)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 9353857 9395871 10093907 10934306 11068401 11840460 12139956 14134723 14534737 14816701 16300368 17641732 17992524 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23791323 24279912 25575249 25579531 26280684 26831716 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 40620697 41666801 41667867 42880205 43800122 44014004 44811375 46058364 46251188 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 51804442 52561368 53663432 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57872920 58140380 58689469 60399235 60935573 61409218 62624620 63424582 63555923 64592814 65128072 65473972	
26	116	0.959159 (0.870367)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 9353857 9395871 10093907 10934306 11068401 11840460 12139956 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23791323 24279912 25575249 25579531 26280684 26831716 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 40620697 41666801 41667867 42880205 43800122 44014004 44811375 46058364 46251188 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 51804442 52561368 53663432 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57872920 58140380 58689469 60399235 60935573 61409218 62624620 63424582 63555923 64592814 65128072 65473972	
26	117	0.961063 (0.871440)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 9353857 9395871 10093907 10934306 11068401 11840460 12139956 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23791323 24279912 25575249 25579531 26280684 26831716 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 40620697 41666801 41667867 42880205 43800122 44014004 44811375 46058364 46251188 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 51804442 52561368 53663432 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57872920 58140380 58689469 59556412 60399235 60935573 61409218 62624620 63424582 63555923 64592814 65128072 65473972	
26	118	0.962887 (0.872455)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 9353857 9395871 10093907 10934306 11068401 11840460 12139956 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23791323 24279912 25575249 25579531 26280684 26831716 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 40620697 41666801 41667867 42880205 43800122 44014004 44811375 46058364 46236459 46251188 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 51804442 52561368 53663432 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57872920 58140380 58689469 59556412 60399235 60935573 61409218 62624620 63424582 63555923 64592814 65128072 65473972	
26	119	0.964641 (0.873421)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 10093907 10934306 11068401 11840460 12139956 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23791323 24279912 25575249 25579531 26280684 26831716 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 40620697 41666801 41667867 42880205 43800122 44014004 44811375 46058364 46236459 46251188 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 51804442 52561368 53663432 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57872920 58140380 58689469 59556412 60399235 60935573 61409218 62624620 63424582 63555923 64592814 65128072 65473972	
26	120	0.966319 (0.874340)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 10093907 10934306 11068401 11840460 12139956 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23791323 24279912 25575249 25579531 26280684 26831716 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 40620697 41666801 41667867 42880205 43800122 44014004 44811375 46058364 46236459 46251188 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 50284074 51804442 52561368 53663432 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57872920 58140380 58689469 59556412 60399235 60935573 61409218 62624620 63424582 63555923 64592814 65128072 65473972	

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
26	121	0.967926 (0.875215)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 10093907 10934306 11068401 11840460 12139956 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23791323 24279912 25575249 25579531 26280684 26831716 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 40620697 41666801 41667867 42880205 43800122 44014004 44811375 45447850 46058364 46236459 46251188 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 50284074 51804442 52561368 53663432 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57872920 58140380 58689469 59556412 60399235 60935573 61409218 62624620 63424582 63555923 64592814 65128072 65473972
26	122	0.969465 (0.876045)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 10093907 10934306 11068401 11840460 12139956 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23791323 24279912 25575249 25579531 26280684 26831716 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 40620697 41666801 41667867 42880205 43800122 44014004 44811375 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 50284074 51804442 52561368 53663432 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57872920 58140380 58689469 59556412 60399235 60935573 61409218 62624620 63424582 63555923 64592814 65128072 65473972
26	123	0.970939 (0.876833)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 10093907 10351530 10934306 11068401 11840460 12139956 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23791323 24279912 25575249 25579531 26280684 26831716 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 40620697 41666801 41667867 42880205 43800122 44014004 44811375 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 50284074 51804442 52561368 53663432 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57872920 58140380 58689469 59556412 60399235 60935573 61409218 62624620 63424582 63555923 64592814 65128072 65473972
26	124	0.972350 (0.877579)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 10093907 10351530 10934306 11068401 11840460 12139956 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23791323 24279912 25575249 25579531 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 40620697 41666801 41667867 42880205 43800122 44014004 44811375 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 50284074 51804442 52561368 53663432 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57872920 58140380 58689469 59556412 60399235 60935573 61409218 62624620 63424582 63555923 64592814 65128072 65473972
26	125	0.973699 (0.878290)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 10093907 10351530 10934306 11068401 11840460 12139956 13170394 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23791323 24279912 25575249 25579531 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 40620697 41666801 41667867 42880205 43800122 44014004 44811375 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 50284074 51804442 52561368 53663432 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57872920 58140380 58689469 59556412 60399235 60935573 61409218 62624620 63424582 63555923 64592814 65128072 65473972
26	126	0.974989 (0.878962)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 10093907 10351530 10934306 11068401 11840460 12139956 13170394 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23791323 24279912 25575249 25579531 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 40620697 41666801 41667867 42880205 43800122 44014004 44811375 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 50284074 51804442 52561368 53663432 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57264850 57872920 58140380 58689469 59556412 60399235 60935573 61409218 62624620 63424582 63555923 64592814 65128072 65473972
26	127	0.976223 (0.879600)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 10093907 10351530 10934306 11068401 11840460 12139956 13170394 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23791323 24279912 25575249 25579531 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 40620697 41666801 41667867 42880205 43800122 44014004 44811375 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 50284074 51804442 52561368 53663432 54235120 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57264850 57872920 58140380 58689469 59556412 60399235 60935573 61409218 62624620 63424582 63555923 64592814 65128072 65473972

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
26	128	0.977401 (0.880204)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 10093907 10351530 10934306 11068401 11840460 12139956 13170394 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 40620697 41666801 41667867 42880205 43800122 44014004 44811375 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 50284074 51804442 52561368 53663432 54235120 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57264850 57872920 58140380 58689469 59556412 60399235 60935573 61409218 62624620 63424582 63555923 64592814 65128072 65473972
26	129	0.978528 (0.880777)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 10093907 10351530 10934306 11068401 11840460 12139956 13170394 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 40620697 41666801 41667867 42880205 43800122 44014004 44811375 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 50284074 51804442 52561368 53663432 54235120 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57264850 57872920 58140380 58689469 59556412 60399235 60935573 61409218 62184461 62624620 63424582 63555923 64592814 65128072 65473972
26	130	0.979602 (0.891678)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 10093907 10351530 10934306 11068401 11840460 12139956 13170394 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18009315 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 40620697 41666801 41667867 42880205 43800122 44014004 44811375 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 50284074 51804442 52561368 53663432 54235120 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57264850 57872920 58140380 58689469 59556412 60399235 60935573 61409218 62184461 62624620 63424582 63555923 64592814 65128072 65473972
26	131	0.980629 (0.895487)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 10093907 10351530 10934306 11068401 11840460 12139956 13170394 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18009315 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 40620697 41666801 41667867 42880205 43800122 44014004 44811375 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 50284074 51804442 52561368 53663432 54235120 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57264850 57872920 58140380 58689469 59556412 60399235 60935573 61409218 62184461 62624620 63424582 63555923 64592814 65128072 65473972 66840159
26	132	0.981609 (0.898516)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 10093907 10351530 10934306 11068401 11840460 12139956 13170394 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18009315 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 40620697 41666801 41667867 42880205 43800122 44014004 44811375 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 50284074 51804442 52561368 53663432 54235120 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57264850 57872920 58140380 58689469 59556412 60399235 60935573 61409218 62184461 62624620 63424582 63555923 64592814 65128072 65473972 66840159
26	133	0.982544 (0.900969)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 10093907 10351530 10703320 10934306 11068401 11840460 12139956 13170394 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18009315 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 40620697 41666801 41667867 42880205 43800122 44014004 44811375 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 49724963 50284074 51804442 52561368 53663432 54235120 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57264850 57872920 58140380 58689469 59556412 60399235 60935573 61409218 62184461 62624620 63424582 63555923 64592814 65128072 65473972 66840159
26	134	0.983437 (0.902983)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 10093907 10351530 10703320 10934306 11068401 11840460 12139956 13170394 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18009315 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 40620697 41666801 41667867 42409869 42880205 43800122 44014004 44811375 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 49724963 50284074 51804442 52561368 53663432 54235120 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57264850 57872920 58140380 58689469 59556412 60399235 60935573 61409218 62184461 62624620 63424582 63555923 64592814 65128072 65473972 66840159

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
26	135	0.984288 (0.904657)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 10093907 10351530 10703320 10934306 11068401 11840460 12139956 13170394 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18009315 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 40620697 41666801 41667867 42409869 42880205 43800122 44014004 44811375 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 49724963 50284074 51804442 52561368 53663432 54235120 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57264850 57872920 58140380 58689469 59556412 60399235 60935573 61409218 62184461 62624620 63424582 63555923 64592814 64953738 65128072 65473972 66840159
26	136	0.985099 (0.906060)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 10093907 10351530 10703320 10934306 11068401 11840460 12139956 13170394 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18009315 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 40620697 41666801 41667867 42409869 42880205 43800122 44014004 44811375 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 49724963 50284074 51804442 52561368 53663432 54235120 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57264850 57872920 58140380 58689469 59556412 59884131 60399235 60935573 61409218 62184461 62624620 63424582 63555923 64592814 64953738 65128072 65473972 66840159
26	137	0.985872 (0.906487)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 10093907 10351530 10703320 10934306 11068401 11840460 12139956 13170394 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18009315 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 40620697 41666801 41667867 42409869 42880205 43800122 44014004 44811375 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 49724963 50284074 51200270 51804442 52561368 53663432 54235120 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57264850 57872920 58140380 58689469 59556412 59884131 60399235 60935573 61409218 62184461 62624620 63424582 63555923 64592814 64953738 65128072 65473972 66840159
26	138	0.986609 (0.907655)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 10093907 10351530 10703320 10934306 11068401 11840460 12139956 13170394 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18009315 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 40620697 41666801 41667867 42409869 42880205 43800122 44014004 44811375 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 49724963 50284074 51200270 51804442 52561368 53663432 54235120 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57264850 57872920 58140380 58689469 59556412 59884131 60399235 60853359 60935573 61409218 62184461 62624620 63424582 63555923 64592814 64953738 65128072 65473972 66840159
26	139	0.987309 (0.908040)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 9976550 10093907 10351530 10703320 10934306 11068401 11840460 12139956 13170394 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18009315 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 40262360 40620697 41666801 41667867 42409869 42880205 43800122 44014004 44811375 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 49724963 50284074 51200270 51804442 52561368 53663432 54235120 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57264850 57872920 58140380 58689469 59556412 59884131 60399235 60853359 60935573 61409218 62184461 62624620 63424582 63555923 64592814 64953738 65128072 65473972 66840159
26	140	0.987977 (0.909016)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 9976550 10093907 10351530 10703320 10934306 11068401 11840460 12139956 13170394 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18009315 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 39823167 40262360 40620697 41666801 41667867 42409869 42880205 43800122 44014004 44811375 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 49724963 50284074 51200270 51804442 52561368 53663432 54235120 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57264850 57872920 58140380 58689469 59556412 59884131 60399235 60853359 60935573 61409218 62184461 62624620 63424582 63555923 64592814 64953738 65128072 65473972 66840159

Continued on next page

Table B.1 – Continued from previous page						
m	n	R_e	d	R_c	Packed integer form of the parity check matrix	
26	141	0.988613 (0.909363)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 9976550 10093907 10351530 10703320 10934306 11068401 11840460 12139956 12936984 13170394 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18009315 18692090 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 39823167 40262360 40620697 41666801 41667867 42409869 42880205 43800122 44014004 44811375 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 49724963 50284074 51200270 51804442 52561368 53663432 54235120 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57264850 57872920 58140380 58689469 59556412 59884131 60399235 60853359 60935573 61409218 62184461 62624620 63424582 63555923 64592814 64953738 65128072 65473972 66840159	
26	142	0.989218 (0.910183)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 9976550 10093907 10351530 10703320 10934306 11068401 11840460 12139956 12936984 13170394 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18009315 18692090 18960733 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 39823167 40262360 40620697 41666801 41667867 42409869 42880205 43800122 44014004 44811375 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 49724963 50284074 51200270 51804442 52561368 53663432 54235120 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57264850 57872920 58140380 58689469 59556412 59884131 60399235 60853359 60935573 61409218 62184461 62624620 63424582 63555923 64592814 64953738 65128072 65473972 66840159	
26	143	0.989794 (0.910889)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 9976550 10093907 10351530 10703320 10934306 11068401 11840460 12139956 12936984 13170394 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18009315 18692090 18960733 19074231 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 39823167 40262360 40620697 41666801 41667867 42409869 42880205 43800122 44014004 44811375 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 49724963 50284074 51200270 51804442 52561368 53663432 54235120 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57264850 57872920 58140380 58689469 59556412 59884131 60399235 60853359 60935573 61409218 62184461 62624620 63424582 63555923 64592814 64953738 65128072 65473972 66840159	
26	144	0.990341 (0.911502)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 9976550 10093907 10351530 10703320 10934306 11068401 11840460 12139956 12936984 13170394 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18009315 18692090 18960733 19074231 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 39823167 40262360 40620697 41666801 41667867 42409869 42880205 43800122 44014004 44811375 45386586 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 49724963 50284074 51200270 51804442 52561368 53663432 54235120 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57264850 57872920 58140380 58689469 59556412 59884131 60399235 60853359 60935573 61409218 62184461 62624620 63424582 63555923 64592814 64953738 65128072 65473972 66840159	
26	145	0.990861 (0.912038)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 9976550 10093907 10351530 10703320 10934306 11068401 11840460 12139956 12936984 13170394 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18009315 18692090 18960733 19074231 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 39823167 40262360 40620697 41666801 41667867 42409869 42880205 43800122 44014004 44811375 45386586 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 49724963 50284074 51200270 51292883 51804442 52561368 53663432 54235120 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57264850 57872920 58140380 58689469 59556412 59884131 60399235 60853359 60935573 61409218 62184461 62624620 63424582 63555923 64592814 64953738 65128072 65473972 66840159	
26	146	0.991355 (0.912508)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 9976550 10093907 10351530 10703320 10934306 11068401 11840460 12139956 12936984 13170394 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18009315 18692090 18960733 19074231 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 25649268 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 39823167 40262360 40620697 41666801 41667867 42409869 42880205 43800122 44014004 44811375 45386586 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 49724963 50284074 51200270 51292883 51804442 52561368 53663432 54235120 54332396 54744803 54780620 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57264850 57872920 58140380 58689469 59556412 59884131 60399235 60853359 60935573 61409218 62184461 62624620 63424582 63555923 64592814 64953738 65128072 65473972 66840159	

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
26	147	0.991825 (0.912925)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 9976550 10093907 10351530 10703320 10934306 11068401 11840460 12139956 12936984 13170394 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18009315 18692090 18960733 19074231 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 25649268 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 39823167 40262360 40620697 41666801 41667867 42409869 42880205 43800122 44014004 44811375 45386586 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 49724963 50284074 51200270 51292883 51804442 52561368 53663432 54235120 54332396 54744803 54780620 54813602 55355562 55989894 55996779 56459293 56474461 56772244 57104360 57264850 57872920 58140380 58689469 59556412 59884131 60399235 60853359 60935573 61409218 62184461 62624620 63424582 63555923 64592814 64953738 65128072 65473972 66840159
26	148	0.992272 (0.913297)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 9976550 10093907 10351530 10703320 10934306 11068401 11840460 12139956 12936984 13170394 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18009315 18692090 18960733 19074231 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 25649268 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 39823167 40262360 40620697 41666801 41667867 42409869 42880205 43800122 44014004 44811375 45386586 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 49724963 50284074 51200270 51292883 51804442 52561368 53663432 54235120 54332396 54744803 54780620 54813602 55355562 55989894 55996779 56459293 56474461 56772244 56900023 57104360 57264850 57872920 58140380 58689469 59556412 59884131 60399235 60853359 60935573 61409218 62184461 62624620 63424582 63555923 64592814 64953738 65128072 65473972 66840159
26	149	0.992696 (0.913630)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 9976550 10093907 10351530 10703320 10934306 11068401 11356330 11840460 12139956 12936984 13170394 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18009315 18692090 18960733 19074231 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 25649268 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 39823167 40262360 40620697 41666801 41667867 42409869 42880205 43800122 44014004 44811375 45386586 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48559824 49569716 49724963 50284074 51200270 51292883 51804442 52561368 53663432 54235120 54332396 54744803 54780620 54813602 55355562 55989894 55996779 56459293 56474461 56772244 56900023 57104360 57264850 57872920 58140380 58689469 59556412 59884131 60399235 60853359 60935573 61409218 62184461 62624620 63424582 63555923 64592814 64953738 65128072 65473972 66840159
26	150	0.993098 (0.913930)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 9976550 10093907 10351530 10703320 10934306 11068401 11356330 11840460 12139956 12936984 13170394 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18009315 18692090 18960733 19074231 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 25649268 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 39823167 40262360 40620697 41666801 41667867 42409869 42880205 43800122 44014004 44811375 45386586 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48023181 48559824 49569716 49724963 50284074 51200270 51292883 51804442 52561368 53663432 54235120 54332396 54744803 54780620 54813602 55355562 55989894 55996779 56459293 56474461 56772244 56900023 57104360 57264850 57872920 58140380 58689469 59556412 59884131 60399235 60853359 60935573 61409218 62184461 62624620 63424582 63555923 64592814 64953738 65128072 65473972 66840159
26	151	0.993480 (0.914201)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 9976550 10093907 10351530 10703320 10934306 11068401 11356330 11840460 12139956 12936984 13170394 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18009315 18692090 18960733 19074231 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 25649268 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 39417953 39823167 40262360 40620697 41666801 41667867 42409869 42880205 43800122 44014004 44811375 45386586 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48023181 48559824 49569716 49724963 50284074 51200270 51292883 51804442 52561368 53663432 54235120 54332396 54744803 54780620 54813602 55355562 55989894 55996779 56459293 56474461 56772244 56900023 57104360 57264850 57872920 58140380 58689469 59556412 59884131 60399235 60853359 60935573 61409218 62184461 62624620 63424582 63555923 64592814 64953738 65128072 65473972 66840159
26	152	0.993843 (0.914447)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 9976550 10093907 10351530 10703320 10934306 11068401 11356330 11840460 12139956 12936984 13170394 14134723 14534737 14816701 16300368 16713067 17641732 17992524 18009315 18692090 18960733 19074231 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 25649268 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 39417953 39823167 40262360 40620697 41666801 41667867 42409869 42880205 43800122 44014004 44811375 45386586 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48023181 48559824 49569716 49724963 50284074 51200270 51292883 51804442 52561368 53574895 53663432 54235120 54332396 54744803 54780620 54813602 55355562 55989894 55996779 56459293 56474461 56772244 56900023 57104360 57264850 57872920 58140380 58689469 59556412 59884131 60399235 60853359 60935573 61409218 62184461 62624620 63424582 63555923 64592814 64953738 65128072 65473972 66840159

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
26	153	0.994187 (0.914673)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 9976550 10093907 10351530 10703320 10934306 11068401 11356330 11840460 12139956 12936984 13170394 14134723 14534737 14816701 16194875 16300368 16713067 17641732 17992524 18009315 18692090 18960733 19074231 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 25649268 26280684 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 39417953 39823167 40262360 40620697 41666801 41667867 42409869 42880205 43800122 44014004 44811375 45386586 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48023181 48559824 49569716 49724963 50284074 51200270 51292883 51804442 52561368 53574895 53663432 54235120 54332396 54744803 54780620 54813602 55355562 55989894 55996779 56459293 56474461 56772244 56900023 57104360 57264850 57872920 58140380 58689469 59556412 59884131 60399235 60853359 60935573 61409218 62184461 62624620 63424582 63555923 64592814 64953738 65128072 65473972 66840159
26	154	0.994513 (0.914880)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 9976550 10093907 10351530 10703320 10934306 11068401 11356330 11840460 12139956 12936984 13170394 14134723 14534737 14816701 16194875 16300368 16713067 17641732 17992524 18009315 18692090 18960733 19074231 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 25649268 26280684 26355497 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 39417953 39823167 40262360 40620697 41666801 41667867 42409869 42880205 43800122 44014004 44811375 45386586 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48023181 48559824 49569716 49724963 50284074 51200270 51292883 51804442 52561368 53574895 53663432 54235120 54332396 54744803 54780620 54813602 55355562 55989894 55996779 56459293 56474461 56772244 56900023 57104360 57264850 57872920 58140380 58689469 59556412 59884131 60399235 60853359 60935573 61409218 62184461 62624620 63424582 63555923 64592814 64953738 65128072 65473972 66840159
26	155	0.994822 (0.915071)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 9976550 10093907 10351530 10703320 10934306 11068401 11356330 11840460 12139956 12936984 13170394 14134723 14534737 14816701 16194875 16300368 16713067 17641732 17992524 18009315 18692090 18960733 19074231 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 25649268 26280684 26355497 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38278580 39088097 39417953 39823167 40262360 40620697 41666801 41667867 42409869 42880205 43287833 43800122 44014004 44811375 45386586 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48023181 48559824 49569716 49724963 50284074 51200270 51292883 51804442 52561368 53574895 53663432 54235120 54332396 54744803 54780620 54813602 55355562 55989894 55996779 56459293 56474461 56772244 56900023 57104360 57264850 57872920 58140380 58689469 59556412 59884131 60399235 60853359 60935573 61409218 62184461 62624620 63424582 63555923 64592814 64953738 65128072 65473972 66840159
26	156	0.995115 (0.915223)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 9976550 10093907 10351530 10703320 10934306 11068401 11356330 11840460 12139956 12936984 13170394 14134723 14534737 14816701 16194875 16300368 16713067 17641732 17992524 18009315 18692090 18960733 19074231 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 25649268 26280684 26355497 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38200398 38278580 39088097 39417953 39823167 40262360 40620697 41666801 41667867 42409869 42880205 43287833 43800122 44014004 44811375 45386586 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48023181 48559824 49569716 49724963 50284074 51200270 51292883 51804442 52561368 53574895 53663432 54235120 54332396 54744803 54780620 54813602 55355562 55989894 55996779 56459293 56474461 56772244 56900023 57104360 57264850 57872920 58140380 58689469 59556412 59884131 60399235 60853359 60935573 61409218 62184461 62624620 63424582 63555923 64592814 64953738 65128072 65473972 66840159
26	157	0.995393 (0.935938)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 9976550 10093907 10351530 10703320 10934306 11068401 11356330 11840460 12139956 12936984 13170394 14134723 14534737 14816701 16194875 16300368 16713067 17641732 17992524 18009315 18692090 18960733 19074231 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 25649268 26280684 26355497 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38200398 38278580 39088097 39417953 39823167 40262360 40620697 41666801 41667867 42409869 42880205 43287833 43800122 44014004 44811375 45386586 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48023181 48559824 49569716 49724963 50284074 51200270 51292883 51804442 52561368 53574895 53663432 54235120 54332396 54744803 54780620 54813602 55355562 55989894 55996779 56459293 56474461 56772244 56900023 57104360 57264850 57872920 58140380 58689469 59556412 59884131 60399235 60853359 60935573 61409218 62184461 62279477 62624620 63424582 63555923 64592814 64953738 65128072 65473972 66840159
26	158	0.995656 (0.936091)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 9976550 10093907 10351530 10703320 10934306 11068401 11356330 11840460 12139956 12936984 13170394 14134723 14534737 14816701 16194875 16300368 16713067 17641732 17992524 18009315 18692090 18960733 19074231 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 25649268 26280684 26355497 26831716 27780357 27845706 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38200398 38278580 39088097 39417953 39823167 40262360 40620697 41666801 41667867 42409869 42880205 43287833 43800122 44014004 44811375 45386586 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48023181 48559824 49569716 49724963 50284074 51200270 51292883 51292883 51804442 52561368 53574895 53663432 54235120 54332396 54744803 54780620 54813602 55355562 55989894 55996779 56459293 56474461 56772244 56900023 57104360 57264850 57872920 58140380 58689469 59556412 59884131 60399235 60853359 60935573 61409218 62184461 62279477 62624620 63424582 63555923 64592814 64953738 65128072 65473972 66840159

Continued on next page

Table B.1 – Continued from previous page

m	n	R_e	d	R_c	Packed integer form of the parity check matrix
26	159	0.995904 (0.936236)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 9976550 10093907 10351530 10703320 10934306 11068401 11356330 11840460 12139956 12936984 13170394 14134723 14534737 14816701 16194875 16300368 16713067 17641732 17992524 18009315 18692090 18960733 19074231 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 25649268 26280684 26355497 26831716 27780357 27845706 28108540 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38200398 38278580 39088097 39417953 39823167 40262360 40620697 41666801 41667867 42409869 42880205 43287833 43800122 44014004 44811375 45167283 45386586 45447850 46058364 46236459 46251188 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48023181 48559824 49569716 49724963 50284074 51200270 51292883 51804442 52561368 53574895 53663432 54235120 54332396 54744803 54780620 54813602 55355562 55989894 55996779 56459293 56474461 56772244 56900023 57104360 57264850 57872920 58140380 58689469 59556412 59884131 60399235 60853359 60935573 61409218 62184461 62279477 62624620 63424582 63555923 64592814 64953738 65128072 65473972 66840159
26	160	0.996140 (0.936372)	7	6	1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152 4194304 8388608 16777216 33554432 1160185 1564702 2585117 3419307 4492287 4810283 9353857 9395871 9976550 10093907 10351530 10703320 10934306 11068401 11356330 11840460 12139956 12936984 13170394 14134723 14534737 14816701 16194875 16300368 16713067 17641732 17992524 18009315 18692090 18960733 19074231 20131180 20858618 22426699 22460065 22562674 23680920 23707076 23791323 24279912 25575249 25579531 25649268 26280684 26355497 26831716 27780357 27845706 28108540 28552180 28936460 32368082 32564036 34048695 34254884 35283464 35985048 35986053 35986898 36286430 38200398 38278580 39088097 39417953 39823167 40262360 40620697 41666801 41667867 42409869 42880205 43287833 43800122 44014004 44811375 45167283 45386586 45447850 46058364 46236459 46251188 46326534 46910965 47221140 47347900 47361840 47389686 47437732 47592948 47926580 48023181 48559824 49569716 49724963 50284074 51200270 51292883 51804442 52561368 53574895 53663432 54235120 54332396 54744803 54780620 54813602 55355562 55989894 55996779 56459293 56474461 56772244 56900023 57104360 57264850 57872920 58140380 58689469 59556412 59884131 60399235 60853359 60935573 61409218 62184461 62279477 62624620 63424582 63555923 64592814 64953738 65128072 65473972 66840159