UNIVERSITY OF SOUTHAMPTON

Advanced Channel Coding for 5G Wireless System

by

Shuai Shao

Supervisors:

Prof. Robert G. Maunder and Prof. Lajos Hanzo

A thesis submitted in partial fulfillment for the degree of Doctor of Philosophy

in the

Faculty of Engineering and Physical Sciences

School of Electronics and Computer Science

June 2021

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF ENGINEERING, SCIENCE AND MATHEMATICS
SCHOOL OF ELECTRONICS AND COMPUTER SCIENCE

Doctor of Philosophy

by Shuai Shao

Channel coding may be viewed as the best-informed and most potent component of cellular communication systems, which is used for correcting the transmission errors inflicted by noise, interference and fading. The powerful turbo code was selected to provide channel coding for Mobile Broad Band (MBB) data in the 3G UMTS and 4G LTE cellular systems. However, the 3GPP standardization group has recently determined to replace it by Low Density Parity Check (LDPC) and polar codes in 5G New Radio (NR), which adopted the LDPC code family for enhanced Mobile Broad Band (eMBB) data and polar codes for eMBB control. Against this background, this thesis has deeply researched and reviewed the three main type of channel coding system, including turbo, LDPC and polar codes. We exploit these three channel coding systems from their encoding characterisations to the decoding performance. By the time of publishing of the 5G NR standardisation from the 3RD Generation Partnership Project (3GPP) group, we summarised the factors that influenced this standardisation, with a particular focus on the Application specific integrated circuit (ASIC) implementation of the decoders of these three codes. As presented in this thesis, we show that the overall implementation complexity of turbo, LDPC and polar decoders depend on numerous other factors beyond their computational complexity. More specifically, we compare the throughput, error correction capability, flexibility, area efficiency and energy efficiency of ASIC implementations drawn from 110 papers and use the results for characterising the advantages and disadvantages of these three codes as well as for avoiding pitfalls and for providing design guidelines.

We then focus on one of the main channel coding systems in the 5G NR standard, namely the NR LDPC code. In this thesis, the full standard about the 5G NR LDPC code that is specified in TS38.212 [1] is presented with straight-forward context and corresponding illustrations. By contrast to the LDPC codes adopted in the previous standards, the NR LDPC code can be considered to be a concatenation of a LDPC code and a Low Density Generator Matrix (LDGM) code. This particular character has the advantage of offering a flexible coding rate, which actually effects the correctness of the conventional 2D Extrinsic Information Transfer (EXIT) chart analysis. In the thesis, we commence with the conventional 2D EXIT chart analysis for regular LDPC code as the background. Then, we characterise the incorrect 2D EXIT chart's analysis and addresses it using a novel reinterpretation of the NR LDPC factor graph. Based on this factor graph, a novel 3D EXIT chart technique is conceived for our three-stage scheme, which facilities the visual characterisation of the NR LDPC decoder's iterative decoding convergence process for the first time. We also introduce a novel 2D projection of the EXIT chart for NR LDPC decoder, which directly solve the problem that resulted by the conventional 2D EXIT chart.

Since our novel EXIT chart analysis is able to describe the exchange of mutual information (MI) in NR LDPC decoder correctly, we research implementations relying-on our

novel EXIT charts. We demonstrate this technique by using it to design a novel iterative decoding activation order for the NR LDPC decoder, which reduces the decoding complexity by approximately 17% compared to a conventional flooding-based decoder without any degrading of its error correction capability. We conclude by discussing several other opportunities for exploiting the proposed 3D EXIT chart technique to improve the design of concatenated LDPC and LDGM codes. We also use this novel EXIT chart analysis to improve the NR Hybrid Automatic Repeat reQuests (HARQ) system for LDPC coded data transmission that is specified in the NR standard by attaching two proposed scheme, namely the Deferred Iteration (DI) and the Early Abandon (EA). The proposed HARQ scheme significantly reduce the complexity of the conventional HARQ system without any degrading on the performance of effective throughput and error-correction.

Dedicated to my beloved family and friends ...

Contents

List of Publications

1. **S. Shao**, P. Hailes, T. Wang, J. Wu, R. G. Maunder, B. M. Al-Hashimi and L. Hanzo, "Survey of Turbo, LDPC, and Polar Decoder ASIC Implementations," *IEEE Communications Surveys Tutorials*, vol. 21, no. 3, pp. 2309-2333, 2019.

2. **S. Shao**, Y. Zhang, R. G. Maunder and L. Hanzo, "3D EXIT Charts for Analysing the 5G 3GPP New Radio LDPC Decoder," *IEEE ACCESS*, , vol. 8, pp. 188797-188812, 2020.

3. C. Chang, J. Wu, M. Yang, T. Wang, **S. Shao** and R. G. Maunder, "Low-latency compressive active user identification over frequency-selective fading channels," *2018 IEEE Wireless Communications and Networking Conference (WCNC)*,2018.

4. **S. Shao**, Y. Zhang, R. G. Maunder and L. Hanzo, "Low-Complexity NR LDPC HARQ Design with aided of 3D EXIT analysis," *IEEE ACCESS*, in preparation.

5. Y. Zhang, **S. Shao**, R. G. Maunder, A. F. Al Rawi, and L. Hanzo, "Low-Complexity HARQ Design for LDPC-Coded Metallic Access Networks," *IEEE ACCESS*, in preparation.

6. J. Chen, **S. Shao**, S. X. Ng, R. G. Maunder and L. Hanzo, "EXIT Chart Analysis of Joint Source Channel Coding using Protograph LDPC Codes", in preparation

DECLARATION OF AUTHORSHIP

I, **Shuai Shao**, declare that the thesis entitled **Advanced Channel Coding for 5G Wireless System** and the work presented in the thesis are both my own, and have been generated by me as the result of my own original research. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University;
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
- Where I have consulted the published work of others, this is always clearly attributed;
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
- I have acknowledged all main sources of help;
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly What was done by others and what I have contributed myself;
- Parts of this work have been published in the provided list of publications.

Signed: .....
Date: .....

Acknowledgements

This thesis would not have been possible without the help, support and guidance of many people. I would like to express my heartfelt gratitude to my supervisors, Prof. Robert Maunder and Prof. Lajos Hanzo, for their all supports in any kind. Thanks to Prof. Robert Maunder for his insightful guidance and technical supports through all the time of the research. Thanks to Prof. Lajos Hanzo for his guidance for the research and also varied out-research activities.

I would also like to thank my colleagues Yanqing Zhang, Luping Xiang, Yusha Liu and all of whom have provided helpful discussion, assistance and motivations. I am glad to spend my PhD times in our Next Generation Wireless research group for the most inspiring and friendly atmosphere. As a difficult year of 2020, I would also like to thanks my flat mate Huaiyang Zeng, who companied with me during the period of self-quarantine. And finally, I would like to express my warmest gratitude to my parents for their unconditional love, understanding and support.

List of Symbols

Chapter 2

K	Information block length
R	Coding rate
$\prod$	The interleaver in turbo encoder
O	Representation of propotional to
d_v	Degree of variable nodes
d_c	Degree of check nodes
N	Encoded block length
$\mathbf{x}$	Information block
$\mathbf{G}$	Generator matrix
$\mathbf{U}$	Encoded block
μ	The averaging calculation
M	Original encoded block length of polar code
L	List size of SCL polar code

Chapter 3

T	Decoded throughput
E_b/N_0	Channel SNR per bit
η	Effective throughput

Chapter 4

A	Transport block length
K'	Original information block length
K	Information block length after zero padding
a	Original transport block of NR LDPC system
b	Transport block after transport-block CRC attachment
c'	One block segmentation of transport block
d'	Encoded block segmentation
f'	Rate-matched encoded block segmentation
$\tilde{a}$	Extracted information block after CRC de-attachment
$\tilde{b}$	Extracted information block with CRC-attachment
$\tilde{c}'$	One of the decoded block segmentations in the received block
$\tilde{d}'$	One of the rate-dematched block segmentations in the received block
$\tilde{f}'$	One of the block segmentations in the received block
i_{LS}	Set index
Z_c	Lifting size
w	Parity bits
V	PCM derivation matrix
E	Encoded bits
N	Transmitted bits (after rate matching)
rv_{id}	Redundancy version
k_0	Starting point of re-transmission in circular buffer
σ^2_{ch}	Variance of channel

Chapter 5

$\tilde{d}'_{(t-1)}$	The rate-dematched block segmentations in the received block from last transmission
$\tilde{d}'_{memory}$	The output of HARQ external memory
I_{conv}	Threshold value of improvement of *extrinsic* MI to check the state of convergence

Chapter 1

Introduction

Fuelled by its immersive applications, wireless communication systems developed rapidly to satisfy the requirements of different users. In cellular communication systems such as 3G UMTS [1] and 4G LTE [2], wireless transmission is used to convey data between user equipment and basestations, where the latter act as gatekeepers to the Internet and telephone networks. However, the received data typically differs from the transmitted data, owing to transmission errors caused by noise, interference and fading. In order to correct these transmission errors, cellular communication systems use forward error correction channel codes. More specially, channel coding increases the original information block length K to the encoded block length N by adding parity check bits. In the receiver, the additional $(N-K)$ bits provide the channel decoder with redundancy that allows the decoder to detect and correct transmission errors, which affect the original K information bits.

In the past couple of decades, wireless communication systems have substantially benefited from the employment of iterative decoding algorithms. For example, the classic turbo codes [2] were adopted by the 3G Universal Mobile Telecommunication System (UMTS) [3] and 4G Long Term Evolution (LTE) [4] mobile broadband standards. By contrast, the Low Density Parity Check (LDPC) codes [5] were ratified by the WiFi [6], WiMax [7], WiGig [8], DVB-S2 [9] and 10GBase-T [10] standards. A more recent scheme is constituted by polar codes [11]. Both the turbo and LDPC codes employ an iterative decoding process, in which each successive decoding attempt informs the next, until the process converges to a legitimate codeword. By contrast, polar codes select the recovered information block from a list of candidates obtained from the associated parallel successive cancellation decoding processes, in which the decoding of each successive information bit informs the decoding of the next. There was a technical debate concerning which one of these three fundamental channel codes will be adopted by the 5th Generation (5G) NR standardized channel coding system. Therefore, a survey is motivated to research the characteristics of these three channel codes leaving in mind the specific requirements of next-generation of wireless communication scenario.

In this context we conducted a survey of the ASIC implementations of three different types of channel decoders. For all three codes, the channel decoder has a much higher complexity than the corresponding encoder, since the decoder uses iterative or parallel decoding processes, which rely on probabilistic representations of the encoded bits to overcome the uncertainty introduced by noise, interference and fading. Owing to this, it is the error correction performance and implementation characteristics of the channel decoder that are typically the main concerns when designing a channel code. These implementation characteristics include reconfiguration flexibility, processing throughput, processing latency, energy efficiency and hardware efficiency. More specifically, this survey was particularly focused on their capability to meet the different requirements associated with the eMBB, Ultra Reliable Low Latency Communication (URLLC) and massive Machine Type Communication (mMTC) applications of 5G [12].

After a few years of discussion, the 3GPP meeting determined the channel coding standard for the 5G NR wireless communication. In the standard, the LDPC code was adopted for data transmission, while the polar code was chosen as the NR channel coding technique for control signals. In contrast to the LDPC codes adopted in previous standards, the NR LDPC code can be considered to be a concatenation of a core LDPC code and a Low Density Generation Matrix (LDGM) code. This particular feature has the advantage of offering a flexible choice of coding rates, but it prevents the application of conventional EXIT chart analysis. Given this problem, a new method of EXIT chart analysis is required for the research of the NR LDPC decoder. Our proposed solution relies on a novel reinterpretation of the NR LDPC factor graph. Based on this factor graph, a novel 3D EXIT chart technique is introduced, which enables the visual characterisation of the NR LDPC decoder's iterative decoding convergence process for the first time. Furthermore, an insightful 2D EXIT chart projection is also conceived from the 3D EXIT chart in order to directly facilitate conventional EXIT chart analysis.

In order to demonstrate the capability offered by this novel EXIT chart analysis, we conceived beneficial NR LDPC applications for the improvement of the conventional NR standard. The NR LDPC decoder relying on optimal decoder activation order is one of the successful examples of exploiting our 3D EXIT chart analysis. The proposed scheduled LDPC decoder reduces the decoding complexity by approximately 17% compared to a so-called conventional flooding-type solution [13], which is achieved without degrading the error correction capability. More specifically, the proposed schedule exploits the observation emerging from our EXIT chart analysis that the MI [14] in the LDGM part of the 3GPP NR LDPC decoder will converge within a few iterations, with any further iterative decoding processing performed in the LDGM part becoming redundant. Inspired by this observation, the proposed scheduled NR LDPC decoder is designed to use only the LDPC part of the Parity Check Matrix (PCM) during the later decoding iterations.

The other EXIT chart aided application introduced in this thesis is the NR HARQ system proposed for LDPC coded data transmission. In the 3GPP standard of NR wireless communication [1], the HARQ is applied to the LDPC coded data transmission by transmitting rate-matched encoded code blocks constructed from the entire encoded bits stored in the circular transmit buffer. Given our novel EXIT analysis, the proposed NR HARQ system introduced the Deferred Iteration (DI) and Early Abandon (EA) concept of [15] for enhancing the conventional HARQ system, including the dispensable re-transmission and decoding iterations. As a result, the proposed NR HARQ system significantly reduces the computational complexity without degrading the error-correction performance and throughput.

1.1 Related work

This section presents our state-of-the-art review. Section 1.1.1 summarises the previous contributions on channel coding implementations, while Section 1.1.2 highlights the previous contributions in the filed of EXIT chart analysis and its applications, with an emphasis on the LDPC codes.

1.1.1 Literature review of channel coding implementations

[13, 15–22], which discussed the implementation of channel decoders. For example, Roth *et al.* [17] reviewed the trade-offs of area, throughput and energy-efficiency in the context of the Very-large-scale Integration (VLSI) implementation of LDPC decoders. Brejza *et al.* [21] disseminated a comprehensive treatise on the design of energy-aware turbo coding and its related applications. Furthermore, Hailes *et al.* [13] reviewed the LDPC decoder implementations based on Field-programmable gate arrays (FPGAs) at a glance. Hence the main channel decoding implementation oriented survey papers are listed in Table 1.1. However, none of them have considered the specific ASIC hardware implementations employed in user equipment. The literature review of different channel coding implementation survey papers gives the motivation of the research of channel coding ASIC implementations in order to further explore the debate concerning the choice of channel coding for next-generation wireless communication. More specifically, our survey will focus on the throughput, error correction capability, flexibility, area efficiency and energy efficiency of ASIC implementations drawn from 110 papers and will use the results for characterising both the advantages and disadvantages of these three codes, whilst avoiding potential pitfalls and providing tangible design guidelines.

TABLE 1.1: Comparison of different channel coding implementation survey papers.

Year	Paper	Focus/Main contribution
2010	H. Mercier *et al.* [16]	Survey of error correcting codes for channels corrupted by synchronization errors with potential applications and challenges
2011	C. Roth *et al.* [17]	Survey of area, throughput, and energy-efficiency trade-offs in the VLSI implementation of LDPC decoders
2012	Nasruminallah and L. Hanzo [18]	Source and channel decoding and their iterative decoding and transmission schemes offered to multimedia communication systems
2013	H. Chen and R. G. Maunder and L. Hanzo [15]	Low-complexity turbo coding techniques and a holistic hybrid ARQ design
2015	Y. Fang *et al.* [19]	Protograph LDPC codes and their applications
2015	E. Ankan *et al.* [20]	Challenges and new directions in spatially coupled LDPC codes, non-bionary LDPC codes, and polar coding
2016	M. F. Brejza *et al.* [21]	Turbo coding and energy-aware design guidelines for energy-constrained wireless applications
2016	P. Hailes *et al.* [13]	LDPC decoder implementations based on FPGA
2017	H. Ben Thameur *et al.* [22]	LDPC convolutional codes design based on different scheduling techniques

1.1.2 Literature review of EXIT analysis and its applications to LDPC codes

Table 1.2 presents the reliant previous contributions on the EXIT chart analysis of LDPC decoders, for our novel EXIT chart analysis as a springboard in Chapter 4. Since LDPC codes [23] and EXIT chart analysis [24] have been invented, many researches aimed for finding the most appropriate method of describing or illustrating the MI exchange within different types of LDPC decoders. In 2001, after introducing the EXIT charts for the convergence analysis of convolutional turbo codes [24], ten Brink [25] also introduced the EXIT chart analysis of LDPC decoders. Three years later, in 2004, ten Brink, Kramer and Ashikhmin identified the relationship between the *a priori* MI and the *extrinsic* MI for the Variable Node (VN) decoder and Check Node (CN) decoder of LDPC decoders [26]. In that paper, the MI exchange within a regular LDPC code was modelled by simple functions, which makes the EXIT chart analysis of LDPC decoding possible. Besides the regular LDPC code, researchers leave further explored the EXIT chart analysis of different types of LDPC decoders [27–29]. For example, Ardakani and Kschischang introduced the EXIT chart analysis of irregular LDPC code in [27]. In 2007, Liva and Chiani conceived a novel EXIT analysis technique for protograph-based LDPC codes, which cannot rely on the conventional EXIT analysis. In Chapter 4, we face the same limitation, namely that the conventional EXIT analysis fails to correctly

TABLE 1.2: Previous related works of the EXIT chart analysis and LDPC code

Year	**Paper**	**Contributions**
2001	S. ten Brink [25]	Introduced the EXIT chart analysis and trajectories to describe the convergence behaviour in the LDPC decoder
2004	M. Ardakani and F. R. Kschischang [27]	EXIT chart analysis for the irregular LDPC code
2004	S. ten Brink and G. Kramer and A. Ashikhmin [26]	First given the equations to generate the EXIT charts for regular LDPC codes
2007	G. Liva and M. Chiani [28]	Introduced the EXIT analysis to photographed LDPC codes
2011	X. Zuo and R. G. Maunder and L. Hanzo [29]	Fixed-points LDPC code design based on the EXIT analysis
2014	M. El-Hajjar and L. Hanzo [30]	Conclusion of EXIT charts analysis for the design of iterative system
2018	3GPP Meeting [1]	The standard of NR Multiplexing and channel coding
2018	T. Richardson and S. Kudekar [31]	Systematically described the code design and structure of the NR LDPC decoder
2018	Y. Julian and R. P. Astuti and K. Anwar [32]	Using the simulation value to generate the 2D EXIT projection of the NR LDPC decoder

describe the MI exchange within the NR LDPC code. This knowledge-gap inspired our novel EXIT analysis.

Many researchers used EXIT charts for designing efficient near-capacity LDPC codes. In 2011, Zuo, Maunder and Hanzo designed a fixed-point LDPC code based on EXIT-chart analysis [29]. In 2014, El-Hajjar and Hanzo published a Communication Survey and Tutorial (CST) paper to discuss the utilization of EXIT chart analysis in the design of iterative system, including channel coding and general two-stage concatenated systems. After few years of discussion, the 3GPP meeting finally decided to use LDPC codes as the channel coding for data transmission in next-generation wireless communication [1]. In the same year, Richardson and Kudekar comprehensively summarised the structure and code design of the NR LDPC code, including the detailed construction of its PCM and the characterization of its error-correction performance. Julian, Astuti and Anwar published a manuscript that shows the mismatch between the conventional EXIT-chart analysis and that of the NR LDPC decoder. In [32], a novel 2D EXIT chart was generated to correctly illustrate the MI exchange behaviour in the NR LDPC decoder. This paper inspired us to produce the 2D EXIT projection of our novel 3D EXIT charts that will directly show the open/closed EXIT-tunnel state of EXIT charts.

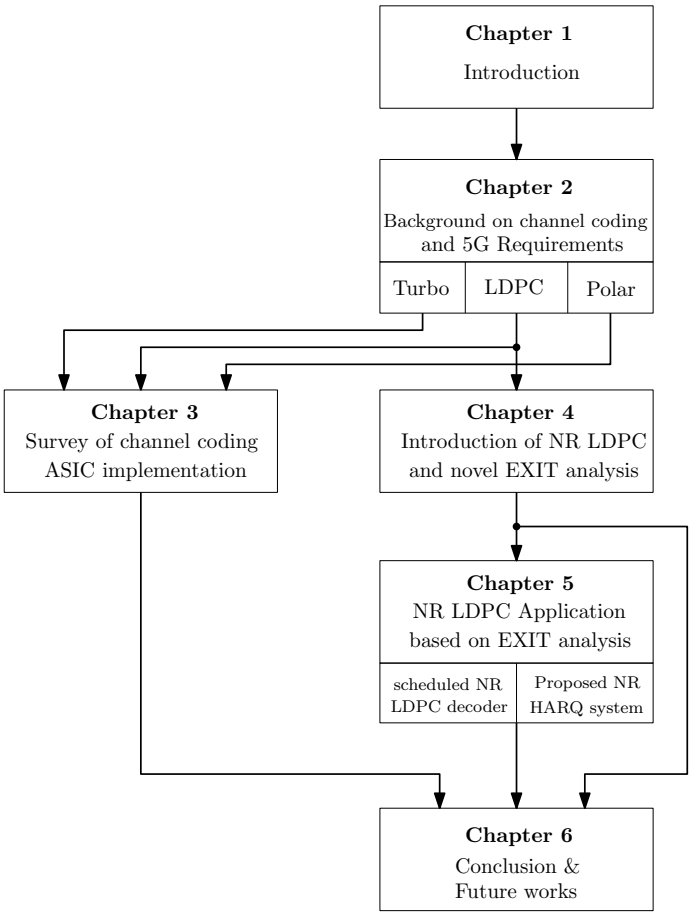


FIGURE 1.1: The outline of the thesis structure.

1.2 Motivation and thesis outline

The literature review of Section 1.1 provided a knowledge-gap analysis and gave us the motivation to conduct further research on these topics in this thesis.

This thesis is structured as seen in Figure 1.1, which is described in further detail as following.

Chapter 2 outlines the requirements of next-generation wireless communication for different applications, including the eMBB, URLLC and mMTC scenarios. This chapter also offers the fundamental background knowledge on three types of channel coding, including turbo, LDPC and polar codes with the aid of illustrations and simple equations. This lays the foundation for the survey of channel coding ASIC implementations in Chapter 3. Furthermore, the LDPC encoding and decoding algorithms form the bases of further research on NR LDPC schemes discussed in Chapter 4 and Chapter 5.

Chapter 3 summaries the characteristics of turbo, LDPC and polar decoder ASIC implementations from 110 papers. More specially, the survey compares the throughput,

error correction capability, flexibility, area efficiency and energy efficiency of ASIC implementations and use the results for characterising the advantages and disadvantages of these three codes for avoiding potential pitfalls and for providing design guidelines to be followed in the rest of the thesis. In order to meet the specifications of the 5G systems, we also present some suggestions and recommendations for designers and manufacturers to enhance their ASIC channel decoder implementations.

Chapter 4 presents the essential components in the physical layer of the 3GPP standardised NR LDPC data transmission. More specifically, the process of transmitting NR LDPC coded data frames is described in the order of transport block Cyclic Redundance Check (CRC) attachment, Base Graph (BG) selection, code block segmentation, code block CRC attachment, NR LDPC encoder, rate matching and the code block concatenation. Since the design of the NR LDPC decoder is not regulated by 3GPP meeting, we use the conventional flooding based NR LDPC decoder philosophy of [13] as our benchmark scheme. By observing the MI exchange in the NR LDPC decoder via conventional 2D EXIT charts, we found that the special structure of the NR LDPC factor graph makes the conventional 2D EXIT analysis inadequate for visualising the MI behaviour correctly. Therefore, we introduced a novel 3D EXIT chart and its 2D EXIT projection to solve this problem by re-arranging the factor graph of the NR LDPC decoder.

Chapter 5 introduces a pair of novel designs for NR LDPC applications relying on our 3D EXIT chart analysis. The first half of Chapter 5 describes the activation order-aided NR LDPC decoder, which applies EXIT analysis to switch off part of the nodes in the LDPC decoder. Hence, the proposed scheduled NR LDPC decoder reduces the complexity by 19% compared to the flooding decoder without degrading its error-correction capability. The other design is the proposed NR HARQ system conceived for LDPC coded data transmission. Inspired by [15] and our 3D EXIT charts, the proposed HARQ system appropriately adopts the DI and EA techniques. Since our novel EXIT chart analysis accurately describes the MI exchange, the proposed HARQ system allows us to eliminate superfluous retransmissions and decoding iterations. As a result, the proposed NR HARQ system significantly reduces the computational complexity of data transmission and decoding without any throughput and error-correction capability degradation.

Chapter 6 summaries our novel findings, followed by ideas for future research opportunities.

1.3 Novel contributions

The novel contributions of this thesis are listed as follows.

In **Chapter 3**, we characterised a diverse set of 110 ASIC implementations of turbo, LDPC and polar decoders. We have characterised the fundamental trade-offs between the various performance characteristics of channel decoder ASICs. Furthermore, we have demonstrated that the overall implementation complexity of a channel code depends not only on its computational complexity, but also on its interconnection complexity and its inherent flexibility. By summarising the performance of different channel coding implementations, we have also made some recommendations for manufacturers and designers to improve their implementation-oriented research.

In **Chapter 4**, we proposed a novel reorganisation of the NR LDPC decoder factor graph, which separates the decoder into four elements, namely core VNs, core CNs, extension VNs and extension CNs. To complement our novel factor graph, we modified the MI equations of [26], in order to develop a novel 3D EXIT chart representation of the NR LDPC decoder. We demonstrated that our 3D EXIT chart accurately characterises the MI exchange during each decoding iteration, as well as the EXIT chart's open/closed tunnel state. For increased convenience, this chapter also introduced a novel 2D EXIT projection of the proposed 3D EXIT chart, which makes it clear, if the EXIT tunnel is open or closed.

In **Chapter 5**, we proposed a pair of attractive designs relying on our novel EXIT chart analysis. Based on the MI exchange illustrated by our 3D EXIT chart pairs, we suggest to deactivate the extension CN decoding, when the correspond MI convergence reaches 99.99% of its fully converged value. This idea is then exploited for controlling the decoding activation order of the NR LDPC decoder components, which will reduce the decoding complexity. As demonstrated by our simulation results, the proposed scheduled NR LDPC decoder saves around 19% of complexity compared to that of the flooding-based decoder without eroding its error-correction capability. Since our EXIT chart analysis correctly characterizes the MI exchange of NR LDPC coded data transmission, we proposed a novel NR HARQ system relying on our EXIT analysis. Firstly, the observation of whether the EXIT tunnel for certain transmissions is open or not directly predicts the potential outcome of its decoding. This expectation of our EXIT analysis, which is introduced as our DI technique, allows the proposed HARQ system to skip the dispensable transmissions before the iterative decoding process its activation. Moreover, we introduced the EA technique to eliminate redundant decoding iterations if the CRC Early Termination (ET) check is failed. By using our EXIT chart, the EA technique is able to accurately curtailing the decoding, if the MI exchange has reached its convergence as indicated by the EXIT trajectories. As a result, our proposed NR HARQ system significantly reduces the computational complexity without any degradation error-correction performance and throughput.

Chapter 2

Background on the 5G Wireless Communication and Channel Coding

2.1 Introduction

In cellular communication systems such as 3G UMTS [3] and 4G LTE [4], wireless transmission is used to convey data between user equipment and basestations, where the latter act as gatekeepers to the Internet and telephone networks. However, the received data typically differs from the transmitted data, owing to transmission errors caused by noise, interference and fading. In order to correct these transmission errors, cellular communication systems use forward error correction channel codes. More specifically, a channel encoder is used in the transmitter (be it the user equipment or the basestation) to convert each so-called information block comprising K data bits into a longer encoded block comprising $N > K$ encoded bits, which are transmitted. In the receiver, the additional $(N - K)$ encoded bits provide the channel decoder with redundancy that allows it to detect and correct transmission errors within the original K information bits.

If the noise, interference or fading is particularly severe, then a low coding rate $R = K/N$ will be required for the channel decoder to successfully detect and correct all transmission errors. However, a low coding rate implies the transmission of a high number N of encoded bits, which consume precious transmission time, energy and bandwidth resources. Therefore, desirable channel codes allow the successful detection and correction of transmission errors at coding rates R that approach the theoretical channel capacity [33].

In the past couple of decades, several near-capacity channel codes have emerged, including the classic turbo codes [2] adopted by the 3G UMTS [3] and 4G LTE [4] mobile broadband standards, the LDPC codes [5] ratified by the WiFi [6], WiMax [7], WiGig [8], DVB-S2 [9] and 10GBase-T [10] standards as well as the more recent polar codes [11]. Both the turbo and LDPC codes employ an iterative decoding process, in which each successive attempt at decoding the information block informs the next, until the process converges to a legitimate codeword. By contrast, polar codes select the recovered information block from a list of candidates obtained from the associated parallel successive cancellation decoding processes, in which the decoding of each successive information bit informs the decoding of the next. For all three codes, the channel decoder has a much higher complexity than the corresponding encoder, since it uses iterative or parallel decoding processes, which rely on probabilistic representations of the encoded bits to overcome the uncertainty introduced by noise, interference and fading. Owing to this, it is the error correction performance and implementation characteristics of the channel decoder that are typically the main concerns when designing a channel code. These implementation characteristics include reconfiguration flexibility, processing throughput, processing latency, energy efficiency and hardware efficiency.

At the time of writing, the 3GPP standardization group is deliberating on the 5G specifications under the terminology of NR, where the turbo code of the operational 3G UMTS and 4G LTE systems has been replaced by an LDPC code for its eMBB data mode, supported by a polar code in the eMBB control mode. To elaborate a little further, the turbo code has been replaced, because it is considered to be incapable of efficiently achieving the multi-Gbps processing throughput required for eMBB. In addition to having an increased throughput, the eMBB mode aims for attaining an improved coverage, which will enable faster file-downloads and more reliable video streaming, for example. Furthermore, 5G is also targeting both URLLC and mMTC [34] applications. Explicitly, URLLC will offer significantly improved error correction capability and latency, in order to support mission-critical applications, such as autonomous vehicles [35]. By contrast, mMTC will offer significantly improved energy efficiency for the Internet of Things (IoT).

Against the background described above, this chapter provides an overview and comparison of turbo, LDPC and polar codes, with particular focus on their capability to meet the different requirements associated with the eMBB, URLLC and mMTC applications of 5G, as detailed in Section 2.2. The operation and characteristics of turbo, LDPC and polar codes are detailed in Section 2.3. And finally, a conclusion will be conducted in Section 2.4.

2.2 The requirements for 5G

Like 3G UMTS and 4G LTE, the aim of the 3GPP 5G NR is to continue the trend of offering substantially improved user experience and more diverse applications for cellular communications. However, this will impose stricter requirements upon all system components, including the channel code [36], as summarized in the following discussions. Note that the different eMBB, and mMTC applications for 5G impose different requirements, which may be impossible to meet simultaneously using a single channel code. Therefore, it may be expected that different channel codes are adopted for different applications.

2.2.1 Processing throughput

The strictest throughput requirements are imposed by the eMBB applications of 3GPP 5G NR. A peak transmission throughput of 20 Gbps is targeted for these eMBB applications, which is much higher than the 1 Gbps achieved by 4G LTE. During video streaming, this significantly improved throughput will enable opportunistic forward buffering, when the channel conditions are favorable, for example. This will substantially increase the reliability of streamed video, which currently suffers from 'stutter' when the channel conditions become unfavorable.

Since all received information has to pass through the channel decoder, it must offer an information throughput of $T = 20$ Gbps. Note that 20Gbps is the peak throughput of a cell, which could (in principle) be allocated all to one user who has (very) favourable channel conditions. Typically, the throughput of the cell varies with time, is lower than 20 Gbps and is shared by multiple users having a range of channel conditions. For our convenience, we assume the required throughput in this case is $T = 20$ Gbps. Achieving this information throughput will require a high degree of parallel processing. If we (perhaps optimistically) assume that $I = 10$ decoding steps (namely iterations or successive cancellation steps) are required for the decoder that are not using pipeline technique. While this channel decoder processors can operate at a clock frequency of $F = 1$ GHz, which is able to complete one decoding iteration in one clock cycle. Then at least $P = I \cdot T/F = 200$ parallel processors will be required. This parallelism may be implemented internally, by using an array of P processors that collaborate during the processing of each block. Alternatively, a high information throughput can be achieved using external parallelism, where separate channel decoders are used for processing multiple blocks at the same time, or where multiple blocks are 'unrolled' and pipelined through the same decoder at the same time [37]. However, this external parallelism

approach does not achieve the same low latency benefit of internal parallelism, as it will be discussed in the following Section 2.2.2.

2.2.2 Processing latency

The strictest latency requirements are imposed by the URLLC applications of 3GPP 5G NR. An end-to-end latency of 0.5 ms is targeted for these URLLC applications, which is much lower than the 10 ms achieved in 4G LTE [38]. This significantly improved latency will allow user inputs made on a user equipment to be delivered to the cloud, processed on a cloud computer and then returned to update the display of the user equipment, without the user perceiving an objectionable delay, for example. This will enable new applications in user-specific 3D video rendering, augmented reality, remote control and mobile gaming, among others. Furthermore, since machines are more sensitive to latency than humans, these ultra-low latencies will support new mMTC applications, such as swarm robotics, factory automation, as well as vehicular control and safety applications.

However, an end-to-end latency of 0.5 ms implies a physical layer latency of 8 to 67 μs [39], which allows data to be transmitted immediately after receiving control information, within the guard interval duration of 8 to 67 μs in the same time slot. Furthermore, the channel decoder must share this latency budget with many other physical layer components, such as channel estimation and demodulation. Owing to this, the channel decoder should target processing latencies as low as 0.665 μs [39]. This is achieved by channel decoders employing internal parallelism, which comprise different processors that collaborate on the processing of each block. Using this approach, an information throughput of $T = 20$ Gbps for data blocks comprising $K = 10,000$ bits implies a processing latency of $L = K/T = 0.5\mu$s, which meets the requirement described above. Note that while a high throughput can be achieved by using separate channel decoders to process multiple blocks at the same time, this approach does not improve the processing latency beyond that of each individual channel decoder. In other words, this relationship only holds under the assumption that no parallel processing of multiple frames is carried out at the same time, no pipelined operation of successive decoding steps is employed

2.2.3 Error correction capability

The strictest error correction capability requirements are imposed in the URLLC applications of 3GPP 5G NR. The target for these URLLC applications is for only 1 block in every 100,000 to suffer from excessive transmission errors that cannot be corrected by the channel decoder, when the transmission throughput approaches the capacity of the communication link between the user equipment and basestation. More specifically, the

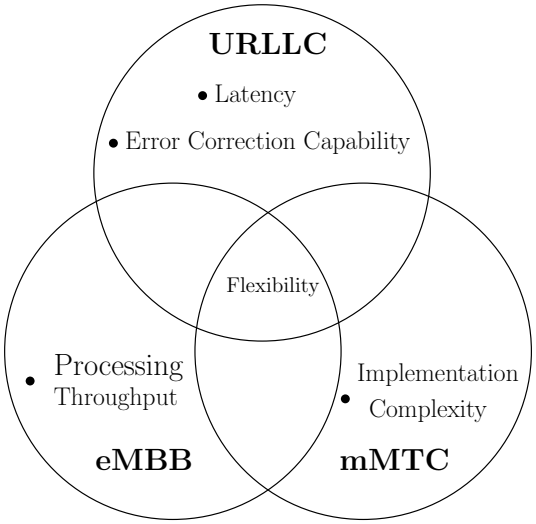


FIGURE 2.1: Summary of requirements for 5G based on different scenario

URLLC channel code must not exhibit an error floor above a Block Error Rate (BLER) of 10^{-5} [38]. This represents an order of magnitude improvement upon that of the 4G LTE turbo code, which has to guarantee a BLER below 10^{-4} [40]. This additional improved error correction requirement further reinforces the improved latency requirement of Section 2.2.2, since it makes it ten times less likely that the receiver will have to activate HARQ [15, 41] to request a retransmission of erroneously decoded information, which imposes a significant additional latency. Despite this however, HARQ will remain a vital component of 5G, in order to enable ultra-reliable communication.

2.2.4 Flexibility

The summary of requirements for 5G NR channel decoder has been illustrated in Figure 2.1. The design of channel decoder has higher priority for variety characteristics based on different scenarios. Unlike other specifications, flexibility is necessary among three main dimensions of 5G NR, owing to the wide range of use cases targeted by each of eMBB, URLLC and mMTC. Owing to this, the channel code must support a wide variety of information block lengths K, as well as a wide variety of coding rates $R = K/N$. For example, short blocks comprising as few as tens or hundreds of information bits may be expected to be typical in URLLC, mMTC and control applications, while long blocks comprising as many as thousands of bits are typical in eMBB applications.

Likewise low coding rates will be required in rural areas, where basestations are deployed sparsely for covering large cells, while high coding rates may be used for ultra-dense urban deployments having strong Line-of-Sight (LOS). If the channel code does not support a wide variety of information block lengths K, then each information block may

have to be padded with a high number of wasteful dummy bits for ensuring that its length becomes one of those legitimate ones supported by the channel code. Likewise, if the channel code does not support a wide variety of coding rates R, then it may be necessary to select a rate that is lower than it is actually required by the current level of noise, interference and fading. Again, that implies the transmission of a high number of wasteful encoded bits. In both cases, the wasteful bits translate into wasted bandwidth. More specifically, the waste results in each transmission having a higher bandwidth, duration or energy than it is actually required, preventing other users from communicating at the same frequency, time or location, without suffering from increased interference. This will therefore degrade the throughput, latency and error correction capability that can be offered by the 3GPP 5G NR. For this reason, the 5G flexibility requirement is key to unconditionally fulfilling the other challenging 5G requirements of Sections 2.2.1 to Section 2.2.3.

2.2.5 Implementation complexity

The implementation complexity of a channel code determines its hardware resource requirement and energy consumption. The strictest implementation complexity requirements are imposed by the mMTC applications of 3GPP 5G NR. In these mMTC applications, low-cost IoT devices have to operate continuously without recharging, while requiring very low chip area and energy consumption. In the uplink, these requirements are imposed upon the channel encoder, while they are imposed on the decoder in the downlink. In eMBB and URLLC applications, the hardware and energy efficiencies of the channel encoder and decoder have to be at least as good as those of 4G LTE. Here, the hardware (or area) efficiency quantifies a channel encoder's or decoder's information throughput as a ratio to its ASIC area, which is measured in Mbps/mm^2. Meanwhile the energy efficiency quantifies the number of information bits that may be decoded per nJ of energy consumed, which is equivalent to Mbps/mW.

2.3 Background on the candidate channel codes

This section provides background discussions of the three mainstream channel codes considered in this chapter, namely the turbo, LDPC and polar codes. In particular, we will detail the decoders of those codes, where the decoders have much higher complexity than the corresponding encoders. The following subsections discuss each code in turn, with reference to the structures illustrated in Figure 2.3.

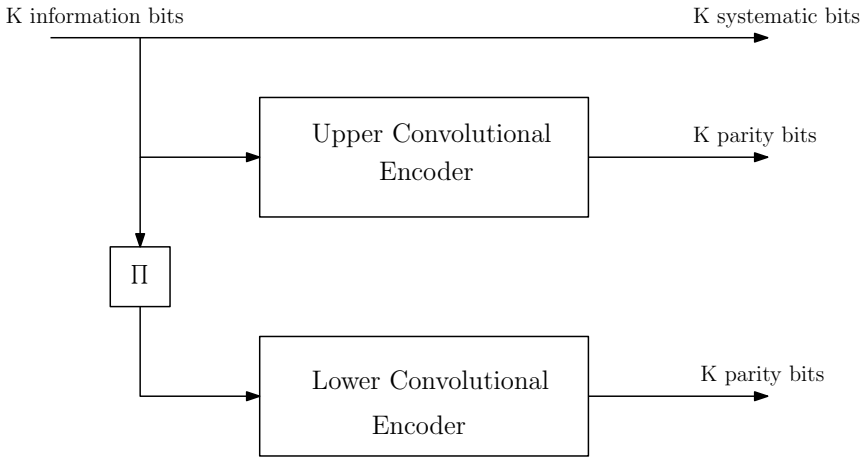


FIGURE 2.2: Turbo encoder schematic.

2.3.1 Turbo codes

Turbo codes may be considered to be very mature in mobile broadband applications, since they were selected to provide flexible channel coding in the 3G UMTS [3] and 4G LTE [4] standards, which have been widely adopted world wide. Furthermore, turbo codes have been selected to provide channel coding in the 4G Narrow Band Internet of Things (NB-IoT) standard [42] for machine type communications.

A turbo encoder [43] comprises a parallel concatenation of two convolutional encoders [44], which are separated by an interleaver. The interleaver creates a replica of the K-bit information block, but rearranges the order of its bits according to a predetermined interleaving pattern. Following this, the two convolutional encoders encode the pair of differently ordered information block and each generates a K-bit parity block, as shown in Figure 2.2. These are typically concatenated together with a third replica of the K-bit information block, which is referred to as the systematic block. Puncturing and repetition [45] may also be used to remove or introduce additional bits for obtaining the resultant N-bit encoded block. The N-bit encoded block is then modulated and transmitted over the wireless channel, where it is exposed to noise interference and fading. Owing to the uncertainty introduced by these effects, the demodulator will typically be unable to express absolute confidence in the values of the N bits in the encoded block. Instead, the demodulator may express its confidence in the value of each bit using the corresponding Log Likelihood Ratio (LLR) [46]. Each LLR provides the logarithm of the ratio of the probabilities of the corresponding bit having the value 0 and 1. This approach enables soft turbo decoding, which iteratively exploits all received information, despite the uncertainty introduced by the wireless channel.

More specifically, the N LLRs of the encoded block may be decomposed into a systematic block and two parity blocks, each comprising K LLRs. These may be entered into a pair of convolutional decoders, which operate on the basis of the Maximum A-Posteriori (MAP) algorithm, also known as the the Logarithmic Bahl-Cocke-Jelinek-Raviv (Log-BCJR) algorithm [47]. This employs a trellis diagram [44] for describing the relationships between the encoded bits, which may be recursed in forward and backward directions [48] to obtain a block of K extrinsic LLRs pertaining to the information block. These K extrinsic LLRs are obtained by combining the received LLRs using both addition and Jacobian logarithm [47] operations, which are also known as the max* operation. The max* operation used for combining two LLRs x and y is given by

$$\max^*(x, y) = \max(x, y) + \ln(1 + e^{-|y-x|}), \tag{2.1}$$

which is simplified as $\max^*(x, y) \approx \max(x, y)$ in the so-called max-log-MAP algorithm [47]. The order of these K extrinsic LLRs may be rearranged by the interleaver and entered into the other convolutional decoder as a block of K *a priori* LLRs. This supports the operation of the other convolutional decoder, enabling it to generate its own block of K extrinsic LLRs pertaining to the information block. In return, these extrinsic LLRs can be reordered by the interleaver and forwarded as *a priori* LLRs to the first convolutional decoder. These additional *a priori* LLRs may be exploited during a second decoding attempt of the first convolutional decoder, in order to improve its operation. This process may continue, with the two convolutional decoders iteratively exchanging successively ever-higher-quality extrinsic LLRs, until the process converges to a legitimate codeword. The LLRs pertaining to the information block may then be converted into the most likely bit values and output [21].

As shown in Figure 2.3(a), turbo codes may be considered to employ a regular structure, comprising two rows of K trellis stages corresponding to the two convolutional decoders, connected by the interleaver. In Figure 2.3(a), the vertical exchange of information between the trellis stages via the interleaver corresponds to the iterative exchange of extrinsic LLRs, while the horizontal exchange of information along the two rows of trellis stages corresponds to the forward and backward recursions of the Log-BCJR algorithm. Since every trellis stage is identical, the turbo code equally protects the K bits in each information block. The complexity of the turbo code structure scales with K, since the interconnections of the interleaver pattern may be described by a $1 \times K$ vector. Owing to the regularity of this structure, turbo codes can be readily designed to reuse hardware to flexibly support a wide range of information block lengths K. However, the forward and backward recursions within the Log-BCJR algorithm impose data dependencies, which many can limit the degree of parallel processing in practical implementations. This impediment has motivated substantial research efforts, which

TABLE 2.1: Decoding schedule for one iteration of fully serial turbo decoder.

Clock cycle	1^{st}	2^{nd}	3^{rd}	...	K^{th}	$K+1^{th}$	...	$2K^{th}$
Processor 1	U_1	U_2	U_3	...	U_K	U_{K-1}	...	U_1
Clock cycle	$2K+1^{th}$	$2K+2^{th}$	$2K+3^{th}$	...	$3K^{th}$	$3K+1^{th}$	...	$4K^{th}$
Processor 1	L_1	L_2	L_3	...	L_K	L_{K-1}	...	L_1

TABLE 2.2: Decoding schedule for one iteration of fully parallel turbo decoder.

Processor	1^{st}	2^{nd}	3^{rd}	...	K^{th}	$K+1^{th}$	...	$2K^{th}$
Clock cycle 1	U_1	U_2	U_3	...	U_K	L_1	...	L_K

have mitigated the data dependencies and increased the degree of parallel processing using the techniques such as Non-Sliding Window (NSW) [49], radix-4 [49] and Fully-Parallel Turbo Decoder (FPTD) [50]. The impact of the turbo decoder architecture on the error correction capability, throughput, area efficiency and energy efficiency will be characterised in the scatter plots of Section IV. Furthermore, it is typically necessary to define a different interleaver pattern for each supported information block length K. In particular, the 4G LTE turbo code employs a Quadratic Polynomial Permutation (QPP) construction technique for generating 188 different interleaver patterns for information block lengths K in the range of 40 to 6144 bits [51].

In order to contrast the two extreme architectures of fully-serial and fully-parallel turbo decoding, their corresponding decoding schedules are illustrated in Tables 2.1 and 2.2. As shown in Table 2.1, the fully-serial turbo decoder uses a single processor operating over K clock cycles to complete each of the forward and backward recursions on each of the upper set of trellis stages and lower set of trellis stages. In total, each decoding iteration of the fully serial turbo decoder requires $4 \times K$ clock cycles, owing to the data dependencies imposed by the Log-BCJR algorithm. However, these data dependencies are ignored in the fully-parallel turbo decoder of Table 2.2, where a separate processor is used for each individual algorithmic block, enabling the forward and backward recursions of both upper and lower sets of trellis stages to be completed at the same time. Therefore, one iteration of fully-parallel turbo decoding can be finished in a single clock cycle, using a total of $2 \times K$ processors.

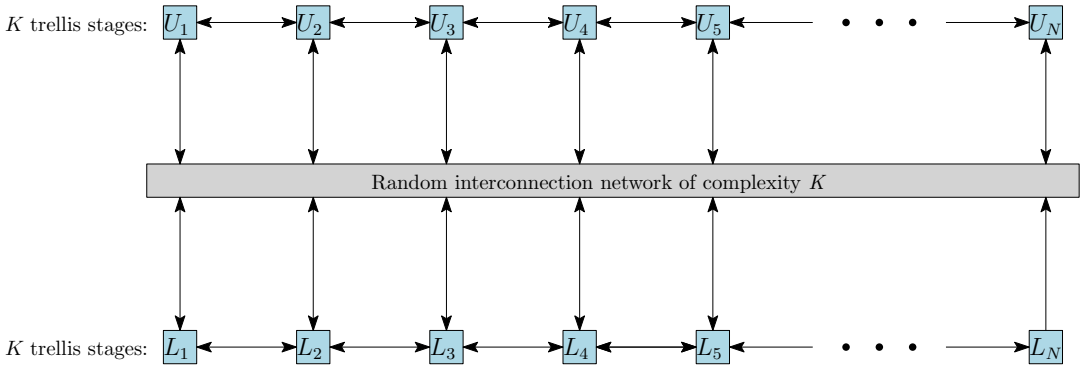
The examples of Tables 2.1 and 2.2 illustrate the contrast between serial and parallel turbo decoders. The serial schedule efficiently propagates information along the forward and backward recursions, enabling a particular error correction capability to be achieved with fewer iterations and lower complexity than the parallel schedule. Despite this, the parallel schedule requires significantly fewer clock cycles overall, leading to superior throughput and latency. The Area and energy requirement of ASIC implementations of turbo decoders tends to be dominated by the capacity of the memory, which is the same for both the serial and parallel schedules. Owing to this, the high throughput of

the parallel schedule typically leads to superior area and energy efficiency. However, the reconfiguration flexibility typically remains limited when implementing a parallel schedule due to the requirement for a fully laid out interconnection network that supports all interleaver designs, as will be detailed in Section IV.

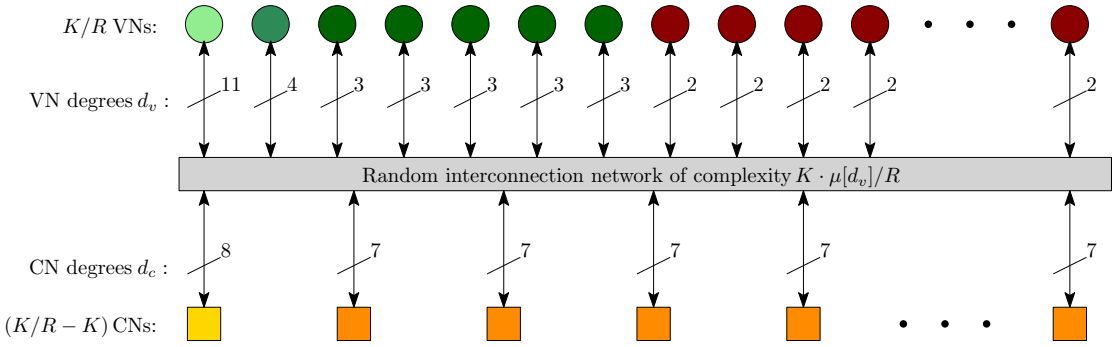
Since the turbo code structure scales with the information block length K, the turbo decoder can be said to decode the information bits directly, rather than recovering the encoded bits and then extracting the information bits. Owing to this, the decoding complexity depends mainly on the information block length K and hence it does not vary significantly when the coding rate R is varied using puncturing and repetition, as shown in Figure 2.5. More specifically, the computational complexity C in terms of Add Compare Select (ACS) operations can be seen to be linearly proportional to the information block length K in Figure 2.6. In other words, the complexity of a turbo decoder is on the order of $C = O(K)$. In this way, the information throughput, and hence the decoding latency, hardware efficiency and energy efficiency do not vary significantly with the coding rate R, as shown in Figure 2.4a. This is illustrated by the analogy of Figure 2.4b, which shows that the information throughput of a turbo decoder remains constant, when the coding rate R is adjusted by varying the puncturing or repetition.

2.3.2 LDPC codes

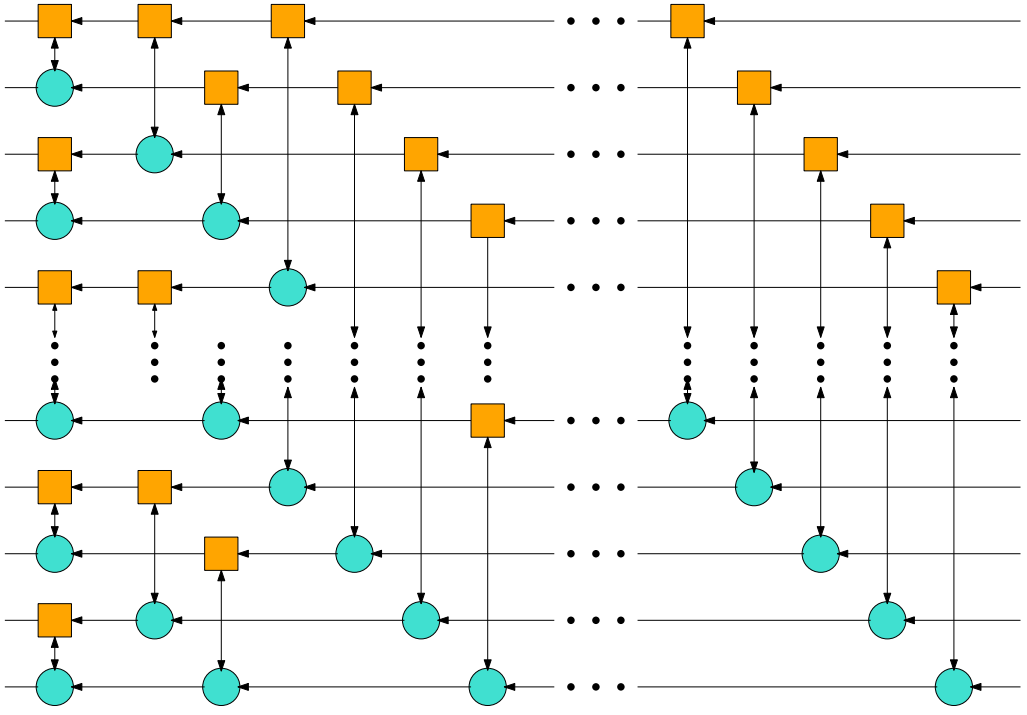
LDPC codes belong to an even more mature channel coding family than turbo codes, since they were proposed by Gallager in his PhD thesis half-a-century ago! They have also been selected for a variety of standards conceived for diverse applications. Perhaps most famously, LDPC coding is employed in WiFi [6], which is designed for wireless local area networking offering support for 12 combinations of medium information block lengths K and high coding rates R. The WiMAX standard [7] designed for mobile broadband communication employs an LDPC code that supports a similar range of information block lengths K and coding rates R, but with 144 options within that range. In the WiGig standard [8] ratified for Millimeter Wave (mmWave) communication, an LDPC code supporting 4 combinations of medium information block lengths K and high coding rates R is employed. The DVB-S2 standard [9] for satellite communication employs an LDPC code, which supports 29 combinations for high information block lengths K and medium to high coding rates R. In the Ethernet 10GBASE-T standard [10], an LDPC code supporting only a single fixed medium information block length K and a high coding rate R is employed for facilitating a high processing throughput. The impact of these different standards [7–10] upon the error correction capability, throughput, area efficiency and energy efficiency, will be characterized in the scatter plots of Section IV.



(a) Turbo decoder structure



(b) LDPC decoder structure



(c) Polar decoder structure

FIGURE 2.3: Structures of turbo, LDPC and polar decoders.

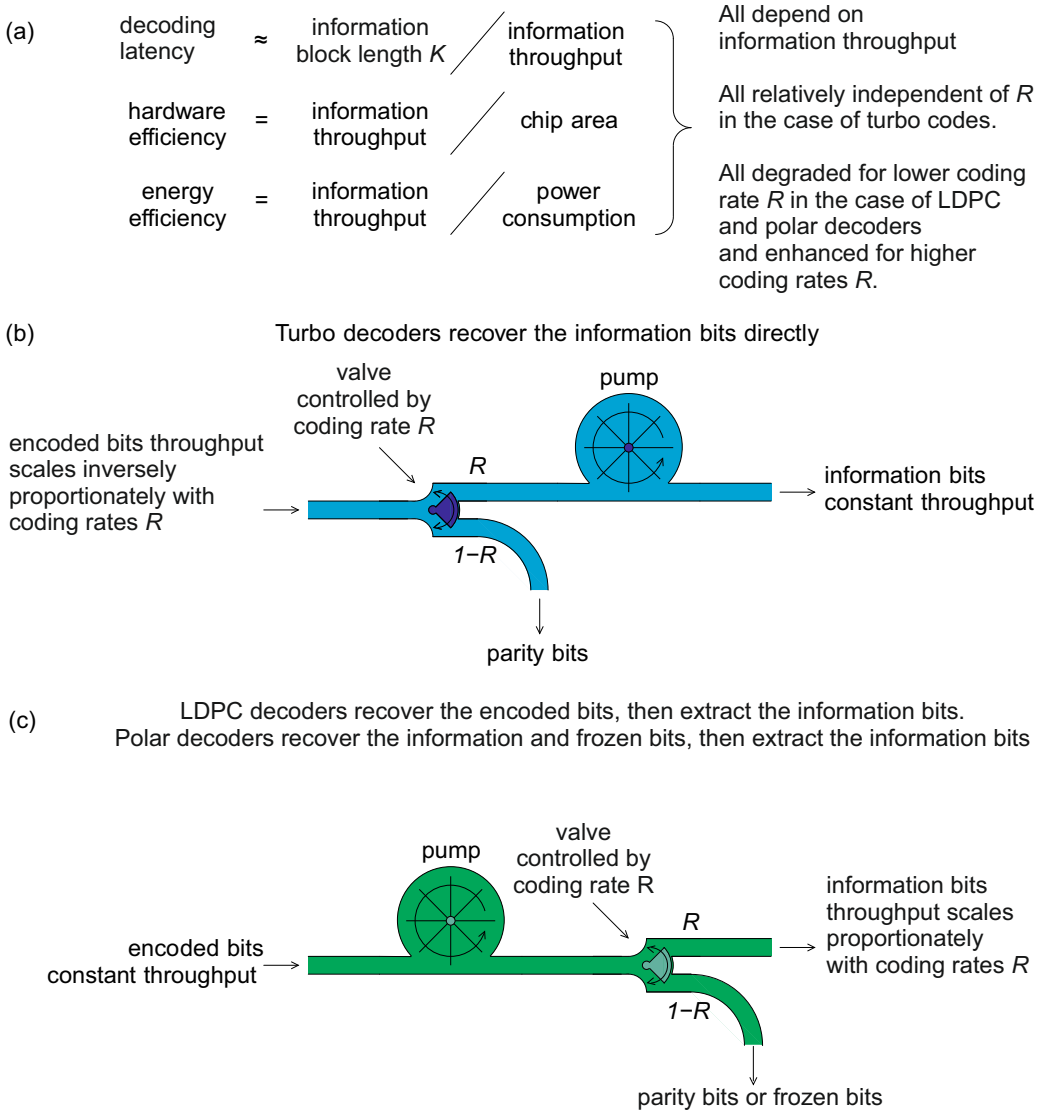


FIGURE 2.4: (a) The relationship between information throughput and decoding latency, hardware efficiency and energy efficiency, as well as an analogy using pumps, valves and pipes, to illustrate how the coding rate R of (b) turbo and (c) LDPC or polar decoders affects their encoded and information throughputs.

An LDPC encoder obtains an N-bit encoded block $\mathbf{U}$ by multiplying a block of K information bits $\mathbf{x}$ by a $(K \times N)$ generator matrix $\mathbf{G}$ in GF(2), according to $\mathbf{U} = \mathbf{G} \cdot \mathbf{x}$. In contrast to a turbo decoder, an LDPC decoder employs an irregular factor graph structure [13], which applies unequal protection to the K bits in each information block. More specifically, N LLRs pertaining to the encoded block are forwarded to a corresponding set of N VNs, which are connected to $(N - K)$ CNs. In detail, the V^{th} VN is connected to d_v number of CNs, where d_v is referred to as the degree of the VN. Similarly, the C^{th} CN is connected to d_c number of VNs, where d_c is referred to as the degree of the CN. In a similar manner to a turbo decoder, an iterative sum product decoding process [13] is used for exchanging extrinsic LLRs between the connected VNs and CNs. More specifically, the extrinsic LLR provided by a VN for each of its connected CNs is obtained as the sum of the LLRs provided by all other connected CNs plus the LLR provided by the channel. Meanwhile, the extrinsic LLR provided by a CN for each of its connected VNs is obtained by the so-called box-plus summation of the LLRs provided by all other connected VNs. To elaborate a little further, the box-plus sum of two LLRs x and y is given by

$$\begin{aligned} \min^*(x, y) =& \text{sign}(x) \cdot \text{sign}(y) \cdot \min(\mid x \mid, \mid y \mid) \\ &+ \ln(1 + e^{-|x+y|}) - \ln(1 + e^{-|x-y|}) \end{aligned} \tag{2.2}$$

which is simplified to

$$\min^*(x, y) \approx \text{sign}(x) \cdot \text{sign}(y) \cdot \min(\mid x \mid, \mid y \mid) \tag{2.3}$$

in the so-called min-sum algorithm [23].

The complexity of this factor graph structure scales with the encoded block length N, since the interconnections between the VNs and CNs in the LDPC decoder may be described by a $(N - K) \times N$ matrix, which is referred to as the PCM [13]. Note that the dimensions of the PCM grow as the coding rate R is reduced.

The computations performed within each LDPC VN and CN typically impose a significantly lower complexity than the computations performed for each trellis stage of a turbo code, hence typically leading to a lower LDPC decoder complexity, particularly at high coding rates. However, while a turbo decoder has K random interconnections in its interleaver, an LDPC factor graph has $K \cdot \mu[d_v]/R$ random interconnections, where $\mu[d_v]$ is the average number of CNs that each VN is connected to. Depending on the design of the LDPC code, an LDPC decoder may have several times more random interconnections than an equivalent turbo decoder, particularly at lower coding rates. For example, in the case of the WiFi LDPC codes, where we have mean$[d_v] = 3.5$ [12], the $R = 1/2$ interconnection complexity is 7 times higher than that of a turbo decoder having the

same information block length K. Since different components of an LDPC decoder have different irregular numbers of connections, it can be a significant challenge to implement flexible LDPC decoders that support various block lengths K and coding rates R, at a high throughput, as it will be demonstrated in Section 3.2.3. This flexibility is typically achieved by defining a different PCM for each supported combination of information block length K and coding rate R, when requiring the decoder to support a wide variety of PCMs.

However, in contrast to turbo decoders, LDPC decoders only have connectivity through the interleaver. More specifically, LDPC decoders do not suffer from the data dependencies that are imposed by the serial nature of the forward and backward recursions used by the Log-BCJR algorithm during turbo decoding. Owing to this, the designers of LDPC decoders are free to implement parallel processing to a wide variety of degrees and using a wide variety of techniques, including block parallel [52], row parallel [53] and fully parallel [54] arrangements. In analogy to the fully-serial and fully-parallel turbo decoding of Tables 2.1 and 2.2, LDPC decoding can be completed using the more-serial layered belief propagation schedule [55] or the extreme of the fully-parallel flooding schedule [54]. As in turbo decoding, a higher degree of parallelism leads to higher throughput, area and energy efficiency, but degraded error correction performance and flexibility, as it will be discussed in Section IV.

Furthermore, as shown in Figure 2.5, the complexity, throughput, latency, hardware efficiency and energy efficiency of LDPC decoders are degraded at low coding rates R, owing to two fundamental reasons. Firstly, the number of rows in an LDPC PCM grows as the coding rate R is reduced, which implies having a higher number of CNs requiring computation. Secondly, the number of columns in the PCM and the number of VNs in the factor graph is given by the encoded block length N, which dictates the input and output interface to the LDPC decoder, as shown in Figure 2.3(b). Since LDPC decoders must recover $1/R$ encoded bits in order to decode each information bit, their information throughputs scale down proportionately with the coding rate R, as illustrated by the analogy of Figure 2.4a. This analogy shows that the information throughput of an LDPC decoder varies when the coding rate is adjusted, since this controls the specific fraction of the N recovered bits that correspond to information bits. As the coding rate of an LDPC is increased, the dimensions of its PCM are reduced and its complexity C is decreased, as shown in Figure 2.5. However, when the coding rate R is fixed, increasing the information block length K will increase the dimensions of the PCM, hence increasing the computational complexity of the LDPC decoder, as shown in Figure 2.6. In general terms, it may be said that the complexity of an LDPC decoder is on the order of $C = O(K/R)$.

2.3.3 Polar codes

While turbo and LDPC codes have been researched and developed over the last 20 years, polar codes [11] were not proposed until 2009. Owing to this, polar codes were not adopted in any standards or consumer devices before the control channel of 3GPP NR eMBB. This limited their scope to proof of concept demonstrators and academic publications.

During polar encoding, the K information bits are interleaved with $(M-K)$ frozen bits, which have a fixed value of zero. The frozen bits are positioned according to a prescribed bit pattern, which should be optimised for each supported combination of K and R. Here, the number of frozen bits $(M-K)$ should be chosen for ensuring that the number of bits in the resultant bit sequence is a power of two, yielding $M = 2^{\lceil \log_2(N) \rceil}$, where $\lceil \cdot \rceil$ is the ceiling function. These bits are combined using XOR operations according to a Kronecker matrix [11], in analogy to Figure 2.3(c), which illustrates the regular recursive structure of a polar code. This process may be represented by the multiplication of the information bit vector $\mathbf{x}$ by a generator matrix $\mathbf{G}$ in GF(2), obtaining the encoded bit vector as $\mathbf{U} = \mathbf{G} \cdot \mathbf{x}$. For example, in the case of $M = 8$, the corresponding Kronecker matrix is given by [1]

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

Following this, $(M-N)$ of the resultant bits are punctured or shortened, in order to reduce the length of the resultant encoded block to N bits. Alternatively, repetition may be used for increasing N above M in cases, where N is slightly higher than a power of two. A polar decoder comprises a structured graph of VNs and CNs, all having a degree of no more than 3, as shown in Figure 2.3(c). At the start of the decoding process, the N LLRs provided by the demodulator are depunctured and forwarded to the inputs on the right of Figure 2.3(c). During the decoding process, the LLRs are propagated through the check nodes from right to the left of Figure 2.3(c). Here, each check node combines a pair of LLRs using the box-plus operation of Equation (2.2) or its min-sum approximation as shown in Equation (2.3). After the propagation of LLRs, hard decisions are made for each of the information bits on the left of Figure 2.3(c) in the order commencing from

[1] Given the inclusion of frozen bits, the matrix becomes an $[M \times M]$-element matrix, rather than $[N \times K]$ matrix.

top to bottom. More specifically, these hard decisions are propagated from left to right, to the VNs, allowing them to pass LLRs to the CNs to propagate from right to left. Here, each VN either adds or subtracts an LLR from another, depending on the value of the bit propagated to it. In this way, each successive hard decision enables further hard decisions for subsequent bits. In order to guard against erroneous hard decisions affecting all subsequent decisions, a list of the L best hard decision based sequences can be maintained during the so-called Successive Cancellation List (SCL) based decoding process [56] where the best candidate is then output at the end of the process. Following this, the $(M-K)$ frozen bits can be removed and the K recovered information bits can be output. Note that in the special case of $L = 1$, the SCL algorithm becomes the Successive Cancellation (SC) algorithm. In analogy to the fully-serial and fully-parallel turbo decoding schedules of Tables 2.1 and 2.2, polar decoding can be completed using the more-serial SC algorithm and its variants [57], or the fully-parallel belief propagation algorithm [58]. As in turbo and LDPC decoding, a higher degree of parallelism leads to a higher throughput, area and energy efficiency, at the cost of a degraded error correction performance and flexibility, as it will be discussed in Section IV.

Owing to the low degrees of its VNs and CNs, the computational complexity of a polar decoder is relatively low compared to that of an LDPC decoder, as determined by the list size L, which scales the complexity linearly, as shown in Figure 2.5. Like LDPC codes, the structure of polar codes scales with the encoded block length N, rather than with the information block length K as in turbo codes. Owing to this, the complexity and encoded throughput of a polar decoder typically remains constant, when the encoded block length N is kept constant, but the information block length K is changed. However, since $1/R$ encoded bits must be decoded in order to recover each information bit, the information throughput typically scales proportionately with the coding rate R, as illustrated by the stylized analogy of Figure 2.4c. Owing to this, as shown in Figure 2.5, the complexity, hardware efficiency and latency of polar decoders are degraded for lower coding rates and improved for higher coding rates, following a similar trend to that of LDPC decoders. On the same note, similarly to LDPC codes, the computational complexity of a polar code grows with the encoded block length N, as shown in Figure 2.5 and 2.6. More specifically, the dimensions of the polar code graph and hence the complexity grows according to the order of $C = O\left(\frac{K}{R}\log\frac{K}{R}\right)$. Note however that a simplified successive-cancellation list based algorithm [59] may be used for reducing the computational complexity of polar decoders having low coding rates, for example. More specifically, this technique reduces the complexity of processing the frozen bits, which become more prevalent at low coding rates R. Note however that fast simplified successive-cancellation list based algorithm [59] may be used for reducing the computational complexity of polar decoders having low coding rates, for example. More specifically, this technique reduces the

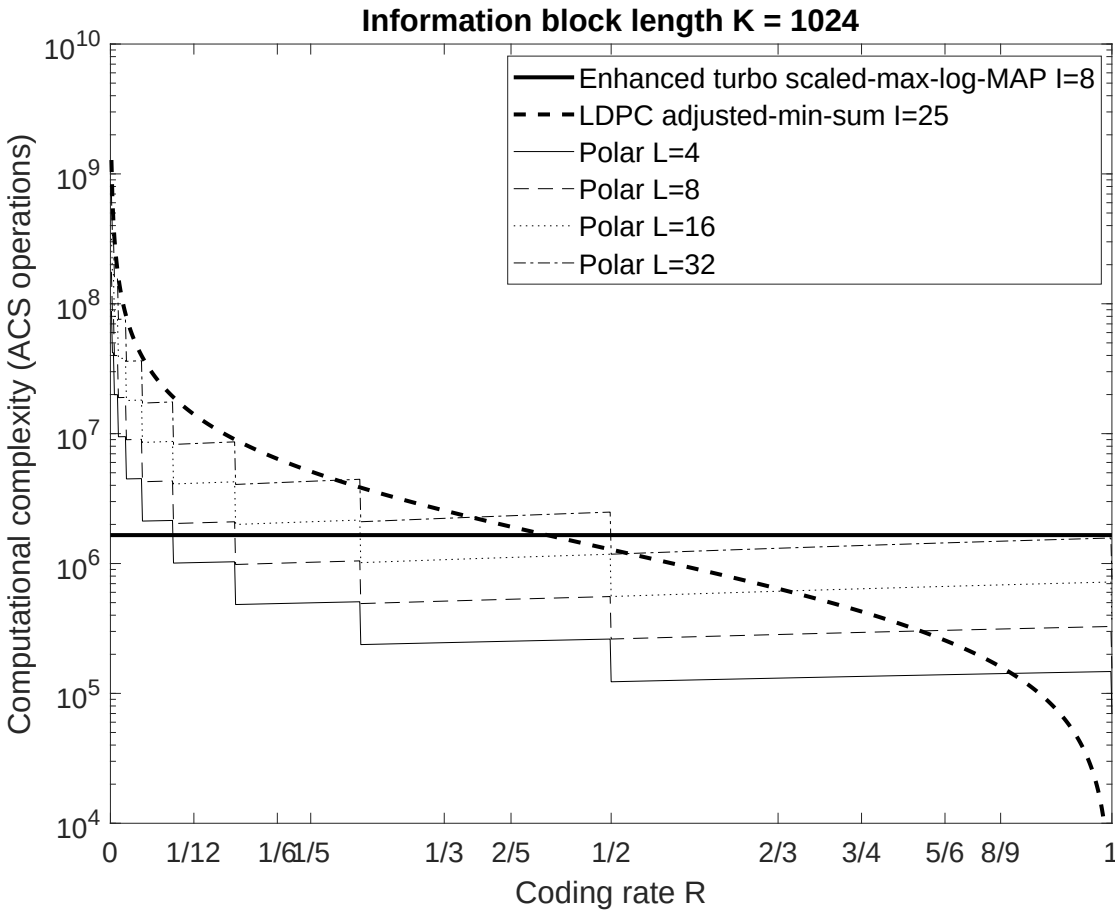


FIGURE 2.5: Computational complexity versus coding rate R for various enhanced turbo, LDPC and polar decoders having information block lengths of K = 1024. Here the enhanced turbo code of [60] employs I = 8 decoding iterations, the adjusted min-sum LDPC decoder of [61] employs I = 25 iterations and the polar decoder of [62] employs list sizes of $L \in \{4, 8, 16, 32\}$.

complexity of processing the frozen bits, which become more prevalent at low coding rates R.

2.4 Conclusion

In this chapter, we introduced the role of channel coding in the modern wireless system. In the 5G NR wireless communication system, channel codes are required to meet variant requirements to ensure their capability on the physical layer. In order to give a comprehensive understanding to audiences, we described the 5G requirements in this chapter. Following this, the algorithm of both encoding and decoding sides of three main channel coding candidates, including turbo, LDPC and polar codes are presented. We found out that for the long code block length, turbo and LDPC codes have more

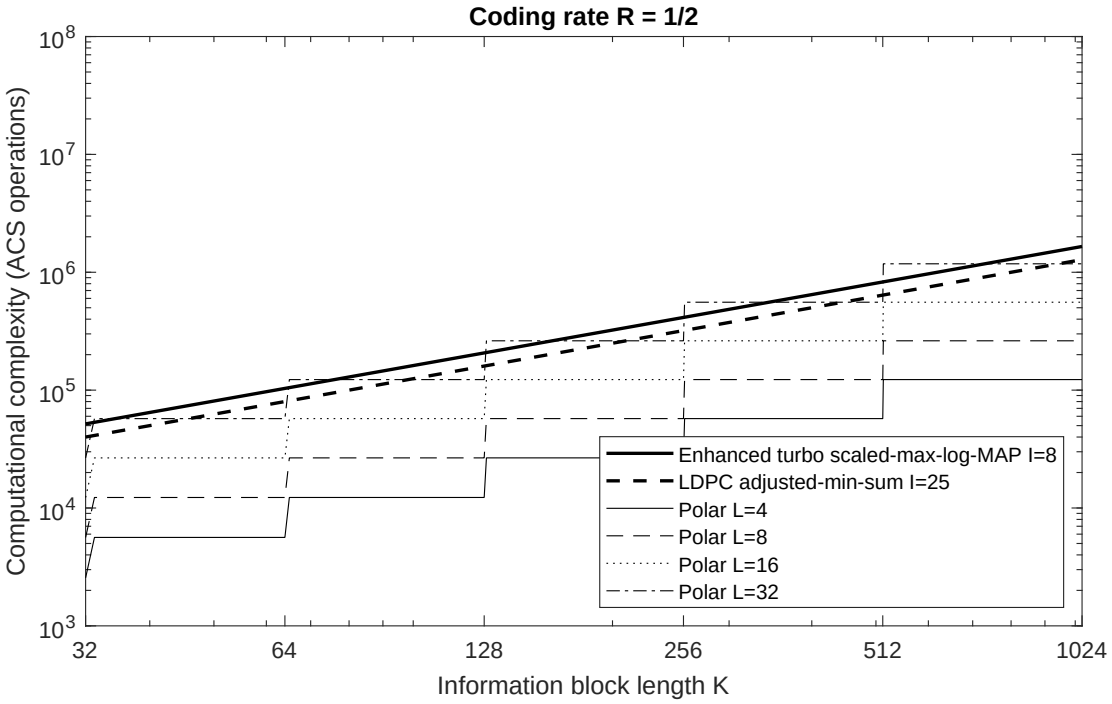


FIGURE 2.6: Computational complexity versus block length K for various enhanced turbo, LDPC and polar decoders having coding rate R = 1/2. Here the enhanced turbo code of [60] employs I = 8 decoding iterations, the adjusted min-sum LDPC decoder of [61] employs I = 25 iterations and the polar decoder of [62] employs list sizes of $L \in \{4, 8, 16, 32\}$.

suitable working parameters than polar codes. By contrast, polar code is usually applied for low code block length. According to their coding algorithm, all these three channel codes are highly flexible on the change of coding rate, which can be achieved by puncturing, shortening and repetition techniques. In order to check the qualification of other requirements, we exploited a survey among these three channel codes on their ASIC implementations to check the throughput and complexity specifications. This survey is presented in Chapter 2 to show the difference between variant channel codes ASIC implementations.

Chapter 3

Survey of Turbo, LDPC and Polar Decoder ASIC Implementations

3.1 Introduction

As we described in Chapter 2, turbo, LDPC, and polar codes are used to be the controversial candidates of the channel coding technique of the 5G NR wireless communication standard. This survey is motivated by this debate and the demand of the 5G NR wireless communication. Our survey summarises several factors that influences the debate, with a particular focus on the ASIC implementation of the decoders of these three codes. Note that in general LLRs are represented by fixed-points numbers, which comprised by short number of bits to provide its precision [63]. Therefore, the computational complexity of channel decoder's implementation are able to be evaluated by simple ACS operations. We show that the overall implementation complexity of turbo, LDPC and polar decoders depends on numerous other factors beyond their computational complexity. More specifically, we compare the throughput, error correction capability, flexibility, area efficiency and energy efficiency of ASIC implementations drawn from 110 papers. Note that due to the lack of information from published papers, our survey is not capable of analysing the latency of various implementations. However, we made our best effort by using other results for characterising the advantages and disadvantages of these three codes as well as for avoiding pitfalls and for providing design guidelines. Although the channel coding technique of 5G NR wireless communication has been determined by 3GPP group as LDPC codes adopted for data transmission and polar codes adopted

for control channel, this survey still gives some new observations and recommendations on the ASIC implementations.

Based on the above introduction, the rest of this chapter will be organized as follows. In Section 3.2, the survey of turbo, LDPC and polar codes is presented as several comparisons in different areas. The processing throughput, error correction capability and flexibility of 110 published ASIC based turbo, LDPC and polar implementations will be characterised in Section 3.2 as a function of their area- and energy-efficiency. Based on the survey, Section 3.3 outcomes some design recommendations for researchers and manufacturers. And finally, a conclusion of this chapter will be summarised in Section 3.4.

3.2 Comparison of turbo, LDPC and polar decoders

This section provides a comprehensive comparison of 100+ state-of-the-art ASIC implementations of turbo, LDPC and polar decoders. The key performance characteristics of a channel decoder are its processing latency, information throughput, error correction capability, flexibility, area efficiency and energy efficiency, as illustrated in Figure 3.1. When employed in a particular application, certain minimum performance requirements are imposed upon the information throughput (and on the resultant processing latency), error correction capability and flexibility of the channel decoder ASIC implementations. Within the constraints imposed by meeting these requirements, the design of a channel decoder ASIC should focus on optimising the area- and energy- efficiency, since they directly determine the implementation and running cost or the battery lifetime of base-stations and user equipment. Motivated by this, the following subsections consider the information throughput, error correction capability and flexibility of the ASIC implementations in turn, each as functions of the area- and energy-efficiency. Each subsection includes plots, which are derived from the data presented in Table 1 of the Appendix A.

3.2.1 The information throughput vs. area- and energy-efficiency

Figure 3.2 characterises the relationship between the information throughput and area-efficiency of 100+ ASIC implementations of channel decoders, where the shape of the data points distinguishes the turbo, LDPC and polar decoders. Where a channel decoder supports multiple coding rates R and/or block sizes K, the corresponding information throughputs have been averaged in Figure 3.2. Furthermore, in order to present a fair comparison amongst ASICs implemented at different technology scales, the results of Figure 3.2 have all been scaled to 65 nm. Explicitly, 65 nm was chosen, since it is the standard technology scale that is closest to the average of the ASICs considered in this

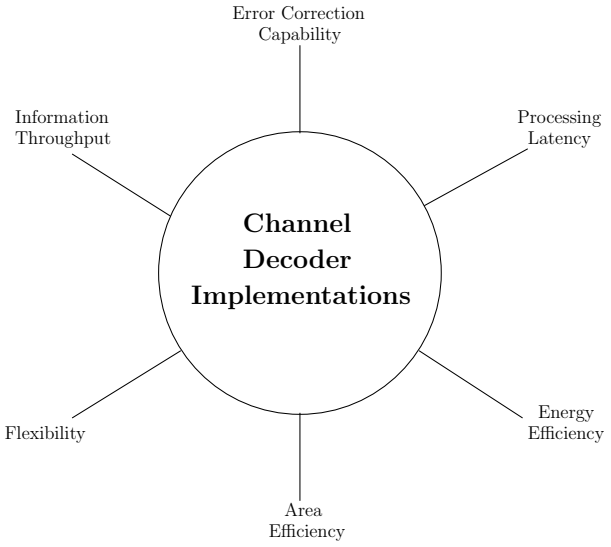


FIGURE 3.1: Key characteristics of ASIC channel decoder implementations

chapter. Here, scaling is applied to the performance characteristics of an ASIC implementation using a scaling factor, which is calculated as the ratio between the ASIC's technology scale and the target technology scale of 65 nm. The scaled information throughput, area and power consumption are obtained by multiplying the ASICs information throughput by the scaling factor that described in [64], the inverse square of the scaling factor and the inverse of the scaling factor, respectively. As shown in [65], the power of ASIC implementations are approximately proportional increase to the increase of the technology nodes. Therefore, we are scaling different technology nodes of investigated ASIC implementation to the same level of technology node, which is 65nm in this case. Owing to this, the area-efficiency and energy-efficiency are proportional to the cube and the square of the scaling factor, respectively.

Figure 3.2 illustrates the difference between the information throughputs achieved by the turbo, LDPC and polar decoders. The considered turbo decoder implementations appear to have the largest portion of decoders associated with low information throughputs, which may be attributed to their low degrees of parallelism imposed by the data dependencies of the Log-BCJR algorithm, as discussed in Section 2.3.1. Likewise, the parallelism of polar decoder implementations is typically limited, owing to the strict data dependencies of the SC and SCL decoding algorithms, as we discussed in Section 2.3.3. Finally, the LDPC decoder implementations have the highest portion of decoders associated with high information throughputs, owing to the high degree of parallelism achieved by the min-sum algorithm, as discussed in Section 2.3.2.

As mentioned in Section 2.2.1, attaining a peak information throughput of 20 Gbps is a key requirement for 5G communication. As shown in Figure 3.2, the only turbo decoder

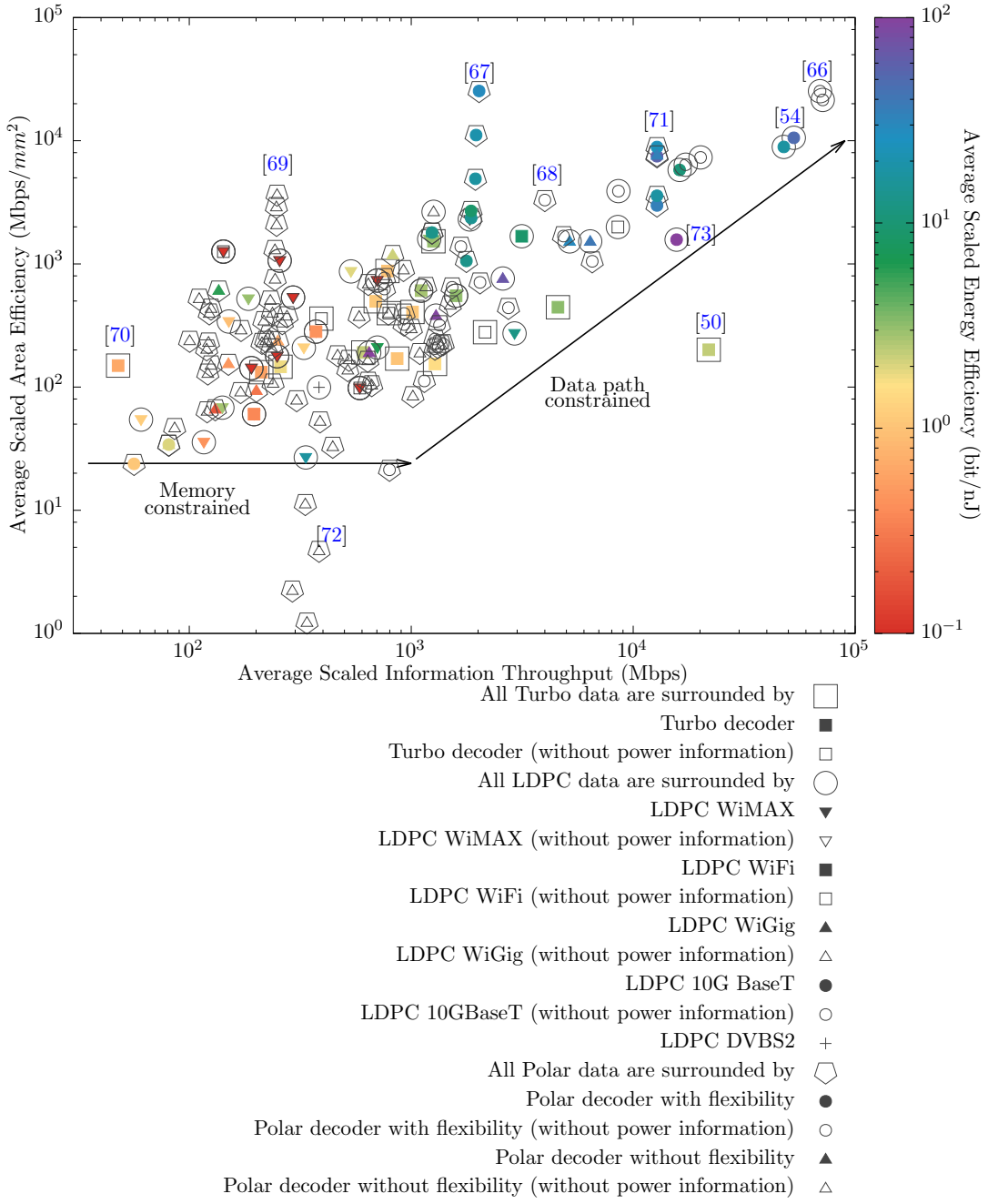


FIGURE 3.2: Area and energy efficiency versus information throughput for various channel decoder implementations.

that achieves the 20 Gbps information throughput is that of [50], which uses a fully-parallel architecture for the turbo decoding. Likewise, the only LDPC decoders that have reached information throughputs in excess of 20 Gbps have adopted fully parallel architectures. In particular, a decoded throughput of T = 52 Gbps and a latency of 0.045 μs have been demonstrated in [54] for the 10GBASE-T LDPC code, which has a block length of $K = 1723$ bits and a coding rate of $R = 0.84$. Likewise, the state-of-the-art

LDPC ASIC implementation of [73] achieves not only a high information throughput, but also an outstanding energy efficiency of 987 bit/nJ. However, the LDPC decoders of [54] and [73] only support this single combination of block length K and coding rate R, which reflects the relatively inflexible nature of LDPC decoders in general and the rigidity of fully-parallel LDPC decoders in particular. The only polar decoders that achieve a decoded throughput in excess of 20 Gbps have managed this by unrolling hundreds of data blocks and pipelining their successive cancellation decoding through the same hardware [71], as indicated in Table 1 in Appendix A. Using this approach, an extremely high throughput of 208 Gbps has been demonstrated for a coding rate of $R = 1/2$, although the latency experienced by each $K = 1024$-bit data block is 3.21 μs. However, this pipelining approach severely limits the flexibility of the polar decoder. Furthermore, the SC technique for polar decoding results in a degraded error correction capability, compared to the more complex SCL decoding technique, as it will be discussed in Section 3.2.2.

However, a high degree of parallel processing implies having a large chip area, and a high cost. For this reason, the key consideration in the selection of the 5G channel code is the area-efficiency, which quantifies the ratio of information throughput to chip area, as seen in Figure 3.2. At first sight, it may be expected that the area efficiency should remain constant upon varying the degree of parallel processing, since this scales both the information throughput and the area.

However, Figure 3.2 shows that decoders having higher information throughputs and degrees of parallel processing tend to have higher area efficiencies. This may be explained by the observation that while the datapath area of a decoder scales linearly with the parallelism, the memory and controller parts tend to increase slower than linearly. Owing to this, decoders having a high information throughput tend to have high area-efficiencies, that are dominated by the particular parts that scale with the grade of parallelism, such as the datapath. By contrast, decoders having a low information throughput tend to have low area-efficiencies that are dominated by the specific parts that do not scale with the grade of parallelism, such as the memory and controller. For similar reasons, the same trend is shown in Figure 3.2 in terms of power-efficiency, with higher information throughputs leading to higher power-efficiency.

However, there are some special cases that do not follow the tendency observed above, such as the fully parallel turbo decoder implementation of [50]. Here, the number of clock cycles required per decoding iteration indeed scales down linearly with the grade of parallelism. However, the number of iterations required by a fully parallel turbo decoder is higher than that of a conventional turbo decoder, when aiming for achieving the same error correction capability. Owing to this, the achievable information throughput does

not scale up linearly with the grade of parallelism, leading to reduced area and power efficiency.

3.2.2 Error correction capability vs. area- and energy-efficiency

As described in Section 2.2.3, there is a 5G requirement to facilitate reliable error correction at channel Signal-to-Noise Ratios (SNRs) per bit (E_b/N_0) that are as close as possible to the channel capacity bound, where a BLER of 10^{-2} is targeted as a complement to HARQ in eMBB applications and 10^{-5} is targeted for URLLC applications. Figure 3.3 characterises the discrepancy between the channel capacity E_b/N_0 bound and the specific E_b/N_0 value, where a Bit Error Rate (BER) of 10^{-4} is obtained for various channel decoder implementations, when using Binary Phase Shift Keying (BPSK) modulation for communication over an Additive White Gaussian Noise (AWGN) channel. This approach is adopted, since nearly all of the papers considered present BER plots, rather than BLER plots, and consider BPSK and AWGN. Figure 3.3 also presents the scaled area-efficiency, while the shape and color of the data points indicate the type of decoder and the scaled energy-efficiency. Note that the legend of Figure 3.3 is inherited from Figure 3.2, which is provided below this figure again for convenience. Note that fewer data points are presented in Figure 3.3 than were presented in Figure 3.2, since many papers do not characterise the error correction capability of their proposed decoder.

Rather than comparing the minimum E_b/N_0 value required for each channel decoder implementation to achieve the target BER, Figure 3.3 compares the discrepancy between this E_b/N_0 value and the channel capacity E_b/N_0 bound, which gives a fairer comparison of the error correction capability of various channel decoder implementations having different coding rates R. In order to illustrate this, Figure 3.4 plots the Discrete-input Continuous-output Memoryless Channel (DCMC) capacity associated with using BPSK for communication over an AWGN channel as a function of E_b/N_0. This is compared with points corresponding to each channel decoder implementation, which plot the spectral efficiency of each decoder versus the minimum E_b/N_0 value that it requires to achieve a BER of 10^{-4}, when using BPSK to communicate over an AWGN channel. If ideal Nyquist pulse shaping having zero excess-bandwidth is employed, the spectral efficiency of a decoder is numerically equal to its effective throughput of, $\eta = R \times \log_2(M)$ [78], where R is the coding rate of the decoder and $M = 2$ is the number of constellation points employed by BPSK, resulting in $\eta = R$. Note that while many of the papers considered present flexible channel decoders that support different coding rates and hence different spectral efficiencies, they typically only provide hardware performance results for a single coding rate and spectral efficiency, as plotted in Figure 3.4.

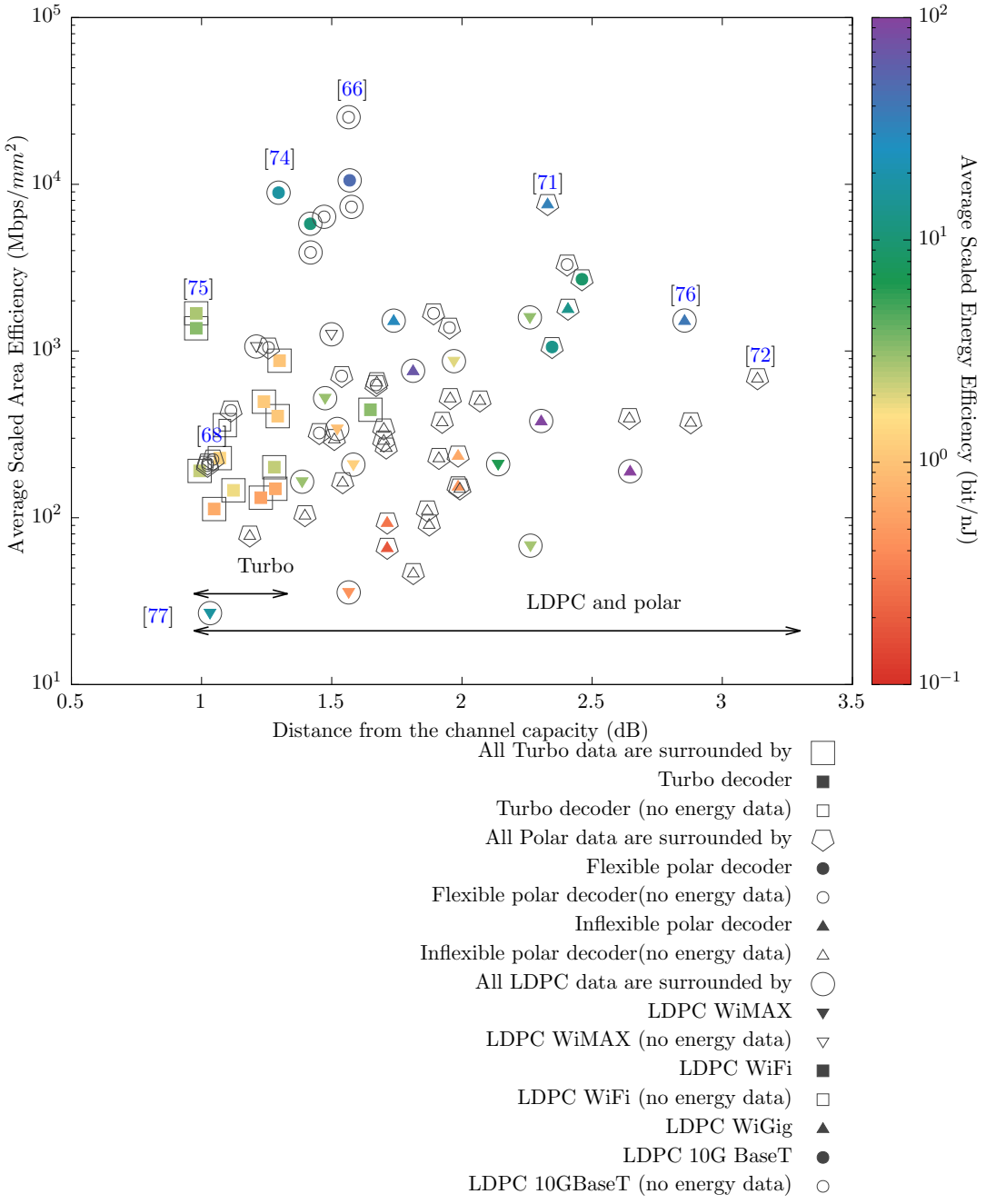


FIGURE 3.3: Area-efficiency ($Mbps/mm^2$) versus the discrepancy between the channel capacity E_b/N_0 bound and the E_b/N_0 value where a BER of 10^{-4} is obtained for various channel decoder implementations.

Figure 3.3 shows that turbo decoders tend to have the best error correction capability, with most of them exhibiting less than 1.5 dB discrepancies from the channel capacity. This may be explained by the long block lengths K, comprising thousands of information bits that are supported in the LTE turbo code. This enables stronger error correction than is possible when using blocks comprising only hundreds of information bits, as

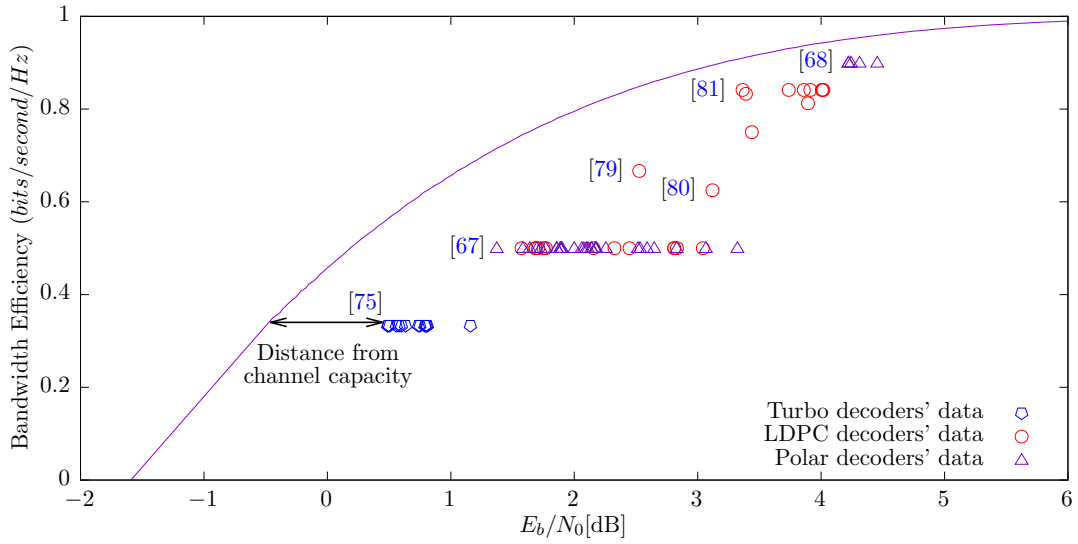


FIGURE 3.4: Bandwidth efficiency versus minimum E_b/N_0 value required to achieve a BER of 10^{-4} for various channel decoder implementations.

is typical for the LDPC and polar decoders considered. Owing to this, most LDPC and polar decoders exhibit more than 1.5 dB discrepancy from channel capacity, which may even exceed 2.5 dB. However, there are some specific LDPC and polar decoder implementations that offer similar error correction capability to turbo decoders. The polar decoder of [57] has a 1.5 dB discrepancy from the channel capacity, as a benefit of using the SCL decoding algorithm with a list size of $L = 4$, rather than the SC algorithm. Outside of the ASIC implementation literature, it has been shown [82] that polar decoders employing large list sizes can achieve superior error correction capability to LDPC and turbo decoders employing large numbers of iterations, particularly at short block lengths K. Furthermore, the LDPC decoder of [74] and several other similar LDPC decoder implementations also exhibit reliable error correction capability as well as extremely high area- and energy-efficiency. These LDPC decoders adopt fully parallel architectures, in order to achieve the high information throughputs supported by the 10GBASE-T Ethernet standard. This allows these decoders to perform a high number of decoding iterations, hence enabling reliable error correction. Note that state-of-the-art LDPC and polar code constructions have been adopted by the 3GPP 5G New Radio standard. It is expected that the error correction performance loss compared to turbo codes will be soon eliminated when implementations of the 5G LDPC and polar codes [12] become available for comparison. Alternatively, the non-binary LDPC ASIC implementation of [77] has largely reduced the error correction performance gap from turbo decoders.

Figure 3.3 suggests that the error correction capability is not directly correlated with the area- and energy-efficiency, as inferred from the various decoders considered here.

However, for a particular decoder, it may be expected that performing more iterations or using a larger SCL list size would improve the error correction capability, albeit at the cost of degrading the information throughput and hence both the area- and energy-efficiency.

3.2.3 Reconfiguration flexibility vs. area- and energy-efficiency

As described in Section 2.2.4, 5G NR requires the channel decoder to support a wide variety of information block lengths K, as well as coding rates $R = K/N$. In this section, we quantify the flexibility of each channel decoder by the number of supported combinations of block length K and coding rate R. Figure 3.5 characterises the flexibility, scaled average area-efficiency and energy-efficiency that is achieved by each turbo, LDPC and polar decoder ASIC considered. In the case of the turbo decoders, the number of information block lengths K supported is determined by the number of interleavers supported, which is 188 for the LTE turbo code. As we mentioned in Section 2.3.1, turbo codes achieve single bit granularly for the encoded block length N using puncturing or repetition, and hence they support a large number of coding rates R. Rather than quantifying the very large number of combinations of K and R supported by the turbo decoders, Figure 3.5 quantifies only the number of supported information block lengths K instead. Hence, the flexibility of the turbo decoders considered is even better than that quantified in Figure 3.5. Note that the 5G LDPC and polar codes [12] flexibly support single-bit granularity for both the information- and encoded-block length, although published implementations are not currently available in the open literature for consideration in this chapter.

By contrast, LDPC codes used in the existing standards require a different PCM for each supported combination of block length K and coding rate R, hence resulting in the relatively low flexibilities observed in Figure 3.5. In principle, polar codes can have single bit granularly of the block length K and coding rate R, using only a single frozen bit selection sequence. However, this high grade of flexibility has not yet been demonstrated in any existing ASIC implementations. Indeed, the majority of polar decoder ASICs in the literature support only a single combination of block length K and coding rate R, as shown in Figure 3.5. This may be because many of the existing polar decoder ASIC implementations in the literature have focused their attention on the implementation of the SC or SCL algorithm, without any consideration of the circuits required for frozen bit insertion or rate matching. This is because these circuits can operate relatively independently of the core decoder, which is reminiscent of the rate matching circuit of a turbo decoder.

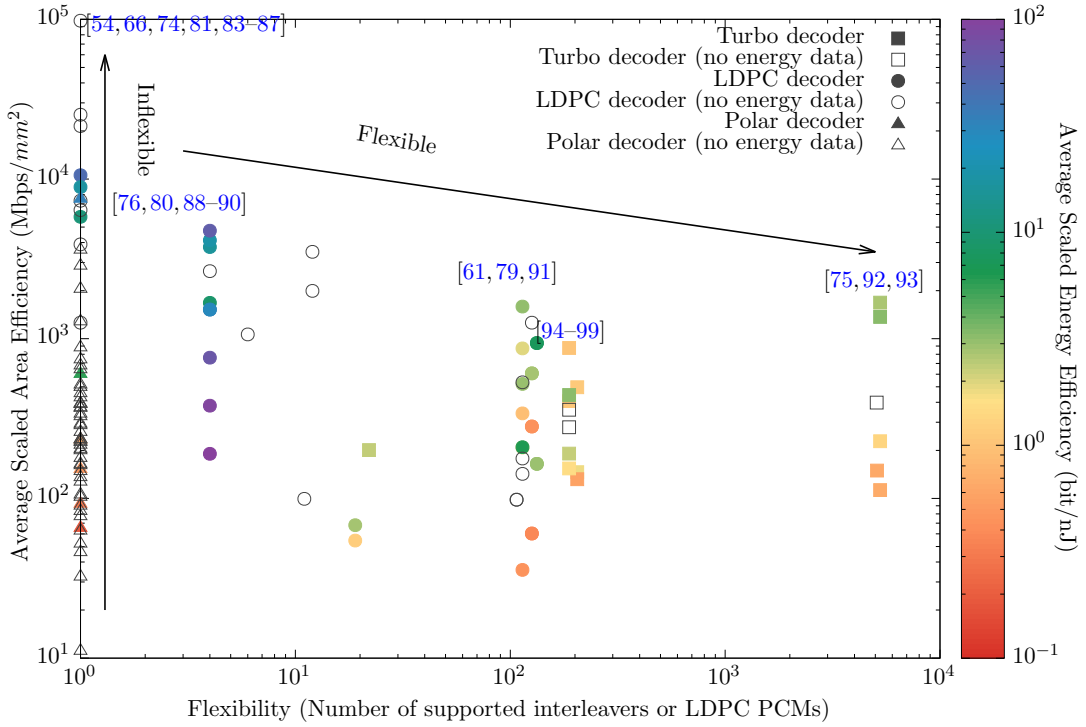
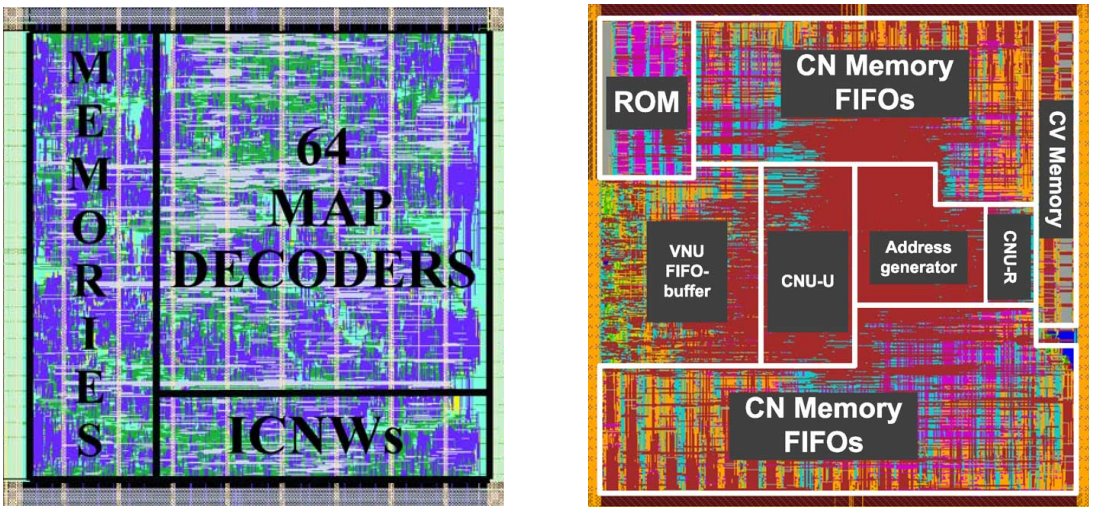


FIGURE 3.5: Area-efficiency ($Mbps/mm^2$) versus the reconfiguration flexibility between state-of-the-art turbo, LDPC and inflexible polar decoder ASICs when scaled to 65 nm.

As shown in Figure 3.5, the turbo decoder implementations considered have the highest flexibility among the three types of channel decoders, as discussed above. The LDPC decoder ASICs that achieve the highest area-efficiencies [54, 66, 74, 81, 83–87] typically support only a single combination of block length K and coding rate R. This is because these LDPC decoders adopt fully-parallel architectures, which only support a single PCM and a high coding rate R. More specifically, these ASICs are physically laid out in a manner that resembles the factor graph structure [100] described by the PCM, using hard-wired connections between registers and dedicated computational hardware for each VN and CN. While this approach prevents flexibility, it allows the LDPC decoding process to be completed using a minimal number of clock cycles and without the requirement for additional memory, switchable interconnections or a complex controller. Owing to this lack of flexibility, it is the LDPC decoders of [54, 66, 74, 81, 83–87] that offer the best throughputs, latencies, hardware-efficiencies and energy-efficiencies among all of the channel decoder ASICs, as described above. By contrast, the flexible LDPC decoder ASICs of Figure 3.5 support more than one PCM by employing partially-parallel architectures, such as the row parallel [53] or block parallel [52] architecture. More specifically, these ASICs employ a bank of computational hardware, which can be flexibly reused at different times to perform the processing associated with different VNs



(a) (b)

FIGURE 3.6: (a) ASIC layout of the partially-parallel turbo decoder of [101], in which the non-computational components (labeled 'MEMORIES' and 'ICNWs') occupy 30% of the area. (b) ASIC layout of the partially-parallel LDPC decoder of [61], in which the non-computational components (labeled 'ROM', 'CN Memory FIFOs', 'CV Memory', 'VNU FIFO-buffer', 'Address generator' and 'CN Memory FIFOs') occupy 75% of the area.

and CNs of different PCMs. However, this approach requires the employment of additional memory, switchable interconnections and a complex controller. As exemplified in Figure 3.6, these components typically occupy around 75% of the chip area in the case of LDPC decoders that support around 100 PCMs, as may be required for meeting the flexibility requirement of 5G. Owing to this, it is these additional hardware components that dominate the throughput, latency, hardware-efficiency and energy-efficiency of the resultant ASICs, particularly as the number of supported PCMs is increased, as shown in Figure 3.5. This may be attributed to the irregular structures and high interconnection complexity, as described in Section 2.3. By contrast, turbo decoders have regular structures associated with significantly reduced interconnection complexities. As a result, the corresponding hardware typically occupies only 30% of the chip area, as exemplified in Figure 3.6. Owing to this, partially-parallel turbo decoders typically offer superior flexibilities, information throughputs, latencies, hardware efficiencies and energy efficiencies than partially-parallel LDPC decoders, as shown in Figure 3.5.

3.2.4 Summary

Having compared numerous turbo, LDPC and polar decoder implementations in terms of their information throughput, error correction performance, flexibility, area efficiency and energy efficiency, some relative advantages and disadvantages can be observed. The

ASIC implementations of turbo decoders offer the best error correction performance owing to their natural support for low coding rates and long block lengths, as exemplified in [75]. However, limited by the data dependencies of the Log-BCJR algorithm, most turbo decoder ASIC implementations have low area- and power-efficiencies compared to LDPC and polar decoders having the same level of information throughput. ASIC implementations of LDPC decoders have been designed for conformance with various standards. For example, the LDPC 10G-BaseT decoders of [54] and [66], both high information throughput, as well as area- and energy-efficiencies, but no flexibility. Polar decoder ASIC implementations offer outstanding error correction performance for short block lengths, offering about 1dB or more coding gain relative to LDPC decoders having short block lengths. However, polar decoders have relatively limited capability to achieve high information throughput, especially at high area- and energy-efficiencies, unless low list sizes resulting in a degraded error correction capability are adopted, as exemplified in [67]. Additionally, polar codes potentially exhibit a high flexibility, although this has not been demonstrated in the open literature as yet. In summary, these types of channel decoder implementations have relative advantages and disadvantages, which must be considered when selecting a channel code for a practical application.

3.3 Design recommendations

Owing to the lack of some important details in some publications, the comparison of the many ASIC channel decoder implementations considered in this chapter has been challenging. Therefore, we recommend valued Colleagues that future ASIC channel decoder designers include the following important details in their implementation-oriented papers:

- Provide values for every parameter of the algorithm and implementation, especially the most important parameters considered in Table 1 in Appendix A.
- State both the average and the worst-case processing latency of the decoder for a variety of block lengths and coding rates, or provide an approximate formulaic relationship between the information throughput and processing latency.
- Provide BLER plots and tabulate the E_b/N_0 value that is required to achieve a BLER of 10^{-2} by the decoder, for a variety of block lengths and coding rates. Identify the type of simulated channel and modulation scheme, which should preferably be BPSK modulation for transmission over an AWGN channel, since this allows direct comparison with the majority of previous publications.

- Provide the power- or energy-consumption for a variety of block lengths and coding rates.
- Quantify the flexibility of the decoder in terms of the number of supported block length and coding rate combinations.

There is still a significant amount of further work that can be completed on the implementation of channel decoders. In particular, there is a need for further research on flexible, high-performance, high-efficiency implementations of turbo, LDPC and polar decoders. For example, the hardware implementation of irregular turbo decoding [102] has not yet been demonstrated, which would improve the attainable error correction capability. Likewise, the performance of LDPC decoders can be significantly enhanced by invoking informed dynamic scheduling [103]. However, this benefit comes at the cost of increased implementation complexity owing to its adaptive decoding schedules. Polar decoding can be enhanced by implementing Single-Input-Single-Output (SISO) decoding [104], which would facilitate ASIC-based turbo detection for the first time for polar codes. Likewise, the ASIC implementation of HARQ [105] for polar decoding is another challenge that remains unsolved at the time of writing. It is not possible to say that any one type of channel code is superior to any other, since all codes have different advantages and disadvantages.

3.4 Conclusions

In closing we present an overview of our design recommendations for future ASIC channel decoder implementations. We commenced by discussing the channel coding requirements of 5G, which include high throughput, low latency, strong error correction capability and low implementation complexity. We have also highlighted the challenging 5G requirements imposed on the channel code to flexibly support a wide variety of block lengths K and coding rates R, in order to ensure the efficient exploitation of the bandwidth available and to address the challenging applications of 5G. We have provided comprehensive discussions on the extent to which the 5G requirements can be met by the turbo, LDPC and polar codes. In order to support our discussions, we also presented several plots to characterise a diverse set of as many as 110 ASIC implementations of turbo, LDPC and polar decoders. We have characterised the fundamental trade-offs between the various performance characteristics of channel decoder ASICs. Furthermore, we have demonstrated that the overall implementation complexity of a channel code depends not only on its computational complexity, but also on its interconnection complexity and its inherent flexibility.

TABLE 3.1: Comparison between turbo, LDPC and polar decoders included in this chapter.

Parameter	Turbo decoder	LDPC Decoder	Polar Decoder
Computational complexity	Higher for most coding rates. Low at low coding rates	Lower for most coding rates. High at low coding rates	Lower for most coding rates
Information throughput	Low. Enhanced by fully-parallel structure	High for high coding rates and parallel structure	High for pipeline structure
Area efficiency	Low for most coding rate. High at low coding rates.	High for most coding rates. Low at low coding rates.	High for most coding rates.
Energy efficiency	Low for most coding rates. High at low coding rates.	High for most coding rate. Low at low coding rates.	High for most coding rates.
Error correction performance	Similar at long block lengths.	Similar at long block lengths.	Superior for short block lengths.
Flexibility	High flexibility	Partial flexibility	Very high potential

Our findings are summarised by the comparison of the three types of channel decoder implementations presented in Table 3.1. We observed that most of existing turbo decoders offer limited information throughput owing to the serial nature of the Log-BCJR algorithm. This leads to lower area- and energy-efficiencies at most of the practical coding rates R, although they maintain these efficiencies at low coding rates, in contrast to the family of LDPC and polar decoders. However, among the decoder ASICs considered in this chapter, the turbo decoders have the best error correction performance and offer the highest flexibility in terms of supporting various combinations of block lengths and coding rates. By contrast, LDPC and polar decoder implementations have both a high information throughput as well as high area- and energy-efficiencies for most practical coding rates. However, the state-of-the-art in LDPC and polar coding has recently been substantially advanced during the design of 3GPP 5G New Radio. More specifically, the New Radio implementations of both codes enable flexibility both in terms of the block length and coding rate with single bit granularity. Furthermore, the error correction performance of both codes has been improved relative to previous implementations. While polar codes offer the best error correction performance at short block lengths, turbo, LDPC and polar codes offer similar performance at block lengths above 1000 bits. By contrast, most of the LDPC and polar decoder implementations considered in this chapter support only shorter block lengths of up to 1024 bits, where their error correction ability is weaker. While it may be expected that flexible LDPC

and polar decoder implementations of 3GPP 5G New Radio codes will emerge in the future, many of the ASIC implementations that were available at the time of writing this chapter do not meet the error correction performance and flexibility requirements of 5G NR. However, the ASIC implementations considered in this chapter do not meet the error correction performance and flexibility requirements of 5G NR. In the case of the LDPC decoders, this may be explained by the limited error correction performance and flexibility of the LDPC codes used in the existing standards. Furthermore, despite their great potential, the implementation of polar decoders having full flexibility has not been demonstrated in the previous ASIC implementations. Likewise, at the time of writing only low list sizes have been considered for the ASIC implementation of SCL polar decoders, while an improved error correction performance can only be expected at higher list sizes.

Chapter 4

Introduction to the New Radio Low Density Parity Check Code and its EXIT Chart Analysis

Based on the survey of channel codes in Chapter 3, LDPC codes tend to exhibit superior performance for the long block length scenarios, which constitute a mature technique invented in 1963 by Gallager [23]. In the 3GPP meeting standardizing the 5G NR, LDPC codes have been adopted as the channel coding method for the data channel. This chapter introduces the encoding scheme of the NR LDPC code specified in the 3GPP NR standard, TS 38.212 that was finally been ratified in 2018 [1]. However, for the sake of achieving high flexibility and superior BLER performance, this NR LDPC code is rather different from conventional LDPC codes, which leads to a failure of the traditional EXIT chart analysis for the NR LDPC decoder. Therefore, a novel EXIT chart analysis technique is introduced in this chapter for a comprehensive understanding of the NR LDPC decoding process. More specifically, this chapter starts with an overview of the NR LDPC code in Section 4.1. Explicitly, Section 4.2 will describe the operation of the NR LDPC code with the aid of block's diagrams. In Section 4.3, simulation results are presented to demonstrate the error-correction performance of the NR LDPC code. Section 4.4 describes the structure of the NR LDPC decoder in more detail with the aid of a rearranged Tanner graph, which also helps our EXIT chart analysis. In this section, our novel solutions will be presented. Finally, Section 4.5 offers our conclusions.

4.1 Overview of the New Radio LDPC code

Proposed by Gallager in his PhD thesis in 1963 [23], LDPC codes have been investigated for around 60 years. As a very mature technique, they have also been adopted for a variety of standards. As we introduced in Chapter 2, LDPC coding is employed in the WiFi [6], WiMAX [7], WiGig [8], DVB-S2 [9], and 10GBASE-T [10] standards. Each of them are employed for diverse applications, including wireless local area networking, mobile broadband communication, mmWave communication, satellite communication and Ethernet. Since the outstanding performance has been validated in many applications, they have also been adopted in the NR 5G wireless communication standard [1] as the channel coding for the data transmission.

Since the 3rd Generation (3G) standardization, the 3GPP body solicited the collaborative contributions to the standardization of the global network protocols, which led to the 4th Generation (4G) standard, also termed as LTE [106–108]. Like 3G UMTS and 4G LTE, the aim of the 3GPP 5G NR is to continue the trend of offering substantially improved user experience and more diverse applications for cellular communications. As we introduced in Chapter 2, the usage scenarios of 5G are basically eMBB, URLLC and mMTC. Although the LDPC code was used in earlier generations of mobile communications, in order to meet the 5G requirements, the LDPC codes had to undergo adjustable designs. For example, the NR LDPC code should support the HARQ as well as a wide range of block lengths and coding rates. Since the LDPC codes have been chosen as the coding technique for data transmission in 5G wireless communication by 2016, enormous research efforts have been investigated in designing the desired NR LDPC code. Finally, in the period between February 2017 to April 2018, the 3GPP specified the NR LDPC standard in TS 38.212 [1]. The philosophy and performance of the NR LDPC code has been demonstrated in [31], which will hence not be described in this treatise.

The aim of this chapter is to present the standardisation of the NR LDPC code included in TS 38.212. Since the standard is full of specific terms and equations, for augmented clarity, we use block diagrams to illustrates most of the coding steps and try to make the explanation comprehensively.

4.2 Standardized NR LDPC components

In this section, the standardised NR LDPC encoder will be described in detail with the aid of figures, with reference to TS 38.212 [1]. Section 4.2.1 gives an overview of the NR LDPC encoding components and of the physical layer channel used in the 5G wireless transmission. This is followed by a step-by-step encoding procedure, including

the transport block CRC attachment, base graph selection, code block segmentation and CRC attachment, the NR LDPC encoder, rate matching and finally the code block segmentation procedure, which are discussed in Section 4.2.2 to Section 4.2.7. Finally, we presented an example of the NR LDPC encoding components in Section 4.2.8 to summarise the physical layer at the transmitter side. The aim of this section is to assist the reader to understand the rationale of the 5G NR LDPC standardisation.

4.2.1 Overview and physical layer channel

For the uplink, the Physical Uplink Shared Channel (PUSCH) is adopted, which is intended to transmit the application data, and also the Uplink Control Information (UCI) messages [109]. By contrast, in the downlink, the Physical Downlink Shared Channel (PDSCH) relies on NR LDPC codes, which carries the application data and paging information [109]. Since the PUSCH and PDSCH have similar structures, we use the PDSCH as an example of the NR LDPC coding components. In order to present this comprehensively, we use the block diagram in Figure 4.1.

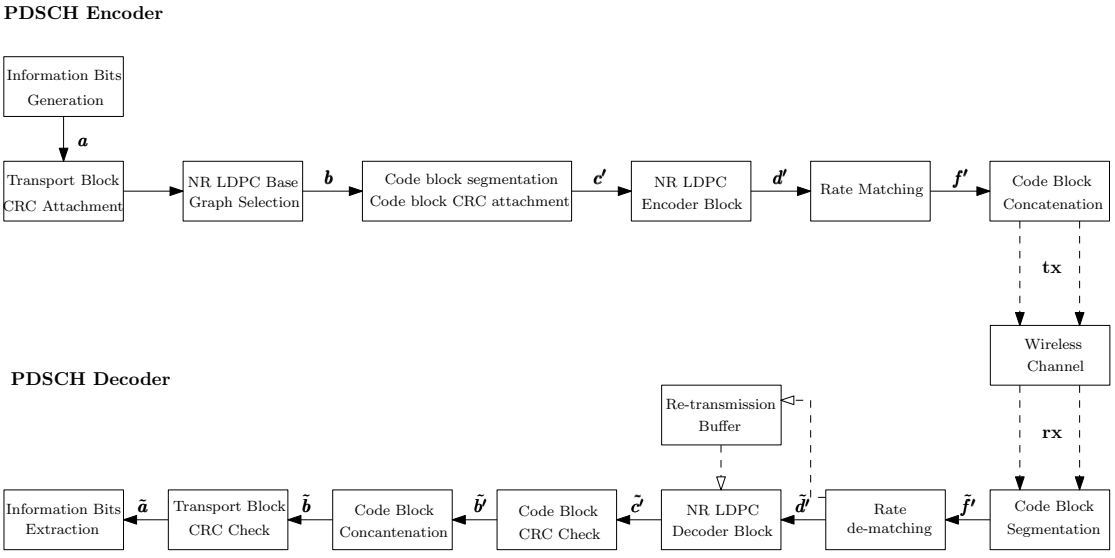


FIGURE 4.1: The block diagram of the NR PDSCH standard.

Before the encoding process, the uncoded block length K and the coding rate R have to be decided for satisfying the different user requirements. In Figure 4.1, the sequence of information bits to be encoded and transmitted is regarded as a transport block $\boldsymbol{a}$, which is extended by a CRC code based on the block length. If the block length is higher than 3824 bits, the system will adopt the CRC24A polynomial of [110] for the CRC processing. By contrast, for the transport block that has a length lower than 3824 bits, the system will employ the CRC16 polynomial [111] for the CRC processing. Then, the encoder will select the Base Graph (BG) based on the block length and

coding rate of the original transport payload $\boldsymbol{a}$. Since both the BG1 and BG2 have a maximum block size, the transport block after the CRC attachment $\boldsymbol{b}$ will be partitioned into several segments. Here, we use $\boldsymbol{c'}$ to represent an example of the code block after the segmentation. If more than one code block are generated, the code block CRC, relying on applying the CRC24B generator will be attached to each code block. Next, the code block $\boldsymbol{c'}$ will be encoded by the NR LDPC encoder. The encoded code block output by the NR LDPC encoder block will be marked as $\boldsymbol{d'}$. Based on the coding rate that was determined by the system requirements, the code block $\boldsymbol{d'}$ will be forwarded to the rate matching block to realize the coding rate required by using a variety of methods, including puncturing, shortening and repetition [45]. Furthermore, since the NR LDPC also supports a HARQ scheme, the coded block $\boldsymbol{d'}$ will be assigned a carefully selected set of encoded bits. Finally, the selected block $\boldsymbol{f'}$ and all other segments will be concatenated into a whole transport block for transmission through the wireless channel. Note that the TS 38.211 standard clarified different modulation and coding scheme for different working scenarios, including $\pi/2$-BPSK, BPSK, QPSK, 16 QAM, 64 QAM and 256 QAM [112]. Since we are focusing on the NR LDPC coded transmission, the NR modulation and coding scheme will not be discussed in detail.

The decoder side of PDSCH is actually not specified in the 3GPP NR standard. This is typical in standards, because different-complexity receives have different costs and this stimulates competition in the market. Therefore, the decoder of Figure 4.1 is simply carries out the inverse operations of the encoder. The PDSCH decoder is aware of the selected block size, coding rate and of the number of re-transmissions. The received transport block will be divided into several code blocks using the same segmentation as the encoder. For example, we consider one of the code blocks $\tilde{\boldsymbol{f}}'$ for further explanation. The code block $\tilde{\boldsymbol{f}}'$ is passed through the rate de-matching block, which converts the code block back to its original block size by using de-puncturing, de-shortening and de-repetition. This recovered code block $\tilde{\boldsymbol{d}}'$ will be decoded by the NR LDPC decoder. Note that, due to the employment of a HARQ scheme, if the decoding took place during the re-transmission scenarios, the former transmission will be stored in a re-transmission buffer and may be used for later decoding. Therefore, there are two inputs to the NR LDPC decoder block, including the current transmission and an input from the buffer that stores information from former transmissions of the same implementation. The decoded code block $\tilde{\boldsymbol{c}}'$ has to pass through a code block CRC check using the same CRC24B polynomial adopted by the encoder side, if the number of received code blocks of the same transport block is more than one. If the CRC check is passed, the decoded code block $\tilde{\boldsymbol{c}}'$ is considered to be correct and its code block CRC attachment is removed, yielding the code block $\tilde{\boldsymbol{b}}'$. Otherwise, the decoder side will ask for a re-transmission of the whole transport block. The physical layer transmission procedure has been specified in TS 38.214 [113], more than one code blocks can be concatenated for producing the

transport block. A transport block can be constituted by 1, 2, 4, 6 or 8 continuous code blocks, which makes it more flexible, especially in the scenario of hostile transmission environments. Until all code blocks succeed in passing the code block CRC check, these code blocks will be concatenated to a whole transport block $\tilde{\boldsymbol{B}}$, which will be passed to the transport block CRC check. Similarly, if the transport block CRC gives positive result, we are able to successfully extract the transmitted information by removing the transport block CRC code. Otherwise, the decoder will ask for a re-transmission of the same transport block.

In the following subsections, all these blocks in Figure 4.1 will be further elaborated on wit the aid of block diagrams, flow charts and examples to unveil the operation standardised for the NR LDPC code.

4.2.2 Transport block CRC attachment

As a commonly used error-detecting code in digital communication, CRC codes were invented by Peterson in 1961 [114]. According to the 3GPP NR standard TS 38.212 Subclause 5.1 [1], the redundant CRC bits will be appended to the transport blocks, which helps the decoder to detect the presence of errors and decide whether to request re-transmission. Based on the different transport block sizes, there are two types of CRC polynomials adopted, including CRC24A and CRC16. As stated in TS 38.212 Subclause 6.1 [1], if the block size of the transport block is higher than 3824 bits, the CRC generator is specified as the 24-bit CRC24A scheme of:

$$g_{CRC24A}(D) = [D^{24}+D^{23}+D^{18}+D^{17}+D^{14}+D^{11}+D^{10}+D^{7}+D^{6}+D^{5}+D^{4}+D^{3}+D+1]. \tag{4.1}$$

As illustrated in Figure 4.2, the output bits of the CRC24A generator will be systematically ordered, which means that the CRC bits are simply concatenated to the original input bits. When the input bits are copied to the output, they are also entered into the shift registers. Seen in Figure 4.2, once all the input bits are fully copied to the output, the switch will be flipped to the bottom and the 24 CRC bits are attached to the original information bits during the next 24 clock cycles. For example, if we have the input bits of

$$[11010110011110001110001011101111],$$

the output bits will be

$$[\underbrace{11010110011110001110001011101111}_{\text{Information bits}}\underbrace{01100001111110001100011}_{\text{CRC bits}}].$$

By contrast, if the block size is equal to or less than 3824, 16 CRC bits are generated by the CRC16 polynomial of :

$$g_{CRC16}(D) = [D^{16} + D^{12} + D^5 + D + 1], \tag{4.2}$$

as illustrated in Figure 4.3. The CRC calculation in Figure 4.3 is similar to that of the CRC24A generator. For instance, if the input bits of the CRC16 generator are,

$$[0001110001011101111],$$

then the output will be

$$[\underbrace{0001110001011101111}_{\text{Information bits}}\underbrace{1011001001010000}_{\text{CRC bits}}].$$

As seen earlier in Figure 4.1, the transport block a is passed through the transport CRC check. Let us assume that the transport payload $\boldsymbol{a}$ has the size of A, which is specified as $\boldsymbol{a} = [a_0, a_1, a_2, ...a_{A-1}]$. At the same time, we have the calculated CRC bits $\boldsymbol{p}$ length of L denoted as $\boldsymbol{p} = [p_0, p_1, p_2, ..., p_{L-1}]$. The resultant payload $\boldsymbol{b}$ output by the transport block CRC will have the size of $B = A + L_{TB}$. The resultant bits of the payload b are denoted as $\boldsymbol{b} = [b_0, b_1, b_2, ..., b_{A+L-1}]$.

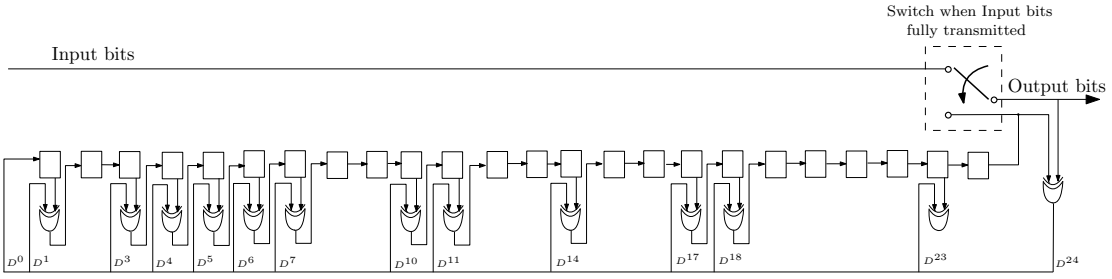


FIGURE 4.2: The block diagram of the CRC24A generator.

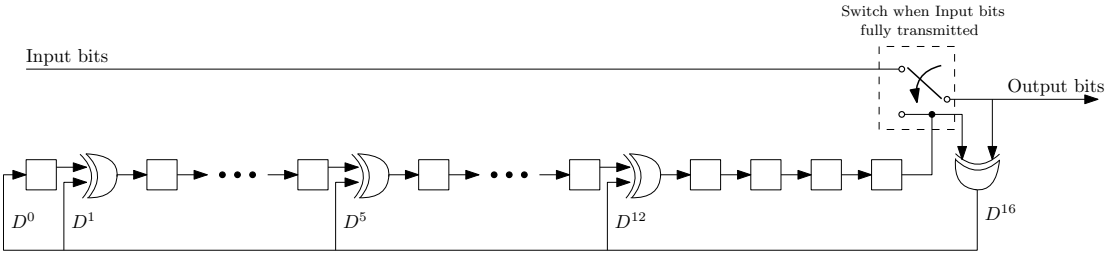


FIGURE 4.3: The block diagram of the CRC16 generator.

4.2.3 Base Graph selection

The key descriptor of the LDPC code is its low density parity check matrix, which is derived from a so-called Base Graph (BG) having the elements 0 and 1. The base graph is more dense than the Parity Check Matrix (PCM) and its role will be discussed in Section 4.2.5. Compared to the conventional LDPC code, the NR LDPC code is implemented with an unique PCM that introduced by the collaboration of different companies in the 3GPP meeting. Suffice to say that the 3GPP NR LDPC code has two different BGs, depending on the block size of A and coding rate R. According to the standard TS 38.212 [1], the adoption of BG2 is suitable for following scenarios:

- $A \leq 292$;
- $A \leq 3824$ and $R \leq 0.67$;
- $R \leq 0.25$;

Otherwise, the NR LDPC code will employ BG1, which is illustrated in Figure 4.4, while BG2 is seen in Figure 4.5. As shown in Figure 4.4, there are 46 rows and 68 columns in BG1, while in Figure 4.5 we observe 42 rows and 52 columns in BG2. All black-filled blocks in these two figures represent a 1 in the block, while all blank blocks contain the value 0. The method of transforming the BG into the PCM will be discussed in Sections 4.2.5. While the associated code design is the main topic of Section 4.4.1.

4.2.4 Code block segmentation and code block CRC attachment

The code block segmentation and code block CRC attachment functions are discussed in the context of Figure 4.6.

As shown in Figure 4.6, the input bits represent the transport payload $\boldsymbol{b}$ having a block size B, which is output by the transport block CRC generator, and it is denoted as $\boldsymbol{b} = [b_0, b_1, ..., b_{B-1}]$. At the beginning, this block will analyse whether this transport payload requires any code block segmentation. For each BG, there is a maximum tolerable code block size of K_{cb}, as determined by the standard TS 38.212. Hence, if B is higher than K_{cb}, code block segmentation will be applied to the transport payload. The maximum code block size for BG1 is $K_{cb} = 8448$, while for BG2, it is $K_{cb} = 3840$ [1]. If the block size B of the transport payload is not larger than the corresponding block size K_{cb} of the selected BG, the total number of code blocks is $C = 1$, hence no the code block segmentation and no code block CRC attachment is required. In this situation,

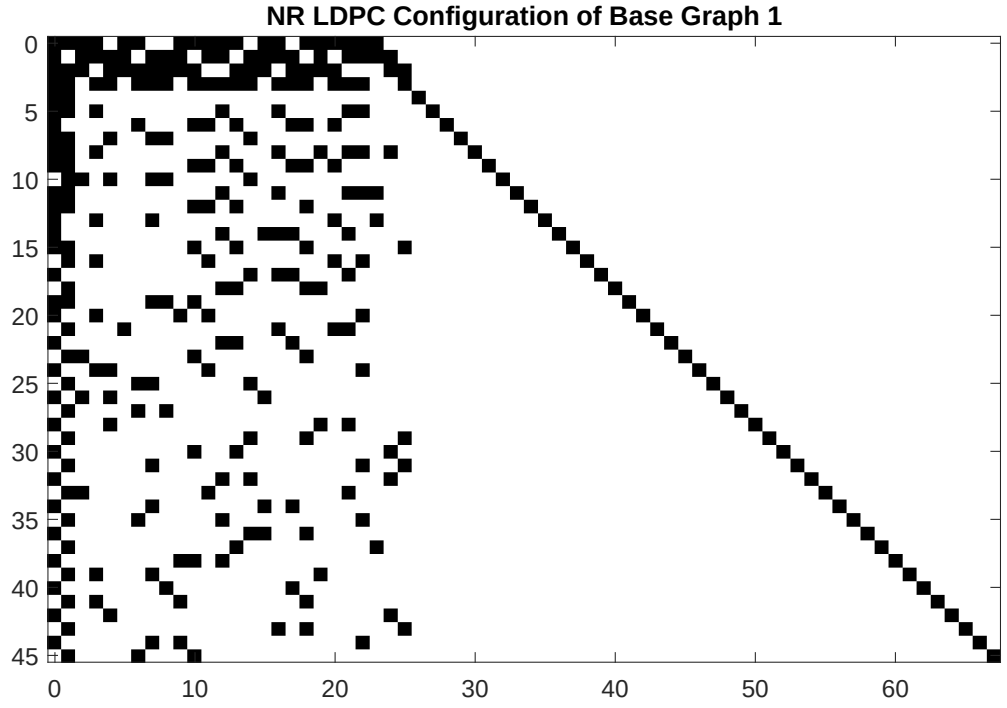


FIGURE 4.4: The NR LDPC configuration of BG1.

the total number of bits in the transport payload B' and the total number of bits in the code block K' are equal to the original block size of the transport payload B.

By contrast, if the block size B of the transport payload is larger than the corresponding block size K_{cb} of the selected BG, the total number is calculated as $C = \lceil B/(K_{cb}-L_{CB})\rceil$, where $L_{CB} = 24$ is the number of code block CRC bits generated by the CRC24B generator. The code block CRC is similar with the transport block CRC. Explicitly, the CRC24B polynomial used for generating the 24 CRC parity check bits is given by

$$g_{CRC24B}(D) = [D^{24} + D^{23} + D^6 + D^5 + D + 1], \tag{4.3}$$

which is shown in Figure 4.7.

For example, if we have the input bit sequence of:

$$[00100111100011110110010111001111],$$

the output of the CRC24B generator will be:

$$[\underbrace{00100111100011110110010111001111}_{\text{information bits}}\underbrace{01000101110000000100111}_{\text{CRC bits}}]$$

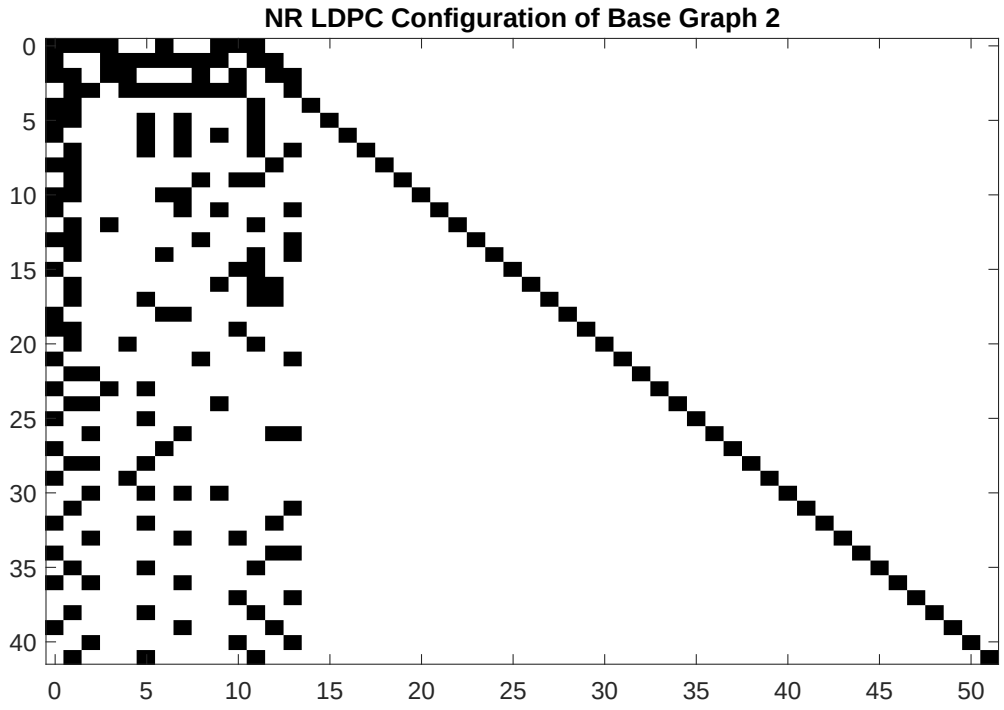


FIGURE 4.5: The NR LDPC configuration of BG2.

After the code block CRC attachment, the total number of bits in the transport payload will become $B' = B + C \cdot L_{CB}$. And the number of bits for each code block is calculated as $K' = B'/C$ [1].

Note that for both scenarios, K' is not the final number of bits in each code block. Since the value of K' and B' have been determined in the former process, this block will use these parameters to determine the number of bits in each code blocks, which is denoted as K. As seen in the Figure 4.6, the value of K_b is going to be determined first, which actually has no specific meaning but only used to help to determine further parameters. The selection of K_b is as follows:

$$K_b = \begin{cases} 22 & \text{if } BG = 1 \\ 10 & \text{if } BG = 2 \text{ and } B > 640 \\ 9 & \text{if } BG = 2 \text{ and } 640 \geqslant B > 560 \\ 8 & \text{if } BG = 2 \text{ and } 560 \geqslant B > 192 \\ 6 & \text{if } BG = 2 \text{ and } 192 \geqslant B. \end{cases} \tag{4.4}$$

Given K_b, the value of the so-called lifting index Z_c can be chosen from Table 4.2.4.

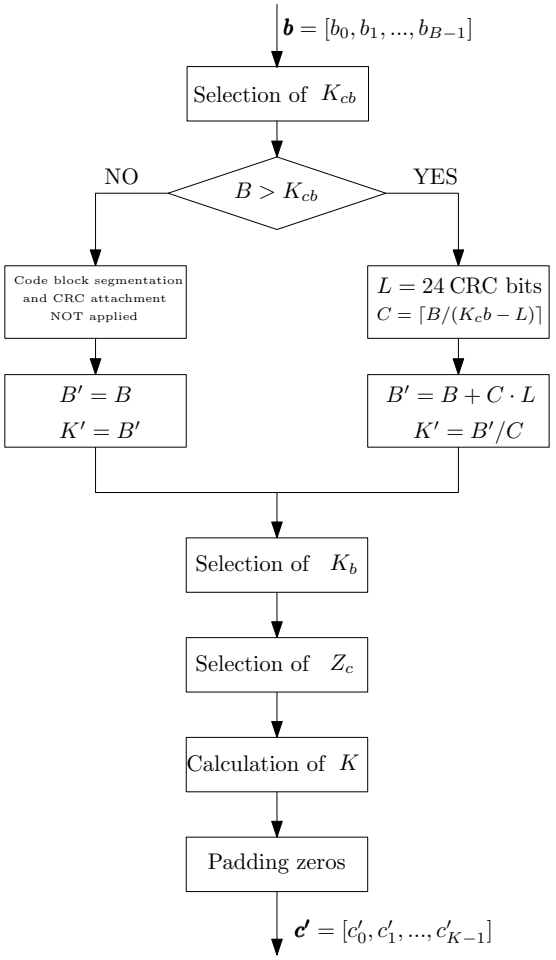


FIGURE 4.6: The structure of code block segmentation and code block CRC attachment functions.

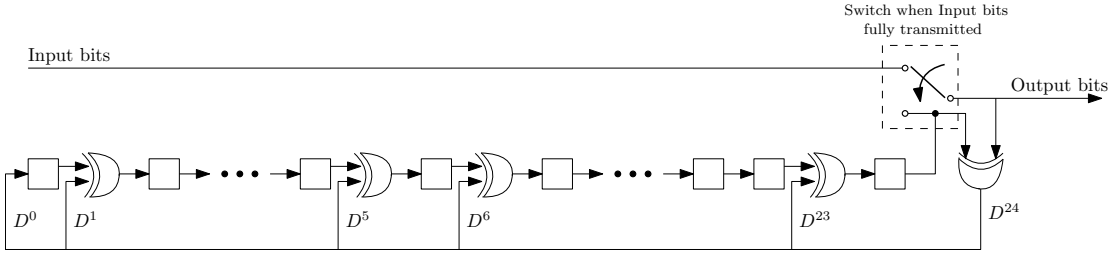


FIGURE 4.7: The block diagram of the CRC24B generator.

The lifting index Z_c is mainly used in the PCM construction, which will be described in detail in the next section. The intention of this step is to ensure that the resultant PCM to be used by the encoder is indeed suitable for the code block. In order to keep the original length of information bits, this value of Z_c has to satisfy that $K_b \cdot Z_c \geqslant K'$, where the lowest value of all qualified Z_c will be selected for the best coding efficiency. Next, the number of bits of each code block can finally be determined. For the encoder associated with the selection of BG1, we have to have $K = 22 \cdot Z_c$, while for the encoder

TABLE 4.1: Sets of NR LDPC lifting sizes Z_c with their set index i_{LS}

Set index (i_{LS})	Set of lifting sizes (Z_c)
0	{2,4,6,8,16,32,64,128,256}
1	{3,6,12,24,48,96,192,384}
2	{5,10,20,40,80,160,320}
3	{7,14,28,56,112,224}
4	{9,18,36,72,144,288}
5	{11,22,44,88,176,352}
6	{13,26,52,104,208}
7	{15,30,60,120,240}

with selection of BG2, $K = 10 \cdot Z_c$. One of the output code blocks of the code block segmentation and code block CRC functions is denoted as $\boldsymbol{c'} = [c'_0, c'_1, ..., c'_{K-1}]$. Finally, in the situation when K' is less than K, zeros will be filled into the missing positions, which makes the bits in the positions $c'_{K'}$ to c'_{K-1} equal to 0. This step basically attaches padding zeros, as presented in the last block of Figure 4.6.

4.2.5 New Radio LDPC encoder

Since the transport payload has been partitioned into code blocks that satisfy the requirement of NR LDPC encoder, the code block $\boldsymbol{c'}$ is passed to the NR LDPC encoder of Figure 4.1. As illustrated in Figure 4.8, the NR LDPC encoder is constituted by three parts, including the Block size determination, the PCM generation and the parity bit generation. In this section, these blocks will be described in detail with the aid of examples and figures.

4.2.5.1 Block size determination

According to the standard TS 38.212 Subclause 5.3.2, the bit sequence output by the NR LDPC has the block size of N denoted as $\boldsymbol{d'} = [d_0, d_1, ..., d_{N-1}]$. Based on the selection of the BG, the number of output bits N has two possible values. For the BG1 LDPC encoder, N is equal to $66 \cdot Z_c$, while for BG2, the N is calculated as $50 \cdot Z_c$. The value of Z_c is selected from Table 4.2.4.

4.2.5.2 PCM generation

Since the first $(K - 2 \cdot Z_c)$ positions of the output bit sequence $\boldsymbol{d'}$ was filled up, the encoder has to generate $(N + 2 \cdot Z_c - K)$ parity bits that are denoted as $\boldsymbol{w} =$

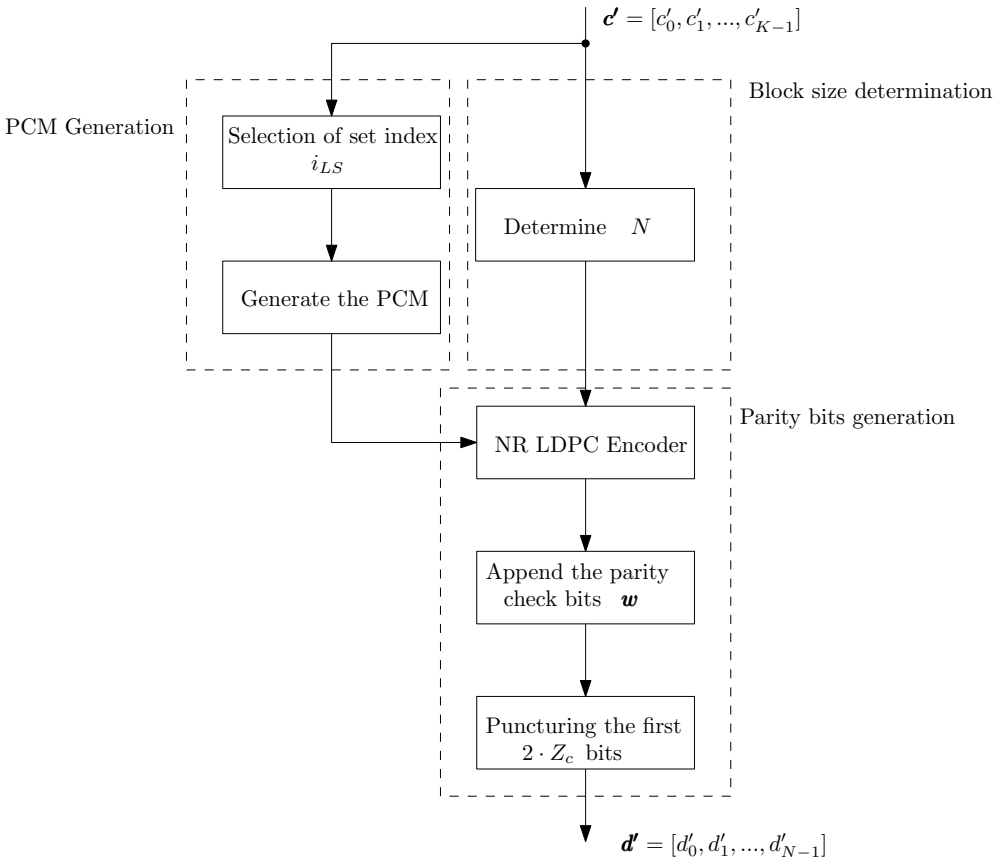


FIGURE 4.8: Flow chart of the NR LDPC Encoder.

$[w_0, w_1, ..., w_{N+2 \cdot Z_c - K-1}]$. The parity bits w have to satisfy that the product of the PCM matrix $\boldsymbol{H}$ and the combination of $\boldsymbol{c'}$ and $\boldsymbol{w}$ is zero, which is formulated as $\boldsymbol{H} \times \begin{bmatrix} \boldsymbol{c} \\ \boldsymbol{w} \end{bmatrix} = \mathbf{0}$, where $\mathbf{0}$ is the full of zeros. Therefore, the PCM H has to be determined before the parity bit generation. Again, the BG of the NR LDPC code was discussed in Section 4.2.3. For each BG, there are 8 different sets of PCMs, which are determined by the index i_{LS}. While the set index i_{LS} leads to the determination of the parity check-instruction matrices that are denoted as $\boldsymbol{V}$ in the TS 38.212 standard Subclause 5.3.2. In order to clarify the process of PCM generation, we provided a simple example in Figure 4.9. As shown in Figure 4.9, the example takes a part of the matrix in BG1 that presented as $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$. Since the lifting index Z_c is already calculated in the previous operation as mentioned in Section 4.2.4, according to the Table 4.2.4, the set index i_{LS} is able be readily determined. In this case, the value of Z_c and i_{LS} are assumed as $Z_c = 4$ and $i_{LS} = 2$. The next step is to enlarge, namely 'lifting' the original BG matrix to the same size as the PCM. Each element of the matrix H_{BG} will be replaced by a $Z_c \times Z_c$ lifted matrix in the PCM matrix H. Explicitly, each 0 element in the $\boldsymbol{H_{BG}}$ is replaced

by a $Z_c \times Z_c$ zero matrix. While each 1 element in the matrix $\boldsymbol{H_{BG}}$ is replaced with a $Z_c \times Z_c$ identity matrix. As a result, the cropped base graph matrix $\boldsymbol{H_{BG}}$ is lifted to a 8-by-8 lifted matrix as seen in Figure 4.9.

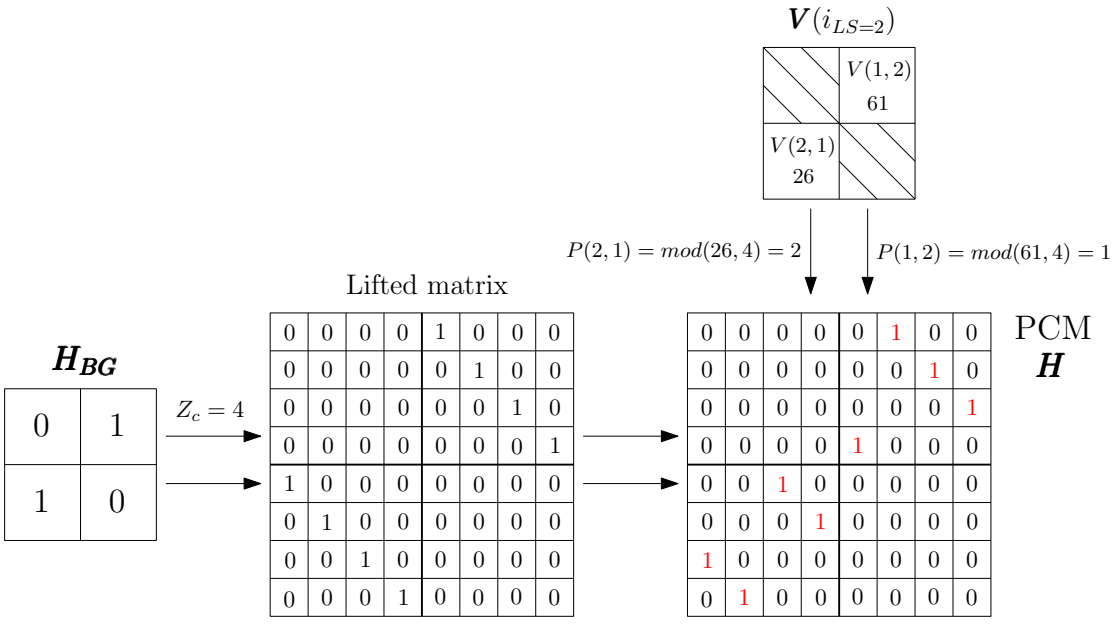


FIGURE 4.9: An example of PCM generation.

The next step of PCM generation is shifting, which is assisted with the parity check-indicator matrices $\boldsymbol{V}$ that are defined in the TS 38.212 standard [1]. As shown in Figure 4.9, corresponding to the cropped $\boldsymbol{H_{BG}}$, the $\boldsymbol{V}$ is cropped by the same size of 2-by-2. For each position has 0 value in the matrix $\boldsymbol{H_{BG}}$, the matrix $\boldsymbol{V}$ keeps the corresponding position vacant, which is programmed as $NULL$ to avoid any confusion. By contrast, for each position which has the value of 1 in $\boldsymbol{H_{BG}}$, there will be a $V_{(i,j)}$ value for the corresponding position in $\boldsymbol{V}$, where i and j are representing corresponding index of row and column. In our case, $V_{(1,2)} = 61$ and $V_{(2,1)} = 26$ are presented in Figure 4.9. As discussed in the standard [1], each lifted identity matrix will be circular shifted to the right by $P_{(i,j)}$ columns, where $P_{(i,j)} = mod(V_{(i,j)}, Z_c)$. Next, we have the value of $P_{(1,2)} = 1$ and $P_{(2,1)} = 2$, which means the top right positioned 4-by-4 matrix will be circular shifted to the right by 1 column. While the bottom left positioned 4-by-4 matrix will be circular shifted to the right by 2 columns. As shown in Figure 4.9, the final PCM $\boldsymbol{H}$, which uses red colour to indicated the change after shifting, is produced from its original base graph matrix $\boldsymbol{H_{BG}}$.

Since we discussed the PCM generation by using only part of the base graph matrix in Figure 4.9, the entire PCM will be transferred and enlarged by the same process. More specifically, the BG1-based original PCM contains $46 \cdot Z_c$ number of rows and $68 \cdot Z_c$ number of columns. While the BG2-based original PCM contains $42 \cdot Z_c$ number of rows

and $52 \cdot Z_c$ number of columns. The structure and characteristics of BG and PCM will be detailed in the Section 4.4.

4.2.5.3 Parity bits generation

Followed by the principle of LDPC encoding, which is noted as $\boldsymbol{H} \times \begin{bmatrix} \boldsymbol{c} \\ \boldsymbol{w} \end{bmatrix} = \boldsymbol{0}$, the parity check bits $\boldsymbol{w}$ can be readily generated. Different from the conventional LDPC encoder, NR LDPC encoder will puncture the first $2 \cdot Z_c$ systematic bits for a better performance [31]. Therefore, the output encoded bits $\boldsymbol{d'}$ will be concatenated as presented in Figure 4.10 with the block length of N.

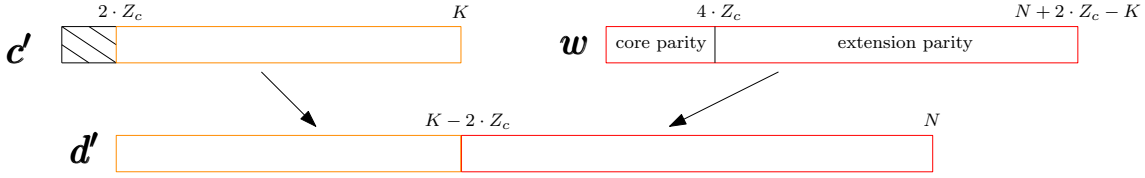


FIGURE 4.10: The illustration of the generation of encoded bits.

4.2.6 Rate matching

The coding rate R is designed to be adjustable in the NR LDPC encoder. By using puncturing and repetition, the coding rate R can be increased or decreased from 1/3 for BG1 and 1/5 for BG2. The coding rate R adopted is usually adjusted to be higher than the original coding rates of the BG, which increases the effective data throughput and reduces the transmission redundancy. Although a high coding rate has a reduced error-correction capability, the 3GPP NR LDPC code is combined with a HARQ scheme. Hence, unsuccessful transmissions will be followed by re-transmissions until error-free decoded bits are received or the maximum number of retransmissions is exhausted.

As shown in Figure 4.1, the input bit sequence of the rate matching block is the encoded code block denoted as $\boldsymbol{d'} = [d_0, d_1, ..., d_{N-1}]$, while the output bit sequence is denoted as $\boldsymbol{f'}$ with the block size of E, which is the code length after rate matching. The next procedure of rate matching is the bit-selection, which is related to the NR HARQ system. In the rate matching block of Figure 4.1, the input bit sequence $\boldsymbol{d'}$ is stored in a circular buffer that has the same block size N, which is $N = 66 \cdot Z_c$ for BG1 and $N = 50 \cdot Z_c$ for BG2. The circular buffer here uses the same standard construction as in the 4G LTE document TS 36.212 [115]. As illustrated in Figure 4.11(a) and (b), the bit sequence d' is input to the circular buffer from position d'_0 to $d'_{(N-1)}$. The process of bit-selection is mainly applied for the HARQ system. At the decoder, if a transmission attempt fails the

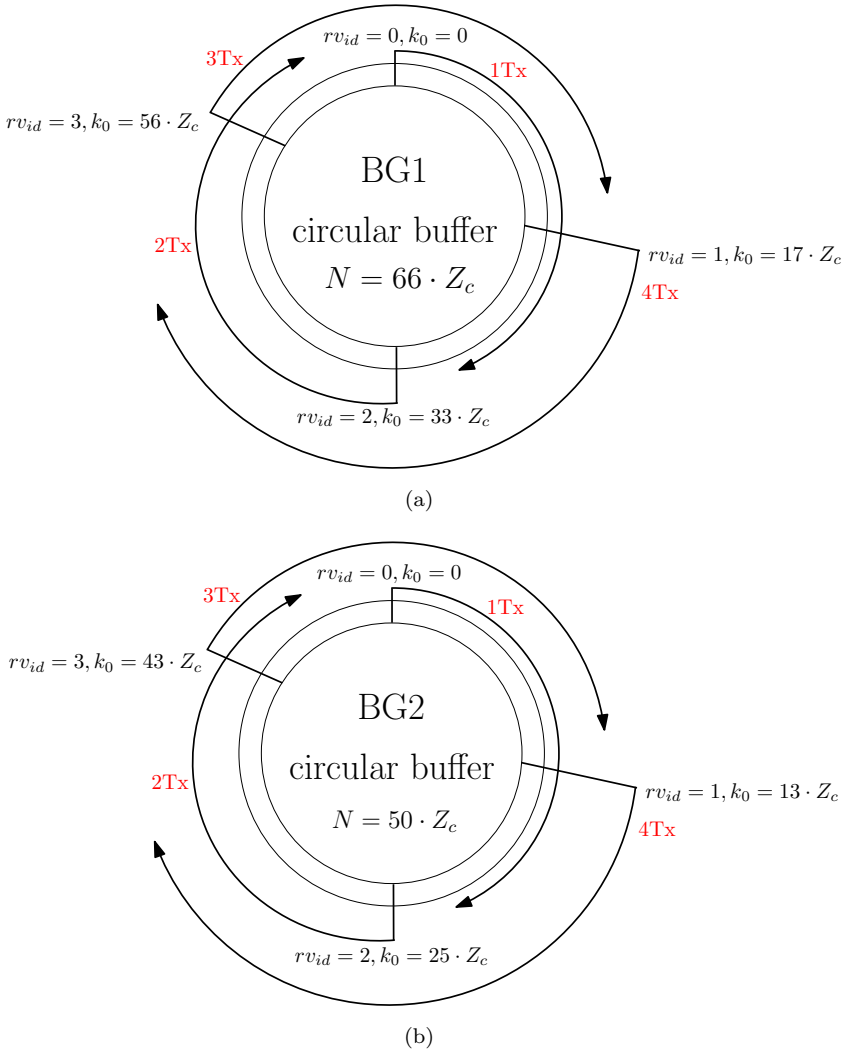


FIGURE 4.11: The illustrations of circular buffer of NR LDPC encoder and the indication of bit-selection with (a) BG1 and (b) BG2.

error-check, the system will request another transmission attempt. As introduced in the standard, the different retransmission attempts have different starting points to obtain the required E number of encoded bits in the circular buffer, which is listed in Table 4.2.6. To elaborate, in Table 4.2.6, rv_{id} represents redundant version of this retransmission, where the variable k_0 represents the starting position of this retransmission in the circular buffer. As suggested by [116], the order of redundant version should be [0,2,3,1]. As illustrated in Figure 4.11(a), the first transmission attempt sends bits starting from $k_0 = 0$, processing E number of bits in the circular buffer, which is denoted as 1Tx in red. More specifically, if the transmitted code block size E is shorter than the size N of the circular buffer, the bits selected for the first transmission attempt will be the input bit sequence $\boldsymbol{d}'$ in the circular buffer, while puncturing the bits from the position of $(N - E - 1)$ to the position of $(N - 1)$. Similarly, for the second transmission attempt,

the bit sequence inside the circular buffer will be selected from $k_0 = 33 \cdot Z_c$. Therefore, the bit sequence selected for the second transmission attempt will be the input bit sequence $\boldsymbol{d'}$ in the circular buffer, while puncturing the bits from the position of $|33 \cdot Z_c + E - N - 1|$ to the position of $(33 \cdot Z_c - 1)$. Likewise, for the third and fourth transmission 3Tx and 4Tx, the output bits are selected from $k_0 = 56 \cdot Z_c$ and $k_0 = 17 \cdot Z_c$ onwards in the circular buffer, which will be punctured correspondingly. The same rule also applies to BG2 as presented in Figure 4.11(b). The performance of this standardised HARQ system has been verified in [117] by comparing it conventional techniques. In Chapter 5, we will conceive a new HARQ scheme.

TABLE 4.2: Table of different start position k_0 in the circular buffer for different retransmission and BG.

rv_{id}	k_0	
	Base graph 1	Base graph 2
0	0	0
1	$17 \cdot Z_c$	$13 \cdot Z_c$
2	$33 \cdot Z_c$	$25 \cdot Z_c$
3	$56 \cdot Z_c$	$43 \cdot Z_c$

Finally, the bits selected from the circular buffer will be interleaved based on the modulation scheme [1]. The final output bits are denoted as $f' = [f'_0, f'_1, ..., f'_{(E-1)}]$.

4.2.7 Code block concatenation

The last step of the NR LDPC encoder is the code block concatenation. Since all code blocks has been processed, this step simply represents concatenating them. Note that, as defined in the NR physical layer recommendation for data, namely TS 38.214 [113], the code blocks can be grouped into different sizes, including 1, 2, 4, 6 and 8 of them. A request for retransmission will not necessarily send the whole original transport block, it may send a fraction of it. In the scenario of poor transmission environments, this method will still maintain a reasonable throughput by reducing the redundancy produced by HARQ.

4.2.8 An entire example of NRLDPC encoding procedure

Followed by discussions of individual encoding components, Figure 4.12 provided an entire example for encoding procedure in the PUSCH physical layer from the transport block generation to the code block concatenation. Here, we presented two scenarios of transport blocks, including a long block length of $A_L = 10000$ bits and a relatively short

block length of $A_S = 1000$ bits. Since we are selecting the coding rate of $R = 5/6$, both transport blocks are using BG1.

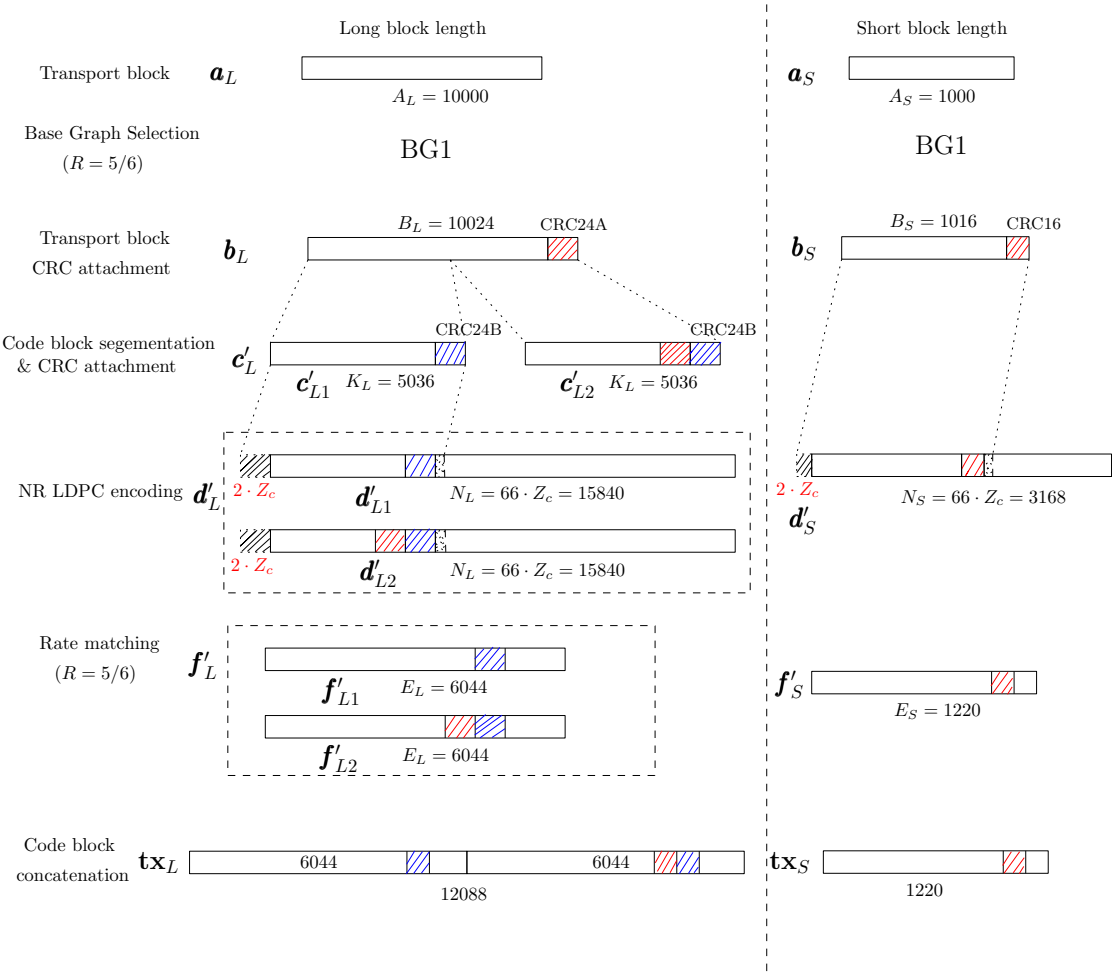


FIGURE 4.12: An example of bit arrangement to the PUSCH physical layer in the transmitter side of NRLDPC code.

As illustrated in Figure 4.12, the CRC24A polynomial and CRC16 polynomial are used for transport block CRC attachment of $\boldsymbol{b}_L$ and $\boldsymbol{b}_S$ respectively. As a result, the number of bits in the transport blocks of two scenarios are increased to $B_L = 10000 + 24 = 10024$ bits and $B_S = 1000 + 16 = 1016$ bits correspondingly. Since the value of B_L is higher than the maximum value of code block length, $\boldsymbol{b}_L$ is split in half and attached with code block CRC by using CRC24B, which leads to the number of bits for each code block segmentation as $K_L = 5012 + 24 = 5036$. By contrast, no code block segmentation and no code block CRC attachment is required for $\boldsymbol{b}_S$, which has short block length. Next, both bit sequences enter to the NR LDPC encoder. With the puncturing of the first $2 \cdot Z_c$ systematic bits and adding some fill-in zeroes as shown in Figure 4.12, the number of bits of $\boldsymbol{d'}_L$ is output as $N_L = 66 \cdot Z_c = 15840$, where Z_c in this scenario is determined as 240. While the number of bits of $\boldsymbol{d'}_S$ is generated as $N_S = 66 \cdot Z_c = 3168$, where Z_c in this

scenario is determined as 48. After that, the rate matching operation selected $E_L = 6044$ and $E_S = 1220$ number of bits respectively for the transmission of each encoded block by using bit-selection circular buffer, which is discussed in Section 4.2.6. Finally, two code blocks are concatenated in the long block length scenario to provide a 12088 bits sequence $\boldsymbol{tx}_L$ for transmission. While in the short block length scenario, the transmitted sequence $\boldsymbol{tx}_S$ keep the same number of bits from the rate-matching operation. The final encoded transmitting frame will be interleaved following the modulation scheme and send to the receiver side.

4.3 NR LDPC performance

In this section, we present the performance of the NR LDPC decoder relying on simple flooding. In Richardson's paper about the introduction of 5G LDPC codes [31], similar BLER performance results have been provided. We apply the same coding rates R and block length K' to verify the accuracy of our NR LDPC system. The AWGN simulation results will be discussed in Section 4.3.1. Apart from the AWGN transmission channel considered in [31], we also employ the uncorrelated Rayleigh fading channel in order to model a more realistic transmission environment. The uncorrelated Rayleigh fading channel simulations will be presented in Section 4.3.2.

4.3.1 Performance of NR LDPC flooding decoder in AWGN channel

In Figure 4.13, we characterize the NR LDPC decoder in AWGN channels using BG1. In order to have a fair comparison with the simulation results in [31], decoders in Figure 4.13 employs same modulation order as Quaternary Phase Shift Keying (QPSK), which contains six different coding rates R, and each of them is combined with three different value of the information block length K'. The selected coding rates are $R = 1/3$, $R = 2/5$, $R = 1/2$, $R = 2/3$, $R = 3/4$ and $R = 5/6$, while the block lengths for each coding rate are $K' = 1000$, $K' = 3000$ and $K' = 6000$. By contrast, Figure 4.14 shows the performance of the flooding NR LDPC decoder using BG2 in the AWGN channels. The coding rates we used for the BG2 simulations are $R = 1/3$, $R = 2/5$, $R = 1/2$ and $R = 2/3$, and each of these coding rates are combined with the information block lengths of $K' = 500$, $K' = 1500$ and $K' = 2500$. As was agreed during the 3GPP NR study item [118, 119] on channel coding, all LDPC decoders are simulated in 50 iterations in order to evaluate their converged decoding performance. By comparing Figure 4.13 and Figure 4.14 with the simulation results in [31], a good agreement is observed. Table 4.3 summarises the results of Figure 4.13 and Figure 4.14.

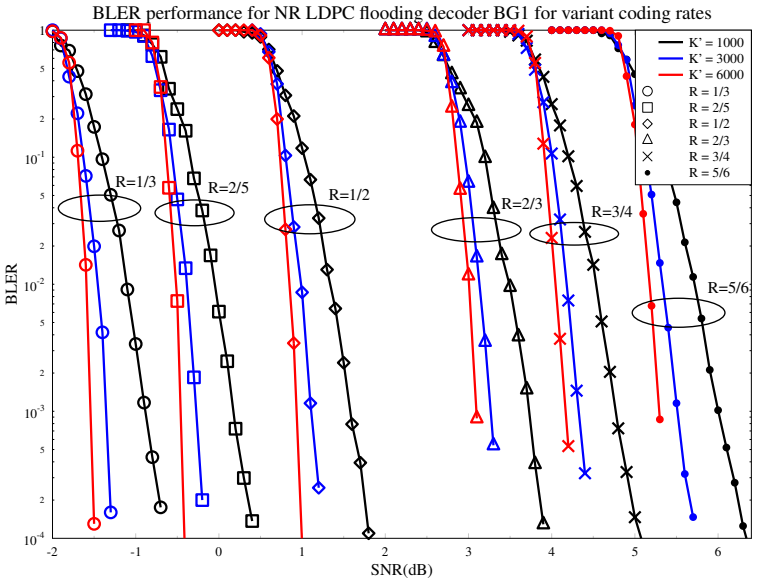


FIGURE 4.13: Pairs of simulation results to show the performance of flooding NR LDPC decoder with **BG1 in the AWGN channel**. Each plots are using different coding rates R with the same combination of information block length K', including 1000, 3000 and 6000. The different value of R for plots are: $R = 1/3$, $R = 2/5$, $R = 1/2$, $R = 2/3$, $R = 3/4$ and $R = 5/6$.

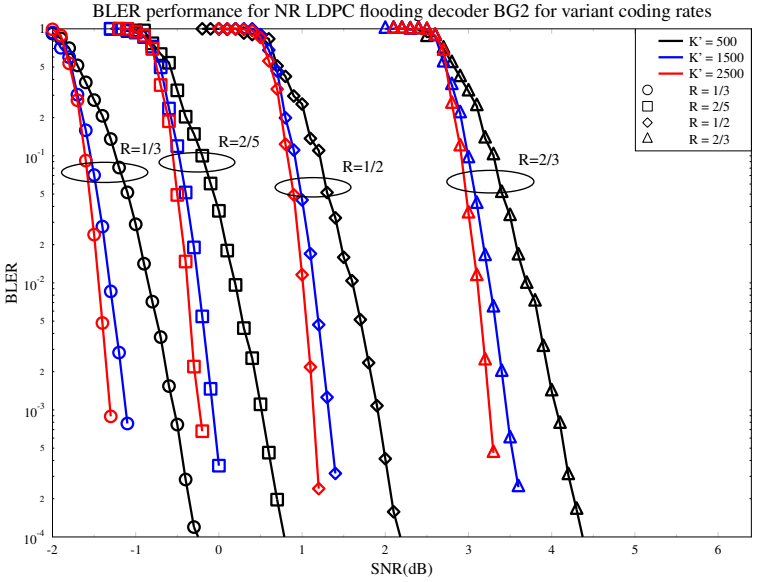


FIGURE 4.14: Pairs of simulation results to show the performance of flooding NR LDPC decoder with **BG2 in the AWGN channel**. Each plots are using different coding rates R with the same combination of information block length K', including 500, 1500 and 2500. The different value of R for plots are: $R = 1/3$, $R = 2/5$, $R = 1/2$ and $R = 2/3$.

By observing the simulation results of Figure 4.13 and Figure 4.14, the following characteristics can be observed. The lower the coding rate, the lower the SNR requirement becomes. However, a low coding rate results in a low effective throughput. Hence,

a promising technique of getting around this problem is to use HARQ, which will be discussed in Chapter 5.

The results of Figure 4.13 and Figure 4.14 are also shown on the classic E_b/N_0 scale in Figure 4.15 as a pair of BER plots by using the same parameters, while the bit-energy E_b was reduced according to the coding rate. For example, $R = 1/3$ obtain the most coding gain in the BG2 NR LDPC decoder. Finally, the coding gain measured on this E_b/N_0 scale at BER=10^{-3} that compared to uncoded QPSK modulation is shown for all six K' values in Figure 4.16 as a function of R.

TABLE 4.3: Simulation results of Figure 4.13 and Figure 4.14.

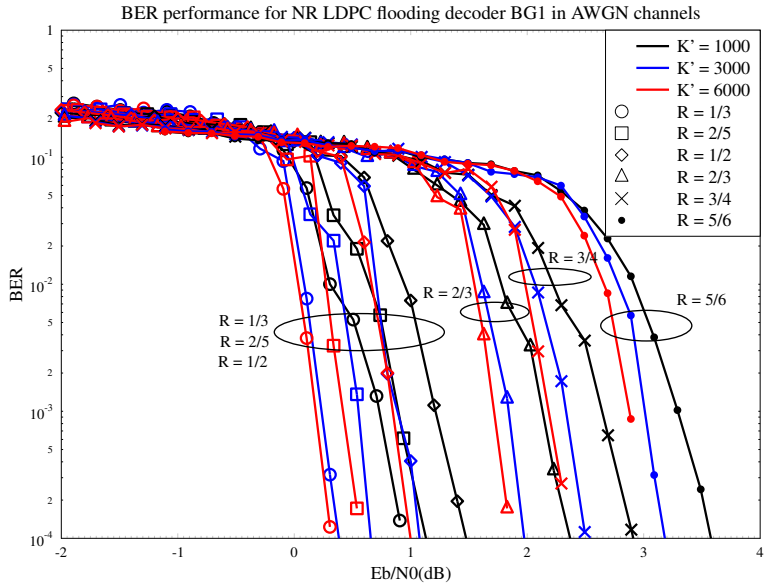
BLER requirement	BG1 BLER=10^{-2}			BG2 BLER=10^{-2}		
	$K' = 1000$	$K' = 3000$	$K' = 6000$	$K' = 500$	$K' = 1500$	$K' = 2500$
$R = 1/3$	-1.1dB	-1.45dB	-1.5dB	-0.9dB	-1.25dB	-1.4dB
$R = 2/5$	-0.5dB	-0.45dB	0dB	0.2dB	-0.15dB	-0.3dB
$R = 1/2$	1.4dB	1dB	0.9dB	1.6dB	1.25dB	1.05dB
$R = 2/3$	3.5dB	3.2dB	3dB	3.8dB	3.3dB	3.15dB
$R = 3/4$	4.5dB	4.3dB	4dB	-	-	-
$R = 5/6$	5.7dB	5.4dB	5.2dB	-	-	-

4.3.2 Performance of the NR LDPC flooding decoder in uncorrelated Rayleigh fading channels

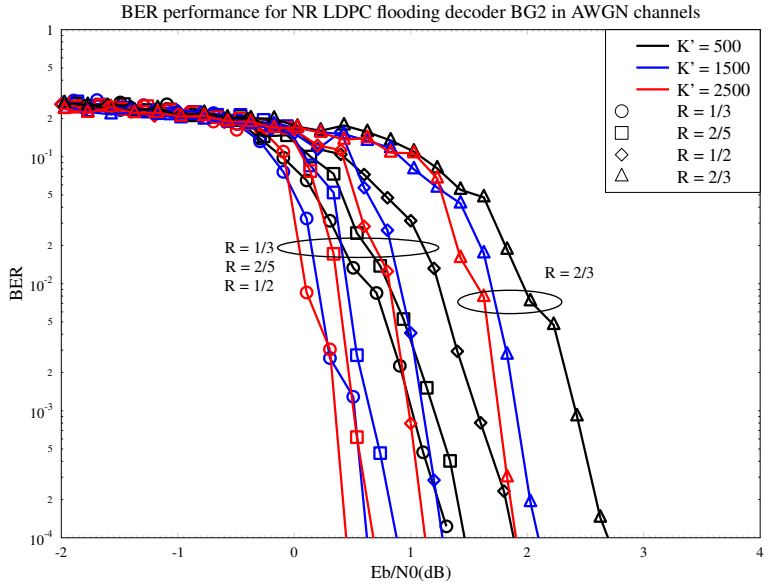
To elaborate a little further, the AWGN channel results of Figure 4.13 and Figure 4.14 may be contrasted to the uncorrelated Rayleigh fading channel results using the same parameters. The simulations plots are presented as Figure 4.17 and Figure 4.18, as well as their E_b/N_0 scaled BER plots in Figure 4.19. The corresponding results are summarized in Table 4.4. Finally, the coding gain measured on this E_b/N_0 scale at BER=10^{-3} is shown for all six K' values in Figure 4.20 as a function of R. As summarized in Figure 4.20, the NR LDPC decoder has much higher coding gain against the uncoded QPSK modulation in uncorrelated Rayleigh fading channels than that in AWGN channels. Again, the lowest coding rate of $R = 1/3$ is the best choice for the highest coding gain in our cases.

TABLE 4.4: Simulation results of Figure 4.17 and Figure 4.18.

Rayleigh Fading	BG1 BLER=10^{-2}			BG2 BLER=10^{-2}		
	$K' = 1000$	$K' = 3000$	$K' = 6000$	$K' = 500$	$K' = 1500$	$K' = 2500$
$R = 1/3$	0.2dB	-0.2dB	-0.4dB	0.5dB	0.95dB	-0.3dB
$R = 2/5$	1.5dB	1.15dB	0.95dB	1.9dB	1.3dB	1.15dB
$R = 1/2$	3.4dB	2.9dB	2.75dB	3.9dB	3.3dB	3dB
$R = 2/3$	6.9dB	6.3dB	6.1dB	7.4dB	6.6dB	6.4dB
$R = 3/4$	8.8dB	8.25dB	7.9dB	-	-	-
$R = 5/6$	11.4dB	10.5dB	10.3dB	-	-	-



(a)



(b)

FIGURE 4.15: Pairs of simulation results to show the performance of flooding NR LDPC decoder with **the AWGN channel** in E_b/N_0 scale. Plot (a) used the same parameters as Figure 4.13. While plot (b) used the same parameters as Figure 4.14.

4.4 NR LDPC Code design and EXIT chart analysis

4.4.1 New Radio LDPC code design

The NR LDPC is specified using a pair of BGs, 51 lifting factors Z_c having values of up to 384, and eight sets of circulants [1], which facilitate the support of a wide

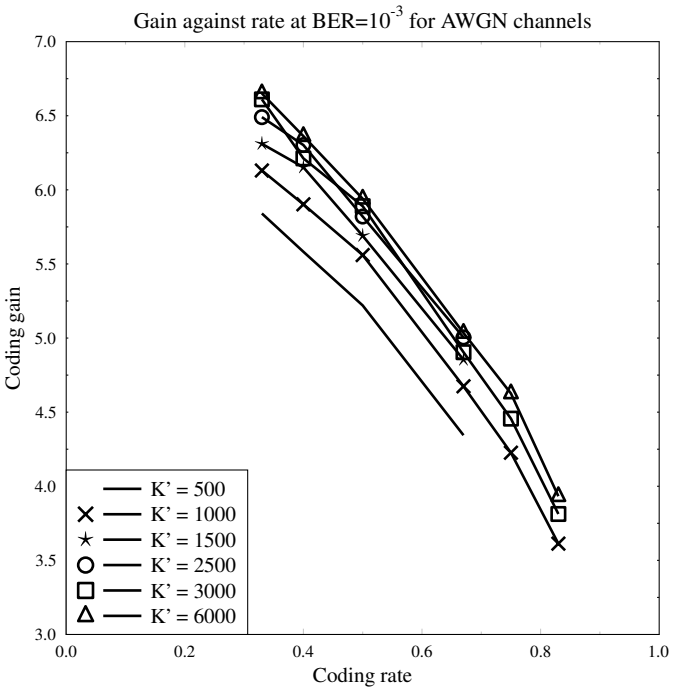


FIGURE 4.16: Coding gain against coding rate for different NR LDPC information block length K' at BER of 10^{-3} over **AWGN channel** compared to the uncoded QPSK modulation.

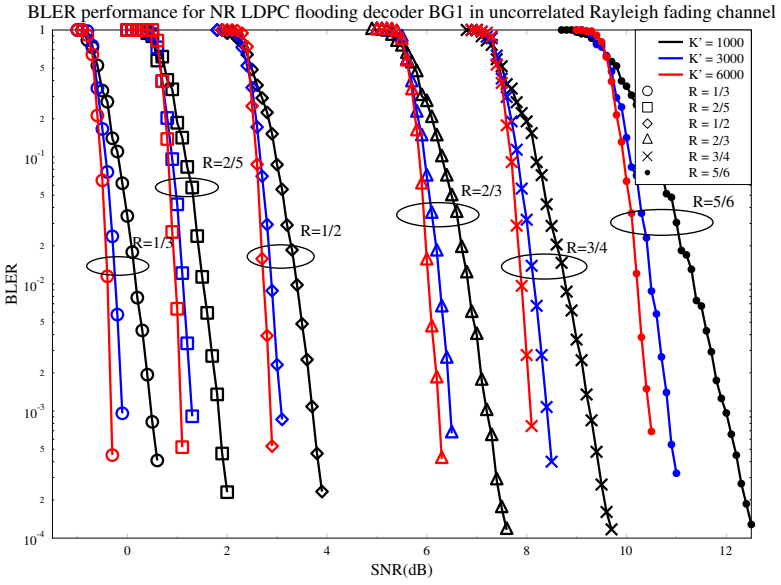


FIGURE 4.17: Pairs of simulation results to show the BLER performance of flooding NR LDPC decoder with **BG1 in the uncorrelated Rayleigh fading channel**. Each plots are using different coding rates R with the same combination of information block length K', including 1000, 3000 and 6000. The different value of R for plots are: $R = 1/3$, $R = 2/5$, $R = 1/2$, $R = 2/3$, $R = 3/4$ and $R = 5/6$.

variety of different combinations of block lengths K' and coding rates R. Each BG is a binary matrix of 0s and 1s that governs the formation of the PCM used at runtime

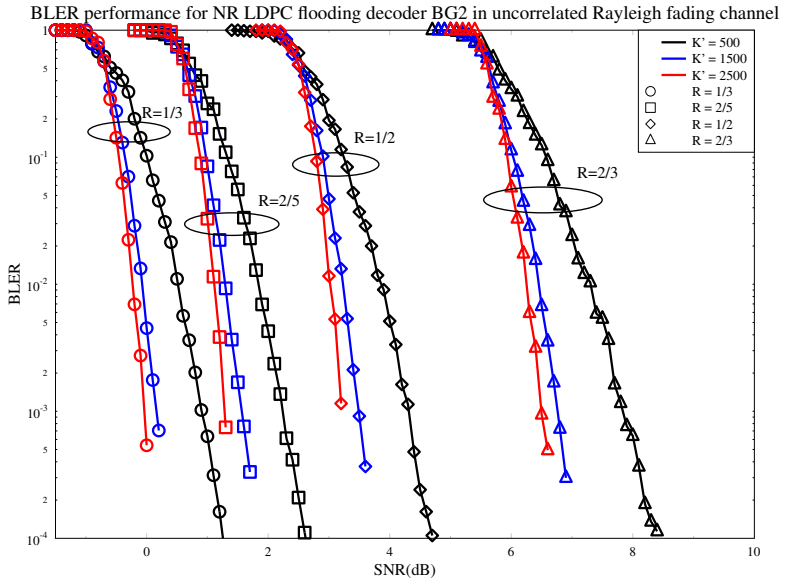
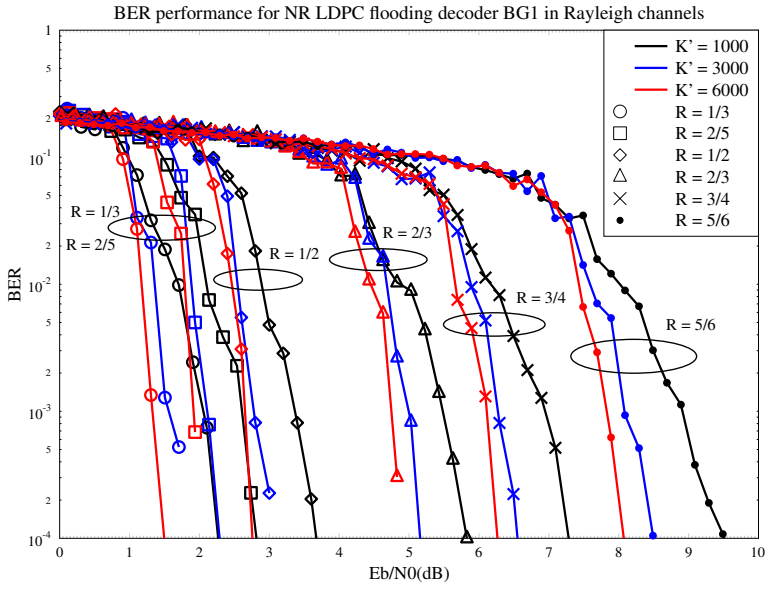


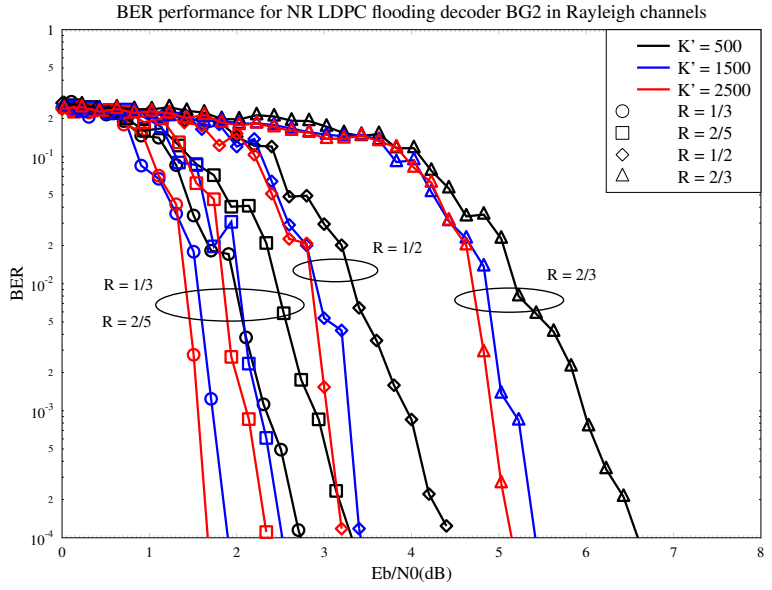
FIGURE 4.18: Pairs of simulation results to show the BLER performance of flooding NR LDPC decoder with **BG2 in the uncorrelated Rayleigh fading channel**. Each plots are using different coding rates R with the same combination of information block length K', including 500, 1500 and 2500. The different value of R for plots are: $R = 1/3$, $R = 2/5$, $R = 1/2$ and $R = 2/3$.

during LDPC encoding or decoding. The two BGs of NR LDPC namely BG1 and BG2, are comprised of four sub-matrices, namely the LDPC matrix, the LDGM matrix, an identity matrix and a zero matrix [31], where Figure 4.21 presents a simplified version of BG2 accordingly. As shown in Figure 4.21, BG2 has $N_c \leq 42$ rows and $N_v \leq 52$ columns, depending on the coding rate R selected at run-time, as it will be detailed below. The LDPC sub-matrix comprises the first $N_{cc} = 4$ rows and the first $N_{vc} = 14$ columns of BG2. Below the LDPC matrix we portray the LDGM matrix, which has $N_{ce} \leq 38$ rows and $N_{vc} = 14$ columns, depending on the coding rate R. Compared to the LDPC sub-matrix, it may be observed that the LDGM sub-matrix has a much sparser distribution of binary values of 1. To the right of the LDPC matrix is a zero matrix, which has all values set to zero and comprises $N_{cc} = 4$ rows as well as $N_{ve} \leq 38$ columns. To the right of the LDGM matrix is an identity matrix, having $N_{ve} \leq 38$ rows and the same number of $N_{ce} \leq 38$ columns. By contrast, BG1 comprises $N_c \leq 46$ rows and $N_v \leq 68$ columns, which has different parameters that are decomposed as $N_{vc} = 26$, $N_{ve} \leq 42$, $N_{cc} \leq 4$ and $N_{ce} \leq 42$.

As mentioned above, the NR LDPC code has a design that supports a flexible coding rate R for supporting adaptive coding and modulation in the face of time-varying channel conditions. More specifically, rate matching is achieved using puncturing in a manner that changes the BG shown in Figure 4.21. To elaborate further, the coding rate $R = K/(N_v - 2)$ is related both to the number of information bits, which is proportional to



(a)



(b)

FIGURE 4.19: Pairs of simulation results to show the BER performance of flooding NR LDPC decoder with **the uncorrelated Rayleigh fading channel** in E_b/N_0 scale. Plot (a) used the same parameters as Figure 4.17. While plot (b) used the same parameters as Figure 4.18.

the number of systematic columns $K = N_v - N_c$, and to the number of encoded bits, which is proportional to $N_v - 2$. Here, the subtraction of 2 is employed since the first two columns of the BG are always punctured for attaining an improved performance [31], regardless of the desired coding rate R. Additional puncturing may also be used for removing a number of extension columns, and an equal number of extension rows that

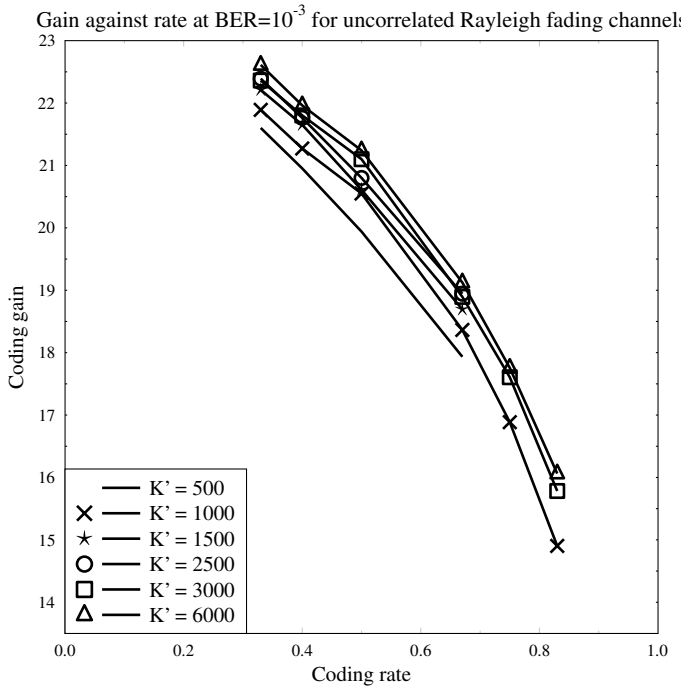


FIGURE 4.20: Coding gain against coding rate for different NR LDPC information block length K' at BER of 10^{-3} over **uncorrelated Rayleigh channel** compared to the uncoded QPSK modulation.

share binary 1 values with these columns in the identity matrix. In this way, N_v may be reduced below 68 in the case of BG1 and below 52 in the case of BG2, hence increasing the coding rate $R = K/(N_v - 2)$.

In order to illustrate the MI exchange in the NR LDPC decoder more comprehensively, this chapter introduces a novel re-interpretation of the factor graph, as shown in Figure 4.22 for the example of BG2. In contrast to the conventional factor graph of an LDPC decoder comprising one group of VNs and CNs [120], Figure 4.22 arranges the VNs and CNs into four groups, namely core VNs, core CNs, extension VNs and extension CNs. More specifically, the $N_{vc} = 14$ VNs included in the LDPC and LDGM matrices of BG2 comprise the core VNs, while the $N_{cc} = 4$ CNs considered by the LDPC and zero matrices comprise the core CNs. Hence, the remaining $N_{ve} \leq 38$ VNs starting from column 15 are the extension VNs, while the $N_{ce} \leq 38$ CNs starting from row 5 are extension CNs, as shown in Figure 4.21. In total, there are $N_v \leq 52$ VNs and $N_c \leq 42$ CNs, in correspondence with the number of columns and rows in BG2, as described above. By considering the effect of puncturing as mentioned above, the number of extension VNs and CNs will be reduced by the application of rate-matching. In the rearranged factor graph of Figure 4.22, the $N_{cc} = 4$ core CNs are positioned at the left-hand side, since they are only connected to the core VNs. By contrast, the $N_{ce} \leq 38$ extension CNs are placed on the right-hand side, where they are connected not only to the core VNs, but

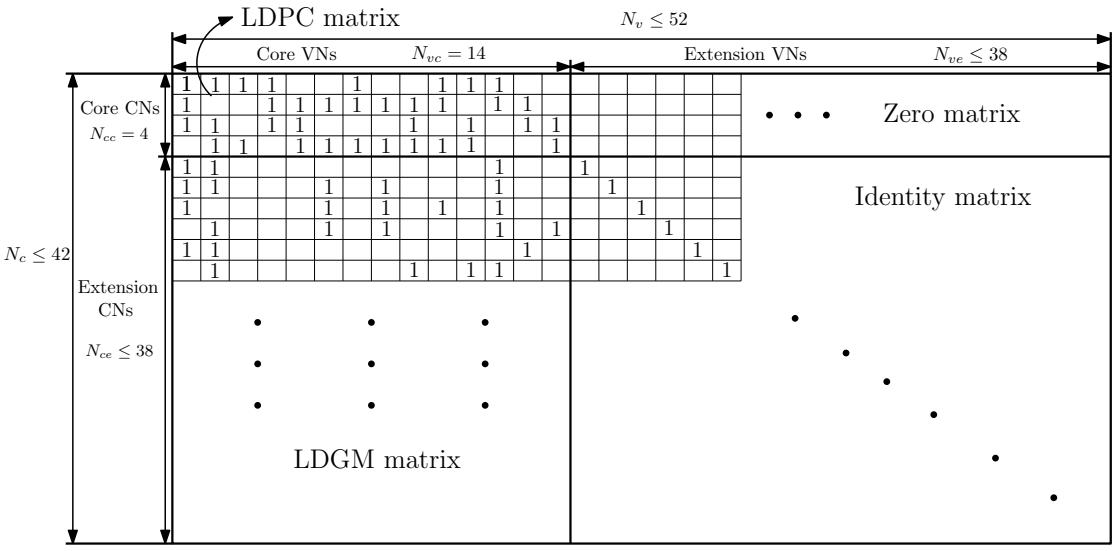


FIGURE 4.21: Structure of BG2 of 5G NR LDPC code. Note that for BG 1 $N_{vc} = 26$, $N_{ve} \leq 42$, $N_{cc} = 4$, $N_{ce} \leq 42$, $N_c \leq 42$ and $N_v \leq 68$ in the absence of puncturing.

also to a single corresponding extension VN each. Finally, the core VNs are positioned in the middle, where they are connected to both sides. In all cases, the edges between the VNs and CNs correspond to the ones shown in Figure 4.21. For example, the 1 in the top 1 left element of Figure 4.21 indicates that V_{c1} and C_{c1} are connected, as shown in Figure 4.22 as well. Additionally, the number of connections spanning from a CN to the VNs or vice versa is referred to as the degree of the respective node, as shown for some examples in Figure 4.22. For instance, there are 8 different VNs connected to C_{c1}, which therefore has a degree of $d_{cc1} = 8$.

With the aid of the BG and factor graph introduced above, the PCM of a NR LDPC decoder can be directly constructed. Generally speaking, the PCM is a "lifted" version of the BG exemplified in Figure 4.21, which expands it to support long information block lengths. As mentioned above, the NR LDPC supports 51 different values for the lifting factor Z_c having values of up to 384. Furthermore, the number of information bits is given by $K = 22 \times Z_c$ for BG1 and $K = 10 \times Z_c$ for BG2. In order to obtain the PCM, the BG exemplified in Figure 4.21 is enlarged by a factor of Z_c, by transforming each 1-valued element in Figure 4.21 into a $Z_c \times Z_c$ rotated identity matrix. Here, the particular rotation applied for each 1-valued element of the BG is selected depending on the value of the lifting factor Z_c. A similar effect may be observed in the factor graph exemplified in Figure 4.22. More specifically, each VN and CN in the BG factor graph corresponds to a set of Z_c VNs or CNs in the PCM factor graph. Furthermore, each edge in the BG factor graph corresponds to a set of Z_c edges in the PCM factor graph, where the corresponding rotation dictates, which of the Z_c VNs is connected to which of the Z_c CNs.

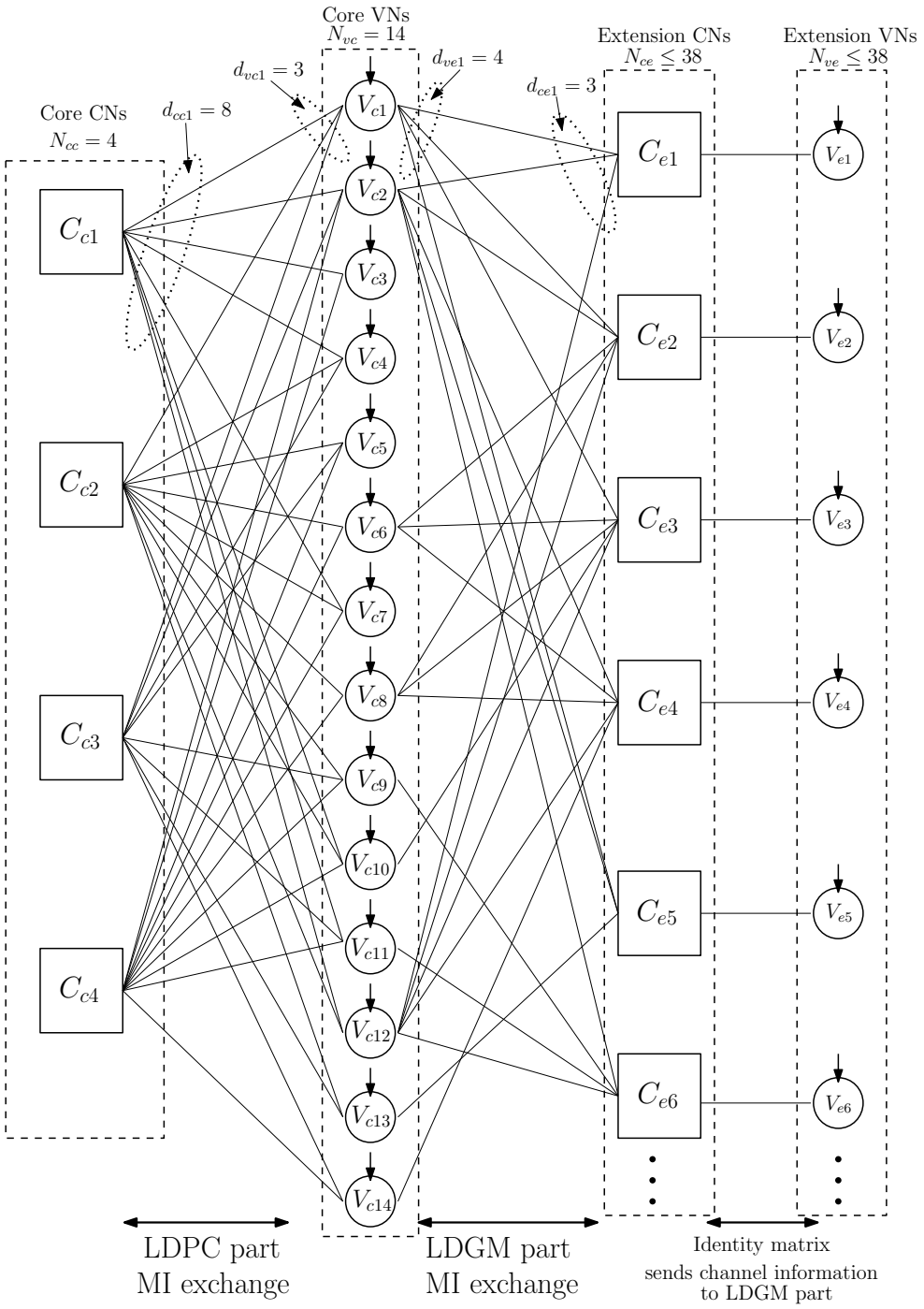


FIGURE 4.22: The factor graph of Base Graph 2 in the case where the number of extension nodes is $N_{ve} = 6$.

4.4.2 EXIT analysis of NR LDPC

In this section, several different forms of EXIT chart analysis techniques are applied to the NR LDPC decoder. Section 4.4.2.1 provides an overview of the EXIT chart from

the perspective of the conventional LDPC decoder. Section 4.4.2.2 characterises the conventional two-dimensional EXIT analysis of the NR LDPC decoder, where we detail why the MI exchange is not characterised correctly. Motivated by this, Section 4.4.2.3 will present the proposed three-dimensional EXIT chart analysis based on the novel factor graph representation of Figure 4.22. For the sake of convenience, we introduce a novel 2D projection of our 3D EXIT chart in Section 4.4.2.4, which gives more direct observation of EXIT chart tunnel, without a complex combination of 3D surfaces. In particular, we show that this projection solves the MI exchange failure we observed in the conventional 2D EXIT chart analysis.

4.4.2.1 Introduction to EXIT chart analysis

The iterative decoding constitutes a powerful technique of achieving near-capacity performance in error-correcting channel decoders. In the process of iterative decoding, each successive decoding iteration improves the quality of the *extrinsic* information exchanged between two or more parts of the decoder, until it converged to unity. As shown in [24], the MI transfer characteristic can be used for visualising the exchange of *extrinsic* information between the constituent decoders. Furthermore, the process of exchanging the *extrinsic* information is strikingly illustrated by the EXIT chart. In [121], an efficient iterative decoder has been developed based on EXIT chart analysis by terminating iterations once a near-unity MI is attained to avoid further futile iterations in the interest of mitigating both the complexity and the power dissipation. In [122], EXIT charts were used to analyse the performance of turbo decoders. While in this section, our main objective is to introduce the EXIT chart analysis of the conventional LDPC decoder.

In order to use EXIT charts for the analysis of an iterative decoder, the decoder has to be of soft-in soft-out nature relying on LLRs, which were firstly introduced by Robertson [123] to describe the information transfer between constituent decoders. The LLR of a bit b_k is specified as

$$L(b_k) \triangleq ln\left(\frac{P(u_k=0)}{P(u_k=1)}\right), \tag{4.5}$$

which is given by the logarithm of the probability ratio of the values of b_k. Since the $P(u_k=0)$ is the numerator in (4.5), an $L(b_k)$ value higher than 0 indicates that the bit u_k is more likely to have value of 0 and vice versa.

According to the [26], the EXIT chart of conventional regular LDPC decoders can be conducted readily described by the parity-check equations. As shown in Figure 4.23, a conventional LDPC decoder is constituted by a Variable Node Decoder (VND) and a Check Node Decoder (CND). At the beginning, the LLRs I_H output by the demodulator

are entered into the VND. Next, the so-called *extrinsic* information I^e_{VND} is generated by the VND and forwarded to the CND in Figure 4.23. After rearranging the LLRs by interleaver of Figure 4.23, this *extrinsic* information is received by the CND as the *a priori* information I^a_{CND}. Since the *extrinsic* information I^e_{VND} has been provided for the CND, the Logarithmic Sum-Product Algorithm (Log-SPA) or the Min-Sum Algorithm (MSA) method may be used by the CND to generate I^e_{CND} [124]. The *extrinsic* information output by CND will be forwarded through the interleaver of Figure 4.23 to the VND as I^e_{CND}. The conventional EXIT chart of the LDPC decoder is hence described by I^a_{VND}, I^e_{VND}, I^a_{CND} and I^e_{CND}.

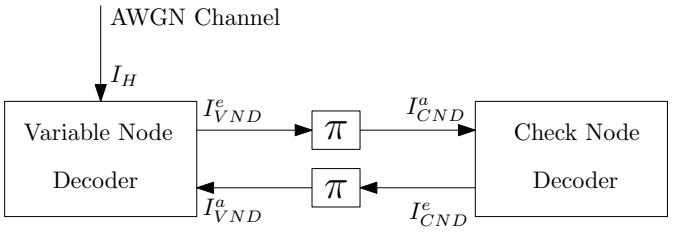


FIGURE 4.23: The block diagram of conventional LDPC decoder for the analysis of 2D EXIT chart.

In order to verify the method of EXIT chart generation in [26], a pair of simulations that model the MI exchange behaviour of CND and VND have been activated individually. Figure 4.24 presents the schematics of the EXIT analysis simulation for both CND and VND in a conventional regular LDPC decoder. As shown in Figure 4.24 (a), in the CND simulation, the information bit sequence $\mathbf{u}$ will be encoded by the generator matrix $\mathbf{G}$. The encoded bit sequence $\mathbf{c}$ will be repeated d_c times in order to imitate the LLRs passed through the edges that are connected with VND. These repeated bits will be interleaved and demoted as bit sequence $\mathbf{p}$. Next, the bit sequence $\mathbf{p}$ and the input MI $I^a(\mathbf{p})$ of the same size of $\mathbf{p}$ across the range of [0,1] will be operated to generate Gaussian distributed *a priori* LLRs $I^a_{CND}(\mathbf{p})$. Since the CND is only connected with VND, the CND operation can be completed with the *a priori* LLRs $I^a_{CND}(\mathbf{p})$ to generate the *extrinsic* LLRs $I^e_{CND}(\mathbf{p})$. The final result of the *extrinsic* MI I^e_{CND} that output from 'Measure MI' block is calculated by both averaging and the histogram method. By increasing the input *a priori* MI $I^a(\mathbf{p})$ from 0 to 1, we are able to obtain a smooth curve of the *extrinsic* MI I^e_{CND} of CND as a function of $I^a(\mathbf{p})$.

For the simulation of VND as is shown in Figure 4.24 (b), besides the same components in CND simulation, the channel information should be considered as well. After the encoded bit sequence $\mathbf{c}$ is generated, it will pass through the BPSK modulator, the AWGN transmission and the BPSK soft demodulator. Finally, the received LLRs $\tilde{\mathbf{c}}$ effected by Gaussian noise are applied to the VND together with *a priori* LLRs $I^a_{VND}(\mathbf{r})$. The generated *extrinsic* LLRs $I^e_{VND}(\mathbf{r})$ will be used to calculated the *extrinsic* MI I^e_{VND} of the VND by both averaging or histogram methods. Again, by increasing the input

a priori MI $I^a(\mathbf{r})$ from 0 to 1, a smooth curve of the *extrinsic* MI I^e_{VND} as a function $I^a(\mathbf{r})$ of will be plotted.

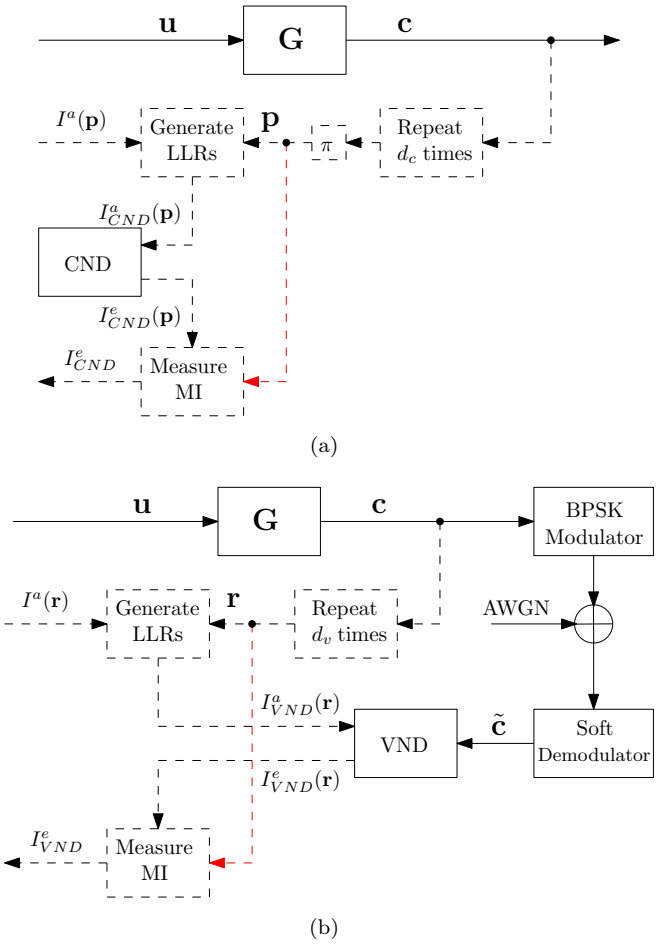


FIGURE 4.24: Schematics of the simulation process of conventional LDPC decoder EXIT chart analysis, including schematic for (a) CND and (b) VND.

The EXIT charts are presented as Figure 4.25, which are exactly the same to the EXIT curves of VND and CND in [26]. These plots compared three methods of EXIT chart analysis, including using equations that mentioned by [26] and running simulations for averaging and histogramming the *extrinsic* MI. As shown in Figure 4.25, all plots with the same parameters are matched with each other, which proves that our equation is able to draw the correct EXIT curves.

Since the EXIT curve of VND and CND has been calculated, the whole EXIT chart of certain regular LDPC decoder is easily plotted. For example, Figure 4.26 gives the EXIT chart for a regular LDPC decoder that has $d_c = 4$ as the degree of CNs and $d_v = 4$ as the degree of VNs. The transmission environment of Figure 4.26 is adopted the same parameter as Figure 4.25, which is SNR = 1dB and coding rate of $R = 1/2$. The choice of this half-coded conventional regular LDPC decoder is aimed for a convenient

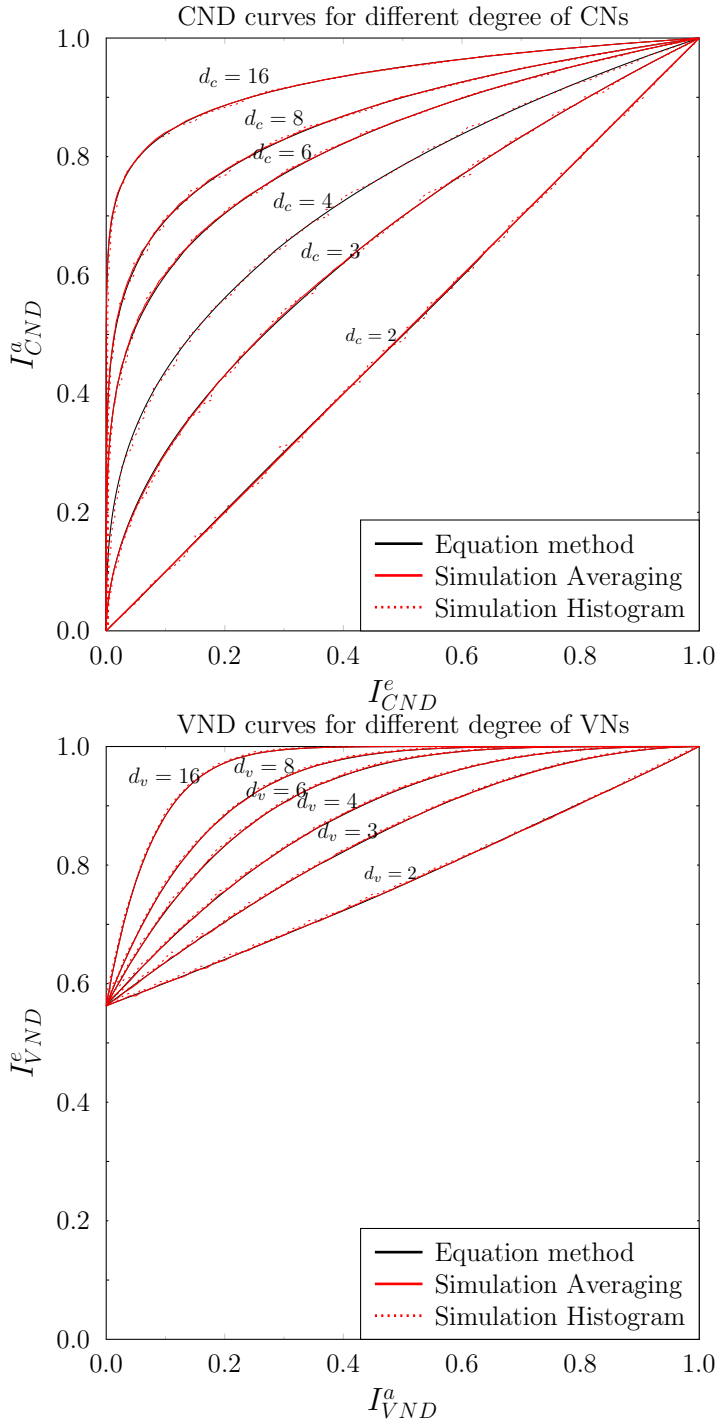


FIGURE 4.25: The simulated EXIT curves for (a) CND and (b) VND in the conventional regular LDPC decoder with different degrees of check nodes and variable nodes with EsN0 = 1dB and coding rate $R = 1/2$.

and straight-forward EXIT chart analysis. In Figure 4.26, the red curve represents the EXIT characteristic of VND that has $d_v = 4$, which is the same curve in Figure 4.25 (a). For the VND curve, the *extrinsic* information I^e_{VND} that output from VND in

Figure 4.23 is dependant on the *a priori* information I^a_{VND} that input to the VND and the channel input I_H. Therefore, the x-axis is determined as I^a_{VND}, while I^e_{VND} is decided as the y-axis for Figure 4.26. Meanwhile, the blue curve in this plot represents the EXIT characteristic of CND that has $d_c = 4$, which is the same corresponding curve in Figure 4.25 (a). As shown in Figure 4.23, the *extrinsic* information I^e_{VND} that output from VND is interleaved and become the *a priori* information I^a_{CND} of CND. Hence, the y-axis of Figure 4.26 is considered as I^a_{CND} when the EXIT curve of CND is drawing. Similarly, the *a priori* information I^a_{VND} of VND is passed and interleaved from the CND as its output *extrinsic* information I^e_{CND}. Therefore, the x-axis of Figure 4.26 is denoted as I^e_{CND} when the CND EXIT curve is plotting. In order to show the MI exchange between CND and VND, the stair-case-shaped trajectory is plotted in Figure 4.26 as well. This trajectory is taken as the average value of MI from 50 times of simulation for the employed regular LDPC decoder. As our expectation, the trajectory is approximately bounded within the EXIT curves, which validates this EXIT chart.

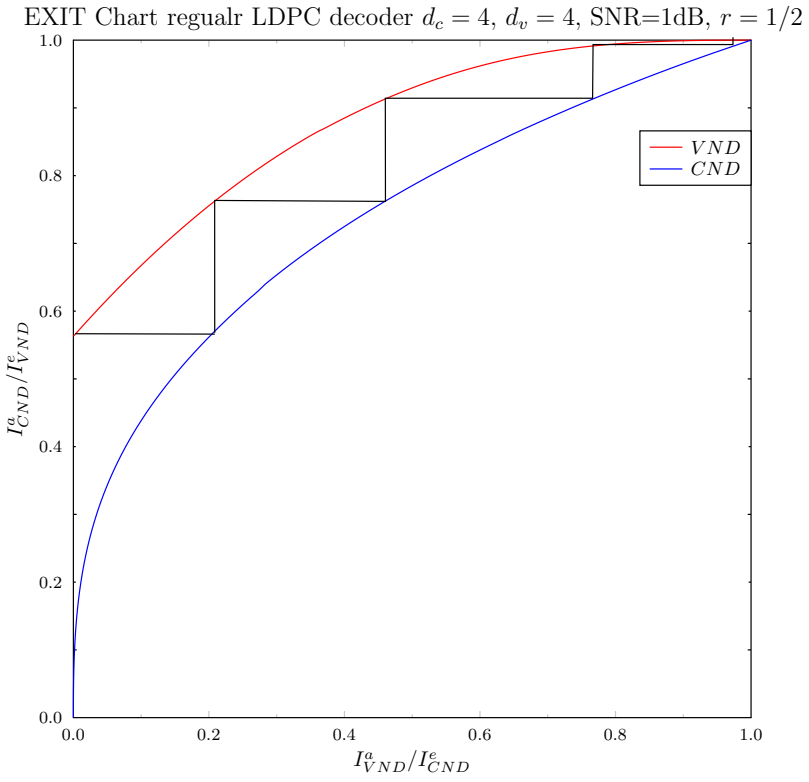


FIGURE 4.26: The simulated EXIT chart for a regular LDPC decoder that has $d_c = 4$ and $d_v = 4$ for EsN0 = 1dB and $R = 1/2$.

4.4.2.2 Conventional two-dimensional EXIT analysis

As discussed in Section 4.4.1, the NR LDPC code has a structure which is different from that of a conventional LDPC code. Motivated by these unique characteristics of the NR LDPC code, we rearranged the factor graph into the novel form presented in Figure 4.22. However, in the conventional view of LDPC decoding and EXIT chart analysis, the core and extension VN decoders should be considered as a single integrated VND. Likewise, the core and extension CN decoders are conventionally considered as a single integrated CND. More specifically, a block diagram of the conventional view of an LDPC decoder used for EXIT chart analysis is shown in Figure 4.27, in which the MI is exchanged between the integrated CN and VN decoders. Here, the information passed from the VNs to the CNs is characterised by the MI I_v^e and regarded as the *extrinsic* information provided by the VN decoder of Fig 4.27, which is characterised by the MI I_c^a and becomes the *a priori* information entered into the CN decoder of Fig 4.27. Similarly, the information passed from the CNs to the VNs is characterised by the MI I_c^e and considered as the *extrinsic* information generated by the CN decoder, which is characterised by the MI I_v^a and regarded as the *a priori* information provided for the VN decoder. Additionally, the VN decoder receives the channel's output information, which is characterised by MI I_H and remains constant throughout the decoding iterations.

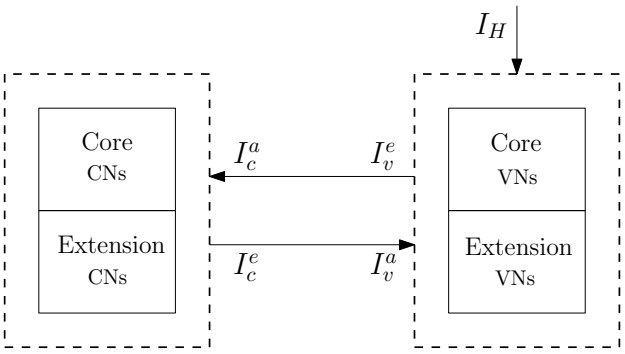


FIGURE 4.27: The block diagram of NR LDPC decoder relying on the conventional structure for analysis using 2D EXIT charts.

The conventional 2D EXIT chart, may be drawn using the semi-analytic method of [26], where the equations may be adjusted for adopting our notation for the NR LDPC code, as presented in (4.6) and (4.7) below:

$$I_v^e\left(I_v^a, I_H\right) = \sum_{i=1}^{N_{vc}+N_{ve}} \frac{J\left(\sqrt{(d_{vc_i}+d_{ve_i}-1)[J^{-1}(I_v^a)]^2+[J^{-1}(I_H)]^2}\right)\cdot(d_{vc_i}+d_{ve_i})}{\sum_{i=1}^{N_{vc}+N_{ve}}(d_{vc_i}+d_{ve_i})} \tag{4.6}$$

$$I_c^e\left(I_c^a\right) = \sum_{i=1}^{N_{cc}+N_{ce}} \frac{\left(1 - J\left(\sqrt{d_{cc_i} + d_{ce_i} - 1} \cdot J^{-1}(1 - I_c^a)\right)\right) \cdot (d_{cc_i} + d_{ce_i})}{\sum_{i=1}^{N_{vc}+N_{ve}} (d_{cc_i} + d_{ce_i})} \tag{4.7}$$

For convenience, we summarise our notation in Table 4.5. Note that the analysis of [26] assumes a flooding-based decoder activation order [13] and the employment of the sum-product decoding algorithm [125]. Here, (4.6) evaluates the average *extrinsic* MI I_v^e generated by the VN decoder as a function of the *a priori* MI I_v^a provided for the VN decoder and the channel's output MI I_H, where $J(\cdot)$ and $J^{-1}(\cdot)$ are empirical functions, as characterised in the Appendix of [26]. In order to support the irregular construction of the NR LDPC PCM, (4.6) considers the degree distribution of different VNs.

Likewise, (4.7) computes the average *extrinsic* MI I_c^e generated by the CN decoder as a function of both the *a priori* MI I_c^a that entered into the CN decoder and of the degrees of the various CNs.

TABLE 4.5: Table of notations and corresponding definitions in Equation (4.6) and (4.7).

Notation	Definition
N_{vc}/N_{ve}	Number of VNs in core/extension part
N_{cc}/N_{ce}	Number of CNs in core/extension part
d_{vci}/d_{vei}	Degree of ith core/extension VN
d_{cci}/d_{cei}	Degree of ith core/extension CN
I_v^a/I_v^e	The *a priori*/*extrinsic* MI of VN decoder
I_c^a/I_c^e	The *a priori*/*extrinsic* MI of CN decoder
I_H	The MI of channel information

Figure 4.28 presents four different conventional 2D EXIT charts obtained using (4.6) and (4.7), where I_v^a and I_c^e are presented on the x-axis, while I_v^e and I_c^a are scaled on the y-axis, since the output of the CND becomes the input of the VND and vice versa during iterative decoding. Figure 4.28 characterises the NR LDPC code having a coding rate of $R = 1/3$ and a block length of $K' = 3000$. Here, Figure 4.28 (a) and (c) adopt BG 1, but different channel SNRs of -1 dB and -3 dB. Likewise, Figure 4.28 (b) and (d) adopt BG 2 and the same pair of channel SNRs. This is motivated by the fact that an SNR of -1 dB is sufficient for a 1/3 coding rate NR LDPC decoder to reach a BLER of 10^{-4}, while the BLER for the same decoder at -3 dB is close to 1.

In each EXIT chart of Fig 4.28, there are two crossing curves, each of which exclusively depends on the operation of the respective VN or CN decoders according to the EXIT function of (4.6) and (4.7), respectively. In the process of iterative LDPC decoding using a conventional flooding schedule, LLRs are exchanged between the CN and VN decoders, as may be characterized by measuring the MIs I_v^a/I_c^e and I_v^e/I_c^a after each operation of

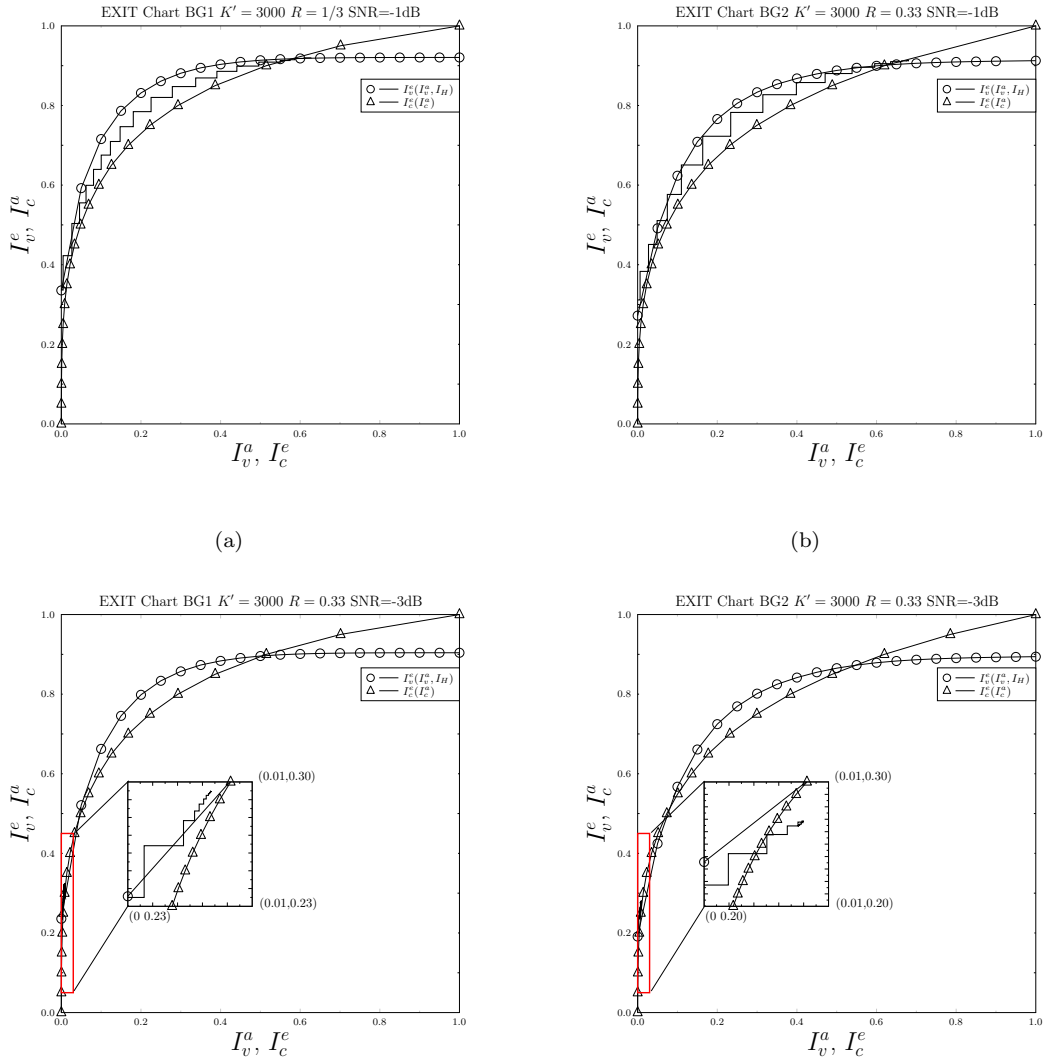
the VND and CND. These measurements may be plotted as a stair-shaped line termed as a trajectory [126] between the VND and CND curves of Figure 4.28. Each successive step in this trajectory characterises the evolution of each successive MI after a single decoding iteration.

Although the strong error-correction performance of the NR LDPC decoder has been confirmed by BLER simulation [31], conventional EXIT chart analysis suggests that the MI exchange in the decoder cannot fully converge to the $(I_c^e, I_v^e) = (1, 1)$ point at the top-right corner of Figure 4.28, which is typically assumed to be a prerequisite for achieving an almost infinitesimally low BLER. However, in the case of VNs having a degree of $d_{vc_i} + d_{ve_i} = 1$, the numerator of (4.6) will become equal to I_H, preventing an I_v^e of 1, even when I_v^a has a value of 1. Hence, it is the presence of degree-1 VNs [127] in the NR LDPC code that prevents convergence to the $(1, 1)$ point in the conventional EXIT chart analysis.

It may however be observed that somewhat unexpectedly, the NR LDPC decoder is capable of attaining a low BLER at channel SNRs that are associated with a closed EXIT chart tunnel. However, a closed EXIT tunnel in conventional 2D EXIT chart always means that the decoder is unable to converge to low BLER performance. The conflict between the BLER simulation results and conventional 2D EXIT chart analysis reveals that a new EXIT chart analysis technique is needed for analysing the MI exchange of the NR LDPC decoder. Motivated by this, we propose a novel technique for the EXIT chart analysis of the NR LDPC decoder, which is discussed in Sections 4.4.2.3 and 4.4.2.4.

4.4.2.3 Proposed three-dimensional EXIT chart analysis

Motivated by the deficiency of the conventional 2D EXIT chart demonstrated in Figure 4.28, a more sophisticated EXIT chart technique is proposed in this section for analysing the MI exchange of the NR LDPC decoder. The proposed method is based on the reorganized factor graph of Figure 4.22, which is characterised using the block diagram of Figure 4.29 for illustrating the MI exchange. Similar to Figure 4.22, Figure 4.29 is organized by relying on the core CNs, core VNs, extension CNs and extension VNs positioned from left to right. The MI provided for the core CNs is referred to as the *a priori* information (I_{cc}^a) of the core CN, which is provided by the *extrinsic* information forwarded from the core VNs to the core CNs (I_{vc}^e). Similarly, the MI (I_{cc}^e) provided by the core CNs of Figure 4.29 is referred to as the *extrinsic* information output of the core CNs, which is entered as the *a priori* information (I_{vc}^a) into the core VNs. Finally, the MI that is received from the channel is termed as I_H. The other MI terms of Figure 4.29 adopt a similar terminology to those described above, as summarised in Table 4.5 and 4.6. Note that the information exchange between the extension VNs and



(a) (b)

(c) (d)

FIGURE 4.28: Conventional 2D EXIT charts for a $K' = 3000$ bit 5G NR LDPC flooding decoder with $R = 1/3$ and different combinations of SNR and base graph, namely: (a) SNR = -1 and BG 1, (b) SNR = 1 and BG 2, (c) SNR = -3 and BG 1, (d) SNR = -3 and BG 2. Each trajectory is a snapshot obtained during the decoding of a single codeblock.

extension CNs is different from the others, since the extension VNs has a degree of 1. Owing to this, the extension VNs similarly pass on the channel's output information I_H, as shown in Figure 4.29. Therefore, the MI exchange between the extension VNs and extension CNs would not be improved by iterative decoding.

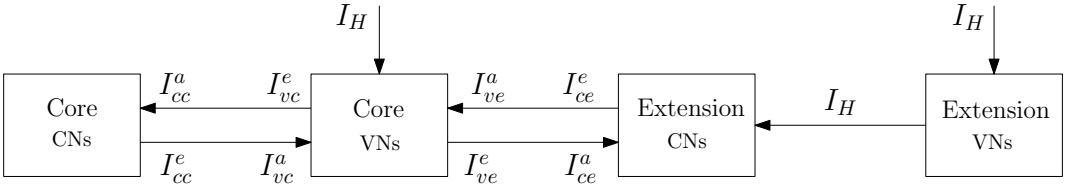


FIGURE 4.29: The block diagram of NR LDPC decoder with structure according to the factor graph of Figure 4.22.

Similar to the scheme of Figure 4.27, the scheme of Figure 4.29 may be characterised by Equations (4.8), (4.9), (4.10) and (4.11), which are appropriately adapted from [26] to evaluate the *extrinsic* information terms of I^e_{vc}, I^e_{ve}, I^e_{cc} and I^e_{ce}. Note that the activation order of decoding nodes in Figure 4.29 is exactly the same as Figure 4.27, which considers the VND and CND as four separated blocks. Similar to (4.6) and (4.7) related to Figure 4.27, these equations characterize the degree distributions of the corresponding blocks in Figure 4.29. More specifically, the information transmitted from the extension VNs to extension CNs is regarded as the channel's output information that affects the *extrinsic* information (I^e_{ce}) of the extension CN decoder. Therefore, I^e_{cc}, I^e_{ce} is dependent on the channel SNR, which is quantified by I_H.

$$I^e_{vc}(I^a_{vc}, I^a_{ve}, I_H) = \sum_{i=1}^{N_{vc}} \frac{J\left(\sqrt{(d_{vc_i}-1)\cdot\left[J^{-1}(I^a_{vc})\right] + d_{ve_i}\cdot\left[J^{-1}(I^a_{ve})\right] + [J^{-1}(I_H)]^2}\right)\cdot d_{vc_i}}{\sum_{i=1}^{N_{vc}} d_{vc_i}} \tag{4.8}$$

$$I^e_{ve}(I^a_{vc}, I^a_{ve}, I_H) = \sum_{i=1}^{N_{ve}} \frac{J\left(\sqrt{d_{vc_i}\cdot\left[J^{-1}(I^a_{vc})\right] + (d_{ve_i}-1)\cdot\left[J^{-1}(I^a_{ve})\right] + [J^{-1}(I_H)]^2}\right)\cdot d_{ve_i}}{\sum_{i=1}^{N_{ve}} d_{ve_i}} \tag{4.9}$$

$$I^e_{cc}(I^a_{cc}) = \sum_{i=1}^{N_{cc}} \frac{\left(1 - J\left(\sqrt{(d_{cc_i}-1)}\cdot J^{-1}(1-I^a_{cc})\right)\right)\cdot d_{cc_i}}{\sum_{i=1}^{N_{cc}} d_{cc_i}} \tag{4.10}$$

$$I_{ce}^{e}(I_{ce}^{a}, I_H) = \sum_{i=1}^{N_{ce}} \frac{\left(1 - J\left(\sqrt{(d_{ce_i} - 1) \cdot \left[J^{-1}(1 - I_{ce}^{a})\right]^2 + \left[J^{-1}(1 - I_H)\right]^2}\right)\right) \cdot d_{ce_i}}{\sum_{i=1}^{N_{ce}} d_{ce_i}} \tag{4.11}$$

TABLE 4.6: Table of additional notations and corresponding definitions in Equations (4.8), (4.9), (4.10) and (4.11).

Notation	Definition
I_{vc}^{a}/I_{vc}^{e}	The *a priori*/*extrinsic* information of core VNs in respect to core CNs
I_{ve}^{a}/I_{ve}^{e}	The *a priori*/*extrinsic* information of core VNs in respect to extension CNs
I_{cc}^{a}/I_{cc}^{e}	The *a priori*/*extrinsic* information of core CNs
I_{ce}^{a}/I_{ce}^{e}	The *a priori*/*extrinsic* information of extension CNs

According to (4.8) to (4.11), in Figure 4.22 the core VN decoder has the pair of outputs I_{vc}^{e} and I_{ve}^{e}, which are affected not only by I_{vc}^{a} but also by I_{ve}^{a}, giving two independent variables and two dependent variables. Hence they cannot be comprehensively characterized by a 2D EXIT chart. Instead, a pair of 3D EXIT charts may be used to visualize the MI exchange of the NR LDPC decoder. 3D EXIT chart analysis is quite common for three-stage concatenated codes, such as turbo codes in [128] and joint system with mapper and encoder in [129], which given us the idea to introduce the 3D EXIT chart analysis for NR LDPC decoder. Indeed, compared with the conventional 2D EXIT chart, the generation of 3D EXIT charts for NR LDPC decoder requires more computations for the sake of two independent variables. However, thanks to the high complexity, the 3D EXIT charts can consider the MI exchange in the decoder more comprehensively. For example, Figure 4.30 presents two pairs of 3D EXIT charts for an NR LDPC decoder using BG 1 and a coding rate of $R = 1/3$. In Figure 4.30 (a) and (b), the channel SNR is -1 dB, while in Figure 4.30 (c) and (d), the channel SNR is adjusted to -3 dB. Here, one of the 3D EXIT charts characterises the MI exchange between the core CNs and the core VNs of Figure 4.29, which is referred to as the LDPC part MI exchange in Figure 4.22.

By contrast, the other 3D EXIT chart characterizes the MI exchange between the core VNs and the extension CNs of Figure 4.29, which is referred to as the LDGM part MI exchange. More specifically, we propose a pair of 3D EXIT charts having an x-axis that characterises I_{vc}^{a}, which is provided by the *extrinsic* information I_{cc}^{e} output by the core CN decoder. By contrast, the y-axis of both EXIT charts is used to characterise I_{ve}^{a}, which is provided by the *extrinisic* information I_{ce}^{e} output of the extension CN decoder. In one of the 3D EXIT charts, the z-axis characterises I_{vc}^{e}, which becomes the *a priori*

information provided for the core CN decoder I_{cc}^{a}. Hence, the EXIT chart characterises the LDPC part MI exchange between the core VN decoder and the core CN decoder. Furthermore, the z-axis of the other 3D EXIT chart characterises I_{ve}^{e}, which becomes the *a priori* information provided for the extension CN decoder I_{ce}^{a}, characterising the LDGM MI exchange. As shown in Figure 4.30 (a), the surface $I_{vc}^{e}(I_{vc}^{a}, I_{ve}^{a}, I_H)$ characterises the *extrinsic* MI I_{vc}^{e} generated by the core VND entered into the core CND, which is dependent on I_{vc}^{a}, I_{ve}^{a} and I_H. To elaborate a little further, the surface $I_{cc}^{e}(I_{cc}^{a})$ represents the *extrinsic* MI I_{cc}^{e} that the core CND outputs as a function of the *a priori* information I_{cc}^{a}, which is then entered into the core CND of Fig 4.29. Similarly, in Figure 4.30 (b), the surface $I_{ve}^{e}(I_{vc}^{a}, I_{ve}^{a}, I^{H})$ gives the *extrinsic* MI I_{ve}^{e} generated by the core VND and entered into the extension CND, which is dependent on I_{vc}^{a}, I_{ve}^{a} and I^{H}. Furthermore, the surface $I_{ce}^{e}(I_{ce}^{a}, I_H)$ gives the *extrinsic* MI I_{ce}^{e} generated by the extension CND as a function of I_{ce}^{a} and I_H in Figure 4.29.

As exemplified in Fig 4.30, both VN decoder surfaces $I_{vc}^{e}(I_{vc}^{a}, I_{ve}^{a}, I_H)$ and $I_{ve}^{e}(I_{vc}^{a}, I_{ve}^{a}, I_H)$ are affected by the channel SNR. By contrast, only the extension part of the CN decoder surface $I_{ce}^{e}(I_{ce}^{a}, I_H)$ is a function of the channel SNR. Due to the existence of degree-one VNs, the value of $I_{ce}^{e}(I_{ce}^{a}, I_H)$ will be directly affected by the channel's output information I_H, which can be observed from (4.11). Furthermore, the value of $I_{cc}^{e}(I_{cc}^{a})$ is calculated by using the conventional equation (4.10).

Using the EXIT surfaces for the VND and CND of both the core and extension parts, a trajectory representing the MI exchange across the successive iterations of the LDPC decoder may be predicted. Figure 4.31 provides a flow chart for illustrating how the predicted trajectory may be generated. At the beginning of the iterative decoding process, the *a priori* MIs I_{vc}^{a}, I_{ve}^{a}, I_{cc}^{a}, I_{ce}^{a} are all initialised to 0, which means that our trajectory is starting from the (0,0,0) point at the corner of both plots in Figure 4.30. As the first step, the channel's output information I_H is forwarded to the VND. Since both the core CND and extension CND of Figure 4.29 are unaffected by this, the MI values of I_{vc}^{a} and I_{ve}^{a} are kept as 0. However, the MI values of I_{ce}^{a} and I_{cc}^{a} are updated by I_{ve}^{e} and I_{vc}^{e}, which can be determined by the VND surfaces seen in Figure 4.30, according to the function $I_{ve}^{e}(I_{vc}^{a}, I_{ve}^{a}, I_H)$ and $I_{vc}^{e}(I_{vc}^{a}, I_{ve}^{a}, I_H)$. Therefore, the trajectory moves upwards from the (0,0,0) point to the lowest corner of the VND surfaces, as shown using solid lines in both plots of Figure 4.30. Next, the NR LDPC decoder begins its iterative decoding process, which starts by operating the core and extension CNDs of Figure 4.29. The iterative decoding will keep operating until the number of iteration i exceeds the maximum number of iteration i_{max}. Since the VND is not affected, the MI values of I_{ce}^{a} and I_{cc}^{a} remain unchanged from the previous step. However, the MI values of I_{vc}^{a} and I_{ve}^{a} are updated by the output of the CNDs, as characterised by the functions of $I_{cc}^{e}(I_{cc}^{a})$ and $I_{ce}^{e}(I_{ce}^{a}, I_H)$, as well as by updating the position of the trajectories in the plots of

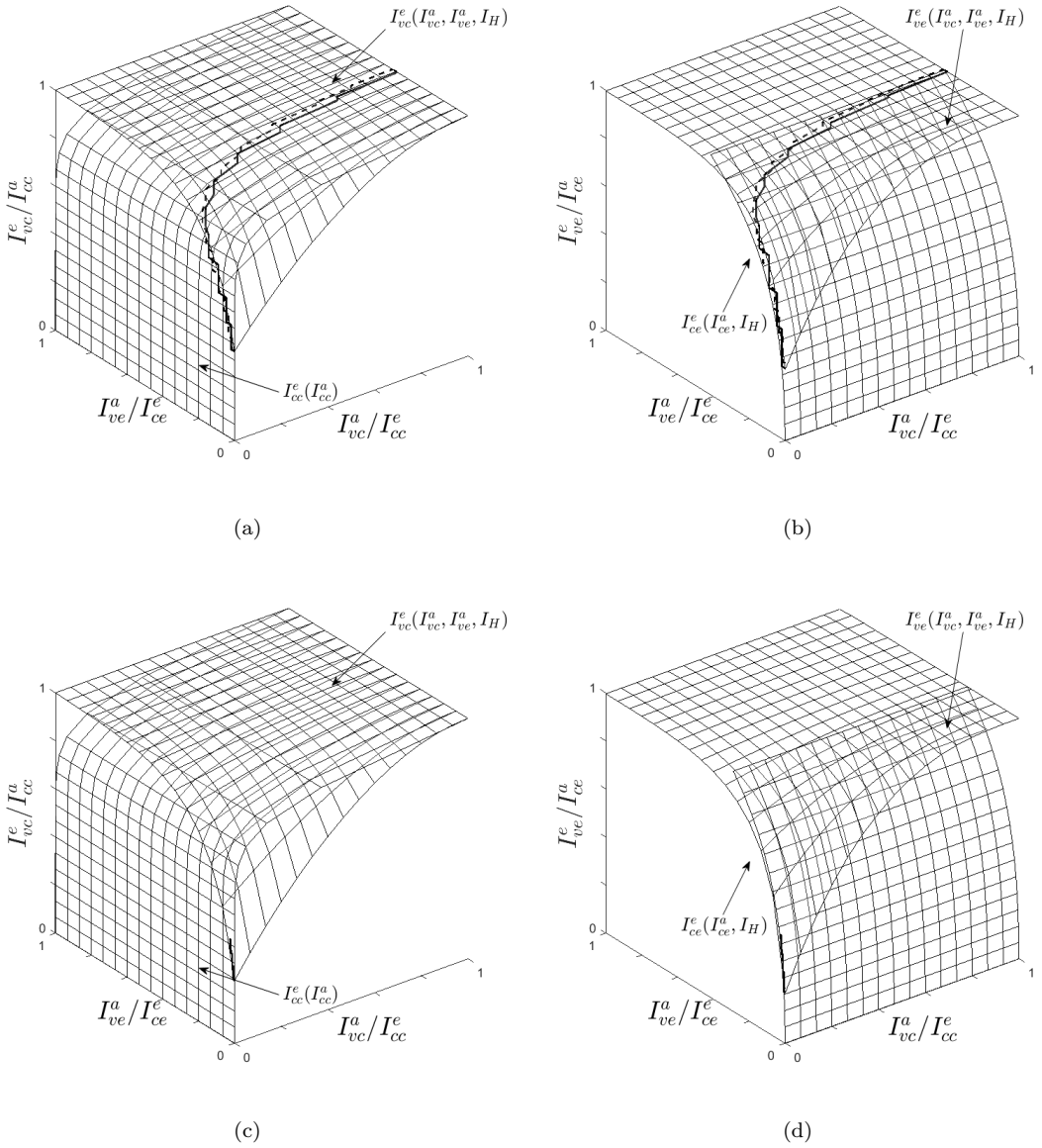


FIGURE 4.30: 3D EXIT chart for a K=3000-bit flooding 5G NR LDPC decoder, using BG 1, R=1/3 and different AWGN channel SNRs, including: (a) LDPC EXIT chart at SNR = -1dB, (b) LDGM EXIT chart at SNR = -1dB, (c) LDPC EXIT chart at SNR = -3dB and (d) LDGM EXIT chart at SNR = -3dB. The continuous line represents to the predicted trajectory that bounced between EXIT surfaces while the dashed line corresponds to the the bit-by-bit Monte-Carlo simulation. The associated activation-order of the components in Figure 4.29 is defined by Figure 4.31.

Figure 4.30 using the CND surfaces. More specifically, the trajectory moves from the corner of the VND surfaces towards the point calculated on the CND surfaces. Following this, the VND may be operated again and similar to the first step, new MI values of I^a_{ce} and I^a_{cc} can be obtained by using the surface functions $I^e_{ce}(I^a_{ce}, I_H)$ and $I^e_{vc}(I^a_{vc}, I^a_{ve}, I_H)$, but this time with the updated values of I^a_{vc} and I^a_{ve} obtained from the previous step.

In this way, the trajectory in both plots of Figure 4.30 moves from the previous position reached on the CND surfaces to the new calculated position on the VND surfaces. As shown in Figure 4.31, these steps will continue until the iterative decoding process is completed. Finally, if the channel SNR is sufficiently high and if a sufficient number of decoding iteration has been completed, then the MI exchange between CND and VND will converge and the MI of I^a_{ce} and I^a_{cc} should approach 1. In summary, the predicted trajectory is generated by bouncing between the EXIT surfaces of Figure 4.30, according to the alternated activation of the VNDs and CNDs. Accordingly, the trajectories of Figure 4.30 have been predicted and plotted as solid lines. In order to validate this prediction, we also introduced simulated trajectories in Figure 4.30 (a) and (b) as dashed lines. These are obtained by performing an iterative decoding process using the alternated activation of the VNDs and CNDs, while measuring the MI of the LLRs exchanged after each step. As shown in Figure 4.30, a good match is observed between the solid and dashed trajectories in each case, configuring the accuracy of the proposed approach.

In contrast to the conventional 2D EXIT chart analysis of Section 4.4.2.2, both 3D EXIT functions of the VN decoders $I^e_{vc}(I^a_{vc}, I^a_{ve}, I_H)$ and $I^e_{ve}(I^a_{vc}, I^a_{ve}, I_H)$ are able to reach $I^e_{vc} = 1$ and $I^e_{ve} = 1$. Therefore, using the proposed 3D EXIT chart, the open/closed state of the EXIT chart's tunnel is a function of the channel SNR, which affects all the functions $I^e_{vc}(I^a_{vc}, I^a_{ve}, I_H)$, $I^e_{ve}(I^a_{vc}, I^a_{ve}, I_H)$ and $I^e_{ce}(I^a_{ce}, I_H)$. For instance, the trajectory shown in Figure 4.30 (a) and (b) reveals that the iterative decoding process can converge to $I^e_{vc} = 1$ and $I^e_{ve} = 1$ at an SNR of -1dB in AWGN channel. In this scenario, the tunnel can be considered to be open, which means that the decoder is expected to obtain low BLER. However, if the channel SNR is decreased to -3 dB as shown in Figure 4.30 (c) and (d), the trajectory will be curtailed at the intersection of the EXIT functions before reaching $I^e_{vc} = 1$ and $I^e_{ve} = 1$. This behaviour reveals that the NR LDPC decoder is incapable of decoding the received bit sequence successfully at a channel SNR of -3 dB. It is also been proved by Figure. 4.13 in Section. 4.3.1 that NR LDPC decoder with same coding parameter can achieve an error-free decoding output with SNR = -1dB. By contrast, the BLER performance of this decoder is relatively poor when SNR is -3dB. In conclusion, the proposed 3D EXIT chart accurately characterizes the MI exchange within the NR LDPC decoder and determines whether the decoder is capable of attaining a low BLER at a prescribed channel SNR.

4.4.2.4 Two-dimension projection on EXIT chart of NR LDPC Decoder

As presented in Section 4.4.2.3, our proposed 3D EXIT charts can determine, whether the EXIT tunnel is open or closed, but this requires simulation of the stair-case shaped

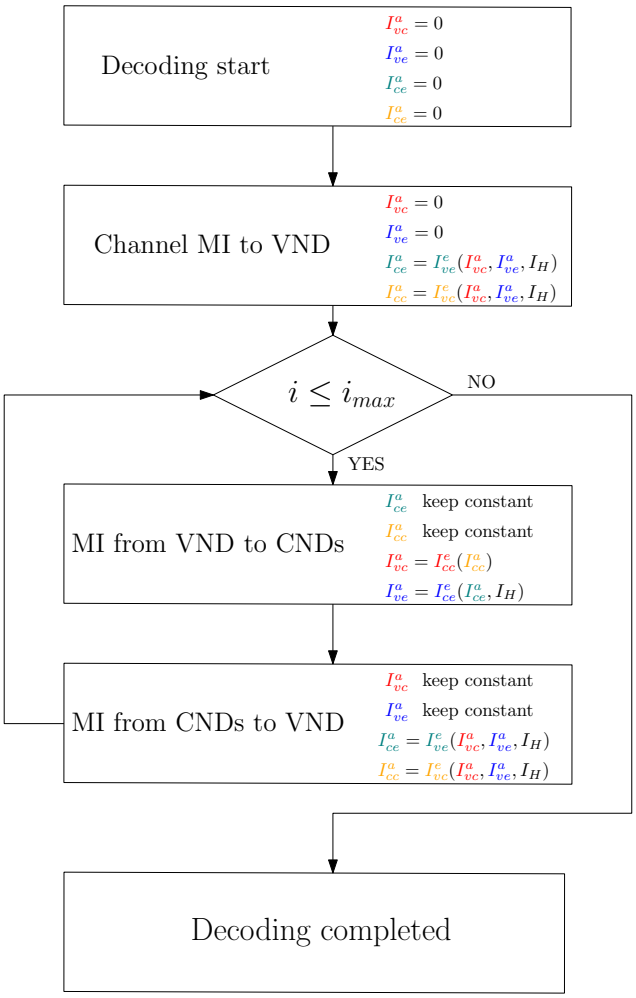


FIGURE 4.31: Flow chart for calculating the NR LDPC decoder's decoding trajectory. Formally defining the activation-order of the components in Fig 4.29.

decoding trajectory and is not immediately apparent from visual inspection. In order to present this result in a straight-forward manner, in this section we propose a 2D projection of our novel 3D EXIT charts, as shown in Figure 4.32 and inspired by [32]. As discussed in Section 4.4.2.2, the failure of conventional 2D EXIT charts to accurately predict the NR LDPC code's performance is due to the degree-one extension VNs, which prevent the extension CNs from providing *extrinsic* LLRs having a mutual information of 1. Hence, in Figure 4.32, the extension CNs and extension VNs are combined with the core VNs, in order to enable the resultant combined VN decoder to provide *extrinsic* LLRs having a MI of 1. A 2D EXIT chart may then be used for characterising the iterative decoding convergence between the core CN decoder and the combined VN decoder. However, in order to facilitate this, a particular decoding schedule must be assumed, which ensures that the combined VN decoder behaves as if it were memoryless, associated with an *extrinsic* MI that depends only on the *a priori* and channel MIs, but not

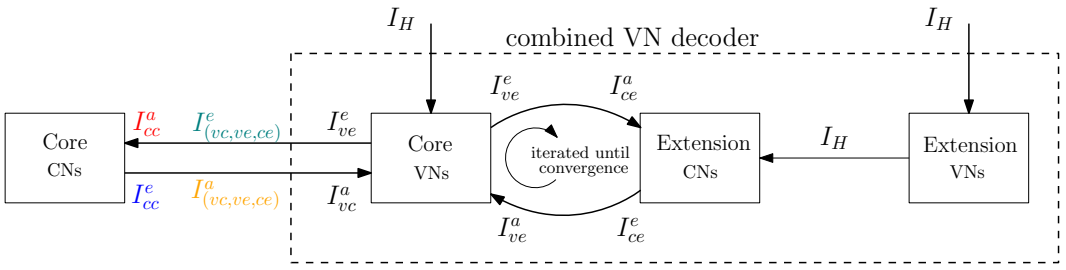


FIGURE 4.32: Block diagram of NR LDPC decoder that considers the core VNs, extension CNs and extension VNs as a combined VN decoder. The activation-order of the components is formally defined in Figure 4.33.

on any internal state. In the first step of this schedule, the channel's output information I_H will be forwarded to all VNs. As shown in Figure 4.32, the proposed schedule performs several MI exchange iterations between the core VNs and the extension CNs, until the I^e_{ve}, I^a_{ce}, I^a_{ve} and I^e_{ce} information becomes fully converged. Then the *extrinsic* information $I^e_{(vc,ve,ce)}$ provided by the combined VN decoder becomes the *apriori* information I^a_{cc} of the core CNs. After the inner calculation within the core CNs, the resultant *extrinsic* information I^e_{cc} will be forwarded to the combined VN decoder as its *a priori* input information $I^a_{(vc,ve,ce)}$. Therefore, the 2D EXIT chart of this system represents the combination of $I^e_{(vc,ve,ce)}$ versus $I^a_{(vc,ve,ce)}$ and I^a_{cc} versus I^e_{cc}.

Instead of using simulations to draw the 2D EXIT projection of the NR LDPC in [32], we conceive a novel method relying on equations (4.8) to (4.11). In order to unambiguously specify this method, Figure 4.33 introduces a flow chart of the corresponding decoding process. As discussed above, the 2D EXIT projection illustrates the combination of $I^e_{(vc,ve,ce)}$ versus $I^a_{(vc,ve,ce)}$ and I^a_{cc} versus I^e_{cc}, which are initialised to 0 as presented in Figure 4.33. At the beginning of the decoding process, the channel's output MI I_H is passed to the core VNs, and extension VNs as shown in Figure 4.33. Hence the channel's output MI that is passed to the extension VNs is directly forwarded to the extension CNs, by the degree-1 extension VNs of Figure 4.32. In the first step, we focus on the effect of passing the channel's output MI to the core VNs. According to the equations of Section 4.4.2.3, the *extrinsic* information I^e_{ve} that is passed from the core VNs of Figure 4.32 to the extension CNs is calculated as $I^e_{ve} = I^e_{ve}(I^a_{vc}, I^a_{ve}, I_H)$ according to (4.9), where I^a_{ve} and I^a_{vc} are initialised to 0.

In the next step of Figure 4.33, the iterative decoding process begins its operation. More specifically, the MI will be first iteratively exchanged between the core VNs and extension CNs, in a process referred to as inner iteration in Figure 4.32 and Fig 4.33. The *extrinsic* information I^e_{ve} generated by the core VNs will then be passed to the extension CNs as the *a priori* information I^a_{ce}. In response, the *extrinsic* MI that is passed from the extension CNs to the core VNs of Figure 4.32 is evaluated by (4.11) as

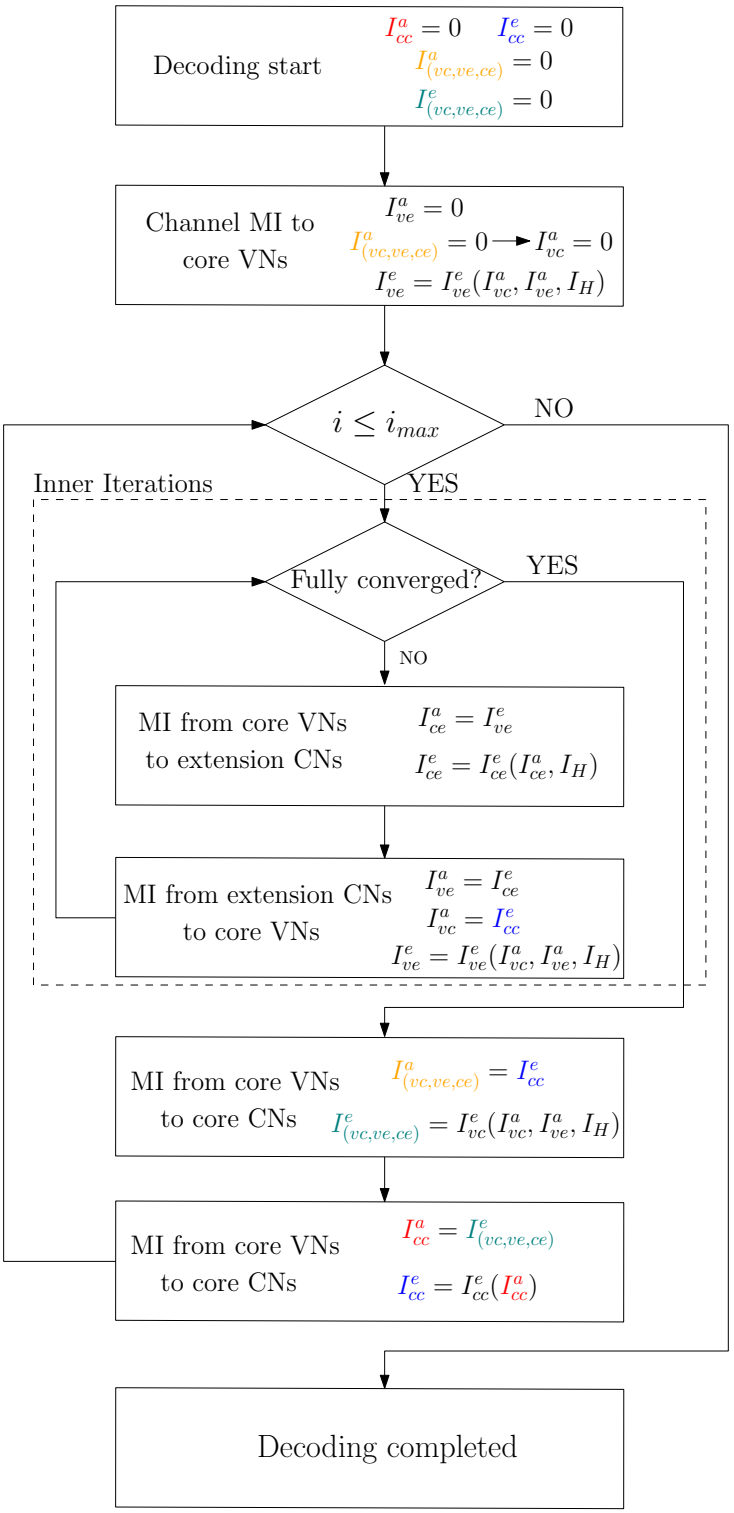


FIGURE 4.33: Flow chart of the decoding process of NR LDPC decoder, when considering the core VNs, extension CNs and extension VNs as the combined VN decoder of Figure 4.32.

$I^e_{ce} = I^e_{ce}(I^a_{ce}, I_H)$. Similar to the first step, the *extrinsic* MI I^e_{ve} will be further updated and passed back to the extension CNs of Figure 4.32 and so on. Note that throughout this inner iteration process, the *a priori* information I^a_{vc} provided by the core CNs of Figure 4.32 will remain constant. In this way, the inner iteration comprises an alternated MI exchange between the core VNs and the extension CNs, which continues until the MI is fully converged and stop improving. For example, in Figure 4.28 of Section 4.4.2.2, the trajectory stops at the cross-over point between $I^e_v(I^a_v, I_H)$ and $I^e_c(I^a_c)$ curve, whereupon convergence is obtained. The aim of the inner iteration of this alternated operation of the NR LDPC decoder is to determine the cross-over point between the $I^e_{ce}(I^a_{ce}, I_H)$ and $I^e_{ve}(I^a_{vc}, I^a_{ve}, I_H)$ curves, as a function of I^a_{vc}.

Once the MI exchange between the core VNs and the extension CNs of Figure 4.32 has become fully converged, the *a priori* information I^a_{ve} can be considered to be a saturated value. According to (4.8) and discussed in Section 4.4.2.2, the *extrinsic* information $I^e_{(vc,ve,ce)}$ that is forwarded by the combined decoder constituted by the amalgamated core VNs, extension CNs and extension VNs to the core CNs may be formulated as $I^e_{vc}(I^a_{vc}, I^a_{ve}, I_H)$. In this way, the EXIT function of the combined VN decoder of Figure 4.32 may be obtained by plotting $I^e_{(vc,ve,ce)}$ versus $I^a_{(vc,ve,ce)}$. This EXIT function may then be compared to the core CN decoder's EXIT function, which is obtained by plotting I^e_{cc} versus I^a_{cc}. Figure 4.34 shows the proposed 2D EXIT projection for several different scenarios. For a comprehensive comparison, Figure 4.34 (a) and (b) correspond to different channel SNR, but the same coding rate of $R = 1/3$. By contrast, Figure 4.34 (c) and (d) employ different coding rates R, but the same channel SNR of -1 dB. As shown in Figure 4.34 (a), the curves that represent the CN decoder and the VN decoder do not cross over until the channel SNR drops to -1 dB. In other words, a 1/3 coding rate NR LDPC decoder with BG1 can be predicted to have adequate error-correction performance when the SNR is higher than -1 dB, which is a conclusion in contrast to the one suggested by the closed tunnel in the conventional 2D EXIT chart of Figure 4.28. Similarly, Figure 4.34 (b) gives the same conclusion when BG2 is utilized. Likewise, Figure 4.34 (c) offers the conclusion that the EXIT chart tunnel will be open when the coding rate is lower than $R = 2/5$ at a channel SNR of -1 dB. Similarly, for BG2, Figure 4.34 (d) has a similar suggestion, namely that a coding rate of less than $R = 2/5$ will give adequate error-correction performance. The reliability and accuracy of the 2D EXIT projection will be verified by simulations that will be presented in Section 5.1.3. As shown in Figure 5.3 (a), the BLER curve for the case of BG1 with $R = 1/3$ begins to decay around an SNR of -1dB, which verifies the EXIT chart result of the Figure 4.34 (b). The 2D EXIT projection not only corrects the deficiency of the conventional 2D EXIT chart of Figure 4.28, but also offers a clear view of whether the EXIT tunnel is open or closed.

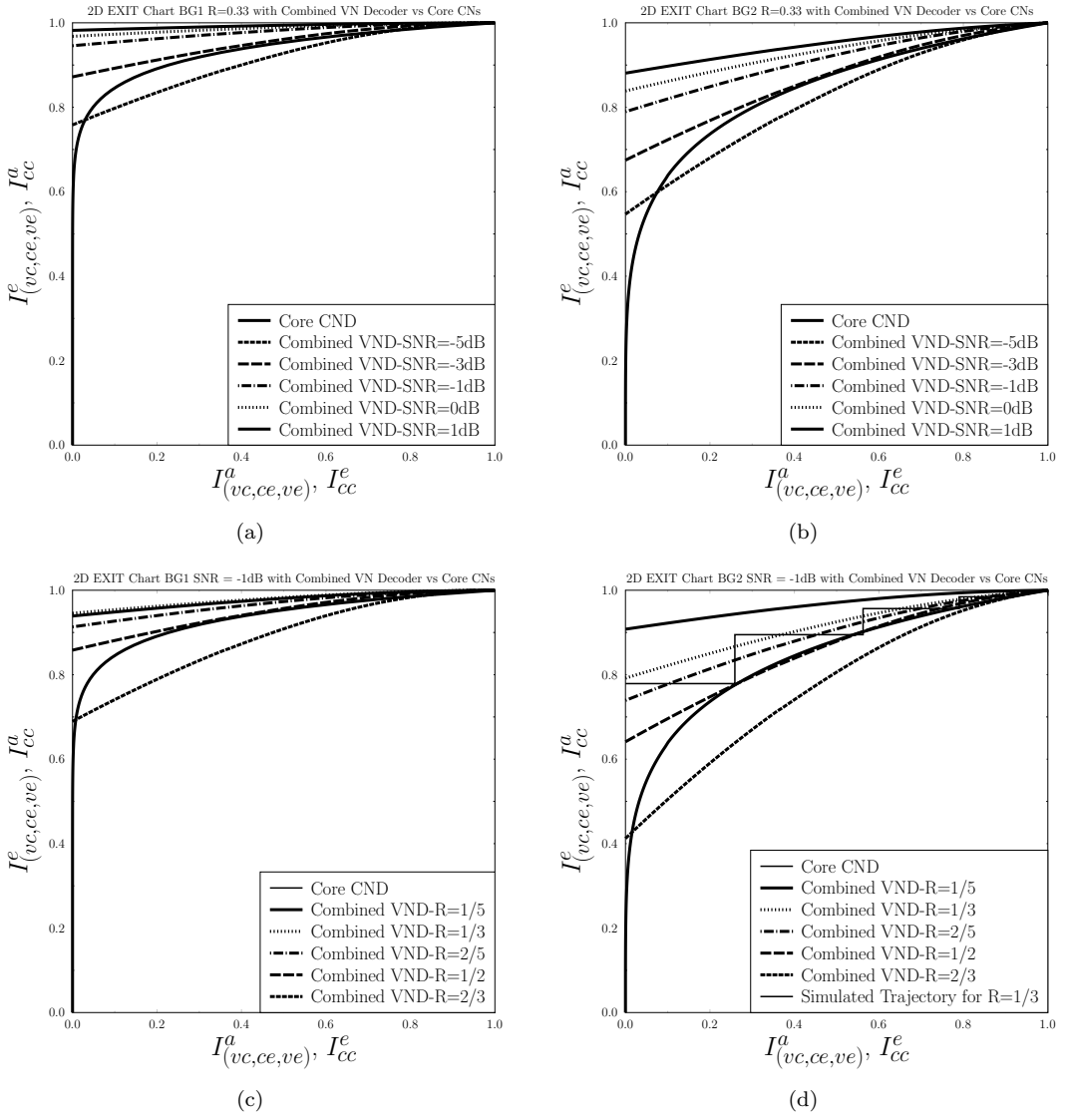


FIGURE 4.34: Proposed 2D EXIT chart projection for the flooding based 5G NR LDPC decoder. R = 0.33 is used in (a) and (b) for BG 1 and BG 2 at different channel SNRs, while SNR = -1dB is used in (c) and (d) for BG 1 and BG 2 at various coding rates R. A trajectory snap-shot is presented in (d), which was obtained during the iterative decoding of a specific LDPC code-block. The schematic of Figure 4.32 and flow chart of Figure 4.33 is used.

In order to further verify the accuracy of our proposed 2D EXIT chart projection, we plot a snap-shot trajectory in Figure 4.34 (d), which characterises the MI exchange of the outer iteration during a simulated iterative decoding process. As illustrated by Figure 4.33, each outer iteration comprises the exchange of the saturated MI $I^e_{(vc,ve,ce)}$ forwarded by the combined VN decoder to the core CN decoder of Figure 4.32. More specifically, the *extrinsic* information gleaned from the inner iteration represents the *a priori* input information I^a_{cc} of the core CN of Figure 4.32 in the outer iteration. During

this exchange, the MI of the exchanged LLRs is quantified and plotted for advancing the trajectory along the $I^{e}_{(vc,ve,ce)}$, I^{a}_{cc} axis of Figure 4.34 (d). After completing the operations of the core CN, the *extrinsic* LLRs that are passed from the core CNs of Figure 4.32 to the combined VN decoder are used for advancing the trajectory along the $I^{a}_{(vc,ve,ce)}$, I^{e}_{cc} axis of Figure 4.34 (d). The *a priori* information $I^{a}_{(vc,ve,ce)}$ produced by the inner iteration will then be used for finding the next fully converged *extrinsic* information $I^{e}_{(vc,ve,ce)}$ before the start of the next outer iteration in Figure 4.33. Hence, the staircase-shaped trajectory will move towards the (1,1) point of the 2D projection of the EXIT chart until the affordable number of iterations is exhausted. The simulated trajectory presented in Figure 4.34 (d) is approximately bounded by the CND and VND EXIT curves, which validates our proposed 2D EXIT projection method.

4.5 Conclusion

In this chapter, we introduced the 3GPP standardised NR LDPC code of the 5G data channel. Next, we briefly highlighted the MI exchange between the VND and CND of the NR LDPC code, which is often characterised by EXIT charts. Based on our simulations, we found that the MI exchange between the VND and CND of the LDPC decoder can be represented by the parity-check equations from [26]. However, we found that the conventional 2D EXIT chart is unable to accurately describe the MI exchange between the VND and CND of the decoder. It also failed to correctly distinguish whether the EXIT tunnel is open or closed. In order to analyse this decoder further, we separated it into four components, namely core VNs, core CNs, extension VNs and extension CNs, which were characterized by our own 3D EXIT chart. This 3D EXIT chart is capable of accurately characterizing the MI exchange of each iteration and to ascertain the EXIT chart's open/closed tunnel state. More specifically, via the 3D EXIT chart analysis, we are capable of predicting the decoding performance and the approximate number of iterations that required for a successful decoding result. For convenience, we also introduced the 2D projection of our 3D EXIT chart, which shows explicitly, whether the EXIT tunnel is open or closed. Although the proposed 3D EXIT chart is capable of analysis all NR LDPC decoder, due to the generation of random distributed input LLRs, a longer block length will bring a more accurate EXIT chart. Therefore, in this thesis, the minimum number of block length is set as 1000 for 3D EXIT chart analysis of NR LDPC decoder. The relationship of the accuracy of 3D EXIT charts and the block length may be investigated as future works.

Chapter 5

New Radio LDPC Code Applications relying on EXIT Chart Analysis

Since the LDPC coding technique was conceived in the 1960s in Gallagher's PhD thesis, it is quite mature. As summarised in [130], the main recent research topics on LDPC codes are related to density evolution [131] and EXIT chart based approaches [25]. This thesis is dedicated to LDPC EXIT chart analysis and to the related applications. Based on our novel EXIT chart analysis conducted for the NR LDPC decoder introduced in Chapter 4, we investigated its implementation. In this chapter, we consider two applications of the NR LDPC EXIT chart analysis, namely the activation order of the scheduled NR LDPC decoder components and of the proposed NR LDPC HARQ system components. More specifically, the proposed NR LDPC decoder employs our novel EXIT chart analysis for controlling the process of MI exchange among its components. We describe the design of this novel iterative decoding schedule for the NR LDPC decoder, which reduces the decoding complexity by approximately 17% compared to a conventional flooding schedule [13], which is adjusted without degrading its error correction capability. The detail of the proposed NR LDPC decoder will be introduced in Section 5.1. Furthermore, the proposed HARQ system relying on the NR LDPC code explicits the open/close nature of the EXIT tunnel to determine the necessity of re-transmissions. The HARQ system advocated improves the throughput of 5G systems. The detailed characteristics and system design of the proposed HARQ system will be presented in Section 5.2. Finally, the conclusion of these two application will be summarised in Section 5.3, as well as some future improvements will be the proposed.

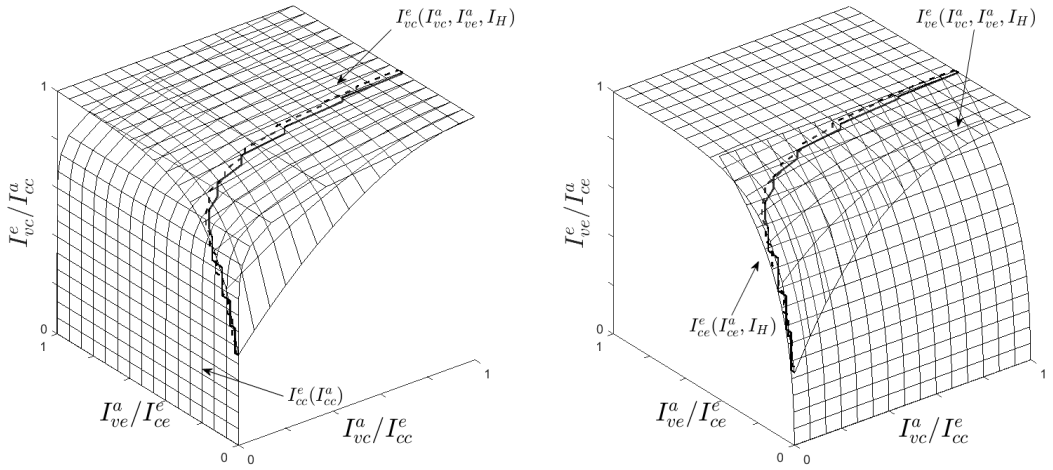
5.1 Fixed schedule decoder system design

In this section, we will introduce a fixed NR component activation order-based LDPC decoder design, which is invented based on the idea of our 3D EXIT chart analysis mentioned in Section 4.4.2.3. Section 5.1.1 commences with the introduction and motivation of our proposed fixed activation-order NR LDPC decoder. Our system design and the associated decoding process is detailed in Section 5.1.2 with the aid of a flow chart. Furthermore, in order to evaluate the performance of the proposed design, we considered both the AWGN channel and uncorrelated Rayleigh fading channel. Section 5.2.3 shows the error-correction performance as a function of the associated complexity. Finally, Section 5.1.4 concludes this section followed by our future research ideas.

5.1.1 Introduction

According to the EXIT chart analysis of Section 4.4.2, a remarkable characteristic of NR LDPC has been revealed, which is related to its MI exchange during the decoding process. Based on these observations, this section will propose a novel LDPC decoder outperforming the conventional flooding LDPC decoder. As the activation order of the decoding components will be determined before the decoding, this method may be regarded as an LDPC decoder having a fixed activation order. In Section 5.1.2, the design of the LDPC decoder will be discussed, while its error correction performance is qualified in Section 5.1.3. A similar idea has been introduced in [132], where the author analysed the computational complexity of the NR LDPC decoder and discussed its decoding component activation order. In contrast to other contributions, we use EXIT charts. Based on Section 4.4.1, the PCM is constituted by two parts, namely the LDPC part and LDGM part. Based on the 3D EXIT chart analysis of Section 4.4.2.3, we found that in the last few decoding iterations, the MI exchange contribution of the extension CNs becomes so small that it can be neglected. Therefore, only the core CNs are activated for the last few decoding iterations.

Let us now introduce the dynamic activation order-based decoder and the fixed activation order-based decoder. The proposed fixed decoder activation order will be introduced in Section 5.1.2 with the aid of its flow charts and its performance is quantified in Section 5.1.3. Finally, a brief summary and future work ideas, including the dynamic activation order-based decoder, will be presented in Section 5.1.4.



(a) (b)

FIGURE 5.1: 3D EXIT chart for a K=3000-bit flooding 5G NR LDPC decoder, using BG 1, R=1/3 and different AWGN channel SNRs, including: (a) LDPC EXIT chart at SNR = -1dB, (b) LDGM EXIT chart at SNR = -1dB. The continuous line represents to the predicted trajectory that bounced between EXIT surfaces while the dashed line corresponds to the the bit-by-bit Monte-Carlo simulation.

5.1.2 System design of fixed activation order-based NR LDPC decoder

The MI exchange within the NR LDPC decoder has been comprehensively illustrated in Fig 4.30. More specifically, as emphasised in Figure 5.1(a) and (b), the MI can be seen to converge to unity along the I^a_{vc}/I^e_{cc} axis during the last few iterations, where it can therefore be reasoned that this is a benefit of the core CNs contribution. By contrast, during those iterations, there is almost no improvement along the axis of I^a_{ve}/I^e_{ce}, which indicates the modest contribution of the extension CNs. As described in Section 4.4.2, a low BLER is achieved when the iterative decoding trajectory reaches $I^e_{cc} = 1$. This phenomenon has been proved for various NR LDPC decoders with different BG selection, block length and coding rates. In order to make this happen using as few iterations as possible, it is desirable to activate the core CNs in every iteration, since the core LDPC code has a strong error correction capability. However, in the later iterations, the extension CNs stop making significant contributions, and so it is desirable to activate the core CNs alone. Therefore, we propose a novel fixed activation order-based NR LDPC decoder in which the extension CNs are deactivated for the last few decoding iterations as they will not contribute beneficially to the MI convergence. The proposed fixed activation order-based LDPC decoder design reduces the cost of complexity without degrading the error-correction performance.

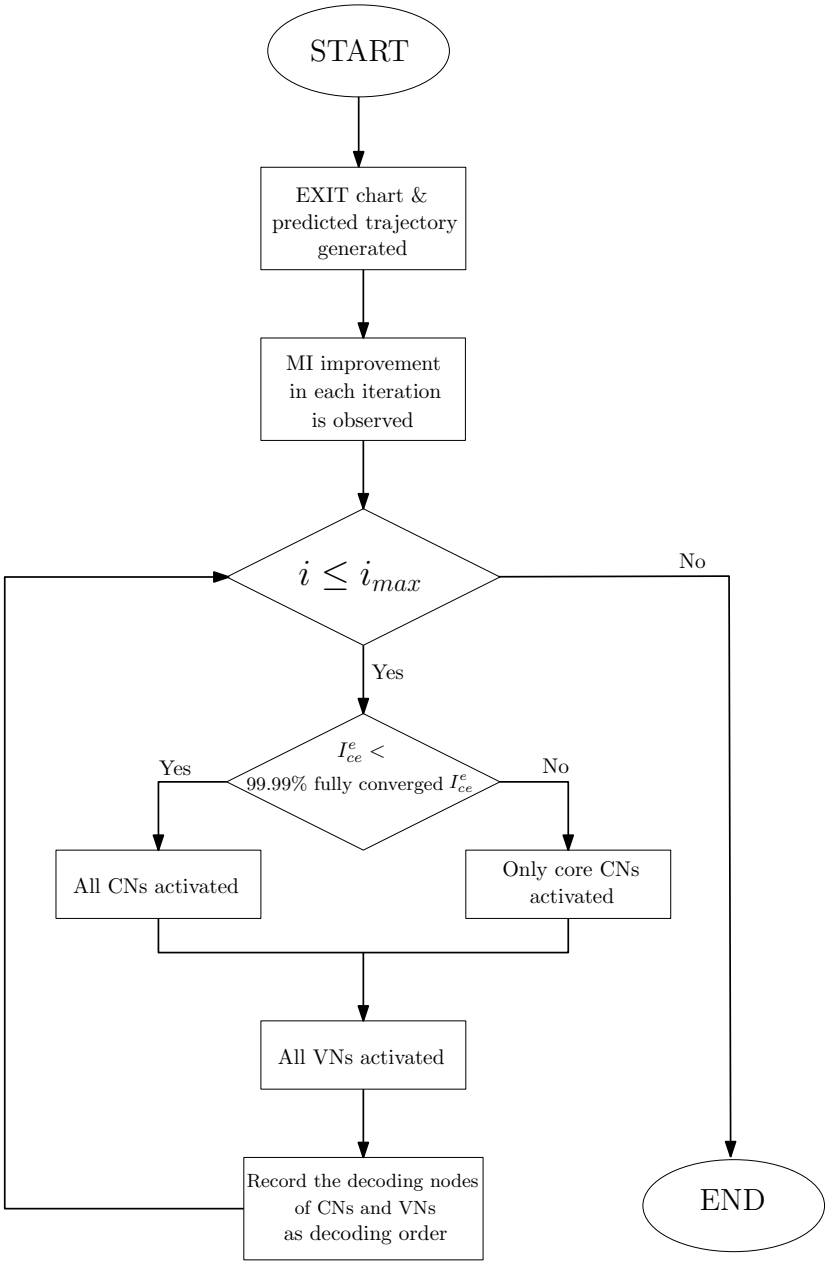


FIGURE 5.2: The proposed process of fixed activation order design for NR LDPC decoder.

The flowchart of the proposed fixed activation order-based NR LDPC decoder is shown in Figure 5.2. Following the construction of the 3D EXIT chart, the next step in Figure 5.2 is that of obtaining the mutual information improvement I_{ce}^{e} in each iteration. We note that I_{ce}^{e} and I_{vc}^{e} explicit similar trends. If the *extrinsic* information I_{ce}^{e} is less than 99.99% of the fully converged I_{ce}^{e}, both the core and the extension check nodes are activated. Otherwise, only the core CNs will be activated.

The above steps will be repeated until the iteration index exceeds the maximum affordable number of iterations. In practice, we can store the number of iterations to be carried out before switching to the core-only mode in a Look-Up Table (LUT), both as a function of the channel SNR as well as the coding rate. The LUT can be populated during an off-line design process using the 3D EXIT chart of Section 4.4.2.3. In order to achieve the best performance, we have also tried the value of 99% and 99.9% as our stopping criterion. However, the 99.99% gives the best complexity versus BLER trade-off.

5.1.3 Performance of proposed LDPC decoder

In this section, the simulation-based performance of the fixed NR components activation order-based LDPC decoder is presented versus its complexity. All simulation results are compared to these of the conventional LDPC flooding decoder, using the same decoding parameters as the simulated proposed decoder. In Section 5.1.3.1, the AWGN transmission channel is applied for characterising the proposed LDPC decoder. By contrast, in Section 5.1.3.2, the uncorrelated Rayleigh fading channel is considered.

5.1.3.1 Performance of proposed LDPC decoder in AWGN channels

This section characterizes the BLER performance of our novel regime of Section 5.1.2, for the case of the NR LDPC code in an AWGN channel. Again, we adopt a flooding based NR LDPC decoder as our benchmark. Our results are presented in Figure 5.3 and were collected by using BG2 of [1] for an information length of $K' = 1000$ bits. In order to demonstrate the versatility of the proposed regime, we use a low coding rate of $R = 1/3$ in Figure 5.3 (a) and (b); $R = 1/2$ is employed for the medium coding rate simulation of Figure 5.3 (c) and (d); $R = 2/3$ is used for the high coding rate simulation of Figure 5.3 (e) and (f).

In this section, we consider two different scenarios. In the first scenario of Figure 5.3 (a), (c), (e), we characterize the BLER performance of the flooding based decoder using 12, 13, 14 iterations [133–135]. Then we compare these to the BLER performance of the proposed decoder having the same complexity. While the flooding decoder performs a fixed number of iterations and has a fixed complexity regardless of the SNR, the complexity of proposed decoder varies with the SNR, as discussed in Section 5.1.2. Therefore, in different SNR environments, a constant number of flooding iterations will correspond to different numbers of iterations for the proposed decoder. In the second scenario of Figure 5.3 (b), (d) and (f), we characterize the complexity required by each decoder to achieve BLERs of 10^{-1}, 10^{-2} and 10^{-3}, respectively. Here, the complexity is

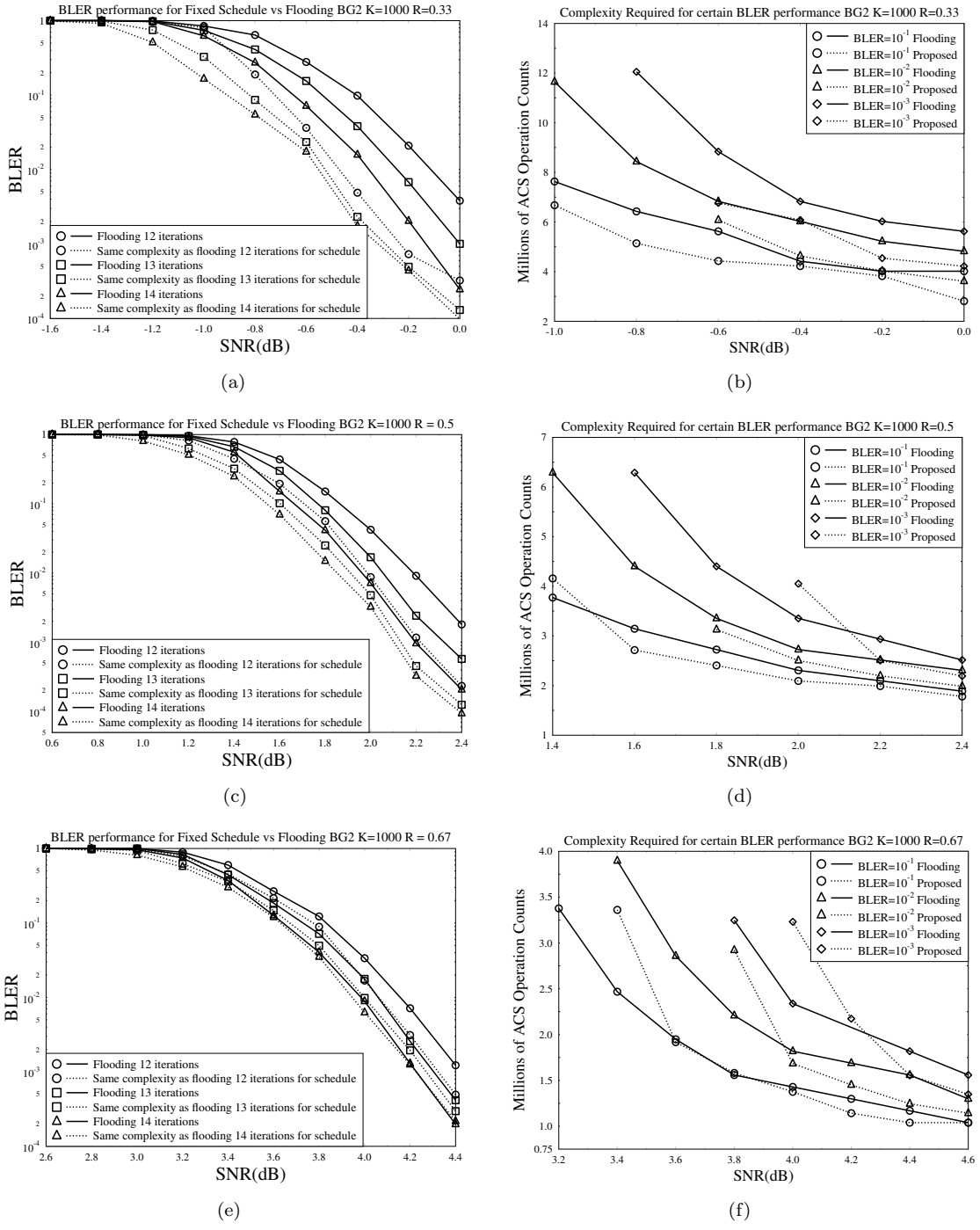


FIGURE 5.3: Pairs of simulation results to characterize the error correction performance of the proposed LDPC decoder in **AWGN channel**, as functions of channel SNR and decoding complexity. All plots are obtained using BG2 and an information block length of $K' = 1000$ bits. Plots (a) and (b) are obtained with 1/3 coding rate. Plots (c) and (d) are simulated with 1/2 coding rate. Finally plots (e) and (f) are using 2/3 coding rate. The schematic of Figure 4.29 and flow chart of Figure 4.31 is used.

quantified by the number of ACS arithmetic operations performed, as commonly used in hardware design evaluation [136]. Generally speaking, Figure 5.3 (a), (c) and (e) compares BLER for same complexity between benchmark and proposed decoder. While Figure 5.3 (b), (d) and (f) compareds complexity for same BLER performance between different decoders.

Figure 5.3 (a) considers a coding rate of 1/3 and shows that the proposed decoder offers an average of approximately 0.3 dB improvement over the flooding decoder at the same BLER performance and the same complexity. In the corresponding complexity plot of Figure 5.3 (b), the proposed decoder is shown to offer around 23% complexity reduction compared to the flooding based decoder. When the coding rate is increased to 1/2, as presented in Figure 5.3 (c), the proposed scheme having the same complexity as 12 flooding iterations achieves the BLER performance of 14 flooding iterations. In the complexity comparison of Figure 5.3 (d), the corresponding complexity reduction observed is 14%. Finally, Figure 5.3 (e) and (f) provide the corresponding simulation results for a high coding rate of 2/3. Here, the BLER results show that the proposed scheme no longer maintains its error-correction capability advantage for a high number of iterations, but it still outperforms the benchmark for a relatively low number of iterations. Furthermore, the same 14% complexity reduction is attained over the benchmarker.

As shown in the complexity versus SNR plots of Figure 5.3 (b), (d) and (e), the proposed decoder is unable to achieve the targeted BLER performance in some low-SNR scenario. Hence at low SNRs the proposed scheme would suffer from an error floor, unless increased the number of iterations above that of the benchmark. However, we observed that this may be mitigated by increasing the convergence threshold beyond the 99.99% value recommended in Section 5.1.2. This increased convergence threshold of selecting the switching point beyond which the extension CNs are neglected, and will achieve an improved BLER at the cost of reducing its complexity advantage. Nonetheless, we recommend 99.99% since the number of flooding decoder iterations required to observe this effect is high.

5.1.3.2 Performance of proposed decoder in uncorrelated Rayleigh fading channel

In this section, we compare the flooding and proposed LDPC decoder in the case of an uncorrelated Rayleigh fading channel. In order to have a comprehensive comparison, Figure 5.4 uses the same parameters as Figure 5.3 for all simulations. As shown in Figure 5.4 (a) and (b), for the low coding rate of $R = 1/3$, the proposed decoder has superior BLER versus complexity performance. When the complexity is fixed for both decoders, the proposed decoder achieves the same BLER at an SNR reduction of 0.3

dB. However, observed in the Figure 5.4 (c) and (e) for the higher coding rates that the proposed decoder no longer maintains its BLER performance gain for a high number of iterations. As for its complexity, Figure 5.4 (b) and (d) shows that the proposed decoder still offers approximately 14% improvement, but in Figure 5.4 (f), the proposed decoder no longer offers any complexity advantage for high coding rates, owing to its reduced error correction capability.

5.1.4 Conclusions and future research

5.1.4.1 Conclusions

In this section, we have characterized the MI exchange of the 5G NR LDPC code using EXIT charts. We have shown that conventional 2D EXIT chart analysis is unable to accurately characterize the MI exchange behaviour of the decoder, hence failing to reliably distinguish whether the EXIT tunnel is open or closed. As a remedy, we reconstituted its factor graph by partitioning the decoder into four elements, namely the core VNs, core CNs, extension VNs and extension CNs of Figure 4.22. In order to suit this new factor graph, we suitably adapted the MI equations of [26] and developed a novel 3D EXIT chart representation of the 5G NR LDPC decoder. We demonstrated that our 3D EXIT chart is eminently capable of characterising both the MI exchange amongst the decoder components, as well as the open/closed state of EXIT chart's tunnel. Furthermore, we also conceived a novel 2D EXIT projection of the proposed 3D EXIT chart for the convenient visualization of whether the EXIT tunnel is open or closed.

Drawing upon the unique insights offered by our EXIT chart analysis, we proposed a novel LDPC decoder activation order. More specifically, we observed that the NR LDPC decoder will typically achieve prompt MI convergence towards that of the extension CNs during the first few iterations, with the extension CNs offering little further benefit during the last few iterations. Therefore, we arranged deactivating for the extension CN decoding when the corresponding MI convergence accelerated by the extension CNs reaches the 99.99% margin of its fully converged value. By comparing the conventional flooding based and the proposed LDPC decoder, we found that the proposed decoder offers an approximately 0.2 dB gain at the same complexity and BLER performance. Viewing these benefits from a different perspective, at a specific BLER and SNR, about 14% to 19% average complexity reduction is attained compared to the flooding based decoder. These saved complexity could be considered as the required energy in the real-time implementation. The saved power are able to be used for other hardware implementations or directly shut down to avoid waste. Besides that, if the hardware implementation is pooled for multiple users or base-stations, the complexity

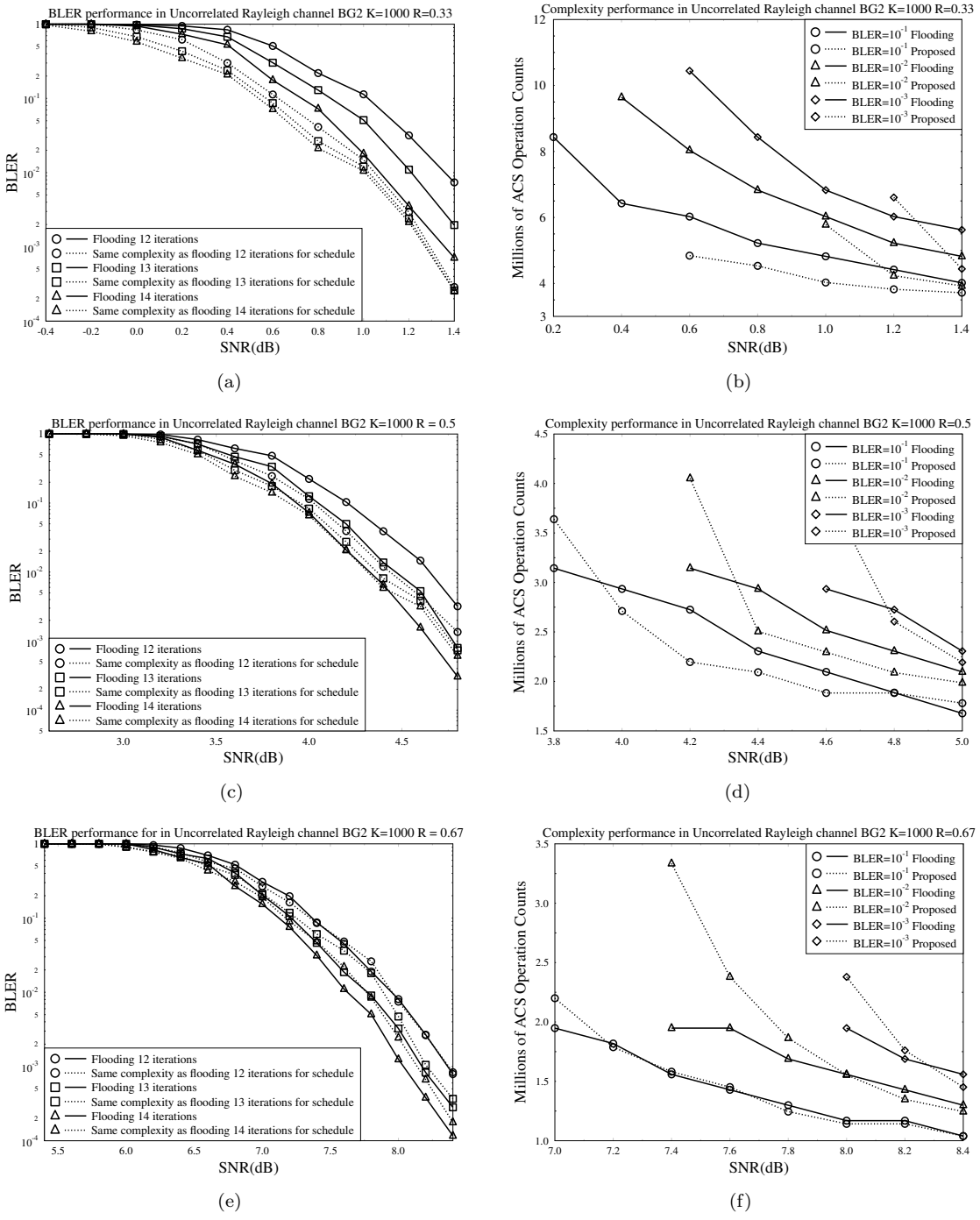


FIGURE 5.4: Pairs of simulation results to characterize the error correction performance of the proposed LDPC decoder in **uncorrelated Rayleigh fading channel**, as functions of channel SNR and decoding complexity. All plots are obtained using BG2 and an information block length of $K' = 1000$ bits. Plots (a) and (b) are obtained with 1/3 coding rate. Plots (c) and (d) are simulated with 1/2 coding rate. Finally plots (e) and (f) are using 2/3 coding rate. The schematic of Figure 4.29 and flow chart of Figure 4.31 is used.

TABLE 5.1: Summary table on the performance of proposed scheduled NR LDPC decoder. Note that all data are corresponding to the same BLER level of benchmark flooding decoder and proposed scheduled decoder.

	AWGN channel		Uncorrelated Rayleigh fading channel	
Coding Rate R	Average improvement on SNR	Reduction on complexity	Average improvement on SNR	Reduction on complexity
$R = 1/3$	0.4dB	23%	0.3dB	14%
$R = 1/2$	0.3dB	14%	0dB	14%
$R = 2/3$	0.1dB	14%	-0.1dB	No reduction

reduction can be expected to remain fairly constant average requirement across a large number of users [137]. Therefore, rather than provisioning the worst-case requirement, the implementation is more likely provision an average requirement for all users. The improvement on performance of the proposed scheduled decoder that presented in the Section 5.1.3 is listed as Table 5.1 for the convenience of audiences.

5.1.4.2 Future research

Again, in addition to the fixed activation order-based decoder considered, we could also conceive a dynamic activation order-based decoder. In contrast to the fixed activation order method that determines the decoding activation order before the iterative operations, the dynamic activation order-based decoder decides during the iterative decoding operations. Figure 5.5 illustrates the decoding process of the dynamic activation order-based NR LDPC decoder. Similar to the fixed activation order operation, we construct the 3D EXIT charts of the dynamic activation order-based decoder for the desired coding rate R and channel SNR by applying the method introduced in Section 4.4.2.3. However, instead of using the LUT in fixed activation order design, the dynamic activation order is decided during the process of decoding. As shown in Figure 5.5, the *extrinsic* information I_{ce}^{e} provided by the core VNs for the extension CNs will be qualified by a simple averaging method before the start of each decoding iteration. If the value of I_{ce}^{e} is lower than the 99.99% of the fully converged I_{ce}^{e}, this decoding iteration activates all CNs. Otherwise, only the core CNs will be activated. These steps will be repeated until the required number of decoding iteration is completed. The dynamic activation order requires more computational complexity than the fixed activation order method. However, exploiting the mutual information during the decoding process will make the decision of the activation order more accurate, which will be reflected by a better error-correction performance.

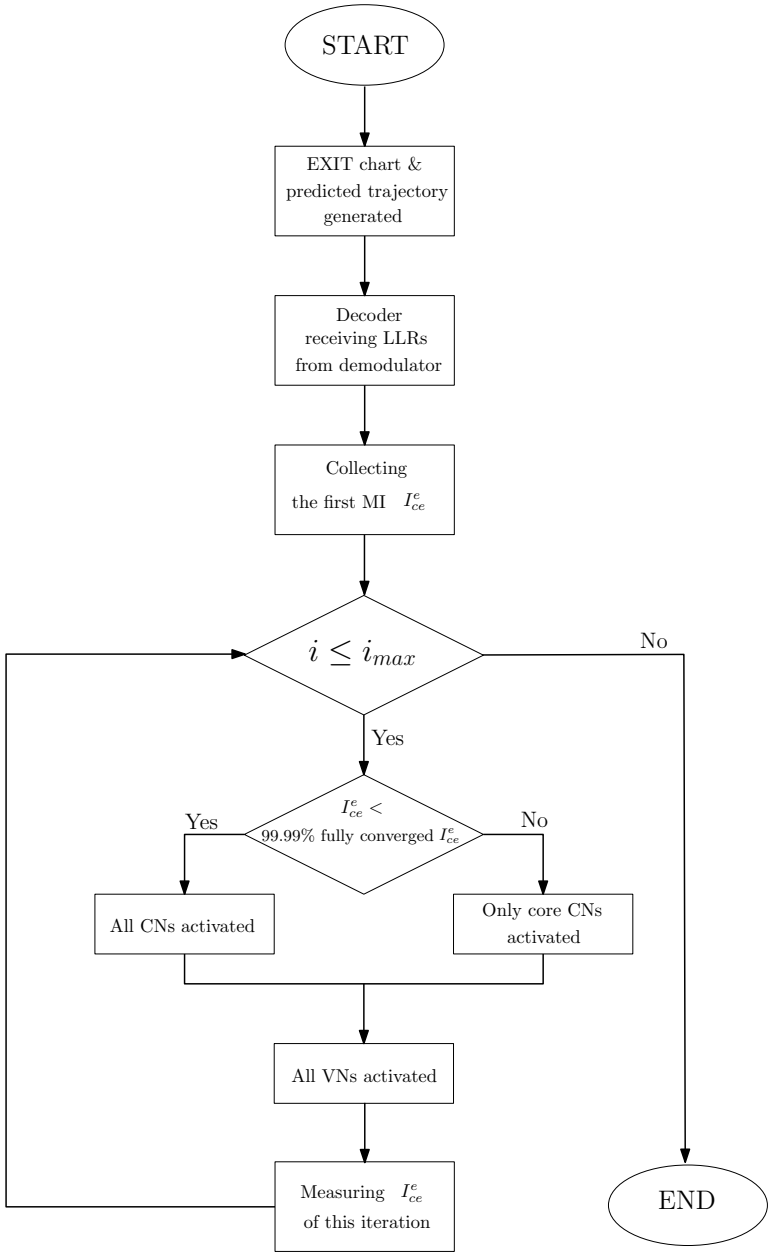


FIGURE 5.5: The designed process of dynamic activation order for the NR LDPC decoder.

There are few more aspects about the fixed activation order-based NR LDPC decoder that can be enhanced in the future. Firstly, an activation order based on layered belief propagation can be developed, which will reduce both the complexity and memory requirement of a practical implementation. Additionally, rather than using the Sum-Product Algorithm (SPA), a computational complexity reduction can be achieved by using a bespoke variant of the fixed-point MSA instead, which uses only binary logical and integer operations in its hardware implementation [124]. In conclusion, unveiling

this novel EXIT chart analysis for the NR LDPC decoder may be expected to offer significant further development of the NR LDPC code and its practical implementation.

5.2 HARQ design relying on EXIT chart analysis

In wireless communication system, retransmission may be used, when a certain transmission cannot be corrected by the channel decoder. The method of retransmissions relies on Automatic Repeat reQuests (ARQ) obeying the control of an acknowledgement(ACK) flag gleaned from the receiver side, which is also combined with the time-out mechanisms [138]. More explicitly, if the receiver of the ARQ system detects errors in the transmitted payload, there will be an negative acknowledgement (NACK) flag sent back to the transmitter to request a retransmission [139]. The drawback of ARQ is that the process of retransmission will eventually continue until a positive feedback ACK flag is received for curtailing retransmissions. However, in order to acquire a positive ACK, the ARQ system might require an excessive number of retransmissions as well as encoding and decoding operations, which may impose enormous cost in terms of reduced effective throughput, increased reception delay and hardware resources. In order to make the retransmissions more efficient, a method called HARQ was introduced in the 1960s by Wozencraft and Horstein [138,140]. The HARQ system combines the ARQ mechanisms with a sophisticated the Forward Error Correction (FEC) regime, as detailed below.

In the 60 years of research, since the numerous HARQ solutions have been conceived [141]. The most popular techniques are chase combining [142] and Incremental Redundancy (IR) [143]. Chase combining is the most fundamental method of establishing the HARQ capability, which simply retransmits the same bits of the former transmission. Then, the LLRs of retransmitted bits will be added to the LLRs of the former transmitted bits for the improved channel decoding. However, chase combining is less efficient than the IR based HARQ. This is because in IR-HARQ additional parity bits are transmitted, and then all LLRs will be intelligently combined together. naturally, this is advised at an increased complexity and memory. In this thesis, we focus on the 5G NR standardized HARQ technique designed for LDPC coded data transmission, as ratified in TS 38.212 [1]. The NR HARQ technique relies on the IR model, which sends different bits during each retransmission. Since all transmitted bits are read from the bit-selection circular buffer of the transmitter, some of the bits transmitted in different retransmission attempts will be overlapped. Therefore, the NR HARQ may be viewed as a mixture of Chase Combining (CC) and IR techniques combining their beneficial features. More details about the process of NR HARQ system will be presented in Section 5.2.1.1.

As described in Section 4.2.6, the NR HARQ designed for LDPC coded data transmission also has a disadvantage because of the retransmitted bits of the different redundant versions may be identical. Hence, we proposed some similar enhancements of the NR HARQ scheme to those in [15]. More explicitly, based on the 3D EXIT chart analysis of the NR LDPC decoder of Section 4.4, we will use the open/close state of the EXIT tunnel for controlling the retransmissions. Furthermore, we will approximately adapt the Deferred Iteration and early stopping techniques of [15] for employing in the 5G NR HARQ scheme. If the EXIT tunnel is considered to be closed in the DI operation, the receiver will not decode this transmission but only store the received LLRs in the retransmission buffer and directly asks for the retransmission of the same frame because any attempts to decode it would be futile. By contrast, in the EA scenario, the receiver curtails the decoding iterations, ensuring the iterations would failed to improve the *extrinsic* MI. Both proposed operations reduce the complexity of the NR HARQ system. Therefore, in this section, we improve the efficiency of the standardized NR HARQ system with the aid of our 3D EXIT chart analysis.

In Section 5.2, we commence with the introduction of the standardized NR HARQ design for LDPC coded data transmission, as a supplement of Section 4.2.6. Next, the design of the proposed NR HARQ system will be described with the aid of illustrations. Section 5.2.2 overviews the novelty of our proposed NR HARQ system, including the DI and EA regimes. Then, Section 5.2.3 portrays the performance of the proposed HARQ system that is also compared to the conventional standardised HARQ system in terms of complexity, packet loss ratio and throughput. Finally, Section 5.2.4 provides our the conclusion and further research ideas.

5.2.1 System design

In this section, we offer the necessary background of the standardized HARQ NR transmission. Section 5.2.1.1 presents a rudimentary introduction of the standardized NR HARQ for LDPC coded data transmission included in the TS 38.212 specifications [1]. Then, Section 5.2.1.2 describes the design of our NR HARQ system with the aid of a flow chart.

5.2.1.1 5G NR physical layer and standardised NR LDPC HARQ system

The standardized NR LDPC coding components of the PDSCH physical layer have been presented in Section 4.2. More specifically, Section 4.2.6 describes the rate-matching and the HARQ operation at the transmitter side. Following on from Section 4.2.6, in this

section, we provide a more detailed description of the NR HARQ for relying on LDPC coded data transmission both at the transmitter and receiver side.

Fig 5.6 illustrates the PDSCH physical layer described in the 5G NR data transmission standard TS 38.212 [1]. Together with the data transmission of **tx** and **rx**, the ACK and NACK flag is also denoted in this figure as the transmission of control signals. Note that the ACK/NACK signals are coded with short block code or polar code and transmitted as UCI (Uplink Control Information) and DCI (Downlink Control Information) via PUCCH (Physical Uplink Control Channel) and PDCCH (Physical Downlink Control Channel) as standardised in TS 38.213 [144]. As for different working scenarios, polar codes are used for ACK/NACK messages above 11 bits. While short block codes are used for shorter ACK/NACK messages. ACK/NACK can be several bits if we have code block group transmission information. ACK/NACK can also be multiplexed with other information before coding, leading to block lengths long enough for polar coding. Since we focused on EXIT charts analysis and NR LDPC decoder in this chapter, further details about control signals will not be considered.

Compared to the Figure 4.1, the bold fonts are used to emphasise the HARQ related blocks. When the LDPC encoded bit sequence $\boldsymbol{d'}$ is obtained by the rate-matching block, these encoded bits will be stored in the bit-selection circular buffer, which is specifically designed for the NR HARQ scheme. Again, the NR HARQ scheme relying on a specific mixture of chase combining and IR techniques, which contains four different types of redundant versions that termed as rv_{id}. Each redundant version has its unique starting position in the circular buffer, from which the rate-matched bit sequence $\boldsymbol{f'}$ having the block size of E_r will be selected for the current transmission. More specifically, as illustrated in Fig 5.7, the encoded bit sequences $\boldsymbol{d'}$ are written into the bit-selection circular buffer and selected for the corresponding output bits by the choice of base graph and the type of redundant version. The 3GPP standard body concluded that, the most beneficial order of the four types of redundant versions is $rv_{id} = [0, 2, 3, 1]$, which represents the chronological order of NR LDPC codewords transmitted. Codewords with redundant versions of 2 and 3 are retransmitted before the codewords with redundant version of 1, which is proved by simulations from researches [145] and [146]. As shown in Fig 5.7, the first transmission labelled as 1Tx by bold fonts is the redundant version $rv_{id} = 0$ in the circular buffer. If this transmission is deemed to have failed, the receiver will ask for a re-transmission termed as 2Tx, which is selected by the redundant version $rv_{id} = 2$ in the circular buffer. Similarly, further re-transmissions will follow the order of redundant versions as mentioned above. Naturally, if the limit of number of re-transmission is higher than 4, the index of redundant version for further retransmission will start again from $rv_{id} = 0$, followed by the original order.

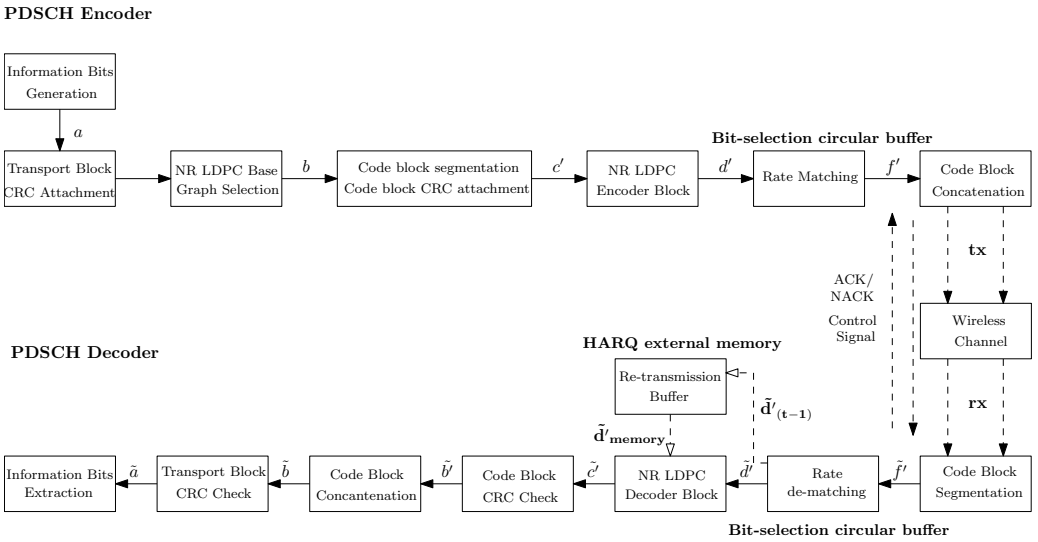


FIGURE 5.6: The block diagram of the standardized corresponding NR LDPC in PDSCH physical layer.

Once the transmitted bits reach the receiver of Fig 5.6, a few more steps are required when the HARQ system is activated. After the original encoded bits $\tilde{\boldsymbol{d}}'$ are output by the rate de-matching block, the conventional receiver will directly proceed to LDPC decoding to obtain the original information bits. However, in order to implement the HARQ scheme, extra memory is required to store all of the information received during former re-transmissions, which contains the channel LLRs for each transmission denoted as $\tilde{\boldsymbol{d}}'_{(t-1)}$ in Fig 5.7. For every re-transmission, the received channel LLRs that are input to the NR LDPC decoder represent the concatenation of channel LLRs $\tilde{\boldsymbol{d}}'$ received in this transmission and the channel LLRs $\tilde{\boldsymbol{d}}'_{memory}$ stored during former re-transmissions. Therefore, upon increasing the number of re-transmissions, the quality of the stored channel LLRs $\tilde{\boldsymbol{d}}'_{memory}$ is improved due to the accumulation. Meanwhile, the error-correction performance of the LDPC decoder is dramatically enhanced since the input LLRs contain more reliable information.

In order to portray the process of NR HARQ more explicitly, in Figure 5.8 we present an example for four transmission attempts of a BG1 NR LDPC decoder. Figure 5.8 (a) illustrates the transmitted bit sequence in the bit-selection circular buffer of the transmitter. By contrast, Figure 5.8 (b) demonstrates the process of the concatenation of received LLRs at the receiver side. In our example of Figure 5.8, we select $E_r = 5/6 \cdot K'$ as the block size of the output rate-matched bit sequence. However, as mentioned in the NR standard TS 38.212 [1], the block size of each transmission can be adjusted by rate-matching based on the user's requirement and the quality of the transmission channels. In this section and our later simulations in Section 5.2.3, we use the same E_r value for the different retransmissions of the same frame as special cases in order to

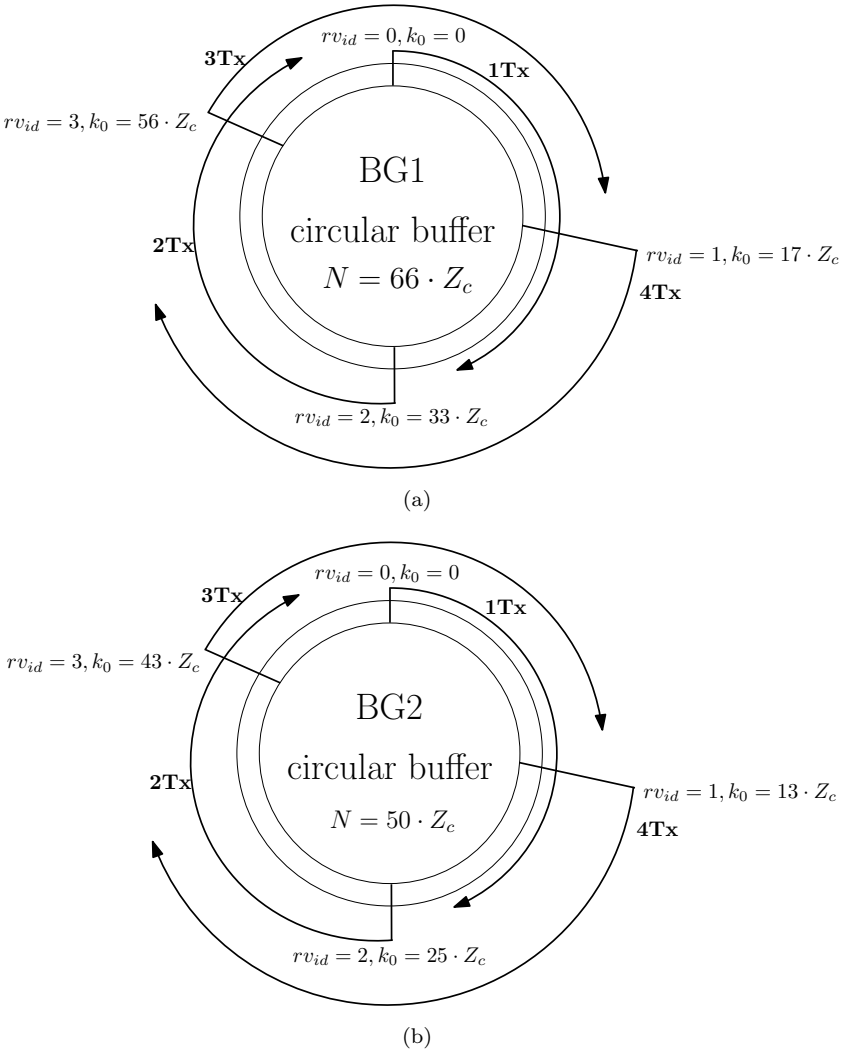


FIGURE 5.7: The illustrations of the NR LDPC encoder's circular buffer and the indication of bit-selection with (a) BG1 and (b) BG2.

make our description more straightforward. As shown in Figure 5.8 (a), the block size of the bit-selection circular buffer is denoted as N, where E_r number of rate-matched bits will be selected from the starting point k_0 that are defined by redundant version. In the first transmission 1TX, the output rate-matched bits are selected from $k_0 = 0$ for the block size of E_r in the circular buffer that are labelled as red color in Figure 5.8 (a). By contrast, Figure 5.8 (b) shows that the corresponding received LLRs are stored in the bit-selection circular buffer in the receiver side. Next, for the second transmission 2TX, the output rate-matched bits are selected from $k_0 = 33 \cdot Z_c$ for the same block size of E_r in the circular buffer as indicated by blue color in Figure 5.8 (a). At this stage, the transmitted bits of the first two transmissions are totally different, which is reminiscent of the IR HARQ model. As shown in Figure 5.8 (b), the LLRs received from the first two transmissions filled into different positions of the circular buffer at the receiver side.

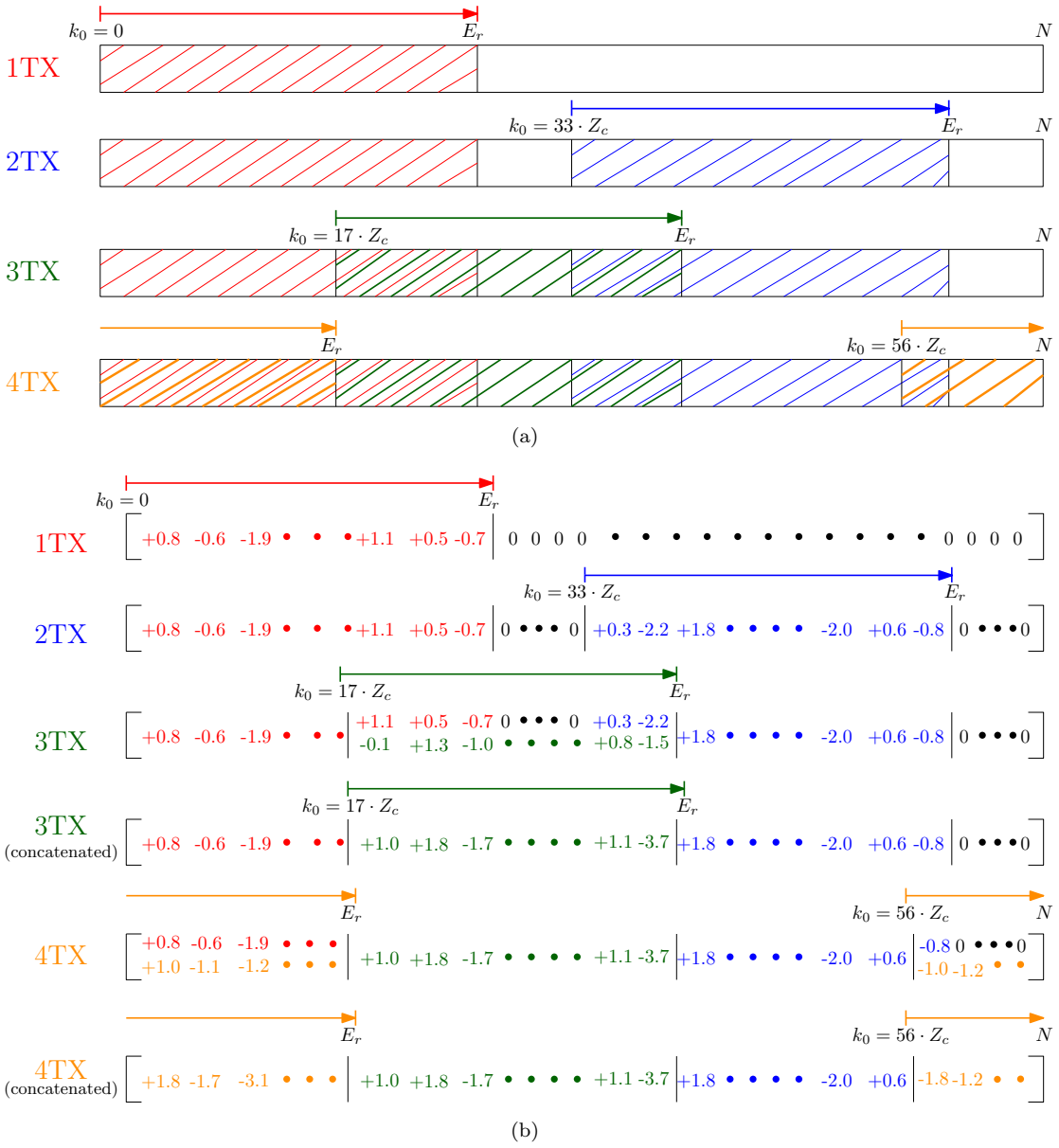


FIGURE 5.8: The illustration of transmitted bits in the NR HARQ system for each transmission, which is generated by LDPC decoder of BG1 with rate-matching output sequence of $E_r = 5/6K'$, where (a) presents the illustrates the selection of transmitted bits for each transmission in the transmitter side, (b) presents the concatenation of received LLRs for each transmission in the receiver side.

The ensuring third transmission 3TX contains the output bits that are selected from $k_0 = 17 \cdot Z_c$ for the block size of E_r in the circular buffer. As seen in the Figure 5.8 (a), the 3TX transmitted bits, which are labelled with green color, are overlapped with parts of the 1TX bits and 2TX bits. When the 3TX transmitted bits are received, the overlapped LLRs will be added up in the bit-selection circular buffer of the receiver. The decoder will use the concatenated sequence of LLRs in the circular buffer in the process

of iterative decoding. Finally, in the fourth transmission 4TX, the output rate-matched bits are selected from $k_0 = 56 \cdot Z_c$ for the block size of E_r in the circular buffer that are labelled by orange color in Figure 5.8 (a). By now, all bits of the circular buffer have been transmitted at least once. The 4TX transmitted bits also overlapped with part of the 1TX and 2TX transmitted bits, which leads to another concatenation of the received LLRs.

5.2.1.2 Design of the proposed HARQ system

Although the NR HARQ system is designed for attaining the best possible efficiency and reliability, the re-transmission impose redundancy, when the transmission is failed. Inspired by [15] portraying a holistic HARQ system design relying on a turbo decoder, we proposed a novel NR HARQ system operating under the control of our new 3D EXIT charts that were designed in Chapter 4 for improving the efficiency and throughput of the entire system. In this section, we will elaborate on the design of our novel NR HARQ system with the aid of flow chart. Furthermore, several schemes will be discussed and compared in this section.

Figure 5.9 portrays the flow chart of our proposed NR HARQ system. Before we discuss this in detail, we briefly introduce the deferred iteration (DI) and early abandon (EA) schemes that are surrounded by dashed lines in Figure 5.9. The fundamental idea of DI hinges on exploiting the state of the EXIT tunnel of the decoder to predict the decoding success or failure of this transmission. If the DI process indicates that the EXIT tunnel is closed, which means that the decoder is unable to success fully converge within the required number of decoding iterations, this transmission is disabled to avoid any extra complexity and power-dissipation. Therefore, the DI aided NR HARQ system will only start decoding the re-transmissions when there is a good chance of successful decoding. In [15], the threshold value of MI to be exceeded by the received LLRs to reach the open EXIT tunnel condition has been explored. Again, if the MI of the received LLRs is lower than this threshold value, this transmission will not be decoded. However, in our proposed DI model, we use our novel 3D EXIT chart analysis of the NR LDPC decoder that is capable of directly predicting the stair-case-shaped decoding trajectory of this transmission. Hence, if the predicted trajectory of this transmission is expected to reach the fully converged (1,1) point in the EXIT chart, the decoding iterations are activated. Note that even if some of the re-transmitted bits are not going to be decoded, their received channel LLRs will be stored in the external HARQ memory as the accumulation of reliable information. The DI aided NR HARQ system will significantly reduce the computational complexity of the NR LDPC receiver by avoiding potential futile iterations. However, due to dispensing with the decoding of some former re-transmissions that may

have been deemed futile, some unlikely successful re-transmissions may be ignored by the system, which leads to some reduction of the effective throughput. In conclusion, if the DI method can be carefully tuned, the throughput may remain reduction modest, while the decoding complexity reduction is expected to be substantial.

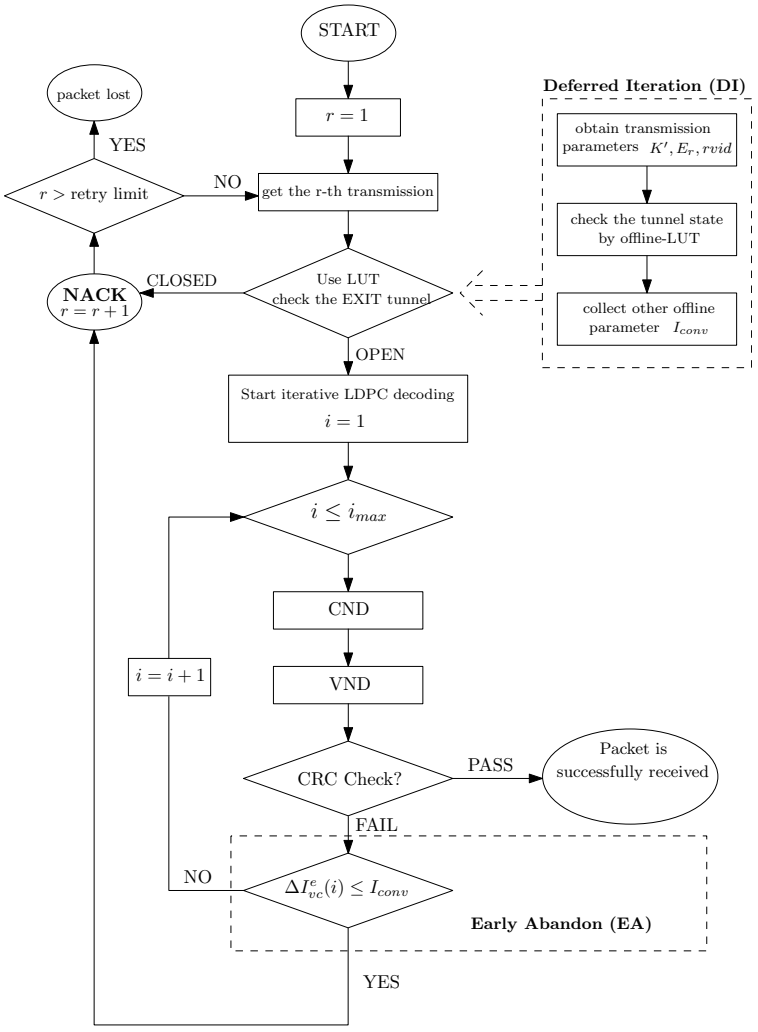


FIGURE 5.9: The proposed HARQ system design.

The other beneficial supplement to the NR HARQ system is the EA technique. Similar to the DI process, the EA process is also contributed by our novel 3D EXIT charts introduced in Section 4.4.2.3. The concept of EA is constituted as the check of convergence state during the decoding process. Generally speaking, the error-correction capability of LDPC codes has its limitations in specific coding parameters and transmission channels. [147]. Explicitly, when the limitation of the LDPC decoder's error-correction capability is reached at a certain SNR, the BER will no longer reduced upon increasing the number of decoding iterations. This observation and some similar ideas to those in [15], which were specifically designed for an IR-aided turbo HARQ system, inspired the idea

of abandoning the redundant decoding iterations in the first few re-transmissions, when the error-correction capability is limited. By predicting decoding trajectories of the 3D EXIT charts, we can determine the number of iteration beyond which no further improvement of the *extrinsic* MI to attained. This can be regarded as the fully converged state of this decoder. We also set a constant threshold value of $I_{conv} = 0.005$ as the smallest improvement of the *extrinsic* MI for all transmissions, which is determined by simulations. The MI improvement chosen for comparison is denoted as $\delta I^e_{vc}(i)$ in Figure 5.9, which is the difference between *extrinsic* LLR output from the VN decoder and towards to CN decoder from this iteration ($I^e_{vc}(i)$) and last iteration ($I^e_{vc}(i-1)$). If the MI improvement after an iteration is less than the value of I_{conv}, we can assume that the decoder has reached its state of fully convergence. If the CRC check is not satisfied, further decoding iterations can be abandoned to reduce the complexity.

With the understanding of DI and EA techniques, we are now ready to describe the proposed NR HARQ system in detail with the aid of Figure 5.9. At the beginning, the NR LDPC decoder receives the first transmission associated with the index of $r = 1$. By applying the DI operation, the proposed NR HARQ scheme will check the EXIT tunnel state of this transmission with aid of the LUT, which stored all the off-line data required by the system's configuration. The instructions gleaned from the DI control will decide whether this transmission should or should not be decoded. If the EXIT tunnel is deemed to be open or nearly open, the decoding operations is commenced. Otherwise, the proposed NR HARQ system will directly send a NACK signal to the receiver and asks for the next re-transmission. Note that the 3D EXIT chart is mainly depended on the MI of received LLRs [148]. Therefore, fading channels did not required be specially considered for any effect on the EXIT charts, and we do not need to considered the CSI (Channel State Information) or the SNR. Our LUT are purely depended on the *extrinsic* information of received LLRs. After that, the decoder will activate to the iterative LDPC decoding process by alternatively operating the VN decoders and CN decoders. However, each HARQ retransmission can be separated in several time slots, while the channel distribution may change in this time. Therefore, our 3D EXIT chart re-considered the EXIT chart tunnel state for each re-transmission by evaluating the MI of received LLRs. In contrast to the NR LDPC decoder of Section 5.1 which terminates its iterations after reaching the maximum affordable complexity, the NR LDPC decoder used in this HARQ system relying on the CRC check aided early termination. This means that the decoding of the received codeword will be terminated and flagged as successfully decoded, if the CRC check is passed. By contrast, if the decoded LLRs fail the CRC check, with the aid of EA technique, the decoder will compare the improvement of the *extrinsic* MI I^e_{vc} denoted as ΔI^e_{vc} to the convergence threshold value of $I_{conv} = 0.005$ for convergence checking. If the ΔI^e_{vc} is lower than the convergence threshold value of I_{conv}, which means that the decoder is fully converged in this iteration, the decoder will abandon

any further iterations to mitigate the complexity and send a NACK signal to the receiver for activating the next retransmission. Otherwise, the decoder will continue its iterative decoding until the required number of iterations is reached.

Figure 5.10 illustrates a certain operational scenario of the proposed HARQ system. In the first transmission 1TX of this frame, the 3D EXIT chart tunnel is considered to be closed. Therefore, the DI control disables the decoding of this transmission and sends a NACK flag to receiver of the remote transmitter for requesting a second transmission attempt. For the second transmission 2TX, the 3D EXIT chart tunnel is considered to be open. This time, the DI control seen in Figure 5.10 allowed the process of the decoding. However, due to the hostile transmission environment, the decoding of this transmission failed in a few iterations, as indicated by the CRC check, which in turn that activated EA. The EA operation terminated the decoding process and sent a NACK flag to the transmitter for the next retransmission. Finally, in the third transmission 3TX, the receiver passed DI check and successfully decoded codeword without the activation of the EA operation. The ACK flag was sent back to the transmitter for the transmission of a new frame.

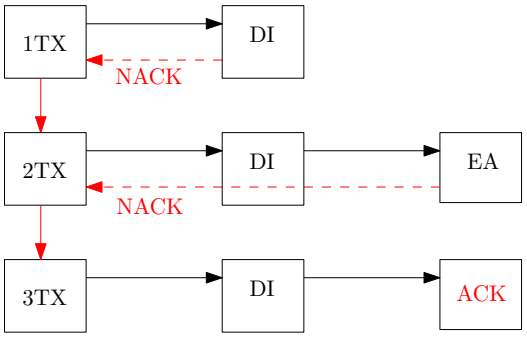


FIGURE 5.10: An example of a three-transmission circumstance.

As we discussed above, our proposed NR HARQ scheme for LDPC coded transmission may have a variety of design options, including the employment of DI only, the employment of EA only, and the combined employment of DI and EA. The method of setting the specific for DI and EA parameters will be detailed discussed in the Section 5.2.2. While the performance of all these NR HARQ systems will be presented in Section 5.2.3 to quantify their benefit versus cost.

5.2.2 The novelty of the NR LDPC HARQ system

In this section, we will introduce our novel NR HARQ system in more detail. Section 5.2.2.1 highlights the implementation of the DI operation, which defers the start of the iterative decoding until sufficient number of retransmission have been received for a

greater chance of successful decoding. Section 5.2.2.2 demonstrate the algorithmic and implementation of the EA operation, especially checking of the state of convergence. The implementation of the EA technique helps the decoder to early abandon decoding iterations if the decoding is not going to succeed. We rely on EXIT charts to illustrate the process of making decisions. Finally, Section 5.2.2.3 details the generation of LUTs, which are implemented for the storage of DI and EA control's configurations.

5.2.2.1 Deferred iteration

The DI concept introduced in Section 5.2.1.2 is designed to receive and buffer the transmitted LLRs from former transmissions of a certain transmission frame, which will only be decoded until sufficient information are accumulated for a successful decoding. In order to make a confident decision, the 3D EXIT chart analysis discussed in Section 4.4.2.3 is applied here to check the EXIT tunnel's open/closed state. With the increase of the number of retransmission in the HARQ system, the received LLRs will have higher and higher quality, while the gap between EXIT surfaces, which is considered as the EXIT tunnel state, will become more and more open. If the EXIT tunnel of the decoder of this transmission is considered to be open, then the decoding of this transmission will be activated since it has a greater chance of successful decoding. The standard of whether the EXIT tunnel is open or not can be decided by the predicted trajectory of the 3D EXIT chart, as discussed in Section 4.4.2.3.

Figure 5.11 and Figure 5.12 present an example of collecting the EXIT tunnel state for the DI operation. Both figures are recorded for a NR LDPC decoder with BG 1, information block length of $K' = 1000$ bit and coding rate of $R = 5/6$, which is simulated using the flooding activation schedule. We found out that for an AWGN SNR of -2 dB as scheme starts to successfully decode the frame after its third re-transmissions associated with $r = 4$. In order to show the evolution of the EXIT tunnel as the number of transmissions, we display four pairs of 3D EXIT charts to represent the *extrinsic* MI of the NR LDPC decoder during its first transmission and its three attempts of re-transmission. These 3D EXIT charts are plotted by the Equations (4.8) to (4.11) of Section 4.4.2.3, albeit they were modified in terms of I_H to match the external buffer input of the HARQ system. As shown in Figure 5.11(a) and (b), the EXIT surfaces are crossing each other at the beginning. There is actually a red point showing the existence to show the immediate convergence of the predicted trajectory. However, due to the lack of sufficient transmitting power, the first transmission of $r = 1$ using this decoder at a -2 dB SNR is considered as failed. Then, Figure 5.11(c) and (d) shows the *extrinsic* MI of the decoder at its first re-transmission attempt $r = 2$. Although the EXIT surfaces are generally distant from each other, the tunnel still becomes closed rapidly. The

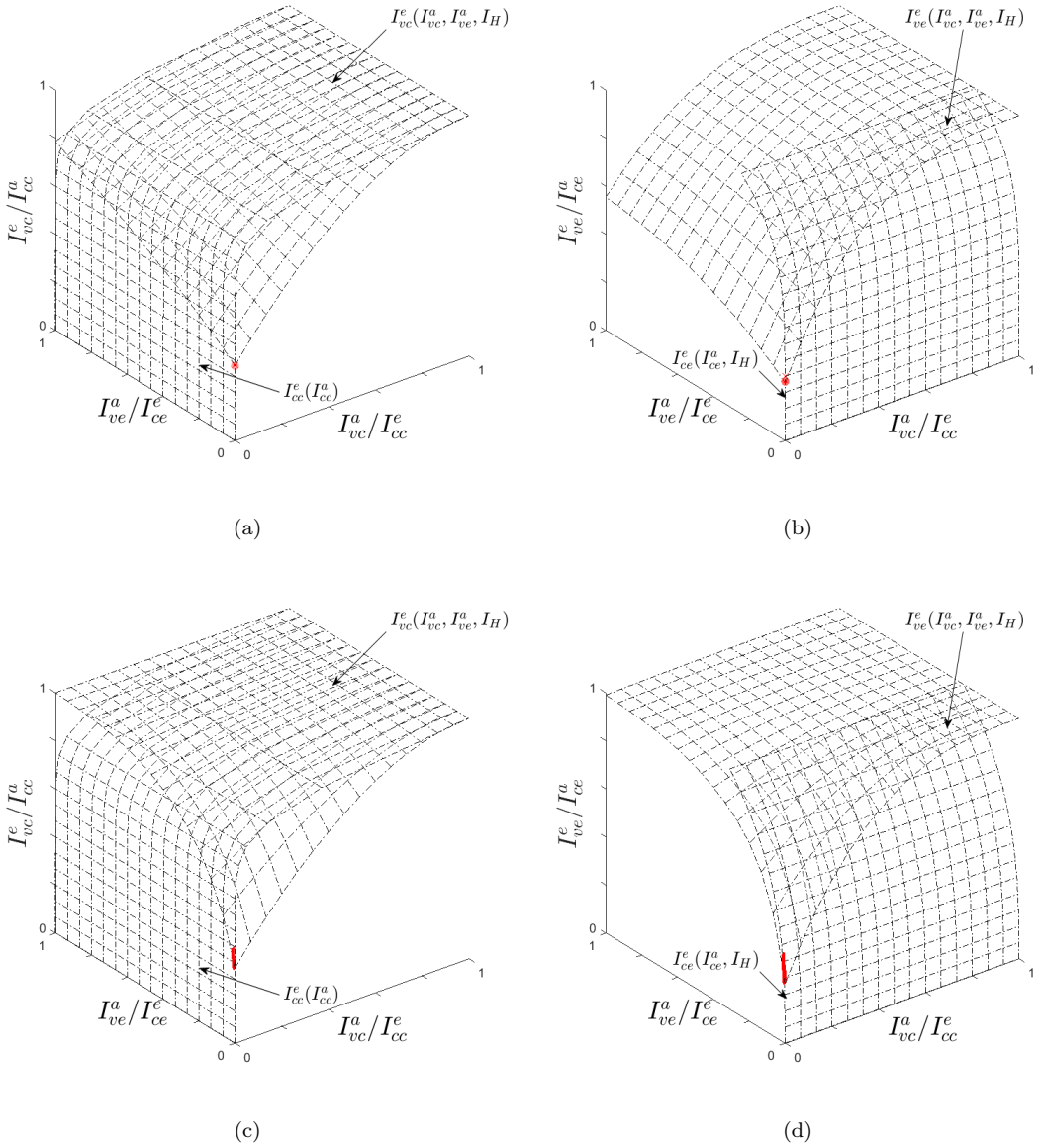


FIGURE 5.11: 3D EXIT chart for an information block length of $K' = 1000$ bit flooding 5G NR LDPC decoder. All plots are obtained using BG 1, $R = 5/6$ and SNR = -2dB. These plots are applying different order of transmissions, including: (a) LDPC EXIT chart of the first transmission $r = 1$, (b) LDGM EXIT chart of the first transmission $r = 1$, (c) LDPC EXIT chart of the first re-transmission $r = 2$ and (d) LDGM EXIT chart of the first re-transmission $r = 2$.

predicted trajectory, which is painted red in between the EXIT surfaces, shows that the decoder fails to reach the state of the fully converged *extrinsic* LLRs for its first re-transmission i.e. for $r = 2$. When the number of retransmissions starts to increase, the EXIT tunnel of the NR LDPC decoder becomes more open. Figure 5.12(a) and (b) presents the 3D EXIT charts of the LDPC decoder during its second re-transmission of

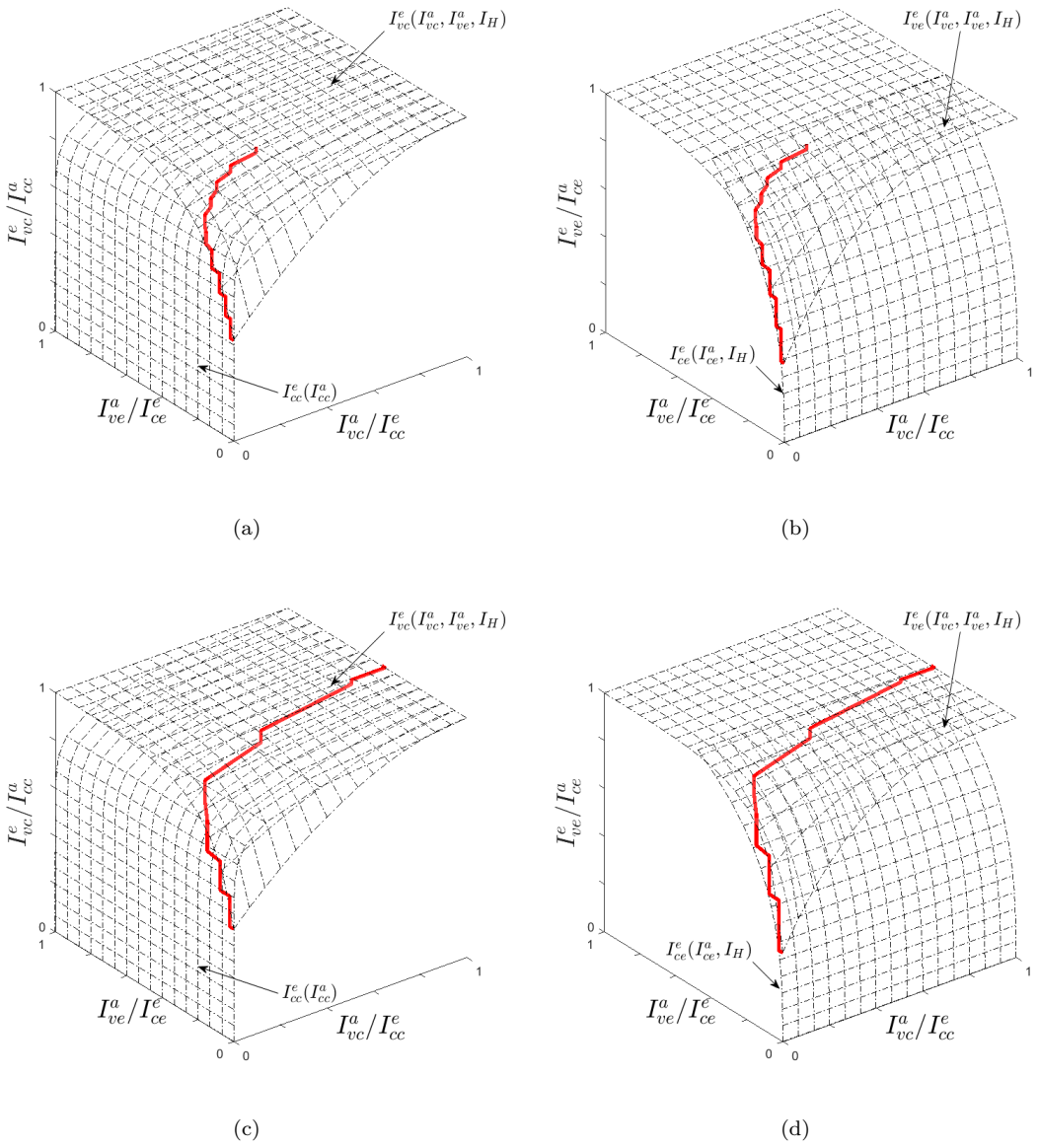


FIGURE 5.12: Continued figures of Figure 5.11. 3D EXIT chart for an information block length of $K' = 1000$ bit flooding 5G NR LDPC decoder. All plots are obtained using BG 1, $R = 5/6$ and SNR = -2dB. These plots are applying different order of transmissions, including: (a) LDPC EXIT chart of the second re-transmission $r = 3$, (b) LDGM EXIT chart of the second re-transmission $r = 3$, (c) LDPC EXIT chart of the third re-transmission $r = 4$ and (d) LDGM EXIT chart of the third re-transmission $r = 4$.

$r = 3$. In these plots, the EXIT tunnel is approximately open. Although the predicted trajectories between these EXIT surfaces suggest that the NR LDPC decoder is unable to fully converge within the required number of decoding iterations, which is set to 10 iterations in this simulation, this second re-transmission of $r = 3$ is still regarded as a potentially successful decoding by the DI regime. Due to the high cost of retransmission

in the HARQ system, we would like to avoid the misjudgement of the DI operation. Therefore, even the chance of successful decoding is limited within the required number of iterations considered by DI, the proposed HARQ system will still try to operate the iterative decoding. Finally, in Fig 5.12(c) and (d), both the EXIT tunnel and the predicted trajectories indicate successful decoding of the third re-transmission of $r = 4$. In conclusion, for this frame of NR LDPC coded bits, only the second and the third re-transmission, which are denoted as $r = 3$ and $r = 4$ will be decoded by the decoder block. The corresponding simulation results will be provided in Section 5.2.3.

Since the DI parameters are gleaned form the analysis of our 3D EXIT charts and predicted trajectories, these data will be recorded in an off-line LUT and directly loaded into the NR LDPC decoder. By applying the DI technique, the LDPC decoder imposed reducing the computational complexity in case of failed transmissions, but directly decodes the potentially successful re-transmissions. In our example, by directly decoding the third re-transmission of $r = 4$, it only requires 1/4 of the complexity imposed by the conventional NR HARQ system. However, as discussed above, the drawback of using DI is that some potentially successful re-transmissions may be neglected by the system without attempting decoding. In some situations, the actual decoder might perform better than predicted, which hence may reduce the effective throughput.

5.2.2.2 Checking the state of convergence in Early Abandon

As shown in Figure 5.9, checking the state of convergence is the next step when the CRC check is failed in the proposed NR HARQ system. The objective of checking the state of convergence of the decoder is to eliminate any redundant decoding iterations that are not contributing to the improvement of the *extrinsic* MI. This operation is usually applied for the first few transmissions of a single frame in a relatively poor transmission environment. For these transmissions, the fully convergence state of the decoder will typical appear before the affordable number of iterations is exhausted, which makes the rest of the decoding iterations expendable. By abandoning the futile LDPC decoding iterations, the total computational complexity of the NR HARQ system will be reduced significantly.

Similar to other techniques introduced for enhancing the proposed NR HARQ system, the decision of whether the EA regime should be operated or not is determined by the 3D EXIT charts and their stair-case-shaped decoding trajectories. Based on our EXIT chart analysis, the reliability of decoded LLRs will directly predetermine the improvement of the *extrinsic* MI. In other words, when the *extrinsic* MI of the NR LDPC decoder is no longer increasing, the decoder is considered as fully converged and can no longer deliver any better decoding result. Therefore, the method of determining whether the

extrinsic MI is fully converged or not is to compare its improvement after each decoding iteration. Here, we opt for using the *extrinsic* MI improvement at the output of the core VN decoder and forwarded to the core CN decoder, which is denoted as $\Delta I_{vc}^{e}(i)$, as our the main parameter. As shown in Figure 5.9, the EA operation will check the state of convergence after a failed CRC check. The improvement of the *extrinsic* MI $\Delta I_{vc}^{e}(i)$ is compared to the minimum threshold value of enhancement, which is denoted as I_{conv}. Inspired by the parameter used in [15], which were configured by our further simulation, we opted for using $I_{conv} = 0.005$. If the value of $\Delta I_{vc}^{e}(i)$ in the current decoding iteration is not higher than I_{conv}, the rest of the decoding iterations are regarded as redundant. After that, the proposed NR HARQ system will directly ask for the next re-transmission. However, if the value of $\Delta I_{vc}^{e}(i)$ in the present decoding iteration is higher than the value of I_{conv}, which implies that the decoder still has a chance to do better during the forthcoming iterations. In this scenario, the iterative decoding will continue until the total number of affordable iteration exhausted.

In order to characterize a range of different state of convergence, Figure 5.13 presents two pairs of 3D EXIT charts and the corresponding decoding trajectories. All plots in Figure 5.13 are obtained using a flooding decoding activation order-based NR LDPC decoder by setting BG as 1, information block length of $K' = 1000$ bits, encoded block length for each transmission of $E_r = 5/6K'$ and SNR of -2dB. Figure 5.13 (a) presents the 3D EXIT chart for the LDPC part of the decoder during the first transmission $r = 1$. In this plot, the EXIT tunnel is totally closed near the edges, which makes the decoding trajectory of this EXIT chart hard to identify. Figure 5.13 (b) provides a magnified version of the hidden trajectory that is surrounded by a red square in the Figure 5.13 (a). As shown by the trajectory, the *extrinsic* MI I_{vc}^{e} of the first transmission only enhanced once by the value of 0.01, which means that the later iterations are not contributing to the decoding process anymore. In other words, only the first two decoding iterations in this transmission are contributing to the MI improvement. Therefore, in the proposed EA operation, this decoder will terminate after the second iteration, where the state of convergence will be detected, and ask for a re-transmission directly. By contrast, Figure 5.13 (c) presents the 3D EXIT chart for the LDPC part of the decoder during the second re-transmission $r = 3$, which gives a clear view of the decoding trajectory. Meanwhile, the magnified version of the trajectory is displayed in Figure 5.13 (d). As shown by this trajectory, the *extrinsic* MI I_{vc}^{e} keeps on accumulating after each decoding iteration. Furthermore, the enhancement of the *extrinsic* MI $\Delta I_{vc}^{e}(i)$ is always higher than the threshold value I_{conv}. In the proposed EA operation, all of these decoding iterations are regarded necessary for passing the CRC check.

The use of EA via checking the state of the convergence of the *extrinsic* MI will significantly reduce the complexity of the decoder by eliminating the redundant decoding

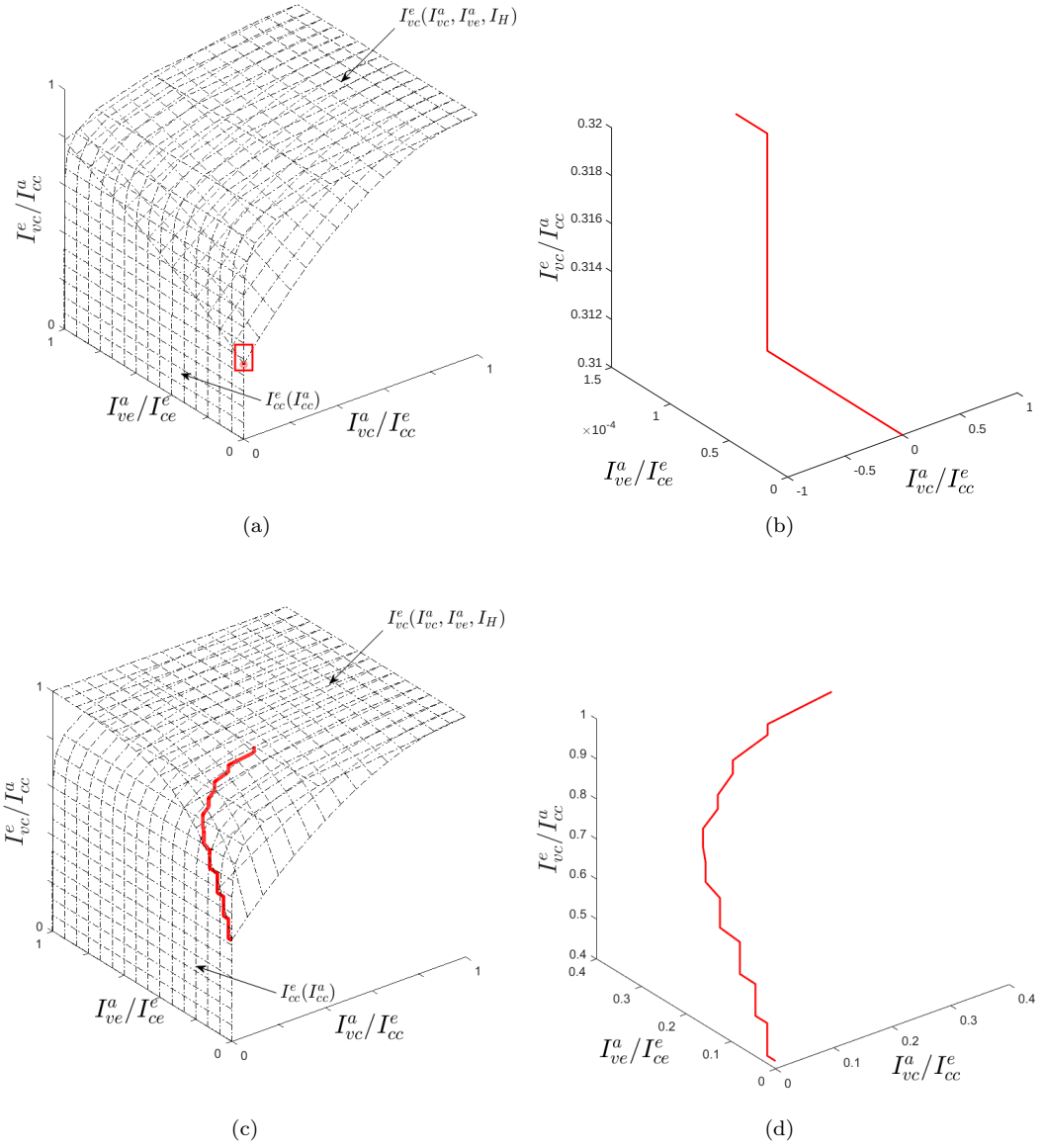


FIGURE 5.13: 3D EXIT chart and the magnified version of trajectories for an information block length of $K' = 1000$ bit flooding 5G NR LDPC decoder. All plots are obtained using BG 1, $E_r = 5/6K'$ and SNR = -2dB. Plots (a) and (b) are EXIT chart and its magnified trajectory of the first transmission $r = 1$. While plots (c) and (d) are EXIT chart and its magnified trajectory of the second re-transmission $r = 3$.

iterations, especially for the relatively unreliable transmissions in the face of hostile transmitting environments. In Section 5.2.3, the simulation results of the proposed HARQ system relying on the EA technique will be provided.

5.2.2.3 Look-up table generation

In order to efficiently utilize the proposed NR HARQ system of Figure 5.9, we opt for using an off-line LUT. Explicitly, the threshold values of I_{dump} and I_{conv} are stored in the LUT of the proposed HARQ system. The off-line LUT is mainly used for recording the predicted and simulated trajectories of the NR LDPC decoder for different retransmission orders. In this section, we will detail the generation and implementation of the LUT in the proposed HARQ system.

Having fixed the above-mentioned parameters, including the information block length K', coding rate R and channel SNR, the corresponding simulations will be carried out for generating the LUT. The first decision in the proposed HARQ LUT is related to the DI operation, which is set before the iterative decoding. As discussed in Section 5.2.2.1, the DI control provides the decision as to whether this transmission will result in passing the CRC check. Therefore, pairs of 3D EXIT charts, representing both the LDPC part and the LDGM part of the LDPC decoder will be generated. Furthermore, in order to determine the open/closed state of the EXIT tunnel, the trajectories of these pairs of EXIT charts are predicted relying on a high number of iterations to ensure the full convergence of the *extrinsic* MI. If the value of $I^e_{(vc)}$ in the predicted EXIT trajectory reaches the threshold value of 0.99, the EXIT tunnel of this transmission is considered as open. Otherwise, the LUT will record the opposite decision for this transmission under the DI regime.

The EA technique is also supported by our EXIT analysis. In contrast to the DI regime, the EA technique considers the state of convergence of the decoder, which requires enormous productions of equation-based EXIT charts, especially in the fading transmission channel. Without any doubt, the memory storage would be too huge to realize in the hardware implementation. Therefore, the value of I_{conv} is determined by the simulating the NR LDPC decoder under devise circumstance. During its normal operation, the proposed NR HARQ decoder will obtain the value I_{conv} from the LUT. While the improvement of *extrinsic* MI $\delta I^e_{vc}(i)$ is collected and compared to I_{conv} during the decoding iterations for controlling the EA operation.

In Figure 5.14, we presents a simple example of the LUT designed for our proposed HARQ scheme. In order to find the required data from LUT, we need to determine five parameters step by step. As shown in our example, the decoder uses BG1, information block length of $K' = 1000$ and encoded block length for single transmission of $E_r = 1200$. By providing the current SNR and index of retransmission, the LUT is able to obtain the EXIT's tunnel state and the threshold value of I_{conv} for the controlling of EA. Here, the EXIT's tunnel state is given as closed, which will deactivate the decoding of this transmission by the controlling of DI operation.

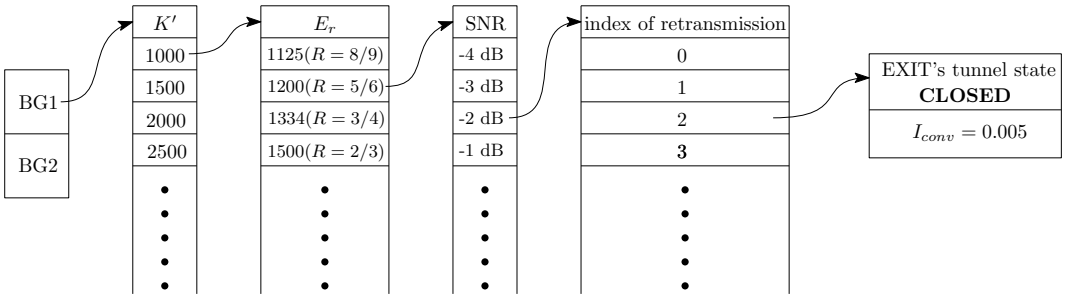


FIGURE 5.14: An example of the off-line LUT designed for proposed HARQ scheme.

As a conclusion, the LUT in this scheme has many layers, which are constructed like a tree structure. Each successive layer of the tree represent different parameters, while traversing the tree through these parameters can help to determine the EXIT's tunnel is open or closed. The conception of this LUT reduces the complexity of the EXIT chart aided analysis of DI and EA operations. Since most of the parameters are stored off-line, its hardware implementation is not required to consider further algorithm units. Since we only considered limited number of K' in our LUT, there are some interpolations can be exploited in the future research to mitigate the difference between the desired and the listed parameters.

5.2.3 Overall performance of the proposed system

This section presents the performance of the proposed NR LDPC HARQ model discussed in Section 5.2.2. The benchmark we adopted is the original NR LDPC coded HARQ system introduced in the 3GPP 5G specification, which was detailed in Section 5.2.1. The proposed HARQ systems are the DI-aided design, the EA-aided design, and the combined HARQ system relying on both the DI and EA techniques. All of these candidates were investigated in both AWGN and Rayleigh block fading channel scenarios for exploring their capability and feasibility, which are discussed in Section 5.2.3.1 and Section 5.2.3.2. Note that in order to obtain the fair result, all error-correction simulations are set to find 200 failed-passing blocks for each set SNR value. While for all complexity and throughput performance, each data point has more than 60 thousands simulated frames.

5.2.3.1 Performance of proposed HARQ system in AWGN channels

As discussed in our system design, the proposed HARQ system relying on DI and EA techniques is expected to significantly reduce the complexity of conventional HARQ

scheme by eliminating redundant decoding process. Figure 5.15 presents the performance of the selected NR HARQ systems in the AWGN transmission environment. All simulated plots were collected by using BG1 for an information block size of $K' = 1000$ and encoded bits for each transmission of $E_r = 1200$ using a decoder which has its maximum number of iterations set as 50. The maximum number of retransmission in this simulation is set as four. All simulation related parameters are summarised as Table 5.2. While each transmission will be selected from the bit-selection circular buffer following the redundant version index of $rv_{id} = [0, 2, 1, 3]$. This choice of K' and E_r is adopted as a high initial coding rate of $R = 5/6$ for each individual transmission, in order to demonstrate the throughput performance of the HARQ system. While the decision of using 50 iterations is to ensure the sufficient convergence of MI in the flooding scheduled LDPC decoder. In Figure 5.15 (a), our three novel designs and the conventional benchmark are compared in terms of their complexity versus channel SNR. The complexity in this plot is valued as the average number of iterations required which is aggregated across all successful transmissions. As shown in Figure 5.15 (a), the benchmark has the highest complexity that decreases upon increasing the channel SNR. In the conventional NR HARQ system, which is the benchmark in the plot, the only method of early termination is based on the CRC check. Therefore, the benchmark curve is gradually decreasing upon increasing the channel SNR.

TABLE 5.2: Simulation parameters related to Figure 5.15 and Figure 5.16.

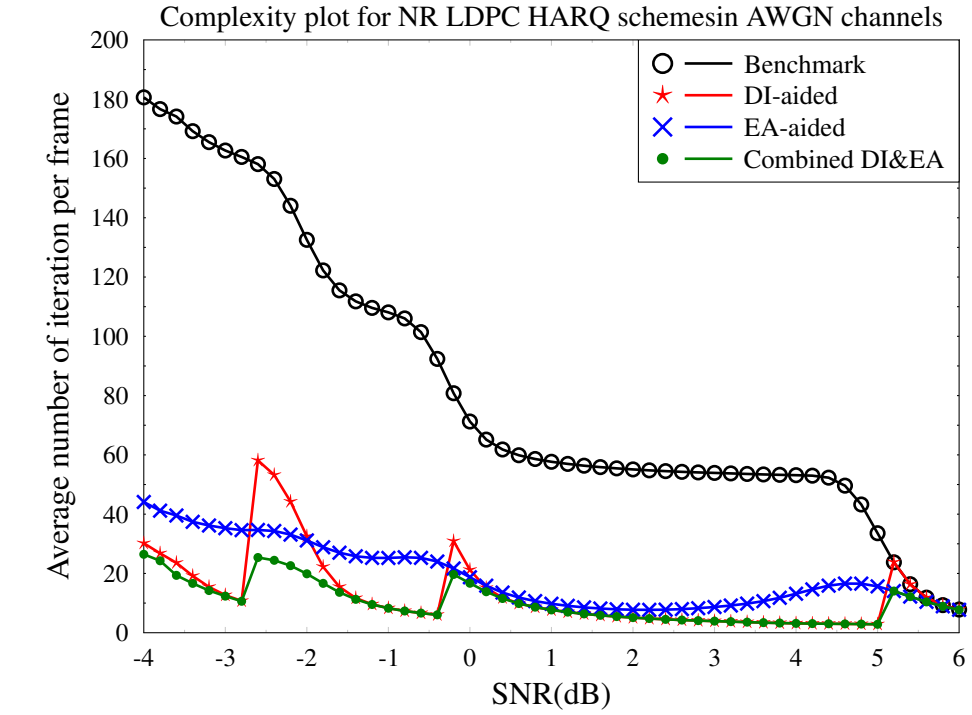
Parameter	Value
Base graph	BG 1
Information block length	$K = 1000$
Encoded block length per transmission	$E_r = 1200$
Maximum number of iteration	$i_{max} = 50$
Maximum number of retransmission	4

As for the three proposed NR HARQ techniques in Figure 5.15 (a), the most distinctive behaviour of the curve of DI-attached design is the existence of peaks. The existence of peaks in this curve represents the change of the decision in the DI operation. During these occasions, the DI operation gives an optimistic estimation of the output of decoding. For example, at the -2.6dB point of the DI-aided design curve, the LDPC decoder may require 4 transmission to have a successfully decoding. However, the DI operation decided to activate the decoding process from the third transmission based on the EXIT chart analysis. Therefore, the redundant operation for decoding the third retransmission dramatically increase the total complexity of the DI-aided scheme, which cause the peak appeared in the complexity plot. Similar behaviors also happens at the SNR=0dB and 5.2dB, where DI-aided design offers optimistic estimation. By contrast, the complexity of the novel EA-aided HARQ system decreased continually. Due to the early-stopping that decided by the state of convergence introduced in Section 5.2.2.2, the

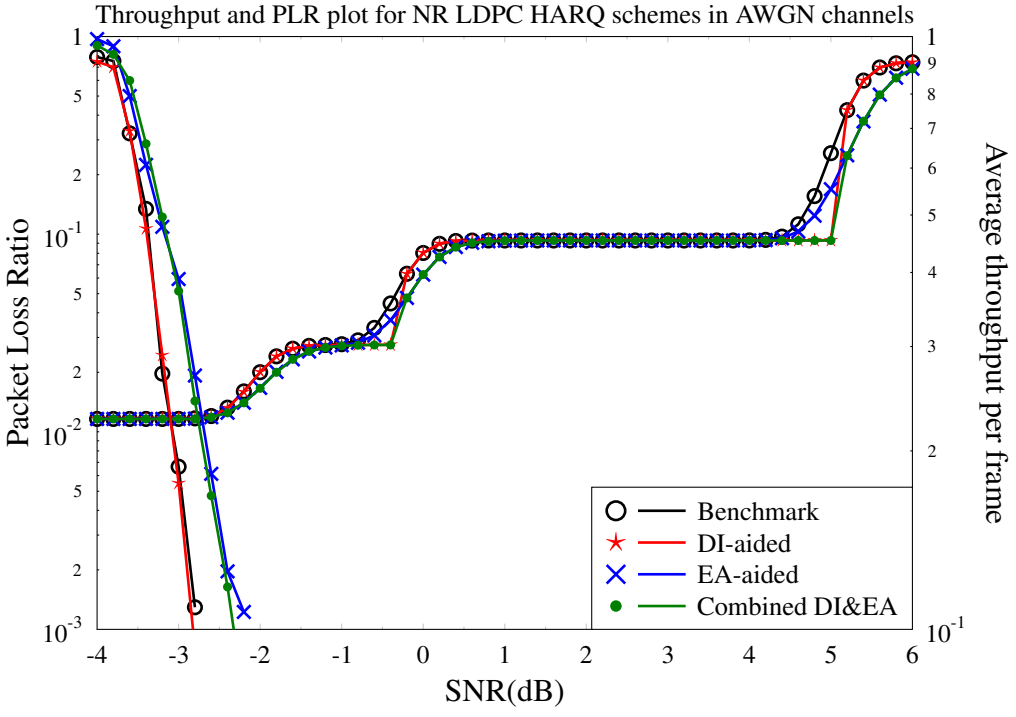
complexity of EA-aided HARQ system reduced significantly compared with the benchmark by eliminating the redundant decoding iterations. Finally, the proposed combined DI and EA design shows a similar behaviour to that of the DI-aided regime. Since the EA technique eliminates the redundant decoding iterations, the proposed HARQ system becomes capable of reducing the complexity further, below that of the DI-attached HARQ system, especially in the low channel SNR range. In conclusion, Figure 5.15 (a) gives the evidence that the proposed combined HARQ system relying on the DI and EA techniques is the lowest-complexity design option.

In order to granting the benefits of the proposed NR HARQ system, Figure 5.15 (b) presents its PLR and average throughput performance. In contrast to the conventional BLER of Figure 5.3, this plot uses PLR for quantifying the HARQ performance. The PLR mentioned in this section is considered as the rate of passing a package successfully within limit number of retransmissions. For example, under the circumstance that the number of retransmission is set as four, if one of ten transmitting packages is successfully decoded at the receiver side, the PLR of this system is 10%. In Figure 5.15 (b), we set the max number of re-transmission of a single LDPC codeword to 4. On the other hand, the effective throughput is another essential parameter reflecting the capability of the HARQ system. The throughput per frame of the HARQ system is calculated as the ratio of the number of information bits and transmitted bits. For instance, this simulation in Figure 5.15 (b) is using an information length of $K' = 1000$ and encoded bits of $E = 1200$ in the NR LDPC decoder. The maximum throughput of this HARQ system is $K'/E_r = 0.83$ as seen at the right y-axis in the plot. In the PLR comparison of these four HARQ system, the benchmark and the DI-aided HARQ designs have approximately 0.5dB SNR gain over the EA-aided HARQ and the combined HARQ designs. This trend is due to the employment of the EA technique, which curtails the decoding process in the interest of improving the PLR error-correction performance. More specifically, the EA process may sometimes abandon the decoding process that would otherwise succeed if given a chance to complete. In order to eliminate this gap of error-correction capability, we have tuned the threshold value of fully convergence I_{conv} for the best balance of complexity reduction and error-correction performance. Our future research will investigate the sensitivity of this threshold value for different block lengths. On the other hand, the average throughput per frame of the four simulated HARQ designs becoming similar. Figure 5.15 (b) gives the confidence that the employment of our proposed techniques reduced the complexity without sacrificing the throughput or PLR.

Although DI-aided HARQ design shows better error-correction performance in Figure 5.15 (b), our combined HARQ system gives approximately 50% complexity reduction in the low-SNR range and 30% complexity reduction in the high-SNR domain as shown



(a)



(b)

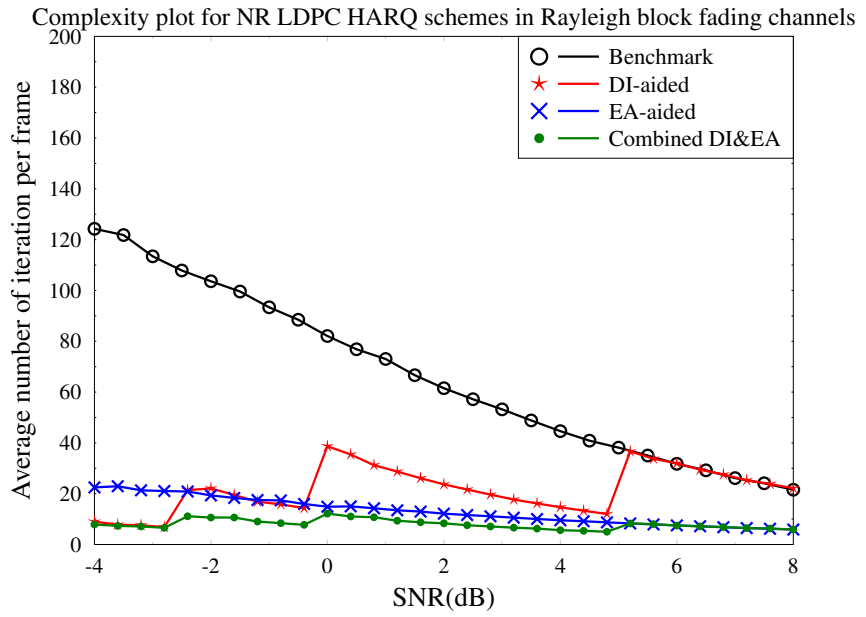
FIGURE 5.15: Comparison of performance between conventional NR HARQ system (benchmark), proposed DI-aided HARQ, proposed EA-aided HARQ and combined proposed system in **AWGN channel**. All simulated HARQ systems are adopted with information bits of $K = 1000$, encoded bits of $E = 1200$ and BG1. The comparison is presented in three perspectives. The average of required complexity for each successful transmitted frame is shown in (a). While the PLR and the average throughput of each HARQ system is shown in (b). All other parameters are summarized in Table 5.2.

in Figure 5.15 (a). Since both schemes provided the same throughput, the proposed combined HARQ system has the edge in the AWGN transmission environment.

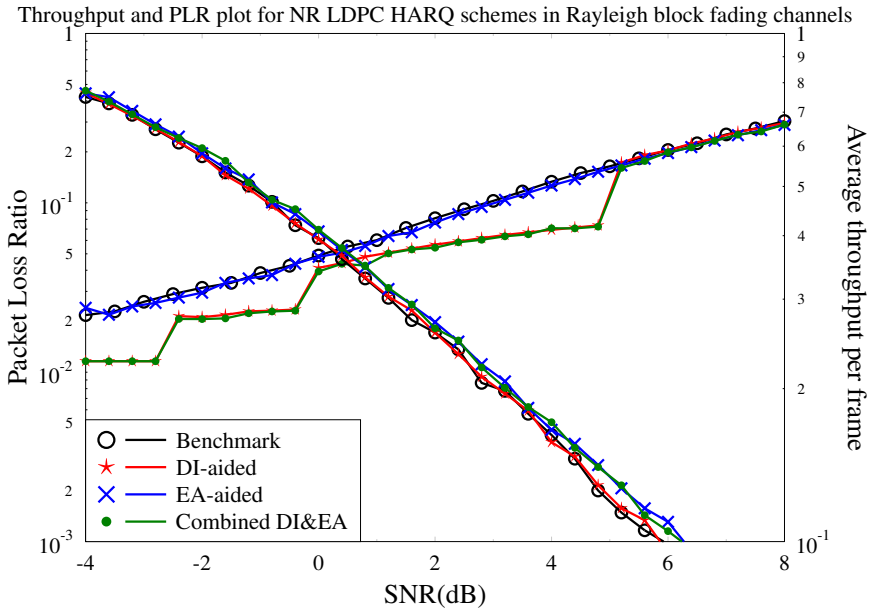
5.2.3.2 Performance of the proposed HARQ system in Rayleigh block fading channels

In order to demonstrate the performance of proposed HARQ technique in real-world wireless communication, we repeated our investigations in Rayleigh block fading transmission channels [149]. In contrast to the uncorrelated Rayleigh fading channel discussed in Section 5.1.3.2, Rayleigh block fading imposes the same fading envelope on all transmitted symbols in the same transmission frame. In this type of transmission channel, the performance of the HARQ-aided decoder will become much more random than in AWGN transmissions. Some early transmission attempts might perform better than later attempts because of their different Rayleigh fading envelopes. However, the block fading will also lead to a slower improvement of the error-correction performance upon increasing the SNR than in the AWGN channel.

Figure 5.16 portrays the performance of the NR HARQ system in Rayleigh block fading channels. Similar to the AWGN simulations, the Rayleigh block fading simulations characterized the throughput, complexity and PLR. Figure 5.16 (a) provides our complexity comparison among the four HARQ systems in terms of average iteration number per frame. Similar to the results of Section 5.2.3.1, the complexity of the DI-aided HARQ system and the proposed combined HARQ system exhibit some ripples caused by the DI technique. However, the complexity of the benchmark no longer has the staircase-shaped reduction upon increasing the SNR. Furthermore, at the beginning of the simulations, the benchmark HARQ system operating in Rayleigh block fading channels requires less iterations for the successful decoding than in the AWGN channel. This trend is caused by the random nature of Rayleigh block fading for each transmission attempt of the same frame, which is due to the different SNR of each re-transmission. By contrast, for the AWGN channel, we have the same SNR for each transmission attempt for the same frame. Due to this specific feature of Rayleigh block fading, the complexity of the EA-aided HARQ system also reduced smoothly, i.e. without any ripples. In conclusion, in the scenario where the HARQ system requires more than one transmission of a single frame, the proposed combined HARQ system has the lowest requirement of complexity, which is approximately valued as 83% reduction compared to that of the benchmark scheme. However, if the SNR is high enough for the decoder to successfully decode the transmitted frame during the first attempt that EA-aided HARQ system requires the same number of decoding iterations as the proposed combined HARQ system.



(a)



(b)

FIGURE 5.16: Comparison of performance between conventional NR HARQ system (benchmark), proposed DI-aided HARQ, proposed EA-aided HARQ and combined proposed system in **Rayleigh block fading channel**. All simulated HARQ systems are adopted with information bits of $K = 1000$, encoded bits of $E = 1200$ and BG1. The comparison is presented in three perspectives. The average of required complexity for each successful transmitted frame is shown in (a). While the PLR and the average throughput of each HARQ system is shown in (b).All other parameters are summarized in Table 5.2.

Figure 5.16(b) portrays the PLR and average throughput per frame for four selected HARQ schemes. Since all PLR curves of these four schemes are similar to each other, our complexity-reduction techniques do not degrade the error-correction capability at the HARQ system. As for the average throughput per frame, due to neglecting some potentially unsuccessful transmission attempts under the control of the DI technique and under the proposed combined HARQ system, they have a lower average throughput per frame than others, until the SNR becomes high enough for the decoder to successfully clean up the transmitted frame in the first attempt. As a result, the benchmark and the EA-aided HARQ system have the best throughput.

Compared to Figure 5.16(a) and (b), we demonstrated that the error-correction capability of the NR HARQ system will not be affected by eliminating the redundant decoding attempts by the DI or EA technique. Clearly, the EA-aided HARQ system and the proposed combined HARQ scheme outperforms the other two in Rayleigh block fading transmission channels. However, in exchange of its significant computational complexity reduction, the proposed combined HARQ scheme has to pay a price in terms of its reduced average throughput per frame compared to the EA-aided design.

5.2.4 Conclusions

In this section, we introduced a novel design of the NR HARQ system with the aid of the EXIT analysis that introduced in the Chapter 4. Inspired by the turbo HARQ design that described in [15], we proposed two techniques to enhance the efficiency of the conventional standardized NR HARQ system, which are termed as DI and EA. The DI employment helps the HARQ system to skip the transmissions that are expected to fail the decoding of the transmitted bits. While the EA technique allows early curtailing the decoding process by checking the state of convergence of *extrinsic* MI in the NR LDPC decoder. As described above, both DI and EA schemes require our EXIT analysis from Chapter 4, which also proves the feasibility of our EXIT analysis for NR LDPC decoder.

Our proposed scheme of NR HARQ system is a combination of both DI and EA implementation. In order to support the performance of our proposed design, we presented the simulation between conventional standardized NR HARQ system, which is referred as benchmark, the DI-aided HARQ system, the EA-aided HARQ system and our proposed HARQ scheme in this section. The simulation is focused on three aspects, including complexity, error-correction performance and average throughput per frame. In the AWGN channels, our proposed NR HARQ design significantly reduces the complexity of successful decoding. As summarised in the Table 5.3, in the particular case of our simulation, when the number of iteration per transmission is set as 50, our proposed scheme can always achieve the successful decoding within the 50 iterations. While all

other three schemes have the higher number of required decoding iterations than the proposed system for all channel SNRs. In the low SNR domain, our proposed system relying on DI and EA techniques will reduce the complexity by 30% compared to the DI-aided and the EA-aided scheme. While comparing to the benchmark, the proposed system significantly decrease the complexity by approximately 80% in the low SNR domain.

For the performance on the throughput, all four selected systems have matched value of the average throughput per received frame as seen in Table 5.3. Finally, for the error-correction performance, which is termed by PLR, the benchmark and the DI-aided HARQ system showed approximately 0.5dB gain compared with EA-aided and our proposed HARQ system. Because of the EA technique is sometimes acting aggressively, and abandoning some decoding process that would otherwise succeed if left to continue. However, the large amount of reduction on complexity still made our proposed NR HARQ system becomes the most efficient scheme in this simulation. In conclusion, in the case where the performance of error-correction and throughput is key criterion, we recommended the DI-aided scheme. While the complexity is considered as the key criterion, we then recommended the proposed scheme relying on DI and EA techniques.

TABLE 5.3: Summary table for NR HARQ schemes simulations in AWGN channels.

Transmissions in AWGN channels						
	Average number of iterations			Average throughput		
HARQ schemes	-4 dB	1 dB	6 dB	-4 dB	1 dB	6 dB
Benchmark	180	60	10	0.22	0.45	0.83
DI-aided	30	10	10	0.22	0.45	0.83
EA-aided	45	13	10	0.22	0.45	0.83
Combine DI&EA	25	10	10	0.22	0.45	0.83

TABLE 5.4: Summary table for NR HARQ schemes simulations in Rayleigh block fading channels.

Transmissions in Rayleigh block fading channels						
	Average number of iterations			Average throughput		
HARQ schemes	-4 dB	1 dB	6 dB	-4 dB	1 dB	6 dB
Benchmark	120	75	40	0.3	0.38	0.58
DI-aided	10	30	40	0.22	0.35	0.58
EA-aided	20	13	10	0.3	0.38	0.58
Combine DI&EA	10	12	10	0.22	0.35	0.58

Table 5.4 summaries the performance of NR HARQ schemes in the Rayleigh block fading channels. Since the PLR performance for all four schemes are matched to each other, we will only discuss the complexity and throughput as shown in the table. In this table, the proposed combined NR HARQ design has the lowest computational complexity among four schemes in a large range of SNR. Due to the nature of uncertainty carried

by the Rayleigh fading, the benefit of our proposed design has been observed as more obvious and stable. The proposed scheme outperforms the benchmark by approximately 83% of reduction in complexity for all simulated SNR domain. Comparing to the DI-aided scheme and the EA-aided scheme, the proposed system also has around 30% of complexity reduction averagely among all SNRs. However, caused by the employment of DI controlling, the DI-aided scheme and the proposed model have lower average throughput per frame than the benchmark and the EA-aided design. This reason for this is the DI control acting aggressively and not attempting the decoding that would be successful if commenced. As shown in Table 5.4, approximately 17% of average throughput is sacrificed by the proposed scheme for the complexity reduction. It makes the selection of NR HARQ system in the Rayleigh block fading environment becomes a trade-off between less complexity and higher throughput.

Here we assume the hardware is provisioned for the worst-case processing requirement, but in case where DI and EA reduces the complexity, we can turn off the hardware to achieve the power saving. Alternatively, the saved hardware resources in the implementation could be re-allocated for other tasks. In the case where the hardware is pooled for multiple users or base-stations, then the complexity reduction can be expected to remain fairly constant on average across a large number of users. Therefore, instead of provisioning the worst-case requirement, the cost of hardware can be reduced for the average-case requirement. Nonetheless, the benefit of saved resources is another worthy topic for future research.

5.3 Conclusion

This section explicitly conclude the entire Chapter 5. In this chapter, we introduced two applications that enhance the performance of the standardized NR LDPC coded system by employing the novel EXIT analysis of Chapter 4. Firstly, we proposed a novel NR LDPC decoder activation order based on the MI exchange behaviour that shown on our novel EXIT analysis. We observed that only the MI that exchanged between core CNs and core VNs can be seen to improve in the last few iterations, where it can therefore be reasoned that it is mainly the core CNs that contribute. Therefore, we decided to deactivating the extension CN operation after the *extrinsic* MI of the extension CN reaches 99.99% of its margin. As a result, the proposed NR LDPC decoder reduced approximately 14% to 19% average complexity compared to the flooding based decoder at a specific BLER and SNR without any extra cost. Secondly, inspired by the unique HARQ design of convolutional code in [15], we proposed a combined NR HARQ system for LDPC coded transmission relying on the DI and the EA techniques. By using our EXIT analysis for NR LDPC decoder in the DI scheme, the receiver is able to skip the

transmission that can be expected to fail the decoding of the transmitted bits. On the other hand, the EA technique focused on the change of *extrinsic* MI in LDPC decoder. If the decoding is not considered as succeed after an iteration, the EA technique allows early curtailing the decoding iterations if the decoder is fully converged. As a conclusion, the proposed HARQ scheme relying on DI and EA designs significantly reduces the complexity compared to that of the conventional HARQ system without performance degrading.

Chapter 6

Conclusions and Future Research

This chapter presents our conclusion and future researches emerging from this thesis. More specifically, Section 6.1 summaries all results and findings, while Section 6.2 describes potential future ideas that related to our observations. Finally, Section 6.3 provides our closing remarks.

6.1 Conclusions and summary

We commenced by providing some background on three fundamental channel coding schemes, including turbo codes, LDPC codes and polar codes, concluding with a summary of the characteristics of channel decoder ASIC implementations. Then, motivated by the global research on 5G NR wireless communication, we focused our efforts on the NR LDPC coding scheme. We comprehensively described the construction of the NR LDPC coding layer ratified by the 3GPP 5G standard [1]. Given the inadequacy of conventional 2D EXIT chart analysis, we designed a novel 3D EXIT analysis technique, which can also be projected to 2D EXIT chart. Then, we introduced an NR LDPC decoder, which utilized our 3D EXIT charts to determine its decoding activation order. This novel NR LDPC decoder reduces the computational complexity of the conventional flooding based LDPC decoder. Moreover, our EXIT analysis is also employed in the NR HARQ system for LDPC coded data transmission. The accurate monitoring of MI exchange assist the HARQ system is identifying the redundant transmissions and decoding iterations, which significantly reduces the complexity of the conventional NR HARQ system. More specifically, we summarize our main contributions as follows.

In **Chapter 3**, we provided comprehensive discussions on the extent to which the 5G requirements can be met by the turbo, LDPC and polar codes. In order to support

our discussions, we presented several plots to characterise a diverse set of as many as 110 ASIC implementations of turbo, LDPC and polar decoders. We have characterised the fundamental trade-offs between the various performance characteristics of channel decoder ASICs. Furthermore, we have demonstrated that the overall implementation complexity of a channel code depends not only on its computational complexity, but also on its interconnection complexity and its inherent flexibility. We also summarised some recommendations for both researchers and potential customers of ASIC channel decoder implementations. As a conclusion, we found out that the LDPC code is superior in the range of long code block length among the three channel decoders, which is also be standardised as the channel coding scheme for the data transmission in 5G NR. While the polar code is been clarified as the most potential channel decoder in the range of short block length in our survey. It is also been standardised as the channel coding scheme for the control signal transmission in 5G NR.

In **Chapter 4**, we introduced the 3GPP-standardised NR LDPC code for the 5G data channel channel coding in [1]. We have shown that conventional 2D EXIT chart analysis is unable to accurately characterize the MI exchange behaviour of the decoder and hence it also fails to correctly distinguish whether the EXIT tunnel is open or closed. In order to develop a comprehensive understanding of the NR LDPC decoder, we introduced a novel reorganisation of its factor graph, which separates the decoder into four elements, namely core VNs, core CNs, extension VNs and extension CNs. To complement our novel factor graph, we modified the MI equations of [26], in order to develop a novel 3D EXIT chart representation of the NR LDPC decoder. We demonstrated that our 3D EXIT chart is able to accurately characterise the MI exchange in each decoding iteration, as well as the EXIT chart's tunnel state. For increased convenience, this chapter also introduced a novel 2D EXIT projection of the proposed 3D EXIT chart, which makes it clear if the EXIT tunnel is open or closed. These EXIT charts correctly illustrate the MI exchange characteristics of the NR LDPC decoder, which inspired further implementations that would be described in Chapter 5.

In **Chapter 5**, we firstly proposed a novel LDPC decoder activation order with the aid of our 3D EXIT chart analysis. More specifically, we observed that in the NR LDPC decoder, the extension CNs offer little further benefit after a few iterations. Therefore, the proposed activation order of decoding nodes is designed to deactivate the extension CN decoding, when the correspond MI convergence reaches 99.99% of its fully converged value. By comparing the simulation results between a flooding LDPC decoder and the proposed activation-order-based LDPC decoder, it was found that the proposed decoder offers approximately 0.2 dB gain, when having the same computational complexity and BLER performance. In other simulations, we show that when achieving the same BLER at a given SNR, the complexity of the proposed decoder is around

19% lower than that of the flooding-based decoder. The performance improvement of the proposed activation order-based decoder presented in Section 5.1.3 is listed in Table 5.1. In the second half of this chapter, we proposed a novel NR HARQ system for LDPC coded data transmission. The proposed HARQ system is combined with DI and EA techniques that eliminate the dispensable re-transmissions and decoding iterations. More specifically, the DI exploits the observation of the EXIT tunnel state to distinguish whether the transmissions are likely to be successfully decoded or not. If the decision from the DI operation is negative, the HARQ system will skip the decoding operation of the current transmission and directly asks for re-transmission, while the EA technique terminates the potentially futile decoding process with the aid of our EXIT charts. As a result, compared to the conventional NR HARQ system, the proposed HARQ system reduces the computational complexity by approximately 50% compared to that of the conventional HARQ scheme without degrading either the throughput or the error-correction capability. These complexity reduction in both of the proposed NR LDPC decoder and NR HARQ scheme are supported to be considered as a method of power saving in the real-time implementation. The saved energy could be used as other hardware resources, or directly shut down to avoid any waste. These applications proved the feasibility of our proposed 3D EXIT chart analysis for the NR LDPC decoder.

6.2 Future research

This section presents some ideas for potential future research related to this thesis. Section 6.2.1 provides suggestions and recommendations for future of channel decoder ASIC implementations. As for the EXIT analysis of NR LDPC decoder, Section 6.2.2 offers some future research ideas based on the 3D EXIT charts, as well as on its 2D projection. Finally, Section 6.2.3 provides potential enhancements for EXIT analysis aided application, including the activation order-aided NR LDPC decoder and the NR HARQ system.

6.2.1 Future ideas on channel coding implementations

Chapter 3 presented a survey of the three fundamental channel coding ASIC implementations. This survey summarised and compared the diverse characteristics of turbo codes, LDPC codes and of polar codes ASIC implementations, including their throughput, computational complexity and flexibility. Based on this, we would like to propose some points for future research on channel coding ASIC implementations based on our survey. There is still a significant amount of further work that can be completed on the implementation of channel decoders. In particular, there is a need for further research on

flexible, high-performance, high-efficiency implementations of turbo, LDPC and polar decoders. For example, the hardware implementation of irregular turbo decoding [102] has not yet been demonstrated, which would improve the attainable error correction capability. Likewise, the performance of LDPC decoders can be significantly enhanced by invoking informed dynamic activation orders [103]. However, this benefit typically comes at the cost of increased implementation complexity owing to its adaptive decoding schedules. Polar decoding can be enhanced by implementing SISO decoding [104], which would facilitate ASIC-based turbo detection for the first time for polar codes. Likewise, the ASIC implementation of HARQ [105] for polar decoding is another challenge that remains unsolved at the time of writing.

6.2.2 Further investigation of EXIT analysis for NR LDPC decoder

Based on the EXIT chart analysis of Section 4.4, there are further related research opportunities that can be explored as future work. The introduction of the proposed 2D EXIT projection create opportunity to improve the design of the NR LDPC code. As narrated in [148], the area properties and the match between the EXIT curves of a 2D EXIT chart can be used to optimise the relationship between coding rate and capacity loss. Furthermore, in the design of irregular LDPC code, the EXIT chart can be used to investigated the degree distribution. Since the 2D EXIT chart is quite mature for this topic, our novel 3D EXIT chart is worth for investigating the specific custom NR LDPC code for different requirements and scenarios.

6.2.3 EXIT chart analysis based NR LDPC applications

In Chapter 5, we introduced the DI and EA design inspired by our EXIT chart analysis for the NR LDPC code, including the activation order-aided NR LDPC decoder and our novel NR HARQ system. Although we demonstrated that our proposed designs are superior to their conventional counterparts, there is still plenty room for enhanced performance. For example, the proposed NR LDPC decoder activation order can be improved for practical ASIC implementation as future work. Firstly, an activation order based on layered belief propagation can be developed, which will reduce the complexity and memory requirement of a practical implementation. Furthermore, rather than using the SPA, a computational complexity reduction can be achieved by using a variant of the fixed-point MSA instead, which uses only binary logical and integer operations in hardware implementation [124]. Due to the standardised NR modulation and coding scheme has been decided, we could have more investigation on the proposed NR LDPC decoder with different modulation methods that mentioned in TS 38.214 [113]. Therefore

we are able to find the best match between our proposed decoder and the modulation scheme.

Furthermore, the utilization of off-line LUT will occupy a large portion of memory in the hardware implementation of the proposed NR HARQ system. There could be a better combination of a LUT and the on-live calculation of decisions during the decoding process. Moreover, the choice of I_{dump} and I_{conv} can be more accurate for further reducing the computational complexity.

6.3 Closing remarks

This thesis has presented the fundamental characteristics channel coding, including turbo codes, LDPC codes and polar codes as well as their ASIC implementations. It showed the reason that LDPC and polar code are chosen for the 5G NR channel coding system. Moreover, this thesis demonstrated that the conventional 2D EXIT analysis is not applicable for the NR LDPC decoder. Hence, it provided solutions, including our novel 3D EXIT charts and their 2D EXIT projections by re-arranging the factor graph of the NR LDPC decoder. We then introduced a pair of inspired applications, namely the activation order-aided NR LDPC decoder and our NR HARQ system conceived for LDPC data transmission. The proposed scheduled NR LDPC decoder has shown superior error-correction capability and reduced complexity compared to the conventional flooding based LDPC decoder. Our NR HARQ system significantly decreased the complexity by removing redundant re-transmissions and decoding iterations without degrading the throughput. As a result, unveiling this novel EXIT chart analysis for the NR LDPC decoder can be expected to offer significant further developments of the NR LDPC code and its implementation. Finally, our proposed designs will reduce the cost of NR LDPC implementations for 5G wireless systems.

Appendix A

Table 1: Comparison of ASIC implementation of channel decoders.

Ref.	Code and Standard	Coding Rate	Information Block Length	Average Scaled Throughput (decoded Mbps)	Scaled Average Area Efficiency (Mbps/mm^2)	Scaled Average Power Efficiency (bit/nJ)	Energy Required (E_b/N_0 in dB)
[75]	Turbo LTE	-	6144	1388	1369.697	3.225806	0.49
[75]	Turbo HSDPA	-	5144	1130	1682.424	2.626197	0.49
[92]	Turbo LTE	-	6144	792	397.9899497	0	-
[93]	Turbo LTE	1/3	6144	1156.154	112.7902	0.65729443	0.5588
[93]	Turbo HSDPA+	1/3	5144	571.8462	228.0382	1.328912467	0.5804
[94]	Turbo LTE	1/3	6144	691.7	496.9109195	1.089291339	0.75
[95]	Turbo LTE	1/3	6144	1013	406.8273092	1.048654244	0.803
[96]	Turbo LTE	1/3	3200	781.2	875.2941176	0.990239574	0.81
[97]	Turbo LTE	1/3	6016	393.7846154	359.4990572	0	0.6
[98]	Turbo LTE	1/3	6144	257.6769231	146.1561624	1.739884693	0.633
[99]	Turbo LTE	1/3	6144	1.426153846	7.811899343	0.34200332	-
[70]	Turbo LTE	1/3	5144	47.8	149.375	0.637758506	0.794
[70]	Turbo LTE	1/3	6144	211.2	132	0.563575717	0.738
[150]	Turbo LTE	1/3	6144	607.6107692	190.9650686	2.233534661	0.503
[150]	Turbo LTE	1/3	6144	4578.923077	444.4823839	3.156789436	1.159
[50]	Turbo LTE	1/3	6144	21900	200.9174312	2.276980661	0.79
[151]	Turbo LTE	0.94	3016	2150	279.2207792	0	-
[152]	Turbo LTE	1/3	6144	1280	154.2168675	1.514792899	-
[153]	LDPC WiFi	2/3	576-2304	586.2307692	98.23210251	-	-
[61]	LDPC WiFi	1/2	336-8640	474.9231	164.6480033	2.884148375	1.573
[61]	LDPC WiFi	5/6	336-8640	2709.692	939.4055106	7.24301611	3.568
[154]	LDPC WiFi	1/2 to 23/24	576-2304	372.2404654	282.0729022	0.434860357	-
[155]	LDPC WiFi	1/2 to 5/6	576-2304	195.6433846	60.30219051	0.370536713	-
[156]	LDPC WiFi	-	576-2304	1107.858528	606.8405124	2.702093971	-
[157]	LDPC WiFi	1/2	1024	142.35	1259.734513	-	2.141
[88]	LDPC WiFi	1/2 to 13/16	1944	3145.88	1677.942495	8.851660101	-
[158]	LDPC WiFi	-	648-1944	8530.510666	1994.433147	-	-

Ref.	Code and Standard	Coding Rate	Information Block Length	Average Scaled Throughput (decoded Mbps)	Scaled Average Area Efficiency (Mbps/mm^2)	Scaled Average Power Efficiency (bit/nJ)	Energy Required (E_b/N_0 in dB)
[159]	LDPC WiFi	1/2,2/3,3/4,5/6	648-1944	1618.294	3499.014054	-	-
[153]	LDPC WiMAX	2/3	576-2304	586.2307692	98.23210251	-	-
[160]	LDPC WiMAX	1/2	576-2304	1205	1590.759076	3.035264484	2.448
[161]	LDPC WiMAX	1/2	576-2304	141	68.03377563	2.711538462	2.81
[61]	LDPC WiMAX	1/2	336-8640	474.9231	164.6480033	2.884148375	1.573
[61]	LDPC WiMAX	5/6	336-8640	2709.692	939.4055106	7.24301611	3.568
[162]	LDPC WiMAX	1/2	576-2304	116.3251454	35.68222213	0.440625551	1.752
[163]	LDPC WiMAX	1/2,2/3,3/4,5/6	576-2304	293.5	533.6363636	-	-
[164]	LDPC WiMAX	1/2,2/3,3/4,5/6	576-2304	190.5	142.4831713	-	-
[165]	LDPC WiMAX	2/3	576-2304	184.6176923	521.2689415	2.921166018	2.527
[154]	LDPC WiMAX	1/2 to 23/24	576-2304	372.2404654	282.0729022	0.434860357	-
[155]	LDPC WiMAX	1/2 to 5/6	576-2304	195.6433846	60.30219051	0.370536713	-
[156]	LDPC WiMAX	1/2,2/3,3/4,5/6	576-2304	1107.858528	606.8405124	2.702093971	-
[91]	LDPC WiMAX	1/2	576-2304	142.35	1259.734513	-	-
[166]	LDPC WiMAX	802.16e	576-2304	60.6	54.47191011	1.158699809	2.809
[167]	LDPC WiMAX	5/6	576-2304	248.7993231	177.9806162	-	-
[168]	LDPC WiMAX	1/2	576-2304	328.0492197	208.2852188	1.21499711	1.77
[79]	LDPC WiMAX	1/2	576-2304	535	869.9186992	1.897163121	2.156
[169]	LDPC WiMAX	5/6	576	255.36	1064	-	2.598
[170]	LDPC WiMAX	1/2	576-2304	150.6687598	340.8331373	0.913143999	1.708
[171]	LDPC WiMAX	1/2,2/3,3/4,5/6	576-2304	703.824	209.4714286	6.120208696	2.326
[172]	LDPC WiMAX	1/2,2/3,3/4,5/6	2304	1835	2293	-	2.25
[173]	LDPC	1/2,5/8,3/4,13/16	672	2920.6	272.3395	12.0972	-
[164]	LDPC DVB-S2	1/4-/10	64800	384	99.45609946	-	-
[76]	LDPC WiGig	1/2	672	6401.893491	1523.01153	38.56562344	3.042
[89]	LDPC WiGig	802.11ad	672	1260	2650.887574	-	-
[88]	LDPC WiGig	802.11ad	672	2140.384615	3752.591918	15.79619642	-
[88]	LDPC WiGig	802.11ad	672	2140.384615	4135.509461	18.66072027	-
[80]	LDPC WiGig	1/2	672	646.1538462	190.3201335	104.2183623	2.833
[80]	LDPC WiGig	5/8	672	1292.307692	380.640267	92.30769231	3.12
[80]	LDPC WiGig	3/4	672	2584.615385	761.2805341	68.01619433	3.439
[80]	LDPC WiGig	4/5	672	5169.230769	1522.561068	28.71794872	3.894
[90]	LDPC WiGig	802.11ad	672	3221.822485	4744.827329	60.10862846	-
[74]	LDPC 10GBASE-T	0.841	2048	17246.82617	6387.713397	-	3.913
[66]	LDPC 10GBASE-T	0.833	2048	71407.6848	21457.67086	-	-
[83]	LDPC 10GBASE-T	0.833	2048	8581.347656	3900.612571	-	3.864
[84]	LDPC 10GBASE-T	0.833	2048	200826.8329	97968.73371	-	-
[85]	LDPC 10GBASE-T	0.833	2048	20124.10156	7317.855114	-	4.018
[54]	LDPC 10GBASE-T	0.841	2048	52909.2503	10566.19644	47.66599126	4.011
[86]	LDPC 10GBASE-T	0.833	2048	69475.26367	25263.73224	-	4.007
[87]	LDPC 10GBASE-T	0.841	2048	47700	8912.096461	17.03571429	3.738
[81]	LDPC 10GBASE-T	0.841	2048	16186.15385	5799.402496	10.38239503	3.86

Ref.	Code and Standard	Coding Rate	Information Block Length	Average Scaled Throughput (decoded Mbps)	Scaled Average Area Efficiency (Mbps/mm^2)	Scaled Average Power Efficiency (bit/nJ)	Energy Required (E_b/N_0 in dB)
[73]	LDPC 10GBASE-T	0.83	3600	15680	1572.764	987.4994	-
[77]	LDPC 10GBASE-T	0.89	8352	334.769	26.80265	16.68112	4.2
[174]	Polar Inflexible	0.5	512	271.3846154	608.5235848	5.608407666	-
[175]	Polar Inflexible	0.5	-	246.1	396.9354839	-	2.83
[175]	Polar Inflexible	0.5	-	244.2	230.3773585	-	-
[175]	Polar Inflexible	0.5	-	242.4	128.9361702	-	-
[175]	Polar Inflexible	0.5	-	240.6	63.65079365	-	-
[176]	Polar Inflexible	0.5	512	692.3076923	-	-	-
[69]	Polar Inflexible	0.5	-	497.28	3656.470588	-	-
[69]	Polar Inflexible	0.5	-	493.48	2902.823529	-	-
[69]	Polar Inflexible	0.5	-	494.74	2078.739496	-	-
[69]	Polar Inflexible	0.5	-	486.06	1299.625668	-	-
[69]	Polar Inflexible	0.5	-	487.36	752.0987654	-	-
[69]	Polar Inflexible	0.5	-	474.07	398.3781513	-	-
[69]	Polar Inflexible	0.5	-	470.59	206.7618629	-	-
[69]	Polar Inflexible	0.5	-	476.69	107.1213483	-	-
[177]	Polar Inflexible	0.5	-	250.6153846	150.1467911	-	2.179
[177]	Polar Inflexible	0.5	-	171.6923077	46.62345382	-	2
[71]	Polar Inflexible	0.5	512	25600	7619.047619	33.16062176	2.516
[72]	Polar Inflexible	0.5	-	1165.153846	373.7928501	-	3.066
[72]	Polar Inflexible	0.5	-	931.8461538	181.4071844	-	-
[72]	Polar Inflexible	0.5	-	776.7692308	52.73338225	-	-
[72]	Polar Inflexible	0.5	-	666	11.24443781	-	-
[72]	Polar Inflexible	0.5	-	582.9230769	2.232327392	-	-
[72]	Polar Inflexible	0.5	-	1280.769231	688.1836422	-	3.322
[72]	Polar Inflexible	0.5	-	1048.153846	166.4577849	-	-
[72]	Polar Inflexible	0.5	-	886.1538462	32.787141	-	-
[72]	Polar Inflexible	0.5	-	768.4615385	4.686787536	-	-
[72]	Polar Inflexible	0.5	-	677.7692308	1.227982273	-	-
[178]	Polar Inflexible	0.5	512	300	154.6391753	0.46728972	2.172
[178]	Polar Inflexible	0.5	512	501	236.3207547	0.634177215	2.172
[178]	Polar Inflexible	0.5	512	262	66.16161616	0.178474114	1.9
[178]	Polar Inflexible	0.5	512	401	93.69158879	0.279247911	1.9
[179]	Polar Inflexible	0.5	512	463.8461538	505.2654446	-	2.256
[179]	Polar Inflexible	0.5	512	425.0769231	228.9158352	-	2.098
[179]	Polar Inflexible	0.5	512	340.6153846	91.20308801	-	2.061
[180]	Polar Inflexible	0.5	480	342	291.1494845	-	1.887
[180]	Polar Inflexible	0.5	480	433.3846154	343.6176785	-	1.887
[180]	Polar Inflexible	0.5	480	459.6923077	264.0214487	-	1.896
[180]	Polar Inflexible	0.5	480	441.6923077	237.5967228	-	-
[180]	Polar Inflexible	0.5	480	551.0769231	376.5155037	-	2.111
[180]	Polar Inflexible	0.5	480	524.7692308	331.1607815	-	-

Ref.	Code and Standard	Coding Rate	Information Block Length	Average Scaled Throughput (decoded Mbps)	Scaled Average Area Efficiency (Mbps/mm^2)	Scaled Average Power Efficiency (bit/nJ)	Energy Required (E_b/N_0 in dB)
[181]	Polar Inflexible	0.5	1024	609.2307692	78.28369419	-	1.372
[182]	Polar Inflexible	0.5	4096	218.8034188	-	-	-
[183]	Polar Inflexible	0.5	512	1273.846154	163.4649659	-	1.729
[184]	Polar Inflexible	0.5	512	221.5384615	523.0598824	-	2.142
[184]	Polar Inflexible	0.5	512	250.6153846	434.4210953	-	-
[184]	Polar Inflexible	0.5	512	200.7692308	237.5967228	-	-
[184]	Polar Inflexible	0.5	512	239.5384615	202.841652	-	-
[185]	Polar Inflexible	0.5	512	1844.307692	892.8869947	-	-
[185]	Polar Inflexible	0.5	512	1578.461538	395.0604373	-	-
[185]	Polar Inflexible	0.5	512	1035.692308	137.5060678	-	-
[185]	Polar Inflexible	0.5	4096	2323.384615	651.2133771	-	1.86
[185]	Polar Inflexible	0.5	4096	2002.153846	297.0935599	-	1.697
[185]	Polar Inflexible	0.5	4096	1326.461538	103.712834	-	1.584
[185]	Polar Inflexible	0.5	16384	2997.129481	461.8951806	-	-
[185]	Polar Inflexible	0.5	16384	2724.94276	219.6867375	-	-
[185]	Polar Inflexible	0.5	16384	2034.480402	84.42476116	-	-
[186]	Polar Inflexible	0.85	32768	1093.4	189.29	-	-
[186]	Polar Inflexible	0.5	32768	639.8	110.75	-	-
[187]	Polar Inflexible	0.5	1024	80.81	34.081	1.957	2.06
[188]	Polar Inflexible	0.5	1024	1240	1797.1	12.9	-
[189]	Polar Inflexible	0.5	1024	825.23	1182.8	2.13	-
[67]	Polar Flexible	0.5	-	4043.076923	25330.79868	28.0466258	-
[67]	Polar Flexible	0.5	-	3918.461538	11112.89467	19.54453217	-
[67]	Polar Flexible	0.5	-	3890.769231	4913.851263	16.97296451	-
[67]	Polar Flexible	0.5	-	3724.615385	2358.217553	14.21488358	-
[67]	Polar Flexible	0.5	-	3544.615385	1057.515405	12.86819348	2.533
[71]	Polar Flexible	0.5	-	25600	7485.380117	56.63716814	-
[71]	Polar Flexible	0.5	-	25600	8888.888889	23.44322344	-
[71]	Polar Flexible	0.5	-	25600	2983.682984	33.77308707	-
[71]	Polar Flexible	0.5	-	25600	3575.418994	17.2972973	-
[188]	Polar Flexible	0.5	-	3720	2695.652174	8.651162791	2.648
[188]	Polar Flexible	0.5	-	2480	1797.101449	12.91666667	-
[68]	Polar Flexible	0.5	-	1514.769231	768.268418	-	-
[68]	Polar Flexible	0.5	-	1514.769231	625.8738405	-	1.857
[68]	Polar Flexible	0.5	-	3345.230769	1382.185739	-	2.139
[68]	Polar Flexible	0.5	-	7994.769231	3303.286613	-	2.591
[68]	Polar Flexible	0.5	-	1855.384615	396.1101036	-	-
[68]	Polar Flexible	0.5	-	1855.384615	322.1982545	-	1.639
[68]	Polar Flexible	0.5	-	4079.076923	708.3552671	-	1.77
[68]	Polar Flexible	0.5	-	9700.615385	1684.567889	-	2.078
[68]	Polar Flexible	0.5	-	2556.923077	245.8390191	-	-
[68]	Polar Flexible	0.5	-	2556.923077	205.2776399	-	4.221

Ref.	Code and Standard	Coding Rate	Information Block Length	Average Scaled Throughput (decoded Mbps)	Scaled Average Area Efficiency (Mbps/mm^2)	Scaled Average Power Efficiency (bit/nJ)	Energy Required (E_b/N_0 in dB)
[68]	Polar Flexible	0.5	-	2636.923077	211.7002857	-	4.221
[68]	Polar Flexible	0.5	-	2786.153846	223.6809903	-	4.242
[68]	Polar Flexible	0.5	-	5483.076923	440.1982603	-	4.311
[68]	Polar Flexible	0.5	-	13075.38462	1049.732047	-	4.455
[187]	Polar Flexible	0.5	1024	80.554	2279.5	220.5	1.8
[186]	Polar Flexible	0.5	1024	797.8	21.295	-	-
[186]	Polar Flexible	0.5	2048	13.29	1.873	-	1.3
[187]	Polar Flexible	0.5	1024	80.812	34.08	1.94	-
[187]	Polar Flexible	0.5	1024	56.38	24.78	1.04	-
[190]	Polar Flexible	0.5	1024	1145	112.12	-	-
[190]	Polar Flexible	0.5	1024	1340	318.4	-	-
[190]	Polar Flexible	0.5	1024	1527	644.9	-	-

Glossary

3D	Three-Dimensional
3GPP	Third Generation Partnership Project
3G	3rd Generation
4G	4th Generation
5G	Fifth Generation
ACS	Add Compare Select
APP	*A Posteriori* Probability
ASIC	Application Specific Integrated Circuit
AWGN	Additive White Gaussian Noise
BCJR	Bahl-Cocke-Jelinek-Raviv
BEC	Binary Erasure Channel
BER	Bit Error Rate
BG	Base Station
BLER	Block Error Rate
BPSK	Binary Phase Shift Keying
BTWC	Better-Than-Worst-Case
CC	Chase Combining
CMP	Column Message Passing
CN	Check Node
CND	Check Node Decoder
CRC	Cyclic Redundancy Check
CST	Communication Survey and Tutorial
DC	Decoding Cycle
DCMC	Discrete-input Continuous-output Memoryless Channel
DFF	D-type Flip Flop
DI	Deferred Iteration
DTS	Double Thresholding Strategy
EA	Early Abandon
eMBB	enhanced Mobile Broad Band
ES	Early Stopping
ET	Early Termination
EXIT	EXtrinsic Information Transfer
FEC	Forward Error Correction
FER	Frame Error Rate
FL	Frame Length
FPGA	Field Programmable Gate Array
FPTD	Fully-Parallel Turbo Decoder
FX	Fixed-point
HARQ	Hybrid Automatic Repeat reQuest
IoT	Internet-of-Things
IR	Incremental Redundancy
LDGM	Low Density Generator Matrix

LDPC	Low Density Parity Check
LLR	Logarithmic Likelihood Ratio
Log-BCJR	Logarithmic Bahl-Cocke-Jelinek-Raviv
Log-MAP	Logarithmic Maximum *A Posteriori*
Log-SPA	Logarithmic Sum-Product Algorithm
LOS	Line-of-Sight
LTE	Long Term Evolution
MAP	Maximum *a-posteriori*
MBB	Mobile Broad Band
MI	Mutual Information
mMTC	massive Machine-Type Communications
mmWave	Millimeter Wave
MSA	Min-Sum Algorithm
NB-IoT	Narrow Band Internet of Things
NR	New Radio
NSW	Non-Sliding Window
PCB	Printed Circuit Board
PCM	Parity Check Matrix
PDSCH	Physical Downlink Shared Channel
PLR	Package Loss Ratio
QPP	Quadratic Polynomial Permutation
QPSK	Quaternary Phase Shift Keying
RCMP	Row-Column Message Passing
RMP	Row Message Passing
RRC	Radio Resource Control
SC	Successive Cancellation
SCL	Successive Cancellation List
SISO	Single-Input-Single-Output
SNR	Signal-to-Noise Ratio
SPA	Sum-Product Algorithm
SSR	Segment Selective Repeat
SW	Sliding Window
UCI	Uplink Control Information
UMTS	Universal Mobile Telecommunication System
URLLC	Ultra-Reliable Low-Latency Communication
VLSI	Very-large-scale Integration
VN	Variable Node
VND	Variable Node Decoder

Bibliography

[1] "NR Multiplexing and channel coding V15.1.0," April 2018. [Online]. Available: http://3gpp.org/ftp/Specs/archive/38{_}series/38.212/

[2] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon Limit Error-Correcting Coding and Decoding: Turbo-Codes. 1," in *Communications, 1993. ICC '93 Geneva. Technical Program, Conference Record, IEEE International Conference on*, vol. 2, May 1993, pp. 1064–1070 vol.2.

[3] Y. Hawwar, E. Farag, S. Vanakayala, R. Pauls, X. Yang, S. Subramanian, P. Sadhanala, L. Yang, B. Wang, Z. Li, H. Chen, Z. Lu, D. Clark, T. Fosket, P. Mallela, M. Shelton, D. Laurens, T. Salaun, L. Gougeon, N. Aubourg, H. Morvan, N. L. Henaff, G. Prat, F. Charles, C. Creach, Y. Calvez, and P. Butel, "3G UMTS Wireless System Physical Layer: Baseband Processing Hardware Implementation Perspective," *IEEE Communications Magazine*, vol. 44, no. 9, pp. 52–58, Sept 2006.

[4] Z. Shen, A. Papasakellariou, J. Montojo, D. Gerstenberger, and F. Xu, "Overview of 3GPP LTE-Advanced Carrier Aggregation for 4G Wireless Communications," *IEEE Communications Magazine*, vol. 50, no. 2, pp. 122–130, February 2012.

[5] D. J. C. MacKay and R. M. Neal, "Near Shannon limit performance of Low Density Parity Check codes," *Electron. Lett.*, vol. 32, no. 18, pp. 457–458, aug 1996.

[6] *IEEE 802.11-2012 Standard for Information Technology - Telecommunications and Information Exchange between Systems - Local and Metropolitan Area Networks - Specific Requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY)*, 2012.

[7] K. H. Teo, Z. Tao, and J. Zhang, "The Mobile Broadband WiMAX Standard [Standards in a Nutshell]," *IEEE Signal Processing Magazine*, vol. 24, no. 5, pp. 144–148, Sept 2007.

[8] C. J. Hansen, "WiGiG: Multi-Gigabit Wireless Communications in the 60 GHz Band," *IEEE Wireless Communications*, vol. 18, no. 6, pp. 6–7, December 2011.

[9] A. Morello and V. Mignone, "DVB-S2: The Second Generation Standard for Satellite Broad-Band Services," *Proceedings of the IEEE*, vol. 94, no. 1, pp. 210–227, Jan 2006.

[10] J. R. Westra, J. Mulder, Y. Ke, D. Vecchi, X. Liu, E. Arslan, J. Wan, Q. Zhang, S. Wang, F. van der Goes, and K. Bult, "Design Considerations for Low-power Analog Front Ends in Full-Duplex 10GBASE-T Transceivers," in *Proceedings of the IEEE 2014 Custom Integrated Circuits Conference*, Sept 2014, pp. 1–8.

[11] E. Arikan, "Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels," *IEEE Trans. Inform. Theory*, vol. 55, no. 7, pp. 3051–3073, July 2009.

[12] R. G. Maunder, "The 5G Channel Code Contenders," *AccelerComm White Paper*, Aug 2016. [Online]. Available: http://www.accelercomm.com

[13] P. Hailes, L. Xu, R. G. Maunder, B. M. Al-Hashimi, and L. Hanzo, "A Survey of FPGA-based LDPC Decoders," *IEEE Commun. Surveys Tutorials*, vol. 18, no. 2, pp. 1098–1122, Second quarter 2016.

[14] A. Kraskov, H. Stögbauer, and P. Grassberger, "Estimating mutual information," *Phys. Rev. E*, vol. 69, p. 066138, Jun 2004. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevE.69.066138

[15] H. Chen, R. G. Maunder, and L. Hanzo, "A Survey and Tutorial on Low-Complexity Turbo Coding Techniques and a Holistic Hybrid ARQ Design Example," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 4, pp. 1546–1566, November 2013. [Online]. Available: https://eprints.soton.ac.uk/347846/

[16] H. Mercier, V. K. Bhargava, and V. Tarokh, "A survey of error-correcting codes for channels with symbol synchronization errors," *IEEE Communications Surveys Tutorials*, vol. 12, no. 1, pp. 87–96, 2010.

[17] C. Roth, A. Cevrero, C. Studer, Y. Leblebici, and A. Burg, "Area, throughput, and energy-efficiency trade-offs in the VLSI implementation of LDPC decoders," in *2011 IEEE International Symposium of Circuits and Systems (ISCAS)*, May 2011, pp. 1772–1775.

[18] Nasruminallah and L. Hanzo, "Near-Capacity H.264 Multimedia Communications Using Iterative Joint Source-Channel Decoding," *IEEE Communications Surveys Tutorials*, vol. 14, no. 2, pp. 538–564, Second 2012.

[19] Y. Fang, G. Bi, Y. L. Guan, and F. C. M. Lau, "A Survey on Protograph LDPC Codes and Their Applications," *IEEE Communications Surveys Tutorials*, vol. 17, no. 4, pp. 1989–2016, Fourthquarter 2015.

[20] E. Ankan, N. ul Hassan, M. Lentmaier, G. Montorsi, and J. Sayir, "Challenges and some new directions in channel coding," *Journal of Communications and Networks*, vol. 17, no. 4, pp. 328–338, Aug 2015.

[21] M. F. Brejza, L. Li, R. G. Maunder, B. Al-Hashimi, C. Berrou, and L. Hanzo, "20 years of turbo coding and energy-aware design guidelines for energy-constrained wireless applications," *IEEE Communications Surveys & Tutorials*, pp. 1–21, June 2015. [Online]. Available: https://eprints.soton.ac.uk/378161/

[22] H. B. Thameur, B. L. Gal, N. Khouja, F. Tlili, and C. Jego, "A survey on decoding schedules of LDPC convolutional codes and associated hardware architectures," in *2017 IEEE Symposium on Computers and Communications (ISCC)*, July 2017, pp. 898–905.

[23] R. G. Gallager, "Low-Density Parity-Check codes," *IRE Transactions on Information Theory*, vol. 8, no. 1, pp. 21–28, Jan 1962.

[24] S. ten Brink, "Convergence of iterative decoding," *Electronics Letters*, vol. 35, no. 10, pp. 806–808, 1999.

[25] ——, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Transactions on Communications*, vol. 49, no. 10, pp. 1727–1737, Oct 2001.

[26] S. ten Brink, G. Kramer, and A. Ashikhmin, "Design of Low-Density Parity-Check codes for modulation and detection," *IEEE Transactions on Communications*, vol. 52, no. 4, pp. 670–678, April 2004.

[27] M. Ardakani and F. R. Kschischang, "A more accurate one-dimensional analysis and design of irregular LDPC codes," *IEEE Transactions on Communications*, vol. 52, no. 12, pp. 2106–2114, 2004.

[28] G. Liva and M. Chiani, "Protograph LDPC Codes Design Based on EXIT Analysis," in *IEEE GLOBECOM 2007 - IEEE Global Telecommunications Conference*, 2007, pp. 3250–3254.

[29] X. Zuo, R. G. Maunder, and L. Hanzo, "Design of Fixed-Point Processing Based LDPC Codes Using EXIT Charts," in *2011 IEEE Vehicular Technology Conference (VTC Fall)*, 2011, pp. 1–5.

[30] M. El-Hajjar and L. Hanzo, "EXIT Charts for System Design and Analysis," *IEEE Communications Surveys Tutorials*, vol. 16, no. 1, pp. 127–153, 2014.

[31] T. Richardson and S. Kudekar, "Design of Low-Density Parity Check Codes for 5G New Radio," *IEEE Communications Magazine*, vol. 56, no. 3, pp. 28–34, March 2018.

[32] Y. Julian, R. P. Astuti, and K. Anwar, "EXIT Analysis for Decoding Behaviour and Performances of 5G NR QC-LDPC Codes," in *2018 21st International Symposium on Wireless Personal Multimedia Communications (WPMC)*, Nov 2018, pp. 437–442.

[33] C. E. Shannon, "The mathematical theory of communication," *Bell System Technical Journal*, vol. 27, pp. 379–423, jul 1948.

[34] K. Mallinson, "The path to 5G: as much evolution as revolution," 2016. [Online]. Available: http://3gpp.org/news-events/3gpp-news/1774-5g-wiseharbour

[35] F.-L. Luo and C. Zhang, *5G Standard Development: Technology and Roadmap*. Wiley-IEEE Press, 2016, pp. 616–. [Online]. Available: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7572796

[36] Samsung, "R1-162181 Channel Coding for 5G New Radio Interface," in *3GPP TSG RAN WG1 #84bis*, Apr 2016.

[37] S. P. L.-E. T. Wehn Norbert, Scholl Stefan and M. Alles, *Challenges and Limitations for Very High Throughput Decoder Architectures for Soft-Decoding*. Cham, Switzerland: Springer International Publishing, 2015, pp. 7–31.

[38] R. S. Groups, "Minimum Requirements Related to Technical Performance for IMT-2020 Radio Interface(s)," February 2017.

[39] Samsung, "R1-1609073 Discussion on Latency of Channel Codes for NR," in *3GPP TSG RAN WG1 #86*, Oct 2016.

[40] A. Nimbalker, Y. Blankenship, B. Classon, and T. K. Blankenship, "ARP and QPP Interleavers for LTE Turbo Coding," in *Wireless communications and networking conference, 2008. WCNC 2008. IEEE*. IEEE, 2008, pp. 1032–1037.

[41] J. Hamorsky, U. Wachsmann, J. B. Huber, and A. Cizmar, "Hybrid Automatic Repeat Request Scheme with Turbo Codes," *Brest,France, International Symposium on Turbo Codes*, pp. 247–250, Sept 1997.

[42] Y. D. Beyene, R. Jantti, K. Ruttik, and S. Iraji, "On the performance of narrow-band internet of things (NB-IoT)," in *2017 IEEE Wireless Communications and Networking Conference (WCNC)*, March 2017, pp. 1–6.

[43] J. P. Woodard and L. Hanzo, *Comparative Study of Turbo Decoding Techniques: an Overview*, Nov 2000, vol. 49, no. 6, pp. 2208–2233.

[44] R. Johannesson and K. S. Zigangirov, *Decoding of Convolutional Codes*. Wiley-IEEE Press, 2015, pp. 550–. [Online]. Available: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7199872

[45] R. Bhownath and H. Xu, "Repeat-punctured turbo trellis-coded modulation," in *AFRICON, 2011*, Sept 2011, pp. 1–6.

[46] T. Kailath, "A general likelihood-ratio formula for random signals in Gaussian noise," *IEEE Transactions on Information Theory*, vol. 15, no. 3, pp. 350–361, May 1969.

[47] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal Decoding of Linear Codes for Minimizing Symbol Error Rate (Corresp.)," *IEEE Transactions on Information Theory*, vol. 20, no. 2, pp. 284–287, Mar 1974.

[48] S. Yoon and Y. Bar-Ness, "A parallel MAP algorithm for low latency turbo decoding," *IEEE Communications Letters*, vol. 6, no. 7, pp. 288–290, July 2002.

[49] T. Ilnseher, F. Kienle, C. Weis, and N. Wehn, "A 2.15GBit/s Turbo Code Decoder for LTE Advanced Base Station Applications," in *2012 7th International Symposium on Turbo Codes and Iterative Information Processing (ISTC)*, Aug 2012, pp. 21–25.

[50] A. Li, L. Xiang, T. Chen, R. G. Maunder, B. M. Al-Hashimi, and L. Hanzo, "VLSI implementation of fully parallel LTE turbo decoders," *IEEE Access*, vol. 4, pp. 323–346, 2016.

[51] *ETSI TS 136 212 LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Multiplexing and Channel Coding*, V13.1.0 ed., 2016.

[52] H. T. Pham, S. Ajaz, and H. Lee, "Parallel Block-Layered Nonbinary QC-LDPC Decoding on GPU," in *2015 IEEE Workshop on Signal Processing Systems (SiPS)*, Oct 2015, pp. 1–6.

[53] K. Shimizu, T. Ishikawa, N. Togawa, T. Ikenaga, and S. Goto, "Partially-Parallel LDPC Decoder Based on High-Efficiency Message-Passing Algorithm," in *2005 International Conference on Computer Design*, Oct 2005, pp. 503–510.

[54] C.-C. Cheng, J.-D. Yang, H.-C. Lee, C.-H. Yang, and Y.-L. Ueng, "A fully parallel LDPC decoder architecture using probabilistic min-sum algorithm for high-throughput applications," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 61, no. 9, pp. 2738–2746, 2014.

[55] D. E. Hocevar, "A reduced complexity decoder architecture via layered decoding of LDPC codes," in *IEEE Workshop onSignal Processing Systems, 2004. SIPS 2004.*, Oct 2004, pp. 107–112.

[56] B. Li, H. Shen, and D. Tse, "An Adaptive Successive Cancellation List Decoder for Polar Codes with Cyclic Redundancy Check," *IEEE Communications Letters*, vol. 16, no. 12, pp. 2044–2047, December 2012.

[57] C. Xiong, J. Lin, and Z. Yan, "A multimode area-efficient SCL polar decoder," *IEEE Trans. VLSI Syst.*, 2016.

[58] Y. S. Park, Y. Tao, S. Sun, and Z. Zhang, "A 4.68Gb/s belief propagation polar decoder with bit-splitting register file," in *Proc. Symp. VLSI Circuits*, June 2014.

[59] S. A. Hashemi, C. Condo, and W. J. Gross, "Fast and Flexible Successive-Cancellation List Decoders for Polar Codes," *IEEE Transactions on Signal Processing*, vol. 65, no. 21, pp. 5756–5769, Nov 2017.

[60] C. Studer, C. Benkeser, S. Belfanti, and Q. Huang, "Design and Implementation of a Parallel Turbo-Decoder ASIC for 3GPP-LTE," *IEEE Journal of Solid-State Circuits*, vol. 46, no. 1, pp. 8–17, Jan 2011.

[61] Y.-L. Ueng, B.-J. Yang, C.-J. Yang, H.-C. Lee, and J.-D. Yang, "An efficient multi-standard LDPC decoder design using hardware-friendly shuffled decoding," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 60, no. 3, pp. 743–756, 2013.

[62] Huawei, "R1-1608865 Design aspects of Polar Code and LDPC for NR," in *3GPP TSG RAN WG1 #86*, Oct 2016.

[63] G. Montorsi and S. Benedetto, "Design of fixed-point iterative decoders for concatenated codes with interleavers," *IEEE Journal on Selected Areas in Communications*, vol. 19, no. 5, pp. 871–882, 2001.

[64] K. C. Saraswat and F. Mohammadi, "Effect of Scaling of Interconnections on the Time Delay of VLSI Circuits," *IEEE Journal of Solid-State Circuits*, vol. 17, no. 2, pp. 275–280, Apr 1982.

[65] A. Stillmaker and B. Baas, "Scaling equations for the accurate prediction of CMOS device performance from 180nm to 7nm," *Integration*, vol. 58, pp. 74–81, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167926017300755

[66] S. S. Tehrani, A. Naderi, G.-A. Kamendje, S. Hemati, S. Mannor, and W. J. Gross, "Majority-based tracking forecast memories for stochastic LDPC decoding," *IEEE Transactions on Signal Processing*, vol. 58, no. 9, pp. 4883–4896, 2010.

[67] O. Dizdar and E. Arıkan, "A high-throughput energy-efficient implementation of successive cancellation decoder for polar codes using combinational logic," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 63, no. 3, pp. 436–447, 2016.

[68] C. Xiong, J. Lin, and Z. Yan, "A multimode area-efficient SCL polar decoder," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 24, no. 12, pp. 3499–3512, 2016.

[69] Y. Fan and C.-y. Tsui, "An efficient partial-sum network architecture for semi-parallel polar codes decoder implementation," *IEEE Transactions on Signal Processing*, vol. 62, no. 12, pp. 3165–3179, 2014.

[70] C.-H. Lin, C.-Y. Chen, and A.-Y. Wu, "Area-efficient scalable map processor design for high-throughput multistandard convolutional turbo decoding," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 19, no. 2, pp. 305–318, 2011.

[71] P. Giard, G. Sarkis, C. Thibeault, and W. J. Gross, "Multi-mode unrolled architectures for polar decoders," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 63, no. 9, pp. 1443–1453, 2016.

[72] J. Sha, X. Liu, Z. Wang, and X. Zeng, "A memory efficient belief propagation decoder for polar codes," *China Communications*, vol. 12, no. 5, pp. 34–41, 2015.

[73] K. Cushon, P. Larsson-Edefors, and P. Andrekson, "A High-Throughput Low-Power Soft Bit-Flipping LDPC Decoder in 28 nm FD-SOI," in *ESSCIRC 2018 - IEEE 44th European Solid State Circuits Conference (ESSCIRC)*, Sept 2018, pp. 102–105.

[74] Q. Zhang, Y. Chen, D. Wu, X. Zeng, and Y.-l. Ueng, "Convergence-optimized variable node structure for stochastic LDPC decoder," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 6535–6539.

[75] Q. Yang, X. Zhou, G. E. Sobelman, and X. Li, "Network-on-chip for turbo decoders," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 24, no. 1, pp. 338–342, 2016.

[76] M. Li, J.-W. Weijers, V. Derudder, I. Vos, M. Rykunov, S. Dupont, P. Debacker, A. Dewilde, Y. Huang, L. Van der Perre, *et al.*, "An energy efficient 18Gbps LDPC decoding processor for 802.11 ad in 28nm CMOS," in *Solid-State Circuits Conference (A-SSCC), 2015 IEEE Asian*. IEEE, 2015, pp. 1–5.

[77] Y. Toriyama and D. Markovic, "A 2.267-Gb/s, 93.7-pJ/bit Non-Binary LDPC Decoder With Logarithmic Quantization and Dual-Decoding Algorithm Scheme for Storage Applications," *IEEE Journal of Solid-State Circuits*, vol. 53, no. 8, pp. 2378–2388, Aug 2018.

[78] S. L. Goff, A. Glavieux, and C. Berrou, "Turbo-codes and high spectral efficiency modulation," in *Communications, 1994. ICC '94, SUPERCOMM/ICC '94, Conference Record, 'Serving Humanity Through Communications.' IEEE International Conference on*, May 1994, pp. 645–649 vol.2.

[79] B. Xiang, D. Bao, S. Huang, and X. Zeng, "A fully-overlapped multi-mode QC-LDPC decoder architecture for mobile WiMAX applications," in *Application-specific Systems Architectures and Processors (ASAP), 2010 21st IEEE International Conference on*. IEEE, 2010, pp. 225–232.

[80] M. Weiner, M. Blagojevic, S. Skotnikov, A. Burg, P. Flatresse, and B. Nikolic, "27.7 A scalable 1.5-to-6Gb/s 6.2-to-38.1 mW LDPC decoder for 60GHz wireless networks in 28nm UTBB FDSOI," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2014 IEEE International*. IEEE, 2014, pp. 464–465.

[81] A. Cevrero, Y. Leblebici, P. Ienne, and A. Burg, "A 5.35 mm^2 10GBASE-T Ethernet LDPC decoder chip in 90 nm CMOS," in *Solid State Circuits Conference (A-SSCC), 2010 IEEE Asian*. IEEE, 2010, pp. 1–4.

[82] I. Tal and A. Vardy, "List Decoding of Polar Codes," *IEEE Transactions on Information Theory*, vol. 61, no. 5, pp. 2213–2226, May 2015.

[83] Q. Zhang, Y. Chen, D. Wu, X. Zeng, and Y.-l. Ueng, "An area-efficient architecture for stochastic LDPC decoder," in *Digital Signal Processing (DSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 244–247.

[84] A. Naderi, S. Mannor, M. Sawan, and W. J. Gross, "Delayed stochastic decoding of LDPC codes," *IEEE Transactions on Signal Processing*, vol. 59, no. 11, pp. 5617–5626, 2011.

[85] D. Wu, Y. Chen, Q. Zhang, Y.-l. Ueng, and X. Zeng, "Strategies for Reducing Decoding Cycles in Stochastic LDPC Decoders," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 63, no. 9, pp. 873–877, 2016.

[86] D. Wu, Y. Chen, Q. Zhang, L. Zheng, X. Zeng, and Y.-l. Ueng, "Latency-optimized stochastic LDPC decoder for high-throughput applications," in *Circuits and Systems (ISCAS), 2015 IEEE International Symposium on*. IEEE, 2015, pp. 3044–3047.

[87] Z. Zhang, V. Anantharam, M. J. Wainwright, and B. Nikolic, "An efficient 10GBASE-T ethernet LDPC decoder design with low error floors," *IEEE Journal of Solid-State Circuits*, vol. 45, no. 4, pp. 843–855, 2010.

[88] M. Li, F. Naessens, M. Li, P. Debacker, C. Desset, P. Raghavan, A. Dejonghe, and L. Van der Perre, "A processor based multi-standard low-power LDPC engine for multi-gbps wireless communication," in *Global Conference on Signal and Information Processing (GlobalSIP), 2013 IEEE*. IEEE, 2013, pp. 1254–1257.

[89] A. Balatsoukas-Stimming, N. Preyss, A. Cevrero, A. Burg, and C. Roth, "A parallelized layered QC-LDPC decoder for IEEE 802.11 ad," in *New Circuits and Systems Conference (NEWCAS), 2013 IEEE 11th International*. IEEE, 2013, pp. 1–4.

[90] M. Li, Y. Lee, Y. Huang, and L. Van der Perre, "Area and energy efficient 802.11 ad LDPC decoding processor," *Electronics Letters*, vol. 51, no. 4, pp. 339–341, 2015.

[91] M. Alles, T. Vogt, and N. Wehn, "Flexichap: A reconfigurable ASIC for convolutional, turbo, and LDPC code decoding," in *Turbo Codes and Related Topics, 2008 5th International Symposium on*. IEEE, 2008, pp. 84–89.

[92] J. Cheng, Q. Yang, and X. Zhou, "Design of a high parallelism high throughput HSPA+ Turbo decoder," in *ASIC (ASICON), 2015 IEEE 11th International Conference on*. IEEE, 2015, pp. 1–4.

[93] G. Wang, H. Shen, Y. Sun, J. R. Cavallaro, A. Vosoughi, and Y. Guo, "Parallel interleaver design for a high throughput HSPA+/LTE multi-standard turbo decoder," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 61, no. 5, pp. 1376–1389, 2014.

[94] X. Chen, Y. Chen, Y. Li, Y. Huang, and X. Zeng, "A 691 Mbps 1.392 mm^2 configurable radix-16 turbo decoder ASIC for 3GPP-LTE and WiMAX systems in 65nm CMOS," in *Solid-State Circuits Conference (A-SSCC), 2013 IEEE Asian*. IEEE, 2013, pp. 157–160.

[95] S. Belfanti, C. Roth, M. Gautschi, C. Benkeser, and Q. Huang, "A 1Gbps LTE-advanced turbo-decoder ASIC in 65nm CMOS," in *VLSI Circuits (VLSIC), 2013 Symposium on*. IEEE, 2013, pp. C284–C285.

[96] C. Studer, C. Benkeser, S. Belfanti, and Q. Huang, "Design and implementation of a parallel turbo-decoder ASIC for 3GPP-LTE," *IEEE Journal of Solid-State Circuits*, vol. 46, no. 1, pp. 8–17, 2011.

[97] A. Ahmed, M. Awais, A. ur Rehman, M. Maurizio, and G. Masera, "A high throughput turbo decoder VLSI architecture for 3GPP LTE standard," in *Multitopic Conference (INMIC), 2011 IEEE 14th International*. IEEE, 2011, pp. 340–346.

[98] C.-H. Lin, C.-Y. Chen, E.-J. Chang, and A.-Y. Wu, "A 0.16 nJ/bit/iteration 3.38 mm^2 turbo decoder chip for WiMAX/LTE standards," in *Integrated Circuits (ISIC), 2011 13th International Symposium on*. IEEE, 2011, pp. 168–171.

[99] L. Li, R. G. Maunder, B. M. Al-Hashimi, and L. Hanzo, "A low-complexity turbo decoder architecture for energy-efficient wireless sensor networks," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 21, no. 1, pp. 14–22, 2013.

[100] H.-A. Loeliger, "An introduction to factor graphs," *IEEE Sig. Process. Mag.*, vol. 21, no. 1, pp. 28–41, jan 2004.

[101] R. Shrestha and R. P. Paily, "High-throughput turbo decoder with parallel architecture for LTE wireless communication standards," *IEEE Trans. Circuits Syst. I*, vol. 61, no. 9, pp. 2699–2710, Sept 2014.

[102] J. Wang, S. X. Ng, A. Wolfgang, L.-L. Yang, S. Chen, and L. Hanzo, "Near-capacity three-stage MMSE turbo equalization using irregular convolutional codes," in *Proc. Int. Symp. on Turbo Codes*, Munich, Germany, Apr 2006.

[103] H. Lee, Y. Su, and Y. Ueng, "A Rate-Compatible Low-Density Parity-Check Convolutional Coding Scheme Using Informed Dynamic Scheduling," in *2017 IEEE 85th Vehicular Technology Conference (VTC Spring)*, June 2017, pp. 1–5.

[104] D. D. Le, D. P. Nguyen, T. H. Tran, and Y. Nakashima, "Joint polar and run-length limited decoding scheme for visible light communication systems," *IEICE Communications Express*, vol. 7, no. 1, pp. 19–24, 2018.

[105] K. Tian, R. Liu, A. Vardy, and R. Wang, "Low-Complexity Hybrid ARQ Scheme for Polar Codes with Higher-Order Modulation," in *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*, Dec 2017, pp. 1–6.

[106] R. Steele and L. Hanzo, *Mobile Radio Communications: Second and Third Generation Cellular and WATM Systems: 2nd.* IEEE Press - John Wiley, May 1999, address: Chichester, UK. [Online]. Available: https://eprints.soton.ac.uk/251446/

[107] L. Hanzo, J. Blogh, and S. Ni, *3G, HSDPA, HSUPA and FDD Versus TDD Networking: Smart Antennas and Adaptive Modulation.* IEEE Press - John Wiley, February 2008. [Online]. Available: https://eprints.soton.ac.uk/264586/

[108] L. Hanzo, H. Haas, S. Imre, D. O'Brien, M. Rupp, and L. Gyongyosi, "Wireless myths, realities, and futures: from 3g/4g to optical and quantum wireless," *Proceedings of the IEEE*, vol. 100, no. 13, pp. 1853–1888, May 2012, invited paper. [Online]. Available: https://eprints.soton.ac.uk/339181/

[109] MATLAB, "Physical channels," http://uk.mathworks.com/help/lte/physicalchannels.html, 2018.

[110] J.-F. Cheng, M. Åhlander, and C. Aalto, "Method and apparatus for providing adaptive cyclic redundancy check computation," Feb. 2013, uS Patent 8,375,271.

[111] T. V. Ramabadran and S. S. Gaitonde, "A tutorial on CRC computations," *IEEE Micro*, vol. 8, no. 4, pp. 62–75, 1988.

[112] "NR; Physical channels and modulation (Release 15)," January 2018.

[113] "Physical layer procedures for data (Release 15)," November 2017. [Online]. Available: https://portal.3gpp.org/ngppapp/CreateTdoc.aspx?mode=view&contributionId=840513

[114] W. W. Peterson and D. T. Brown, "Cyclic codes for error detection," *Proceedings of the IRE*, vol. 49, no. 1, pp. 228–235, 1961.

[115] "Multiplexing and Channel coding (Release)," 2008. [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=2426

[116] "Rate matching scheme for NR LDPC codes," Septmber 2017. [Online]. Available: https://portal.3gpp.org/ngppapp/CreateTdoc.aspx?mode=view&contributionId=822765

[117] Q. Wang, S. Cai, L. Chen, and X. Ma, "A Throughput-Enhanced HARQ Scheme for 5G System via Partial Superposition," *IEEE Communications Letters*, pp. 1–1, 2020.

[118] Q. C. Nokia, Alcatel-Lucent Shanghai Bell, "Summary of [89-24] Email Discussion on LDPC Code Base Graph #1 for NR," *3GPP TSG RAN WG1 NR Ad-Hoc #2, R1-1711545*, June 2017.

[119] ——, "Summary of [89-25] Email Discussion on LDPC Code Base Graph #2 for NR," *3GPP TSG RAN WG1 NR Ad-Hoc #2, R1-1711546*, June 2017.

[120] R. Tanner, "A recursive approach to low complexity codes," *IEEE Transactions on Information Theory*, vol. 27, no. 5, pp. 533–547, Sep. 1981.

[121] M. Tuchler, "Design of serially concatenated systems depending on the block length," *IEEE Transactions on Communications*, vol. 52, no. 2, pp. 209–218, 2004.

[122] L. Hanzo, T. Liew, B. Yeap, R. Tee, and S. X. Ng, *Turbo coding, turbo equalisation and space-time coding: EXIT-chart-aided near-capacity designs for wireless channels*. John Wiley & Sons, 2011, vol. 22.

[123] P. Robertson, "Illuminating the structure of code and decoder of parallel concatenated recursive systematic (turbo) codes," in *1994 IEEE GLOBECOM. Communications: The Global Bridge*, vol. 3, 1994, pp. 1298–1303 vol.3.

[124] H. Chen, K. Zhang, X. Ma, and B. Bai, "Comparisons Between Reliability-Based Iterative Min-Sum and Majority-Logic Decoding Algorithms for LDPC Codes," *IEEE Transactions on Communications*, vol. 59, no. 7, pp. 1766–1771, July 2011.

[125] X.Hu, E. Eleftheriou, D. . Arnold, and A. Dholakia, "Efficient implementations of the sum-product algorithm for decoding LDPC codes," in *GLOBECOM'01. IEEE Global Telecommunications Conference (Cat. No.01CH37270)*, vol. 2, 2001, pp. 1036–1036E vol.2.

[126] M. Franceschini, G. Ferrari, and R. Raheli, "Does the Performance of LDPC Codes Depend on the Channel?" *IEEE Transactions on Communications*, vol. 54, no. 12, pp. 2129–2132, 2006.

[127] D. Divsalar, "Ensemble Weight Enumerators for Protograph LDPC Codes," in *2006 IEEE International Symposium on Information Theory*, 2006, pp. 1554–1558.

[128] F. Babich, "Exit chart aided design of periodically punctured turbo codes," *Electronics Letters*, vol. 46, pp. 1001–1003(2), July 2010. [Online]. Available: https://digital-library.theiet.org/content/journals/10.1049/el.2010.1018

[129] M. F. Uddin Butt, R. A. Riaz, S. X. Ng, and L. Hanzo, "Near-capacity iterative decoding of binary self-concatenated codes using soft decision demapping and 3-d exit charts," *IEEE Transactions on Wireless Communications*, vol. 9, no. 5, pp. 1608–1616, 2010.

[130] A. P. Roberts, Michaelraj Kingston, "A Comparative Review of Recent Advances in Decoding Algorithms for Low-Density Parity-Check (LDPC) Codes and Their Applications," *Archives of Computational Methods in Engineering*, June 2020. [Online]. Available: https://doi.org/10.1007/s11831-020-09466-6

[131] J. Chen and M. P. C. Fossorier, "Density evolution for two improved BP-Based decoding algorithms of LDPC codes," *IEEE Communications Letters*, vol. 6, no. 5, pp. 208–210, 2002.

[132] B. Wang, Y. Zhu, and J. Kang, "Two Effective Scheduling Schemes for Layered Belief Propagation of 5G LDPC Codes," *IEEE Communications Letters*, pp. 1–1, 2020.

[133] A. Amaricai, O. Boncalo, and I. Mot, "Memory efficient FPGA implementation for flooded LDPC decoder," in *2015 23rd Telecommunications Forum Telfor (TELFOR)*, Nov 2015, pp. 500–503.

[134] E. Pisek, D. Rajan, and J. Cleveland, "Gigabit rate low-power LDPC decoder," in *2011 IEEE Information Theory Workshop*, Oct 2011, pp. 518–522.

[135] B. L. Gal, C. Jego, and J. Crenne, "A High Throughput Efficient Approach for Decoding LDPC Codes onto GPU Devices," *IEEE Embedded Systems Letters*, vol. 6, no. 2, pp. 29–32, June 2014.

[136] K. K. Parhi, "An improved pipelined MSB-first add-compare select unit structure for Viterbi decoders," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 51, no. 3, pp. 504–511, March 2004.

[137] F. Meshkati, H. Poor, S. Schwartz, and N. Mandayam, "An energy-efficient approach to power control and receiver design in wireless data networks," *IEEE Transactions on Communications*, vol. 53, no. 11, pp. 1885–1894, 2005.

[138] J. M. Wozencraft and M. Horstein, "Coding for two-way channels," 1961.

[139] E. Dahlman, S. Parkvall, *et al.*, "Hybrid ARQ for packet data transmission," Dec. 20 2005, uS Patent 6,977,888.

[140] J. Wozencraft and M. Horstein, "Digitalised communication over two-way channels," in *Fourth London Symp. Inform. Theory, London, UK*, 1960.

[141] H. A. Ngo and L. Hanzo, "Hybrid Automatic-Repeat-reQuest Systems for Cooperative Wireless Communications," *IEEE Communications Surveys Tutorials*, vol. 16, no. 1, pp. 25–45, 2014.

[142] D. Chase, "Code Combining - A Maximum-Likelihood Decoding Approach for Combining an Arbitrary Number of Noisy Packets," *IEEE Transactions on Communications*, vol. 33, no. 5, pp. 385–393, 1985.

[143] P. Frenger, S. Parkvall, and E. Dahlman, "Performance comparison of HARQ with Chase combining and incremental redundancy for HSDPA," in *IEEE 54th Vehicular Technology Conference. VTC Fall 2001. Proceedings (Cat. No.01CH37211)*, vol. 3, 2001, pp. 1829–1833 vol.3.

[144] "NR; Physical layer procedures for control v16.5.0," March 2021.

[145] CATT, "Rate matching scheme for NR LDPC codes," *3GPP TSG RAN WG1 Meeting NR#3*, Sept 2017. [Online]. Available: https://portal.3gpp.org/ngppapp/CreateTdoc.aspx?mode=view&contributionId=822765

[146] Ericsson, "Redundancy Version for HARQ of LDPC Codes," *3GPP TSG RAN WG1 Meeting NR#3*, Sept 2017. [Online]. Available: https://www.3gpp.org/ftp/TSG_RAN/WG1_RL1/TSGR1_AH/NR_AH_1709/Docs/R1-1715732.zip

[147] T. Richardson, "Error floors of LDPC codes," in *Proceedings of the annual Allerton conference on communication control and computing*, vol. 41, no. 3. The University; 1998, 2003, pp. 1426–1435.

[148] A. Ashikhmin, G. Kramer, and S. ten Brink, "Extrinsic Information Transfer Functions: Model and Erasure Channel Properties," *IEEE Transactions on Information Theory*, vol. 50, no. 11, pp. 2657–2673, Nov 2004.

[149] X. Lagrange, "Throughput of HARQ protocols on a block fading channel," *IEEE Communications Letters*, vol. 14, no. 3, pp. 257–259, 2010.

[150] R. Shrestha and R. P. Paily, "High-throughput turbo decoder with parallel architecture for LTE wireless communication standards," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 61, no. 9, pp. 2699–2710, 2014.

[151] T. Ilnseher, F. Kienle, C. Weis, and N. Wehn, "A 2.15 GBit/s turbo code decoder for LTE advanced base station applications," in *Turbo Codes and Iterative Information Processing (ISTC), 2012 7th International Symposium on*. IEEE, 2012, pp. 21–25.

[152] Y. Sun and J. R. Cavallaro, "Efficient hardware implementation of a highly-parallel 3GPP LTE/LTE-advance turbo decoder," *INTEGRATION, the VLSI journal*, vol. 44, no. 4, pp. 305–315, 2011.

[153] W. Zhang, S. Chen, X. Bai, and D. Zhou, "A full layer parallel QC-LDPC decoder for WiMAX and Wi-Fi," in *2015 IEEE 11th International Conference on ASIC (ASICON)*, 2015, pp. 1–4.

[154] D. Bao, B. Xiang, R. Shen, A. Pan, Y. Chen, and X. Y. Zeng, "Programmable Architecture for Flexi-Mode QC-LDPC Decoder Supporting Wireless LAN/MAN Applications and Beyond," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 57, no. 1, pp. 125–138, Jan 2010.

[155] C. Liu, C. Lin, S. Yen, C. Chen, H. Chang, C. Lee, Y. Hsu, and S. Jou, "Design of a Multimode QC-LDPC Decoder Based on Shift-Routing Network," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 56, no. 9, pp. 734–738, 2009.

[156] Y. Sun and J. R. Cavallaro, "A low-power 1-Gbps reconfigurable LDPC decoder design for multiple 4G wireless standards," in *2008 IEEE International SOC Conference*, 2008, pp. 367–370.

[157] C. Howland and A. Blanksby, "Parallel decoding architectures for Low Density Parity Check codes," in *ISCAS 2001. The 2001 IEEE International Symposium on Circuits and Systems (Cat. No.01CH37196)*, vol. 4, 2001, pp. 742–745 vol. 4.

[158] T. H. Tran, Y. Nagao, H. Ochi, and M. Kurosaki, "ASIC design of 7.7 Gbps multi-mode LDPC decoder for IEEE 802.11ac," in *2014 14th International Symposium on Communications and Information Technologies (ISCIT)*, 2014, pp. 259–263.

[159] K. Gunnam, G. Choi, W. Wang, and M. Yeary, "Multi-Rate Layered Decoder Architecture for Block LDPC Codes of the IEEE 802.11n Wireless Standard," in *2007 IEEE International Symposium on Circuits and Systems*, 2007, pp. 1645–1648.

[160] B. Xiang, D. Bao, S. Huang, and X. Zeng, "An 847–955 Mb/s 342–397 mW Dual-Path Fully-Overlapped QC-LDPC Decoder for WiMAX System in 0.13μm CMOS," *IEEE Journal of Solid-State Circuits*, vol. 46, no. 6, pp. 1416–1432, 2011.

[161] X.-Y. Shih, C.-Z. Zhan, C.-H. Lin, and A.-Y. Wu, "An 8.29 mm^2 52 mW Multi-Mode LDPC Decoder Design for Mobile WiMAX System in 0.13μ m CMOS Process," *Solid-State Circuits, IEEE Journal of*, vol. 43, pp. 672 – 683, 04 2008.

[162] C. Liu, S. Yen, C. Chen, H. Chang, C. Lee, Y. Hsu, and S. Jou, "An LDPC Decoder Chip Based on Self-Routing Network for IEEE 802.16e Applications," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 3, pp. 684–694, 2008.

[163] G. Gentile, M. Rovini, and L. Fanucci, "Low-Complexity Architectures of a Decoder for IEEE 802.16e LDPC Codes," in *10th Euromicro Conference on Digital System Design Architectures, Methods and Tools (DSD 2007)*, 2007, pp. 369–375.

[164] T. Brack, M. Alles, T. Lehnigk-Emden, F. Kienle, N. Wehn, N. E. L'Insalata, F. Rossi, M. Rovini, and L. Fanucci, "Low Complexity LDPC Code Decoders for Next Generation Standards," in *2007 Design, Automation Test in Europe Conference Exhibition*, 2007, pp. 1–6.

[165] Y. Wang, Y. Ueng, C. Peng, and C. Yang, "Processing-Task Arrangement for a Low-Complexity Full-Mode WiMAX LDPC Codec," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 58, no. 2, pp. 415–428, 2011.

[166] X.-Y. Shih, C.-Z. Zhan, C.-H. Lin, and A.-Y. Wu, "A 19-mode 8.29mm^2 52-mW LDPC decoder chip for IEEE 802.16e system," 07 2007, pp. 16 – 17.

[167] D. Yang, G. Yu, X. Zou, Y. Deng, and J. Zhong, "The design and verification of a novel LDPC decoder with high-efficiency," in *2014 International Symposium on Integrated Circuits (ISIC)*, 2014, pp. 256–259.

[168] S. Huang, D. Bao, B. Xiang, Y. Chen, and X. Zeng, "A flexible LDPC decoder architecture supporting two decoding algorithms," in *Proceedings of 2010 IEEE International Symposium on Circuits and Systems*, 2010, pp. 3929–3932.

[169] K. V. Kumar, R. Shrestha, and R. Paily, "Design and implementation of multi-rate LDPC decoder for IEEE 802.16e wireless standard," in *2014 International Conference on Green Computing Communication and Electrical Engineering (ICGCCEE)*, 2014, pp. 1–5.

[170] T. Kuo and A. N. Willson, "A flexible decoder IC for WiMAX QC-LDPC codes," in *2008 IEEE Custom Integrated Circuits Conference*, 2008, pp. 527–530.

[171] X. Peng, Z. Chen, X. Zhao, D. Zhou, and S. Goto, "A 115mW 1Gbps QC-LDPC decoder ASIC for WiMAX in 65nm CMOS," in *IEEE Asian Solid-State Circuits Conference 2011*, 2011, pp. 317–320.

[172] T. T. Nguyen-Ly, V. Savin, K. Le, D. Declercq, F. Ghaffari, and O. Boncalo, "Analysis and Design of Cost-Effective, High-Throughput LDPC Decoders," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 26, no. 3, pp. 508–521, 2018.

[173] M. Milicevic and P. G. Gulak, "A Multi-Gb/s Frame-Interleaved LDPC Decoder With Path-Unrolled Message Passing in 28-nm CMOS," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 26, no. 10, pp. 1908–1921, 2018.

[174] A. Mishra, A. J. Raymond, L. G. Amaru, G. Sarkis, C. Leroux, P. Meinerzhagen, A. Burg, and W. J. Gross, "A successive cancellation decoder ASIC for a 1024-bit polar code in 180nm CMOS," in *2012 IEEE Asian Solid State Circuits Conference (A-SSCC)*, 2012, pp. 205–208.

[175] C. Leroux, A. J. Raymond, G. Sarkis, and W. J. Gross, "A Semi-Parallel Successive-Cancellation Decoder for Polar Codes," *IEEE Transactions on Signal Processing*, vol. 61, no. 2, pp. 289–299, 2013.

[176] B. Yuan and K. K. Parhi, "Low-Latency Successive-Cancellation Polar Decoder Architectures Using 2-Bit Decoding," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 61, no. 4, pp. 1241–1254, 2014.

[177] A. Balatsoukas-Stimming, A. J. Raymond, W. J. Gross, and A. Burg, "Hardware Architecture for List Successive Cancellation Decoding of Polar Codes," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 61, no. 8, pp. 609–613, 2014.

[178] B. Yuan and K. K. Parhi, "Low-Latency Successive-Cancellation List Decoders for Polar Codes With Multibit Decision," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 23, no. 10, pp. 2268–2280, 2015.

[179] A. Balatsoukas-Stimming, M. B. Parizi, and A. Burg, "LLR-Based Successive Cancellation List Decoding of Polar Codes," *IEEE Transactions on Signal Processing*, vol. 63, no. 19, pp. 5165–5179, 2015.

[180] C. Xiong, J. Lin, and Z. Yan, "Symbol-Decision Successive Cancellation List Decoder for Polar Codes," *IEEE Transactions on Signal Processing*, vol. 64, no. 3, pp. 675–687, 2016.

[181] Y. Fan, J. Chen, C. Xia, C. Tsui, J. Jin, H. Shen, and B. Li, "Low-latency list decoding of polar codes with double thresholding," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 1042–1046.

[182] J. Lin, C. Xiong, and Z. Yan, "A reduced latency list decoding algorithm for polar codes," in *2014 IEEE Workshop on Signal Processing Systems (SiPS)*, 2014, pp. 1–6.

[183] Y. Fan, C. Xia, J. Chen, C. Tsui, J. Jin, H. Shen, and B. Li, "A Low-Latency List Successive-Cancellation Decoding Implementation for Polar Codes," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 2, pp. 303–317, 2016.

[184] J. Lin and Z. Yan, "An Efficient List Decoder Architecture for Polar Codes," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 23, no. 11, pp. 2508–2518, 2015.

[185] J. Lin, C. Xiong, and Z. Yan, "A High Throughput List Decoder Architecture for Polar Codes," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 24, no. 6, pp. 2378–2391, 2016.

[186] X. Liu, Q. Zhang, P. Qiu, J. Tong, H. Zhang, C. Zhao, and J. Wang, "A 5.16Gbps decoder ASIC for Polar Code in 16nm FinFET," in *2018 15th International Symposium on Wireless Communication Systems (ISWCS)*, 2018, pp. 1–5.

[187] P. Giard, A. Balatsoukas-Stimming, T. C. Müller, A. Bonetti, C. Thibeault, W. J. Gross, P. Flatresse, and A. Burg, "PolarBear: A 28-nm FD-SOI ASIC for Decoding of Polar Codes," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 7, no. 4, pp. 616–629, 2017.

[188] P. Giard, A. Balatsoukas-Stimming, G. Sarkis, C. Thibeault, and W. J. Gross, "Fast Low-Complexity Decoders for Low-Rate Polar Codes," *CoRR*, vol. abs/1603.05273, 2016. [Online]. Available: http://arxiv.org/abs/1603.05273

[189] R. Shrestha and A. Sahoo, "High-Speed and Hardware-Efficient Successive Cancellation Polar-Decoder," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 66, no. 7, pp. 1144–1148, 2019.

[190] C. Xia, J. Chen, Y. Fan, C. Tsui, J. Jin, H. Shen, and B. Li, "A High-Throughput Architecture of List Successive Cancellation Polar Codes Decoder With Large List Size," *IEEE Transactions on Signal Processing*, vol. 66, no. 14, pp. 3859–3874, 2018.