Error Correcting Codes for Communication, Security and Networks

Thesis submitted by

TIRTHADIP SINHA

Doctor of Philosophy (Engineering)

Department of Electronics and Telecommunication Engineering

Faculty Council of Engineering and Technology

Jadavpur University

Kolkata, India

2025

JADAVPUR UNIVERSITY

KOLKATA – 700 032, INDIA

INDEX NO. 151/19/E

Title of the Thesis:

Error Correcting Codes for Communication, Security and Networks

Name, Designation and Institution of the Supervisor:

Prof. Jaydeb Bhaumik
Professor
Department of Electronics and Telecommunication Engineering (ETCE),
Jadavpur University,
188, Raja S C Mallik Road, Jadavpur, Kolkata-700032

List of Publications

International Journals:

1. **T. Sinha** and J. Bhaumik, "Performance Analysis of NR Polar Codes at Short Information Blocks for Control Channels," *Wireless Personal Communications*, vol. 138, no. 2, pp. 879–890, Aug. 2024, doi: https://doi.org/10.1007/s11277-024-11530-4.

2. **T. Sinha** and J. Bhaumik, "Error-rate analysis of polar code designs using genetic algorithm for additive white Gaussian noise channel," *International Journal of Communication Systems*, vol. 36, no. 18, Aug. 2023, doi: https://doi.org/10.1002/dac.5611.

3. **T. Sinha** and J. Bhaumik, "Enhancing Polar code performance employing Genetic Algorithm for AWGN and Rayleigh Fading Channels," Communicated to Journal of Communication Technology and Electronics, Springer, *(under review).*

4. **T. Sinha**, J. Bhaumik, D. Giri and M. S. Obaidat, "Enhanced Error Correction Strategies for WSNs in IoT: Concatenated codes with interleaving and decoding optimization," Communicated to Journal of Transactions on Emerging Telecommunication Technologies, Willey, *(under review).*

5. **T. Sinha** and J. Bhaumik, "Artificial Noise-aided Secure Polar coding for Wireless networks," Communicated to Journal of Telecommunication Systems, Springer, *(under review).*

List of Presentations in International Conferences:

1. **T. Sinha** and J. Bhaumik, "Efficient and Novel Architecture of Golay Encoder and Decoder for McEliece Cryptosystem," *Proc. of 4th Int. Conf. on Communication, Devices and Computing (ICCDC 2023)*, Haldia Institute of Technology, West Bengal, India, March 1-3, 2023, doi: https://doi.org/10.1007/978-981-99-2710-4_44.

Proceedings Published in Lecture notes in electrical engineering, pp. 547–561, Springer, Singapore, July. 2023.

2. **T. Sinha** and J. Bhaumik, "Performance analysis of CRC-aided Polar codes with SCL decoding algorithm for the Binary Deletion Channel," *Proc. of 2nd Int. Conf. on Communication, Devices and Computing (ICCDC 2019)*, Haldia Institute of Technology, West Bengal, India, March 14-15, 2019,doi: https://doi.org/10.1007/978-981-15-0829-5_2.

Proceedings Published in Lecture Notes in Electrical Engineering (LNEE), vol. 602, pp.13-23, Springer, Singapore, Jan. 2020.

Book Chapters:

1. **T. Sinha**, S. Nayek, and J. Bhaumik, "On relative performances and decoding of CRC concatenated Polar codes with different lists for Solid State Drives," *Lecture Notes in Electrical Engineering (LNEE)*, vol. 602, pp. 377-389, Springer, Singapore, Jan. 2020. doi: https://doi.org/10.1007/978-981-15-0829-5_37.

PROFORMA – 1

"Statement of Originality"

I, **Tirthadip Sinha** registered on **18th June, 2019** do hereby declare that this thesis entitled "***Error Correcting Codes for Communication, Security and Networks***" contains literature survey and original research work done by the undersigned candidate as part of Doctoral studies.

All information in this thesis have been obtained and presented in accordance with existing academic rules and ethical conduct. I declare that, as required by these rules and conduct, I have fully cited and referred all materials and results that are not original to this work.

I also declare that I have checked this thesis as per the "*Policy on Anti Plagiarism, Jadavpur University, 2019*", and the level of similarity as checked by iThenticate software is **5%.**

Tirthadip Sinha
Signature of Candidate:
Date: 01/12/2025

Jaydeb Bhaumik 01/12/2025
Certified by Supervisor:
(Signature with date, seal)

Dr. Jaydeb Bhaumik
Professor
Dept. of Electronics and Telecommunication Engineering
Jadavpur University
Kolkata - 700032

PROFORMA- 2

Certificate from the Supervisor

This is to certify that the thesis entitled "*Error Correcting Codes for Communication, Security and Networks*" submitted by Shri **Tirthadip Sinha**, who got his name registered on **18th June, 2019** for the award of Ph.D. (Engg.) degree of Jadavpur University is absolutely based upon his own work under the supervision of **Prof. Jaydeb Bhaumik, Professor, Department of ETCE, Jadavpur University**, and that neither his thesis nor any part of the thesis has been submitted for a degree/diploma or any other academic award anywhere before.

Jaydeb Bhaumik 01/12/2025

Prof. Jaydeb Bhaumik
Professor
Department of ETCE, Jadavpur University

Dr. Jaydeb Bhaumik
Professor
Dept. of Electronics and Telecommunication Engineering
Jadavpur University
Kolkata - 700032

Signature of the Supervisor and date with Office Seal

Acknowledgements

I would like to express my sincere gratitude and thanks to my supervisor, Prof. Jaydeb Bhaumik, Professor, Department of ETCE, Jadavpur University, Kolkata, West Bengal, India, for his invaluable advice, continuous support and encouragement during my research work. I appreciate all of his immense knowledge, contributions of time, suggestions, motivations and patience for my research.

I am also incredibly grateful to the members of Research Advisory Committee (RAC), Prof. Sudipta Chattopadhyay, Professor, Department of ETCE, Jadavpur University, Prof. Sudhabindu Ray, Professor and Head of the Department of ETCE, Jadavpur University, and the committee chair for their insightful comments and suggestions.

I would like to thank Prof. Manotosh Biswas, Prof. Ananda Sankar Chowdhury and Prof. Sheli Sinha Chowdhury, who were associated as Head of Electronics and Telecommunication Engineering Department at Jadavpur University in different phases of my PhD course, for supporting me with valuable advice and suggestions towards the progress of my research.

I am also thankful to Haldia Institute of Technology, Haldia, West Bengal, for providing me the computing facilities for carrying out my research works. For the editing assistance and emotional support, I am also appreciative of my colleagues at Department of ECE, Haldia Institute of Technology.

I am deeply indebted to my parents, Shri Harekrishna Sinha and Shrimati Anjali Sinha for their unconditional love, support, patience, tolerance and encouragement for my research. I wish to express my deepest sense of gratitude to my wife Mrs. Merry Sinha, my beloved daughter Oishani and beloved son Ishaan for their invaluable patience, endless support and encouragement, without which it would not be possible to conduct this research. I also gratefully acknowledge the contributions of my elder brother Partha and elder sister Shaswati, who have continuously motivated me even in hard times.

Finally, I would like to thank God, for letting me through all the difficulties. I have experienced your guidance day by day and blessings that enable me to complete the task. You are the one who let me finish my degree. I will keep on trusting you for my future.

Tirthadip Sinha

TIRTHADIP SINHA

Abstract

In this digital world, needs of data integrity and confidentiality as well as reliable transmissions are paramount. Error Correcting Codes (ECC) play a vital role in ensuring that transmitted information remains accurate, even in the presence of errors introduced by noise, interference, or malicious activities. In communication systems, ECC are used popularly for detecting and correcting errors. Beside this application, ECC are equally crucial for maintaining data integrity in Solid-State Drives (SSD), where corruption can have serious consequences. With the rise of cyber-security threats, ECC have found applications in physical layer security (PLS) techniques that help to protect data against unauthorized access and manipulation. Advanced ECCs such as Golay and polar codes have demonstrated significant capabilities, where Golay codes are excelling in correcting multiple errors and polar codes are enabling capacity-achieving performance for future communication systems. As communication technologies continue to evolve, the importance of efficient and robust error correction methods will only increase, making the study of these codes essential for future developments in the field.

This thesis explores the decoding techniques and performance of Cyclic Redundancy Check (CRC)-concatenated polar codes, focusing on their application in SSD and Binary Deletion Channels (BDC). It demonstrates that integrating CRC codes with polar coding significantly improves error recovery, even in deletion-prone environments, while maintaining computational efficiency. Through simulations, the research provides insights into optimizing CRC length and list size configurations, contributing to advancements in error correction for communication and data storage systems. This thesis also investigates the Genetic Algorithm (GA) based optimized polar code construction technique for communication systems over Additive White Gaussian Noise (AWGN) and Rayleigh Fading Channel (RFC). Proposed method enhances error-correction performance without relying on CRC. The results show that GA-optimized designs, using Successive Cancellation List (SCL) and Belief Propagation (BP) decoding, achieve lower Block Error Rates (BLER) at reduced Energy per Bit to Noise Power Spectral Density (E_b/N_o) ratios, improving efficiency and performance. Further, this thesis explores advanced coding techniques to improve security and reliability in wireless communication, including the development of efficient Golay encoders and decoders for the McEliece cryptosystem and Physical Layer Security (PLS) scheme based on polar coding method. The proposed McEliece system, leveraging extended Golay codes, addresses vulnerabilities in traditional cryptosystems against quantum threats. Also, the proposed polar coding scheme uses artificial noise to enhance secrecy at physical layer by weakening eavesdropper channels. These contributions provide some solutions for secure and reliable data transmission in wireless networks. Lastly, this thesis examines the optimization of polar codes for 5G New Radio (NR) control channels and Wireless Sensor Networks (WSNs) in the Internet of Things (IoT) applications, focusing on enhancing error correction capability by applying concatenated polar codes with interleaving blindly (I_B). This work highlights the performance improvements of Distributed CRC Aided (DCA-polar) and CRC Aided (CA-polar) polar codes, particularly for short block lengths and low Signal-to-Noise Ratio (SNR) conditions. The results also demonstrate that concatenated BCH-Polar codes with I_B outperform traditional methods in terms of Bit Error Rate (BER) and computational complexity. All these findings provide a robust framework for reliable and secure communication in modern wireless systems showcasing the potential of these techniques to meet the stringent demands of 5G and beyond.

Contents

List of Figures

List of Tables

List of Abbreviations

Abbreviation	Description
3GPP	Third Generation Partnership Project
5G NR	5G New Radio
AGC	Algebraic Geometric Code
AN	Artificial Noise
AWGN	Additive White Gaussian Noise
BD	Bounded Distance
BF	Beamforming
BP	Belief Propagation
BS	Base Station
BCH	Bose-Chaudhuri-Hocquenghem
BDC	Binary Deletion Channel
BDMC	Binary-input Discrete Memoryless Channel
BDMS	Binary-input Discrete Memoryless Symmetric
BEC	Binary Erasure Channel
BER	Bit Error Rate
BI-AWGNC	Binary-input Additive White Gaussian Noise Channel
BLER	Block Error Rate
BPL	Belief Propagation List
BSC	Binary Symmetric Channel
BSMC	Binary-input Symmetric Memoryless Channel
CA-BPL	CRC-aided BPL
CA-polar	CRC-Aided polar code
CA-SCL	CRC-Aided Successive Cancellation List

CBS	Code Block Segmentation
CD	Compact Disc
CDMA	Code Division Multiple Access
CRC	Cyclic Redundancy Check
CSI	Channel State Information
DCA-polar	Distributed CRC-Aided Polar Codes
DCI	Downlink Control Information
DVD	Digital Versatile Disc
DVB-S2	Digital Video Broadcasting Satellite Second Generation
DSA	Digital Signature Algorithm
ECC	Error Correcting Codes
eMBB	Enhanced Mobile Broadband
FAR	False Alarm Rate
FEC	Forward Error Correction
FPGA	Field Programmable Gate Array
GA	Genetic Algorithm
HDD	Hard Disk Drive
IB	Blind Interleaving
IoT	Internet of Things
IR	Random Interleaving
LCPC	Low Complexity Parity Check
LDPC	Low Density Parity Check
LFSR	Linear Feedback Shift Register
LLR	Log-Likelihood Ratio
LTE-MIMO	Long Term Evolution – Multiple Input Multiple Output
LUT	Look Up Table
MISO	Multiple Input Single Output

ML	Maximum-Likelihood
MLC	Multi-Level Cell
mMTC	massive Machine Type Communication
MS	Mobile hand-Set
NTRU	N-th degree Truncated polynomial Ring Units
OFDM	Orthogonal Frequency Division Multiplexing
PCCC	Parallel Concatenated Convolutional Codes
PBCH	Physical Broadcast Channel
PDCCH	Physical Downlink Control Channel
PDSCH	Physical Downlink Shared Channel
PLS	Physical Layer Security
PUCCH	Physical Uplink Control Channel
PUSCH	Physical Uplink Shared Channel
QoS	Quality of Service
QPSK	Quadrature Phase Shift Keying
RC-GenAlg	Genetic algorithm with Redefined Crossover
RFC	Rayleigh Fading Channel
RM	Reed-Muller
RS	Reed Solomon
RSA	Rivest-Shamir-Adleman
RSC	Recursive Systematic Convolutional
SC	Successive Cancellation
SCL	Successive Cancellation List
SNR	Signal to Noise Ratio
SSD	Solid State Drives
TBCC	Tail-Bit Convolutional Codes
TCP/IP	Transmission Control Protocol/Internet Protocol

UAV	Unmanned Aerial Vehicles
UCI	Uplink Control Information
UEP	Undetected Error Probability
URLLC	Ultra-Reliable Low-Latency Communication
VLC	Visible Light Communication
WiMAX	Worldwide Interoperability for Microwave Access
Wi-Fi	Wireless Fidelity
WLAN	Wireless Local Area Network
WSN	Wireless Sensor Network

Chapter 1

Introduction

1.1. Background

Error Correcting Codes (ECC) has become a cornerstone in the fields of communication and storage systems. In a world where data integrity, confidentiality, and reliable transmission are paramount, ECC play a vital role in ensuring that transmitted information remains accurate even in the presence of errors introduced by noise, interference, or malicious activities. The origins of ECC can be traced back to Claude Shannon's pioneering work on information theory, which laid the theoretical foundation for coding techniques that enhance data reliability in noisy environments.

In modern communication systems, data is often transmitted over channels that are susceptible to various impairments such as noise, fading, and interference. These impairments can lead to errors, resulting in the corruption of transmitted information. ECC provide a systematic way of detecting and correcting these errors without the need for retransmission, making them indispensable in a wide range of applications, from deep-space communication, data storage systems to wireless communication networks and secure data transmission protocols.

The role of ECC extends beyond traditional communication systems; it is equally crucial in ensuring data integrity in storage devices, such as hard drives and solid-state drives, where data corruption can have severe consequences. Furthermore, with the rise of cyber-security threats, ECC have found applications in cryptographic and physical layer security protocols, offering enhanced security features that help to protect data against unauthorized access and manipulation.

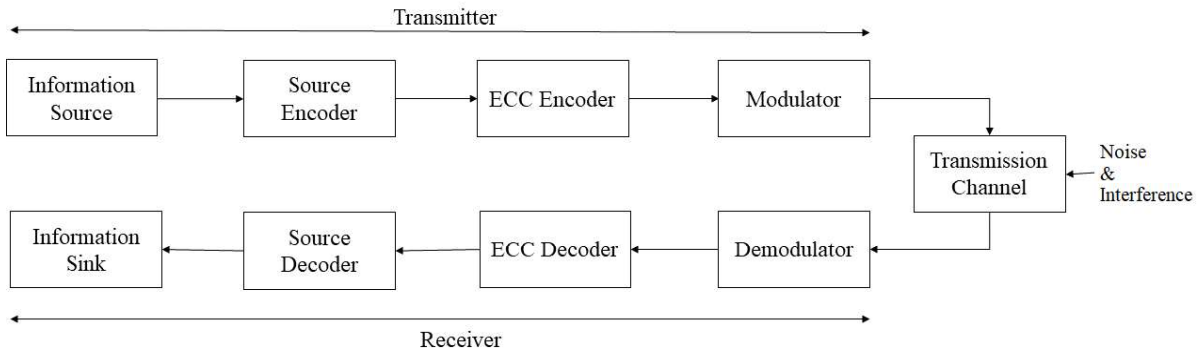


Figure 1.1 Block diagram of a typical communication system.

1.2. Utilization of Error Correcting Codes in Communication Systems, Security and Networks

Error Correcting Codes in Communication Systems: Communication systems rely on the accurate and efficient transmission of data between source and destination. However, communication channels are inherently imperfect, introducing errors during transmission due to noise, interference and fading. ECC are essential in mitigating these errors, ensuring that the received data matches the transmitted data as closely as possible. Several types of ECC, such as block codes, convolutional codes, Goppa codes, turbo codes, Golay codes, polar codes, and Low-Density Parity-Check (LDPC) codes, have been developed to address different communication scenarios. Each of these codes offers unique advantages in terms of error detection and correction capabilities, coding efficiency, and computational complexity. For instance, turbo codes and LDPC codes have gained prominence in recent years due to their near-capacity performance, making them ideal for high-throughput and low-latency communication systems such as 4G/5G mobile networks, satellite communications, and data centers. These codes utilize iterative decoding techniques that leverage soft information to achieve exceptional error correction performance, bringing them close to the theoretical limits defined by Shannon.

Error Correcting Codes in Security: In the realm of security, ECC also contribute to the protection of sensitive data from unauthorized access and tampering. ECC are integral to many security schemes, where they serve not only to detect and correct transmission errors but also to enhance security measures against active and passive attacks. One notable application of ECC is for Physical Layer Security (PLS) techniques, which exploit the inherent randomness of communication channels to secure data against eavesdropping. By integrating ECC into PLS, it is possible to simultaneously achieve reliable communication and security, making unauthorized interception exceedingly difficult. ECC also find use in public-key cryptography, where they help to ensure the robustness of encryption and decryption processes, particularly in noisy or lossy environments. For example, RS codes and other algebraic codes are frequently employed in digital signatures, secure voting systems, and data authentication, enhancing the reliability and security of cryptographic operations. Goppa code is a type of ECC which is used in McEliece cryptosystem. Also, ECC like RS codes find application in secret sharing schemes.

Error Correcting Codes in Networks: Networks, both wired and wireless are the backbone of modern digital communication, connecting billions of devices worldwide. In such a vast and dynamic environment, maintaining data integrity across network nodes is a significant challenge. ECC play a critical role along with network protocols by ensuring that data packets transmitted across multiple hops retain their integrity and accuracy. Network coding, a novel approach that combines the principles of error correction with data transmission, has emerged as a powerful tool for enhancing network performance. By encoding data packets at intermediate nodes, network coding allows for error correction, increased throughput, and improved resilience against packet loss. This technique is particularly beneficial in wireless sensor networks, ad hoc networks, and distributed storage systems, where conventional error correction methods may fall short. ECC can be used alongside network protocols, including those in the TCP/IP stack, to ensure reliable data delivery over unreliable links. In high-speed networks, ECC help to reduce the need for retransmissions, thereby enhancing overall network efficiency and user experience.

1.3 Literature Survey on related research works

The design and implementation of different ECC for communication systems, physical layer security, and wireless networks have been explored extensively in recent research. Golay codes, known for multiple error-correcting capabilities, and polar codes, recognized for achieving Shannon's capacity, are both pivotal in enhancing communication systems' performance and security. This literature survey summarizes several findings obtained from various studies to provide insights into their applications and benefits.

The new era of communication systems has been started with the remarkable invention of Claude E. Shannon [1] in the year 1948. Consequently, various ECC schemes [2-11] have been developed for enhancing the reliability of communication systems. The ECC have been applied in modern wireless communication techniques to increase the reliability of data transfer and reduce energy consumption. One of the leading families of ECC for wireless communication systems is known as LDPC codes. However, the challenges in implementation of LDPC codes are: higher encoding and decoding complexities, long latency and the number of iterations. A novel linear error detection and correction approach for single bit and multiple bits ECC called "Low Complexity Parity Check (LCPC)" codes have been presented in [12]. In [13], RS codes have been used for down link LTE system over LTE-MIMO channel. Short block length codes for ultra-reliable low latency wireless communication have been explored in [14]. Also both convolutional and turbo codes are employed for error correction in wireless communication [15-18]. In case of satellite communication, error detection is normally performed through CRC codes and error correction is usually performed through linear codes [19-20]. One of the commonly employed codes for error correction is RS code [21] which is a linear cyclic non-binary block code. The LDPC codes [22-23] are also employed in satellite communication. The combined version of CRC-LDPC and CRC-RS codes [24-25] are also applicable for satellite communication. Various error control coding techniques are used in digital and personal communication standards. In GSM, CRC Codes are used for error detection, block and convolutional codes are applied for error correction [26]. Convolutional codes have been employed in the second generation CDMA systems [27-29]. In CDMA2000, convolutional codes and turbo codes [30-32] are used for error correction. The LDPC codes are the standard ECC for many wireless communication protocols such as WiMAX (802.16e) and WLAN (802.11) [33-34]. Convolution codes and RS codes are mostly used for space communication systems [35-36].

The polar Codes [37-38] are the near Shannon's capacity approaching codes which have been selected for the control channels of Third Generation Partnership Project (3GPP) [39-41]. Polar codes have been adapted for different scenarios, including wiretap channels, multipath fading, and energy-efficient transmission, showcasing their broad applicability in modern communication systems. Moreover, the concatenation of two or more ECC schemes have been widely used in different communication systems for obtaining better error performances. Various concatenation schemes of polar codes as inner code and RS [42-43], BCH [44], LDPC [45] codes as outer codes have been introduced for the application in communication systems. The concatenation of RS and convolutional codes [46] has been applied in the IEEE802.11b standard for better noise performance. Also various kind of concatenated codes [47-48] are available in the literature for space and satellite communication applications. The performances of CRC-polar concatenated code have been analysed in depth for achieving better error performance [49]. Further, concatenation of polar codes with hash codes [50] and CRC-polar codes [51] have been analysed for 5G applications. Polar codes in Rayleigh Fading Channel (RFC) can be transformed into an

Additive White Gaussian Noise (AWGN) model using channel transformation techniques, enabling effective decoding and significant error rate improvements [52]. An integrated polar coding scheme enhances joint energy and information transfer, optimizing bit allocation based on conditional entropy, leading to notable performance gains over traditional methods [53]. Recent advancements have introduced various techniques to optimize polar codes to trade-off between theoretical capacity and practical performance. Polar codes using GA is a promising area of research that employs optimization techniques in coding theory. This approach leverages the adaptive nature of GAs to enhance the performance of polar codes, particularly in challenging environments such as those with imperfect channel estimation. GA is employed to optimize polar code construction for systems like orthogonal frequency division multiplexing (OFDM) where channel estimation errors are prevalent [54]. The GA approach is robust against variations in channel estimation quality, making it suitable for real-world applications where channel conditions can fluctuate. The design and implementation of polar codes for security purposes have gained significant attention due to their potential in enhancing communication systems, particularly in the context of 5G and beyond 5G. Polar codes not only improve error correction capabilities but also provide robust security frameworks. Polar codes provide a framework for secure communication by ensuring reliability and information-theoretic secrecy against eavesdroppers, making them suitable for modern communication systems. Polar codes can be utilized to generate secret keys and communicate confidential messages securely, even in the presence of eavesdroppers [55]. They achieve a trade-off between secret-key and secret message rates, ensuring strong secrecy through a combination of encoding schemes and one-time pad encryption. Also, the inherent properties of polar codes allow them to provide information-theoretic security, which does not rely on limited computational power assumptions about adversaries. They can be effectively applied in wiretap channels, ensuring that the transmitted information remains statistically independent from what an eavesdropper can observe. They achieve optimal performance in degraded broadcast channels, where legitimate receivers can be prioritized based on channel quality, allowing for tailored reliability and secrecy constraints [56]. While polar codes show great promise in enhancing security and efficiency in communication systems, challenges remain in optimizing their performance under highly dynamic environments and ensuring their practical deployment in diverse applications, necessitating ongoing research to refine their adaptability and efficiency.

Golay codes, known for their robustness, are increasingly integrated into modern communication frameworks, including LTE and emerging 5G/6G systems. While Golay codes offer significant advantages in error correction and power efficiency, their implementation can be complex and may require careful design considerations to optimize performance in various communication scenarios. A novel reinterpretation of Golay codes with additional permutations and puncturing has been suggested in [57], allowing for polar code type decoding and potentially extending this approach to other algebraic codes. Golay codes have been proposed [58] as a superior alternative to Reed-Muller (RM) codes for encoding Uplink Control Information (UCI) in 4G LTE and 5G NR systems in terms of performance and complexity, offering approximately 0.5 dB better performance for 24-bit codeword lengths. A novel Golay encoder architecture has been developed that utilizes clock gating techniques to reduce power consumption by 64.6%, while maintaining system delay and area efficiency [59]. This architecture is particularly beneficial for battery operated devices in wireless communication, where power efficiency is critical. Golay codes, particularly in the context of cryptography and physical layer security, provide robust error correction capabilities while ensuring data integrity and confidentiality highlighting their versatility and effectiveness in enhancing secure communications. Golay codes are also

utilized in secret sharing schemes, providing a robust framework for distributing secret information securely among multiple parties [60]. These schemes leverage the combinatorial properties of Golay codes to establish minimal access structures, enhancing the security of shared secrets.

While Golay and polar codes offer significant advantages in communication systems and security, challenges remain in optimizing their implementation for diverse network conditions and evolving security threats. Further research could explore hybrid approaches and adaptive coding strategies to enhance their robustness and efficiency in practical applications.

1.3. Motivation of the research work

The ECC are employed in almost all kinds of modern communication systems and storage systems as well as in networks. The design of suitable encoder-decoder is very much essential and a challenging area of research. But, design complexity and overheads (computation time, area, delay, power) increase with the increase of error detection and correction capabilities. Also, soft errors rate increases with the technology scaling for storage systems . Hence, modification of existing encoding-decoding schemes and new proposal for efficient ECC schemes are necessary to solve the above mentioned problems for various modern communication systems to ensure reliability and also to employ them for designing security schemes.

1.4. Objectives of the research work

This thesis aims to explore the design, analysis, and application of ECC in the domains of communication, security, and networks. The objectives of this research are summarized as follows.

(i) Performance evaluation of polar codes over different channel models.

(ii) Construction of efficient polar codes using Genetic Algorithm for communication systems.

(iii) Adaptation and implementation of concatenated codes in WSNs and 5G control channels.

(iv) Design of security algorithms based on error correcting codes for networks.

1.6. Organization of the Thesis

An overview, a literature survey of related works, and a list of significant contributions are included in the *chapter1*. The review identifies key research gaps, particularly in the application of ECC for Physical Layer Security (PLS) and resource-constrained environments, motivating the need for novel coding schemes that balance performance, complexity, and security. It highlights the role of ECC in enhancing communication systems' robustness and security. The rest of this thesis is organized as follows:

Chapter 2 provides the fundamentals of some ECC (more emphasis on Golay and polar codes), outlining their basics and design principles. It outlines their structural advantages, such as capacity-achieving properties and efficient decoding algorithms, making them suitable for modern communication applications. Also, an extensive survey on the evolution of ECC, from classical codes like Hamming and RS to modern techniques such as turbo, LDPC and polar codes, is presented. Also, chapter 2 provides an overview of different channel models which are considered in this thesis.

Chapter 3 presents the performance evaluation of polar codes over different channel models covering their design principles and implementation details. Different polar codes have been introduced in the first contributory chapter i.e. *chapter 3* and their performances are evaluated in two different channel models such as Binary Deletion Channel (BDC) and Solid State Drive (SSD) channel. The analysis has demonstrated the robustness of polar codes in handling burst errors and deleted bits. Outcomes show a compromising relationship between the BLER performance and CRC codelength when the polar code rate is same for the BDC. Also, the proposed CRC concatenated polar codes with SCL decoding is shown effective to improve error correction ability in SSD channel.

Chapter 4 offers the construction of efficient polar codes using Genetic Algorithm (GA), including simulation results, performance comparisons, and computational complexity. GA based polar code constructions have been introduced for Additive White Gaussian Noise (AWGN) and Rayleigh Fading channels (RFC) in this chapter. The performance comparisons are presented along with analysis of computational complexity. The proposed GA-based construction method significantly improves polar codes' error correction performance by dynamically selecting the most reliable bit channels. The adaptive nature of the GA allowed for real-time optimization, enhancing polar codes' suitability for varying channel conditions. GA is employed along with polar code construction to minimize the gap between traditional Successive Cancellation List (SCL) and iterative Belief Propagation (BP) decoding performances, while also reducing computation time. Thus, GA based optimization enables the attainment of a target error rate with lesser BP iterations or a reduced list size.

Chapter 5 demonstrates the design of different concatenated polar coding schemes for WSNs and 5G NR control channels, examining their impact on performance. The research demonstrates that polar codes could be optimized through simplified decoding algorithms, concatenated coding with interleaving, and dynamic resource allocation. The practical applicability of these optimizations is validated through case studies, including short block-length polar codes for 5G NR control channels and enhanced error correction strategies for WSNs. The proposed DCA-polar coding emerges as a flexible design capable of supporting various decoding paradigms to accommodate diverse requirements of 5G NR control channels and provides significant error correction performance using short block lengths as well as permits early termination during decoding. Also, the proposed concatenated polar code with interleaving is the best choice in WSNs based IoT system considering performance and decoding complexity.

Chapter 6 discusses the design of ECC based security schemes, highlighting their performance and security benefits. Key contributions include a modified architecture of Golay encoder and decoder for the proposed McEliece cryptosystem and the design of Artificial Noise (AN) -aided secure polar coding for wireless networks. These schemes demonstrate how ECC could provide an additional security layer, enhancing the robustness of secure communication protocols. For secure high-speed communications, the proposed Golay encoder and decoder is a promising option in the McEliece cryptosystem, where high throughput, minimal latency and low area are achieved for both hardware modules. Also, the proposed AN-aided polar codes are successfully employed for higher BLER at the eavesdropper to enhance security and reliability, which is highly desirable as PLS solutions.

Chapter 7 finally concludes the thesis with key contributions, and potential directions for future research.

1.7. Summary

ECC are indispensable tools that enhance the reliability, security, and efficiency of modern communication systems. As the demand for high-speed, secure, and robust data transmission continues to grow, the development of advanced ECC techniques will remain a critical area of research. This thesis seeks to contribute to this evolving field by proposing novel coding schemes and exploring their applications in communication, security, and networks, paving the way for more reliable and secure data communication and storage systems.

Chapter 2

Fundamentals of some Error Correcting Codes and Channel Models

2.1. Introduction

In modern communication systems, ECC are indispensable for ensuring data integrity. In digital communication, data is often transmitted over channels that introduce errors due to noise, interference, or other factors. ECC enable the detection and correction of errors that occur during data transmission and storage. This ensures the reliability of data transmission, even in adverse conditions like high interference or limited signal-to-noise ratios. The effectiveness of an ECC is measured by its ability to minimize the bit error rate (BER) or block error rate (BLER) while maintaining a manageable level of complexity in encoding and decoding processes. As digital communication technologies continue to evolve, the demand for efficient and robust error correction methods has become increasingly critical.

This chapter begins with importance and overview of existing ECC techniques, followed by detailed sections on the fundamentals of Golay and polar codes, including their construction, properties, and applications. Next, the different channel models are discussed in brief. The chapter concludes with a summary and scopes of research.

2.1.1 Overview of Error Correcting Codes

The origins of ECC date back to the 1940s and 1950s, coinciding with the emergence of information theory. Claude Shannon's seminal work in 1948, titled "A Mathematical Theory of Communication," laid the groundwork for the development of coding theory by introducing the concept of channel capacity and demonstrating that reliable communication is possible over noisy channels using appropriate coding schemes. This breakthrough led to the search for practical codes that could approach the theoretical limits set by Shannon. Over the years, numerous coding techniques have evolved, each catering to specific requirements such as bandwidth efficiency, low latency, and high reliability.

ECC techniques can be broadly categorized into two types: block codes and convolutional codes. Block codes, such as Hamming and BCH codes, divide data into fixed-size blocks, whereas convolutional codes introduce memory into encoding, processing continuous streams of data. Advanced codes like turbo and LDPC codes further enhance error correction by leveraging iterative decoding and sparse matrix structures, respectively. Despite their effectiveness, traditional ECC techniques face challenges such as high computational complexity, suboptimal performance, and scalability issues. These limitations drive the need for modification of existing ECC and design of new ECC.

2.1.2 Basics and Historical Development of ECC techniques

Basics of different ECC techniques and related historical development are discussed in this sub-section.

1) Block Codes: These codes operate on fixed-size blocks of data. Each block is encoded into a longer codeword, which can be transmitted over the channel. These are a fundamental class of error-correcting codes used in digital communication and storage systems. Their primary purpose is to detect and correct errors that occur during the transmission or storage of data. By adding redundancy to the original message, block codes enable reliable communication even in the presence of noise or interference. Block codes are characterized by their generator and parity-check matrices.

*(i) Linear Block Codes***:** These block codes are characterized by their linearity, meaning that the sum of any two codewords is also a codeword.

(ii) Cyclic Codes: A subclass of linear block codes, cyclic codes have the property that if a codeword is cyclically shifted, the result is still a codeword. This property simplifies encoding and decoding processes. It is often used in applications like CRC.

(iii) Perfect Codes: These codes achieve the maximum possible error correction capability for a given block length.

The most notable types of block codes include:

Hamming Codes (1950): In the early 1950s, Richard Hamming introduced one of the first practical ECC, known as Hamming codes, which were designed with minimum Hamming distance of 3 to correct single-bit errors or detect double-bit errors. These codes were particularly notable for their simplicity and ease of implementation, marking the beginning of systematic error correction in digital communication.

Bose-Chaudhuri-Hocquenghem (BCH) Codes (1960): These codes are capable of correcting multiple errors by extending the principles of Hamming codes and are widely used in digital communication and data storage systems.BCH codes were introduced in 1959 by Hocquenghem and independently in 1960 by Bose and Chaudhuri. These codes enable efficient detection and correction of multiple errors. BCH codes are constructed over finite fields and are highly versatile, allowing multiple error corrections.

Reed-Solomon (RS) Codes (1960): These codes are defined over finite fields and are widely used in storage devices like CDs and DVDs as well as data transmission. In the 1960s, Irving S. Reed and Gustave Solomon developed Reed-Solomon codes, which are based on polynomial arithmetic over finite fields and became widely recognized for their robustness in correcting burst errors. RS codes gained prominence due to their use in deep-space communication, notably in the Voyager missions.

Golay Codes (1949): These are a specific type of block code known for their ability to correct multiple errors. Golay codes were introduced in 1949 by Marcel J. E. Golay in his paper on error-correcting codes. They are binary and ternary linear codes designed for efficient transmission of data in noisy communication channels. The most well-known Golay codes are the (23,12,7) binary Golay code, which encodes 12 data bits into 23 bits and can correct up to 3 errors, and the (24,12,8) extended binary Golay code [61]**,** which adds a parity bit for better error detection.

2) Convolutional Codes: These codes are a class of error-correcting codes used extensively in digital communication systems to improve the reliability of data transmission. Unlike block codes, which encode data in fixed-length blocks, convolutional codes process data as a continuous stream, making them highly suitable for real-time applications. These codes generate codewords by convolving the input data with a set of generator sequences. Viterbi Algorithm is commonly used for decoding convolutional codes, providing optimal decoding

performance. Convolutional codes have memory, meaning that the encoding of a bit depends on previous bits. They are often used in applications requiring real-time processing, such as mobile networks and satellite communications.

3) Advanced Codes: Advanced coding techniques like turbo codes, LDPC codes, and polar codes have revolutionized modern communication systems, enabling reliable data transmission under challenging conditions. Each of these coding strategies offers unique advantages, making them indispensable for various applications. As communication demands continue to evolve, these codes will remain central to the development of efficient and robust systems.

Turbo Codes (1993): These codes are a class of high-performance ECC that combine two parallel concatenated convolutional codes (PCCCs) using an interleaver. An iterative decoding process is used to achieve near-capacity performance close to Shannon's limit. The invention of turbo codes by Claude Berrou, Alain Glavieux, and Punya Thitimajshima in 1993 marked a significant milestone in coding theory and sparked renewed interest in ECC. Turbo decoding relies on an iterative process using two soft-input soft-output (SISO) decoders, which exchange extrinsic information to refine their estimates iteratively. Turbo codes provide excellent error correction capabilities, especially at low signal-to-noise ratio (SNR). Their performance depends on the design of the interleaver and the number of decoding iterations. These codes found applications in 3G/4G mobile communication, deep-space communication, and other high-demand scenarios.

Low-Density Parity-Check (LDPC) Codes (1990): Although originally proposed by Robert G. Gallager in 1962, these codes were rediscovered in the 1990s and quickly became a leading coding scheme due to their exceptional error correction performance and low complexity. LDPC codes use sparse bipartite graphs for encoding and decoding, offering near-optimal performance and scalability for long block lengths. LDPC codes are characterized by their sparse parity-check matrix, which defines the relationships between codeword bits. The matrix is typically represented using a Tanner graph, a bipartite graph with v*ariable nodes,* representing the codeword bits and c*heck nodes,* representing parity-check equations. LDPC codes are decoded using iterative algorithms, such as the belief propagation (BP) algorithm or its simplified versions. These algorithms use the graphical representation to pass messages between variable and check nodes, updating probabilities iteratively until convergence. LDPC codes exhibit excellent error correction capabilities and approach the Shannon limit in performance. They are computationally efficient for large block lengths and are widely adopted in standards like Wi-Fi (IEEE 802.11n/ac/ax), DVB-S2, and 5G NR.

Polar Codes (2008): These codes, introduced by Erdal Arikan in 2008, are a relatively new class of codes that achieve the capacity of binary-input symmetric memoryless channels (BSMCs). They are distinguished by their unique construction and decoding algorithms. Polar codes are based on the concept of channel polarization, where a set of channels is transformed into polarized channels with either high reliability or low reliability. Polar codes use a structured generator matrix derived from the Kronecker power of a base matrix. Low-reliability channels are assigned frozen bits (fixed, known values), while high-reliability channels carry information bits. The encoding process involves multiplying the input bit vector with the generator matrix. Due to the structured nature of the matrix, the encoding can be efficiently implemented using fast algorithms. Polar codes are decoded using the successive cancellation (SC) decoding algorithm, which sequentially estimates the bits. Variants such as SC list (SCL) decoding and CRC-aided SCL decoding enhance performance. Polar codes are highly efficient for long block lengths and perform close to the Shannon limit. They are particularly

effective in scenarios where low latency is required, making them suitable for 5G ultra-reliable low-latency communication (URLLC).

2.2 Basics of Golay and polar Codes

Among the various coding techniques, Golay codes and polar codes have emerged as significant players due to their unique properties and applications. This chapter aims to provide a comprehensive overview of these two coding schemes, exploring their fundamentals, classifications, and practical implications in error correction. Golay codes are renowned for their efficiency in correcting errors in challenging environments, including deep-space communication and data storage. Their compact size and robustness make them a classic example of efficient linear block codes. Polar codes, on the other hand, represent a breakthrough in modern ECC, achieving Shannon capacity under successive cancellation (SC) decoding. They are characterized by their simple encoding and decoding algorithms, making them suitable for high-performance communication systems, including 5G networks. Understanding these codes provides a dual perspective: the legacy of Golay codes in classical ECC and the innovation of polar codes in modern applications.

2.2.1 Golay Codes

Golay codes [62], which were developed in 1949, have seen extraordinary growth in recent years and implemented as a linear error-correcting code in communication links. Bit interleaving technology can be used with Golay codes to fix burst errors. Golay codes are linear block codes that can correct multiple errors in a codeword. Ternary and binary Golay codes are two different types of Golay codes. Golay codes were famously used in NASA's Voyager missions for transmitting data over vast distances. Their ability to correct multiple errors with minimal redundancy made them ideal for deep-space communication.

Properties of Golay Codes:

(i) Perfect Code Property: Every received codeword falls exactly into one decoding sphere without overlap or gap.

(ii) Cyclic Nature: Codewords can be cyclically permuted, simplifying implementation.

(iii) Error Correction Capability: Golay codes can correct multiple errors, making them suitable for applications where data integrity is critical. They are high resilience in noisy environments, especially in space communication.

(iv) Simplicity of Implementation: The design approach for Golay codes are relatively simple and straightforward, allowing for efficient implementation both in hardware and software.

The binary Golay codes are classified into perfect binary and extended versions such as:

*(i) Perfect Golay Code (23, 12, 7)***:** This code can correct up to three errors or detect up to six errors in a 23-bit codeword with 12-bit information length. It is constructed using a generator matrix and has a minimum Hamming distance of 7.

*(ii) Extended Golay Code (24, 12, 8)***:** This code is an extension of the (23, 12, 7) Golay code, adding a parity bit to create a 24-bit codeword for enhanced error correction. It can correct up to 3 errors or detect up to 7 errors. The parity bit is zero if binary Golay codeword has even weight, otherwise parity bit is 1.

Construction of Golay Codes:

The construction of Golay codes involves the generator matrix G, which transforms information bits into codewords. One of the following generator polynomials produces the perfect binary Golay code and they are $g_1(X) = 1+X^2+X^4+X^5+X^6+X^{10}+X^{11}$ and $g_2(X) = 1+X+X^5+X^6+X^7+X^9+X^{11}$. Thus, the perfect binary Golay code can be produced using a 12×23 generator matrix $G_{23} = [I, B_{23}]$, where I is 12×12 identity matrix and B_{23} is 12×11 matrix. For the (23, 12, 7) Golay code, the generator matrix is designed to maximize the Hamming distance between codewords.

$$B_{23} = \begin{bmatrix} 0&1&1&1&1&1&1&1&1&1&1 \\ 1&1&1&0&1&1&1&0&0&0&1 \\ 1&1&0&1&1&1&0&0&0&1&0 \\ 1&0&1&1&1&0&0&0&1&0&1 \\ 1&1&1&1&0&0&0&1&0&1&1 \\ 1&1&1&0&0&0&1&0&1&1&0 \\ 1&1&0&0&0&1&0&1&1&0&1 \\ 1&0&0&0&1&0&1&1&0&1&1 \\ 1&0&0&1&0&1&1&0&1&1&1 \\ 1&0&1&0&1&1&0&1&1&1&0 \\ 1&1&0&1&1&0&1&1&1&0&0 \\ 1&0&1&1&0&1&1&1&0&0&0 \end{bmatrix} \quad (2.1)$$

To generate extended Golay code (24,12,8), an extra parity bit is added to the binary Golay code (23,12,7).The 12 x 24 generator matrix $G_{24} = [I, B_{24}]$ produces the extended Golay code, where I is 12×12 identity matrix and B_{24} is 12×12 matrix.

$$B_{24} = \begin{bmatrix} 1&1&0&1&1&1&0&0&0&1&0&1 \\ 1&0&1&1&1&0&0&0&1&0&1&1 \\ 0&1&1&1&0&0&0&1&0&1&1&1 \\ 1&1&1&0&0&0&1&0&1&1&0&1 \\ 1&1&0&0&0&1&0&1&1&0&1&1 \\ 1&0&0&0&1&0&1&1&0&1&1&1 \\ 0&0&0&1&0&1&1&0&1&1&1&1 \\ 0&0&1&0&1&1&0&1&1&1&0&1 \\ 0&1&0&1&1&0&1&1&1&0&0&1 \\ 1&0&1&1&0&1&1&1&0&0&0&1 \\ 0&1&1&0&1&1&1&0&0&0&1&1 \\ 1&1&1&1&1&1&1&1&1&1&1&0 \end{bmatrix} \quad (2.2)$$

There are many different encoding-decoding techniques available. Although [63-65] deal with many algorithms, but hardware implementation is difficult with these due to higher complexity. For hardware implementation, LFSR (linear feedback shift register)-based [66-69] approaches are suitable, although they are not acceptable due to high latency and less throughput. CRC-based hardware implementation is suggested in [70]. Several hardware architectures [71-75] have been developed based on various Golay code decoding methods.

2.2.2 Polar Codes

Polar codes are constructed based on the principle of channel polarization, which transforms a set of identical and independent channels into a set of polarized channels with varying levels of reliability. Here channels are splitted into “good” and “bad” subsets, transmitting data only through good channels. Polar codes are the first provable capacity-achieving codes for symmetric binary-input discrete memoryless channels (BDMCs). Polar codes have been adopted in 5G New

Radio (NR) standards, particularly for control channel communication. Their low latency and high throughput make them suitable for IoT and machine-to-machine communication.

Properties of polar Codes:

(i) Capacity Achieving: Polar codes are the first class of codes proven to achieve the capacity of binary-input memoryless channels.

(ii) Low Complexity: The encoding and decoding processes for polar codes are computationally efficient, making them suitable for high-speed communication systems.

The key components of polar codes include:

(i) Channel Polarization: Channel polarization is achieved through recursive splitting. This is a process of transforming a set of channels into polarized channels, where some channels become very reliable while others become unreliable. A binary transformation separates good and bad channels, quantified by the Bhattacharyya parameter. Repeated application of this transformation results in polarization.

(ii) Encoding: The encoding process for polar codes involves a simple linear transformation of the input data using a generator matrix derived from the polarization process.

(iii) Decoding: The decoding of polar codes is typically performed using the SC algorithm, which recursively estimates the transmitted bits based on the received codeword.

Construction of polar Codes:

In coding theory, popular channel codes have fixed structural design and architectures are not depending on the channel characteristics, but polar codes are distinct in this scenario. The concept of channel polarization is at the core of polar code theory. In this framework, we consider $N = 2^n$ duplicate copies of a channel to create N bit-channels. Certain bit-channels in polar codes become entirely noisy, while the remaining ones become noiseless.

The term "polarization" means that transformed channels are either perfect/noiseless i.e., $I(W) \xrightarrow{\text{yields}} 1$ or imperfect/noisy i.e., $I(W) \xrightarrow{\text{yields}} 0$, asymptotically, where mutual information I(W) of a BDMC is defined as:

$$I(W) \triangleq \sum_{y \in Y} \sum_{x \in X} W(y|x) \log \frac{W(y|x)}{\frac{1}{2}W(y|0)+\frac{1}{2}W(y|1)} \tag{2.3}$$

where input $X = \{0, 1\}$, output = Y and transmission probabilities = $W(y|x)$.

According to Shannon capacity calculation, it is well-known that the noiseless channels have lower error probabilities and higher capacities in comparison with noisy channels, thus theoretically it is always preferred to transmit the bits of information through a noiseless channel. For encoding in polar code initially, simple the selection of k information out of N indices are to be done where $N = 2^n$, the code length is N and the length of information is k [76]. For channel polarization exploitation, many algorithms are proposed previously but the simple one to use is the recursive one given as, $z \rightarrow \{2z - z^2, z^2\}$ as shown in Figure 2.1. Then a particular channel is chosen (e.g., AWGN, BEC, BSC). Though any channel can be taken into consideration only knowing the Bhattacharya parameter of the channel as for AWGN channel it is $z = e^{-Ec/No}$. Then two values are created using two different values and so on a chaining method is introduced until the tree has N leaves indexed from top 0,1, …., N-1. Then the k list values are picked from the leaves and their indices are stored in a set J and are shown as the output.

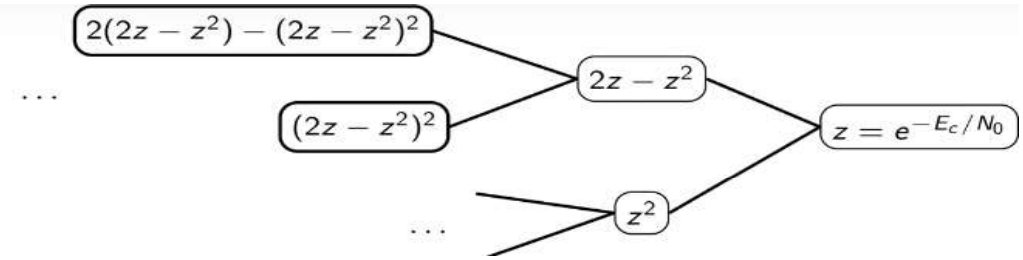


Figure 2.1 Recursive technique for generation of polarization channels.

The capacity of symmetric Binary-input Discrete Memoryless Channel (BDMC) is identical to the mutual information of channel input-output and reliable transmission is feasible at any rate up to I(W). Optimal polar code construction algorithm is hard and therefore many sub-optimal polar code constructions have been proposed at different computation complexities. The earliest polar code construction algorithm is from Arıkan using Bhattacharyya parameter which is defined as:

$$\mathrm{Z(W)} \triangleq \sum_{\mathrm{y}\in\mathrm{Y}} \sqrt{\mathrm{W(y|0)W(y|1)}} \tag{2.4}$$

Arıkan proved that Z(W) of Binary Erasure Channel (BEC) had a pair of upper limits as $\{\mathrm{z},\mathrm{z}\} \xrightarrow{\text{yields}} \{2\mathrm{z} - \mathrm{z}^2, \mathrm{z}^2\}$ on the probability of maximum-likelihood (ML) decision error at each polarizing transform as Z(W) is a measure of reliability of the channel.

The polar codewords are denoted as

$$x_1^N = u_1^N G_N \tag{2.5}$$

where u consists of k information bits and $(N-k)$ frozen bits, codelength $N = 2^n$ and n = 2,…..,10.

Therefore, the generator matrix

$$G_N = B_N F^{\otimes n} \tag{2.6}$$

where B_N represents the bit-reversal permutation matrix and $F^{\otimes n}$ represents the n-th Kronecker power of the matrix F.

Matrix F is defined as:

$$F = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \tag{2.7}$$

and $n = log_2 N$ (2.8)

$$\text{Thus, } G_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} \tag{2.9}$$

An example of polar code construction mechanism and generator matrix is shown in Figure 2.2 for N = 4.

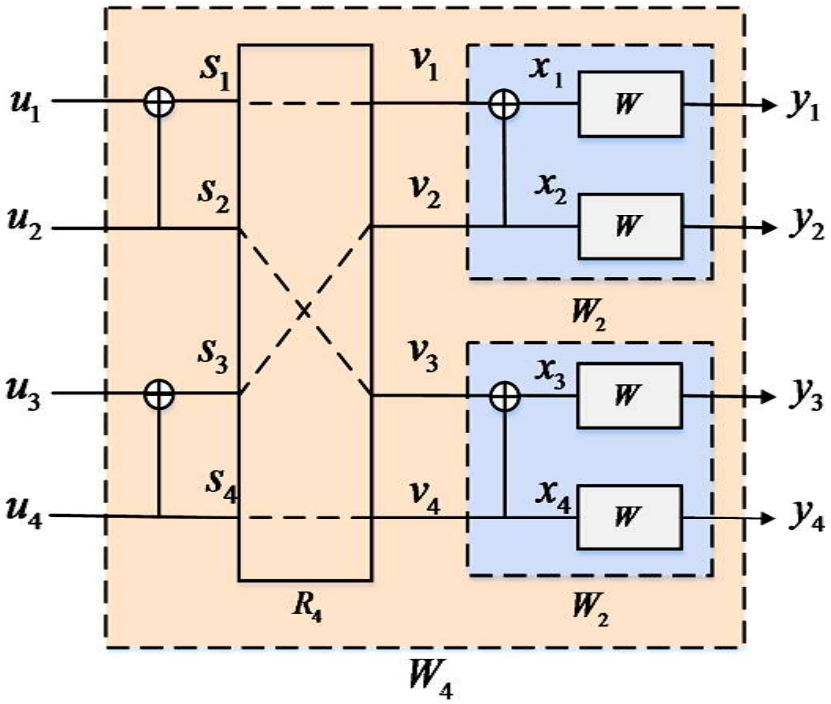


Figure 2.2 An example of polar code construction mechanism for N=4.

Thus, polar code P(N, k) has a code rate R=$\frac{k}{N}$. The encoding is reversed by polar decoding in the receiver. Polar list decoders have the decoding complexity as $\mathcal{O}(LN \log N)$, where the encoded block length and the list size are N and L, respectively. The code structure is constructed by controlling the more reliable k bit-channels for the information and selecting the more unreliable $(N-k)$ bit-channels for the frozen bits. If the code structure comprises some unreliable or weak bit-channels for information bits, then the error correction proficiency would be extremely worsened [77].

Decoding Techniques for polar Codes:

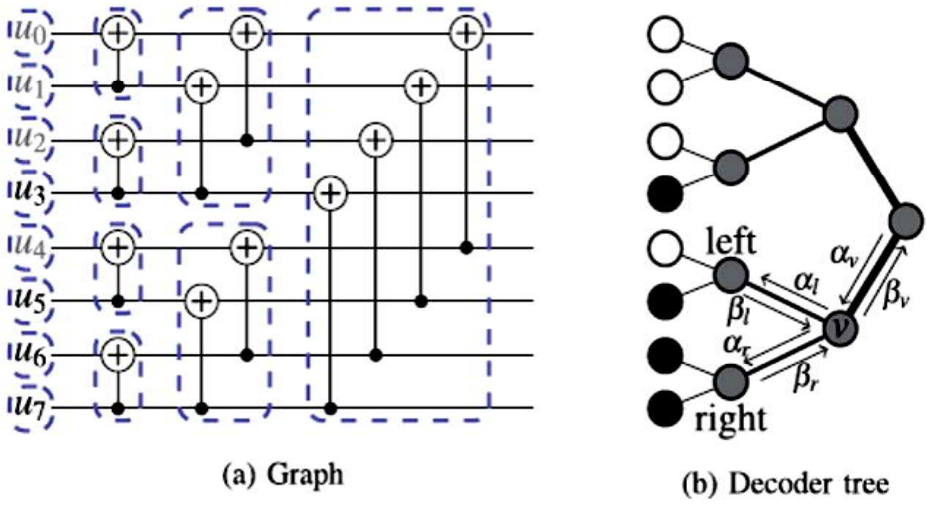
The salient features of several polar decoding systems are listed below:

(i*) SC decoding* [78]: In polar coding, the first decoding algorithm used is SC decoding. It successively hard-decides each of the k bits based on the previous estimation and the condition of the communication channel. However, errors at the decided bits are not corrected later, potentially impacting the following bit decisions. SC decoding scheme for polar codes is introduced by Arikan, considers the channel output as y_1^N and the i-th synthesized subchannel with input u_i and output $\left(y_1^N, u_1^{i-1}\right)$ by transition probability matrix $W_N^{(i)}$ for i = 1,…..*N*. For the given y_1^N and the estimates $\hat{u}_1^{i-1}$ of u_1^{i-1}, the SC decoding algorithm evaluates u_i. The logarithmic likelihood ratios (LLR) are applied for estimation of each u_i where i = 1, …..*N*.

$$L_N^{(i)}\left(y_1^N, \hat{u}_1^{i-1}\right) = \log \frac{W_N^{(i)}\left(y_1^N, \hat{u}_1^{i-1} | u_i=0\right)}{W_N^{(i)}\left(y_1^N, \hat{u}_1^{i-1} | u_i=1\right)} \tag{2.10}$$

The sign of the LLRs controls the estimation of an unfrozen bit u_i so that $\hat{u}_i$=0 if $L_N^{(i)}\left(y_1^N, \hat{u}_1^{i-1}\right) \geq 0$ and $\hat{u}_i$=1 otherwise. The complexity of SC decoder is $\mathcal{O}(N \log N)$.

An example of SC decoding graph and decoding tree are presented in Figure 2.3 (a) and (b) respectively, for N=8.



(a) Graph (b) Decoder tree

Figure 2.3 An example of SC decoding graph and decoding tree for N=8.

(ii) *SCL decoding* [79]: At every decision point, the SCL decoding process divides into two pathways (0 and 1), limiting the complexity by maintaining just a list or L likely paths that are consistent with a specific path metric. In SCL decoding which is basically an advanced version of SC decoding, extra paths are involved by including the different list sizes to increase error correction performance further. Usually, SCL is functioned as a breadth-first search algorithm and permits a maximum of L contestant paths compared to SC where only one path is retained after processing at each level. SCL doubles the quantity of contestants by assigning a bit (0 or 1) to each contestant paths. It also chooses a maximum of L ones with the major metrics saving in a list. In a *lazy copy* memory sharing arrangement, the SCL decoder can be applied with computational complexity of $\mathcal{O}(LN\log N)$ whereas direct implementation will need $\mathcal{O}(LN^2)$. The effective equation is given by

$$L_{i,j} = \begin{cases} 2tanh^{-1}\left[\tanh\left(\frac{L_{i+1,j}}{2}\right).\tanh\left(\frac{L_{i+1,j+2^{i-1}}}{2}\right)\right], \left\lfloor\frac{j-1}{2^{i-1}}\right\rfloor mod\ 2 = 0 \\ \left(1-2s_{i,j-2^{i-1}}\right)\left(L_{i+1,j-2^{i-1}}\right)+L_{i+1,j},\ otherwise \end{cases} \quad (2.11)$$

SCL decoder checks qualified paths related to the input bits in a tree-diagram and holds best L contenders in parallel where L signifies size of the list.

(iii) *CA-SCL decoding* [80]: Additional cyclic redundancy check (CRC) bits are added with SCL decoding to support choosing the ultimate codeword from the L remaining competitors. As a result, performance significantly improves. CRC precoding and SCL decoding are introduced in conventional polar codes to further improve the error correction ability. Here, the SCL decoder outputs the contestant paths into a CRC detector, and the checking outcomes are used to identify the accurate codeword. Adaptive CRC concatenated SCL or CRC-aided SCL (CA-SCL) decoding is suggested by gradually increasing the list sizes to lower the complexity of SCL decoding. The error correction ability is substantially enhanced and outpaces than normal decoding process depending upon CRC length and generator polynomial under CRC-aided decoding structures [81-82]. But, the redundancy linked with CRC codes turn into more prominent and thus decreases the overall competency of the concatenated polar code if the codelength is small.

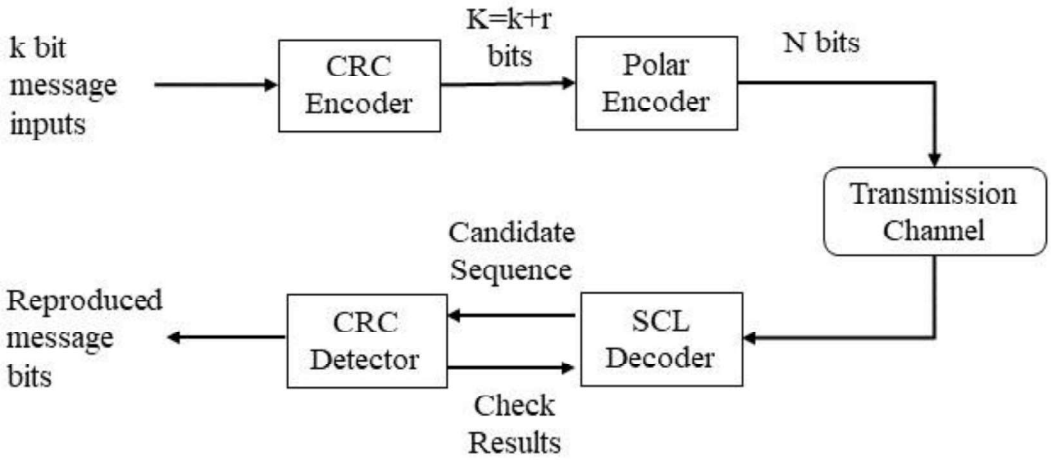


Figure 2.4 Polar codes using CRC-aided SCL decoding.

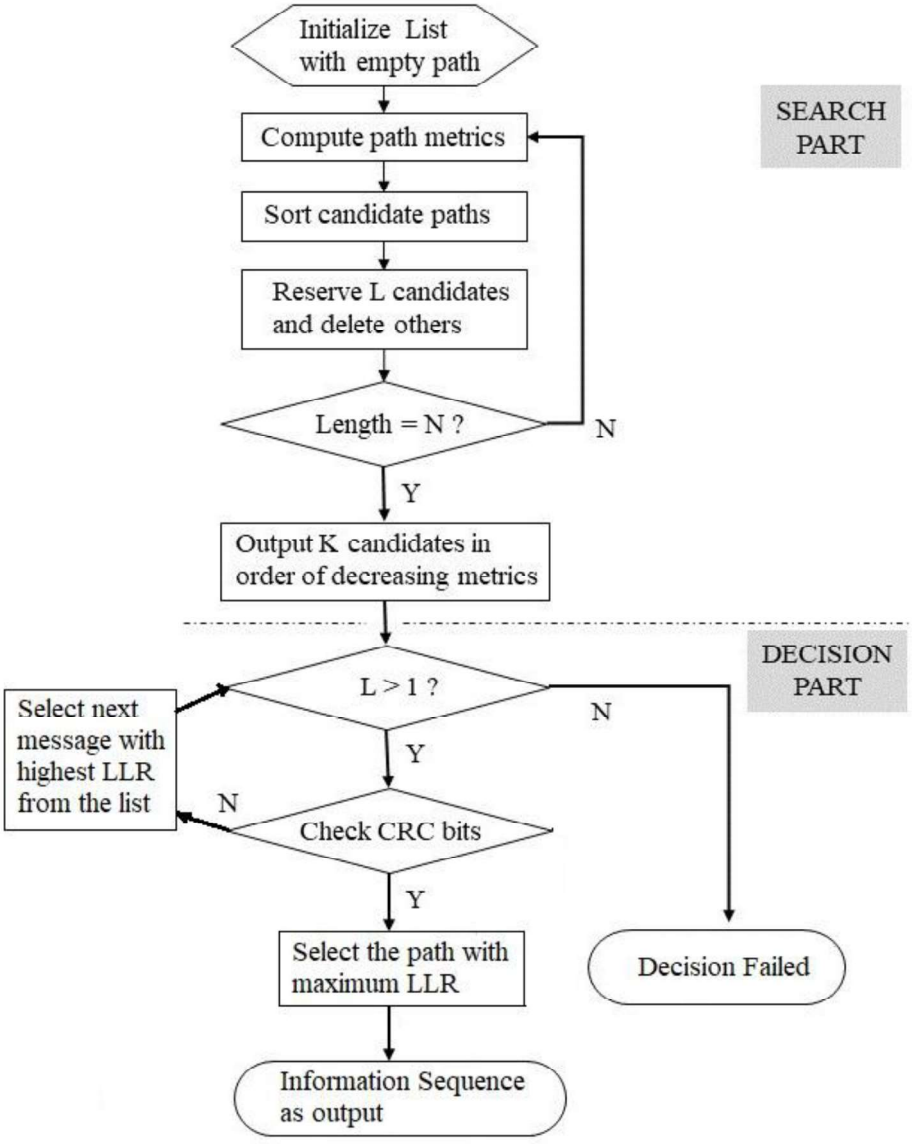


Figure 2.5 Flowchart of CA-SCL decoding algorithm.

In Figure 2.4, the block diagram of polar codes using CRC-aided SCL decoding scenario is shown. Instead of setting all (N – k) frozen bits to zero, (N – k – r) frozen bits can be set to zero and r bit CRC value (or the parity bits) are applied. CRC polynomial of degree r contains (r+1) coefficients and r coefficient bits are affixed after k information bits to produce (K = k+r) bit. Polynomial division is performed to confirm the accuracy of the received message bits if the remainder is zero. The path with the maximum likelihood is selected as the utmost reliable path and CRC detector verifies its information sequence. When the CRC detector identifies the selection as wrong, the verification of the contenders with the next probable likelihood done till any of the *L* contenders pass the CRC test. The next message with highest LLRs selected from the list with predefined list size L and all vectors deleted if failed to pass CRC verification. The corresponding flowchart of CA-SCL decoding algorithm is presented in Figure 2.5.

The r bit CRC is affixed after the k bit information as an external code with code rate $k/(k + r)$ so that the actual polar code rate is increased to R_{polar} = (k+r)/N keeping original information rate R_{info} = (k/N) intact. Hence, polar code rate increases and error correction declines as r incremented to get improved error detection performance. The redundancy introduced by the outer CRC code should be planned realistically, particularly if block size (N) is small. It can falsely identify the wrong one as right candidate if improper CRC length is chosen.

(iv*) BP decoding* [83]: In Log-Likelihood Ratios (LLR) data is repeatedly transmitted over the encoding graph in BP decoding, with an iteration limit or a quick termination criterion. The decoding is stopped by employing G-matrix-based early termination.

(v) *BPL decoding* [83]: Multiple BP decoders with a list of L, each using different factor graphs (FG), work synchronously to decode the data in BPL decoding.

(vi) *CA-BPL decoding* [84]: The error-correction capabilities are leveraged by combining iterative BPL decoding with CRC in CA-BPL decoding.

These existing decoding schemes use different approaches to decode polar codes and vary with respect to complexity and performance characteristics. In the contributory chapters of this thesis, polar and Golay codes are employed for designing schemes for error corrections and security. Therefore, a brief comparison between Golay and polar codes is presented in this chapter in Table 2.1.

Table 2.1 Comparative analysis of Golay and polar codes

Feature	**Golay Codes**	**Polar Codes**
Block Length	Either 23 or 24	Flexible
Complexity	Low	Moderate to High
Performance	Robust in noise	Capacity-achieving
Applications	Space communication	5G, IoT, Wireless networks

2.3 Different Channel Models

Channel models play a vital role in understanding transmission medium and designing practical communication systems. These models simulate the physical medium through which signals propagate, capturing the effects of attenuation, noise, and interference. Commonly used models include the Additive White Gaussian Noise channel**,** which idealizes noise as a white Gaussian process affecting the signal, and is widely used for theoretical performance analysis. The Rayleigh and Rician fading models are employed to characterize wireless channels with multipath propagation, where signal reflections cause constructive and destructive interference. Different channel models used in this research are described briefly in the following subsection.

2.3.1 Binary Deletion Channel (BDC)

In a communication system, synchronization anomalies caused by imperfect sampling can result in the loss of some symbols from the received vector or the insertion of unwanted random symbols among the correctly received ones. An example of a nonstationary and non-ergodic channel with memory is the BDC where information bits or symbols from the sender are either sent to the receiver (with deletion probability p) or deleted without notifying the receiver (with probability 1-p). Symbols can be deleted randomly whereas receiver is unfamiliar with deletion locations. The characteristic of a deletion channel is not similar like an erasure channel where bits may get erased or scrambled due to channel imperfections. For example, if 01010101 was transmitted to the channel, the receiver would acquire 00011 if the second, fourth, and seventh bits were deleted, and would acquire 0?0?01?1 if the bits were erased. Mitzenmacher [85] surveyed and explained deletion channels in detail. There have been several findings regarding upper and lower limits on deletion channel's capacity [85 - 86] but the Shannon capacity of deletion channels are not fully explored yet. The practical channels corrupted by these types of insertion or deletion errors have memory and the polar codes used for memoryless channels cannot be applied straightway as the polarization of a channel with memory has not been fully inspected. In a few recent works [87 - 92], polarization theorems are exploited for processes with memory.

Dolecek et al. [93] showed that substitutions alongside a single deletion can be corrected using Reed-Muller (RM) codes. Due to similar algebraic constructions, polar code is a prospective candidate for correcting deletions too. Polar codes over a BDC can be considered as an adversarial version of the binary erasure channel (BEC) with deletion properties, and a few modifications in SCL decoding algorithm can be able to recover the original message successfully. For BEC with single deletion error, a polar decoding scheme is suggested by Thomas et al. [90] where the deleted symbols recovered by employing pre-coded CRC. All possible deletion patterns are identified and analyzed. Each pattern is then decoded accordingly. The average asymptotic decoding complexity is $O(N^{d+1} \log N)$ for d-deletion symbols. Guess and check codes based upon Reed-Solomon (RS) codes are advocated by Hanna et al. [91] for deletion channel. The complexity of the decoder is $O(k^{d+2} / \log^d k)$, where information-length is k. Tian et al. [92] proposed an alternative method which directly senses the deletion channel outputs to decode with no pre-processing and reducing the decoding complexity to $O(d^2N \log N)$. The method can be suitable to perform CA-SCL decoding (e.g., CRC Aided SCL) for channels with insertion and deletion errors.

Thus, the BDC is characterized by the random deletion of bits during transmission, where the receiver is unaware of the deletion locations. This channel model poses unique challenges for error correction, as conventional polar codes designed for memoryless channels cannot be directly

applied. The CA-SCL decoding algorithm has been proposed to address these challenges by incorporating CRC bits, which help identify deleted positions and improve recovery accuracy.

2.3.2 Solid State Drive (SSD) Channel

The graphical voltage distribution model of 2-bit NAND flash memory cell is presented in Figure 2.6, where four states show all probable combination of 2-bits in a NAND flash memory cell [94]. The state S_0 is termed as *erased state* and other three states (S_1 to S_3) are known as *programmed state*. The voltage distributions in a flash memory cell have a Gaussian-like distribution due to read/program disturbances and interferences [95 - 96]. So, Gaussian distribution is taken in this work as a competent model to portray the flash memory cell. At the start of SSD lifespan, the noise variance is insignificant and the gaps between each distribution are able to offer adequate information to differentiate among states. But, standard deviation of distributions produced by noise is considerably inflated with the rise of program and erase (P/E) cycles [94 -97] and thus stronger protections for stored data are required using ECC to retrieve data properly.

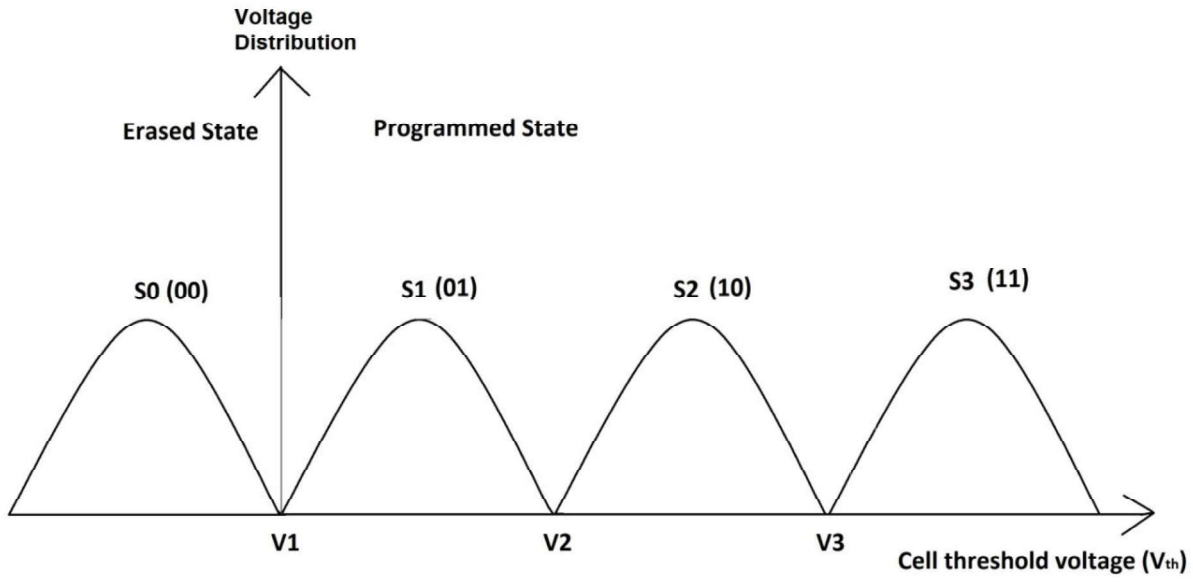


Figure 2.6 The voltage distribution model of 2-bit NAND flash memory cell

Generally, memory subsystems have used Hamming code, whereas RS code is used in HDDs and CD-ROMs. Later, BCH code, LDPC code, or combinations of both [98] are used as ECC. Though they exhibit inadequate error correction capability or extended decoding latency [99]. Polar codes [78] are renowned to attain channel capacity under BDMCs with low computational complexity of O(n log n) for hardware implementation of both encoder and decoder. Also, the performance of polar codes is better than LDPC with small code length and flexible code-rate [100]. Due to all of these advantages, polar code is considered as promising ECC scheme for SSD.

Polar codes are realized by channel polarization which is produced by the method of channel splitting and combining respectively. The information bits are sent only over the good channels and bad channels are discarded. Several kinds of decoding algorithms have been suggested for polar codes and applied very effectively in many applications. Successive Cancellation List (SCL) decoding, offered by Tal and Vardy [95], encourages the use of concatenated polar codes using CRC codes as precoding, so that the right codeword from the list can be retrieved during decoding at the receiver [80]. Still, detecting the best CRC code

length and list size combination for the concatenated polar codes has remained rather unknown [98].

The error correction competency of the polar code is extremely depending upon the perfection of the polar code structure. A code structure is essentially a design layout for input bits, built solely based on the status of channel quality. However, the channel quality status is affected by the number of program/erase (P/E) cycles of data blocks. Flash memory generally implements the array assembly to construct the chip for reducing the area-size and design of the circuit. Although, the array assembly likewise brings severe interference amongst the attached memory cells when the read, write, or erase operation is done. Usually, a bigger unit is adopted to execute these processes [94, 101-102] to escape this situation. Previously, various work has been done to implement the polar codes in the field of flash drive [95 -96, 103].

2.3.3 Additive White Gaussian Noise Channel

The Additive White Gaussian Noise (AWGN) channel is a fundamental model used in communication systems to analyze signal transmission and its degradation due to noise. It assumes that the only impairment affecting the signal is noise, which is additive, white, and Gaussian. The noise is added directly to the transmitted signal, altering its amplitude without introducing any dependency on the signal itself. This makes the model linear and easy to analyze. The term "white" indicates that the noise has a constant power spectral density over a wide range of frequencies, meaning it affects all frequency components of the signal equally. The noise follows a Gaussian probability distribution, characterized by its mean (typically zero) and variance, which defines the noise power. This reflects real-world thermal and electronic noise characteristics. The AWGN model ignores other impairments like fading, interference, or distortion, making it ideal for theoretical and baseline system performance evaluations.

The AWGN channel is extensively used to derive fundamental limits of communication systems, such as channel capacity as defined by Shannon's theorem. It is also employed in testing and comparing modulation schemes, coding techniques, and receiver designs. While the AWGN model is useful for ideal conditions, it does not account for real-world complexities like multipath propagation, shadowing, or time-variant channel effects. The simplicity and analytical tractability of the AWGN model make it a cornerstone in communication theory, forming the basis for understanding more complex channel models.

2.3.4 Rayleigh Fading Channel

Rayleigh fading channels (RFC) are a widely used model in wireless communication to describe the behaviour of a signal propagating through a multipath environment. They are particularly relevant in scenarios where there is no direct line-of-sight (LOS) path between the transmitter and receiver, and the signal reaches the receiver through multiple scattered, reflected, or diffracted paths. Some characteristics of RFC are presented below.

(i) Multipath Propagation: The signal arrives at the receiver through various paths of differing lengths, phases, and amplitudes, causing constructive or destructive interference. This leads to rapid fluctuations in the signal's amplitude and phase over time and frequency.

(ii) No Direct Line-of-Sight (LOS): Rayleigh fading assumes the absence of a dominant direct path. Instead, the received signal is the sum of many independently scattered components.

(iii) Statistical Model: The amplitude of the received signal envelope follows a Rayleigh distribution, which is appropriate when the received signal is composed of a large number of

statistically independent scattered components. The phase of the reflected, random signal at the receiver is uniformly distributed between 0 and 2π.

Rayleigh fading is commonly used to model urban and indoor wireless environments, such as mobile cellular systems, where obstacles like buildings or walls scatter the signal. Rayleigh fading significantly affects system performance, necessitating the use of techniques like diversity schemes (e.g., antenna diversity), equalization, and error-correcting codes to mitigate its effects. The Rayleigh fading model is not suitable for scenarios with a strong LOS component. In such cases, a Rician fading model is more appropriate. Rayleigh fading channels are critical in understanding and designing robust communication systems for multipath environments, helping engineers predict and counteract signal degradation due to fading.

2.3.5 Wiretap Channel

The wiretap channel is a theoretical communication model introduced by Aaron D. Wyner in 1975, which forms the foundation of physical layer security in communication systems. It is designed to analyze and ensure secure communication over a channel that is simultaneously observed by both an intended recipient and an eavesdropper. The wiretap channel is a cornerstone of information-theoretic security, providing insights into achieving secrecy without relying on traditional encryption methods. It continues to influence modern secure communication system design by ensuring confidentiality even in the presence of powerful eavesdroppers. The key concepts of the wiretap channel are as follows.

(i) Channel Structure: The wiretap channel consists of three parties: a transmitter (Alice), an intended receiver (Bob), and an eavesdropper (Eve). Alice sends information to Bob through a main channel, while Eve intercepts the communication via a separate channel known as the eavesdropper's channel. The eavesdropper's channel is typically noisier or weaker than the main channel.

(ii) Secrecy Capacity: Secrecy capacity is a measure of the maximum rate at which data can be transmitted securely from Alice to Bob, ensuring that Eve gains no useful information. Mathematically, the secrecy capacity $C_s$is defined as the difference between the capacities of the main channel C_m and the eavesdropper's channel C_e. If the main channel is less capable than the eavesdropper's channel ($C_m<C_e$), secure communication is not possible.

(iii) Achieving Security: The security of the wiretap channel is guaranteed by ensuring that the mutual information between the transmitted message and Eve's observations is negligible. Techniques such as encoding the message with random noise or using advanced coding schemes (e.g., wiretap codes) are employed to ensure confidentiality.

(iv) Physical Layer Security: Unlike traditional cryptographic methods, which rely on computational complexity, the wiretap channel exploits the physical properties of the communication medium to achieve security. This makes it robust against computational advances, including quantum computing.

(v) Extensions: Multiple-input multiple-output (MIMO) wiretap channels are leverage spatial diversity for enhanced security. *Relay-assisted wiretap channels* use intermediate nodes to improve secure communication. *Quantum wiretap channels* extend the concept to quantum communication systems, ensuring security in quantum networks.

2.3.6 Physical control channels in 5G NR

The control channel in 5G New Radio (NR) is a crucial component of the physical layer, responsible for conveying control information between the base station (BS) and the user equipment (UE). This information ensures the proper functioning of the network by facilitating tasks such as resource allocation, scheduling, and link adaptation. The design of 5G NR control channels emphasizes flexibility, efficiency, and scalability to meet the diverse requirements of 5G applications. The 5G NR control channel is a pivotal element of the 5G ecosystem, enabling efficient and reliable communication across a wide range of use cases. Its advanced features and adaptability make it a key enabler of 5G's promise to transform connectivity and communication. A number of physical channels constitute physical layer in 5G NR, as shown in Figure 2.7.

The information from a UE to a BS is transported in uplink channels and the information from BS to the UE is transferred in downlink channels. These NR physical channels are classified as data channels to transfer user data, or control channels to transfer control information. PUCCH (Physical Uplink Control Channel) and PUSCH (Physical Uplink Shared Channel) are for transmitting control information. Also, PDCCH (Physical Downlink Control Channel) and PDSCH (Physical Downlink Shared Channel) are for receiving control information. PBCH (Physical Broadcast Channel) is for broadcasting control information to many UE from the BS. The control channels transport control information to synchronize all UEs connected with the BS for managing data transmission in data channels. They also support initial connections to the BS. Each UE implements blind decoding, where it decodes numerous hypothesised blocks containing E (encoded block length), A (block length of information), DCI (Downlink Control Information) type and location in downlink channel.

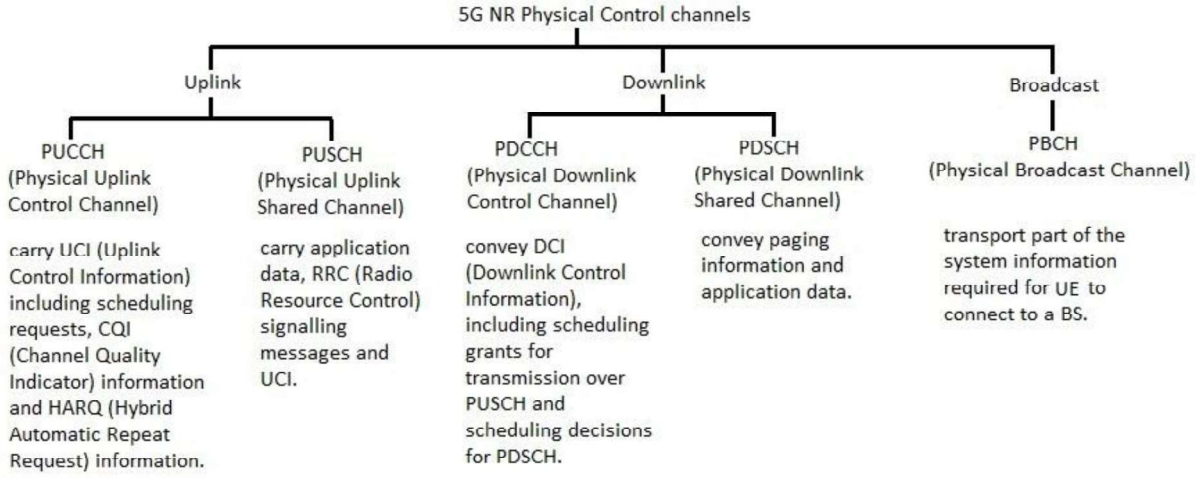


Figure 2.7 Physical control channels in 5G NR.

The key types of control channels in 5G NR are discussed briefly as follows.

(i) Physical Downlink Control Channel (PDCCH): The PDCCH carries downlink control information (DCI), which includes scheduling information for both uplink and downlink data transmissions, resource assignments, and power control commands. It supports multiple formats and aggregation levels to adapt to varying channel conditions and UE capabilities. PDCCH is decoded using a search space mechanism, which defines the potential locations where control information can be found.

(ii) Physical Uplink Control Channel (PUCCH): The PUCCH is used by the UE to send uplink control information (UCI) to the base station.UCI includes hybrid automatic repeat

request (HARQ) feedback, scheduling requests, channel state information (CSI), and acknowledgment (ACK/NACK) responses. It is designed to support various formats to accommodate different payload sizes and latency requirements.

(iii) Physical Broadcast Channel (PBCH): The PBCH transmits the master information block (MIB), which contains essential system information needed for initial access to the network. It is broadcasted periodically to all UEs within the cell.

The key features of 5G NR control channels are presented below.

(i) Flexibility: The design supports diverse use cases, from high-throughput eMBB applications to low-latency and high-reliability scenarios like URLLC. Dynamic scheduling and configurable control resource sets adapt to changing traffic demands and network conditions.

(ii) Beamforming Support: Control channels leverage advanced beamforming techniques to enhance coverage and reliability, especially in high-frequency millimeter-wave bands.

(iii) Higher Reliability: Multiple repetitions and aggregation levels improve decoding performance in challenging conditions, such as at the cell edge.

(iv) Spectral Efficiency: 5G NR employs optimized resource allocation and overhead reduction strategies to minimize control channel usage while maximizing data throughput.

(v) Support for Massive Connectivity: The control channel design accommodates a massive number of connected devices, a requirement for IoT and machine-type communication (MTC).

2.4 Summary

An in-depth exploration of Golay and polar codes has been presented in this chapter. Golay and polar codes represent two significant advancements in the field of error correction coding. Golay codes exemplify the elegance of classical block coding, while polar codes represent the forefront of ECC, achieving Shannon capacity. Golay codes, with their ability to correct multiple errors, have been widely used in various applications with high reliability, while polar codes have emerged as a powerful tool for achieving the capacity of communication channels and have poised to play a crucial role in future communication systems. As communication technologies continue to evolve, the importance of efficient and robust error correction methods will only increase, making the study of these codes essential for future developments in this field. Understanding the principles and applications of Golay and polar codes will provide valuable insights into the design and implementation of modern communication networks. Also, an overview of different channel models used in this thesis is described briefly in this chapter.

Research on concatenated or hybrid coding schemes and polar code extensions could address challenges in high-dimensional and quantum communication systems. The integration of these codes into next-generation technologies like 6G remains a promising area for future exploration. Design and performance evaluation of polar codes for two different channel models (BDC and SSD) have been introduced in the next chapter with the objective of reducing the BLER.

Chapter 3

Performance Evaluation of polar codes over different channel models

3.1 Introduction

The increasing demand for reliable data transmission in modern communication systems necessitates the development of efficient ECC. Polar codes [78], introduced by Arikan in 2009, have emerged as a significant advancement in channel coding, particularly for binary-input discrete memoryless channels (BDMCs). Polar codes have gained attention due to their ability to achieve the symmetric capacity of BDMCs with low encoding and decoding complexities. This chapter aims to evaluate the performance of polar codes for various channel models, emphasizing their application in environments characterized by deletion and noise. These codes are also applicable practically in 5G wireless communication systems [104-105]. Already, several efforts have made to construct polar codes for other channel models [106 - 108], e.g., BSC and AWGN, providing roadmap to polar code construction algorithms for practical transmission channels with memory as well. The successive cancellation (SC) decoding algorithm of polar decoders are not perform well at large finite block lengths and that leads to several alternative SC decoding algorithms [108 - 109] with low computational power by researchers later. The efficient implementation of Successive Cancellation List (SCL) decoding algorithm, invented by Tal and Vardy [79], outperformed LDPC and turbo codes and is useful since then. The SCL decoding algorithm is also suitable for the information pre-coded with CRC bits. The polar SCL decoder can eradicate the codeword symbols mismatched with CRC information and attain reduced error rate even in channel imperfections by selecting correct received vectors. An optimum CRC detection algorithm for polar codes using SCL decoder is presented in [82].

With the necessity of massive data storage applications, SSDs have been improved and accepted gradually as an efficient data storage device to support the huge storage capacity and simultaneously save the cost using NAND flash memory in modern computing systems. In comparison with magnetic HDDs, SSDs convey greater read and write performance, enhancement in random-access input/output (I/O) operations, resistance to physical shock, low form factor and less static power consumption. Also, SSDs are less susceptible to drive failure than HDDs due to absence of mechanical parts. Though HDDs are progressively replaced by SSDs as primary data storage, error correction is still severe to SSDs since NAND flash memories have worsening reliability.

This chapter evaluates the performance of polar codes across various channel models, including the BDC and the SSD Channel. The analysis focuses on the effectiveness of CRC-aided Successive Cancellation List (CA-SCL) decoding, which enhances error correction capabilities in practical applications. The chapter presents some findings from recent studies, highlighting the advantages and limitations of polar codes in different scenarios. The rest of this chapter are arranged in the given fashion. The implementation of the CA-SCL decoder for deletion channel and SSD channel is explained based upon problem formulation in Section 3.2 and 3.3, respectively.

Simulation results are presented in graphs and analysis is done. Lastly, the summary is presented in Section 3.4.

3.2 Design and Implementation of polar codes in Binary Deletion Channel (BDC)

The block diagram of BDC with d-deletions is presented in Figure 3.1. The CA-SCL decoder receives (N - d) bits as channel output in a d-deletion channel for N transmitted bits though the receiver is unaware of deleted bit locations [85]. It is suggested that individual coded bit can be deleted with a deletion probability $P_d \in (0,1)$. Though the capacity of BDC is still not fully known, the upper bound capacity can be taken as $C(W) = (1 - P_d)$ if the deletion locations are considered as known to receiver and the capacity of BDC will become similar as the capacity of BEC.

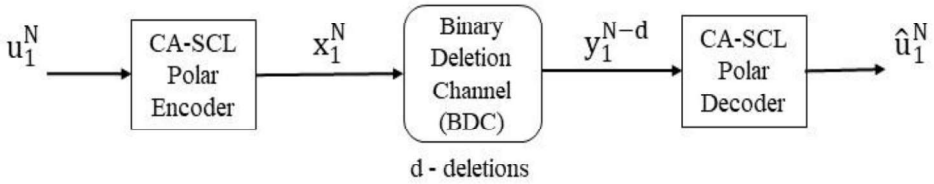


Figure 3.1 Binary deletion channel model with d-deletions where receiver only succeeds to receive (N-d) coded bits instead of N.

3.2.1 Related existing works

The existing works on the reliability in BDC is presented in concise manner in Table 3.1. The existing works show either the ability of single deletion correction or high decoding complexity using traditional polar codes with SC decoding. None of the previous works had used CA-SCL decoding and performed d-deletion corrections.

Table 3.1 Existing works on reliability in BEC and BDC.

Existing work	ECC used	Channel	Performance
Thomas et al. [90]	Polar codes	BEC	single deletion error symbols recovered by employing pre-coded CRC
Hanna et al. [91]	Guess and check codes based upon RS codes	BDC	High decoding complexity
Tian et al. [92]	Polar codes	BDC	decode with no pre-processing
Dolecek et al. [93]	RM codes	BDC	substitutions alongside a single deletion can be corrected

Considering the capacity of BDC is same as the capacity of BEC with deletion probability $P_d \in (0,1)$ for d-deletions, a message vector u_1^N is encoded using CRC aided polar encoder and is transmitted through the channel. The CRC generator polynomials [92, 95] used in this work are given below as

CRC-8: $z^8+z^7+z^6+z^4+z^2+1$

CRC-12: $z^{12}+z^{11}+z^3+z^2+z+1$

CRC-16: $z^{16}+z^{15}+z^2+1$

CRC-23: $z^{23}+z^{9}+z^{7}+z^{5}+z^{3}+1$

CRC-32: $z^{32}+z^{26}+z^{23}+z^{22}+z^{16}+z^{12}+z^{11}+z^{10}+z^{8}+z^{7}+z^{5}+z^{4}+z^{2}+z+1$

The BDC permits erasures along-with d-deletions. The CA-SCL polar decoder is designed and capable to retrieve original message from the received vector with a list L of linear size in N. The list contains the estimation of $\hat{u}_1^N$ for the transmitted message vector u_1^N. All the scaling parameters involved in encoding-decoding process can be saved in memory to reduced real-time decoding complexity further.If L represents the candidate codewords formed by SCL decoder with list L = {1, 2, . . .L} and $\ell^\star$ represents the list index which compares to the right codeword. Without loss of generality, it can be anticipated that smaller indices signify more probable contenders. If the list has no correct codeword i.e., $\ell^\star \notin$ L, it can be assumed that $\ell^\star > L$. Furthermore, let $P_{l^*}(\gamma, L)$ = the probability adapted on γ, i.e., $P_r\{\ell^\star = l \mid \gamma\}$ denote the distribution of $\ell^\star$ which is the probability that $\ell^\star$ concurs with l at a stated Signal-to-Noise Ratio over a transmission channel. Considering the perfect situation where the right codeword in the list is recognized accurately, then the overall error probability $P_{e,\mathrm{id}}(\gamma, L)$ of the decoder can be expressed as

$$P_{e,id}(\gamma, L) = 1 - \sum_{l=1}^{L} P_{l^*}(\gamma, L) \tag{3.1}$$

The error correction ability of CA-SCL decoding is mainly depends on the proper utilization of CRC polynomials regardless of whether the length of CRC codes is same.

3.2.2 Result analysis

The effectiveness of the offered CA-SCL decoding algorithm for the BDC is shown by MATLAB simulation results in this section. Here, deletion errors with erasures are considered. However, the same algorithm might be utilized for the combination of deletion error with additive noise and with symmetric flips in case of AWGN channel and BSC respectively. Since BDC differs from BDMC, the conventional constructional algorithm cannot be used straightway and frozen bits cannot be considered to zero always. Thus, a small modification in polar code constructional algorithm is done and large computational complexity is required to attain high precision results. Frozen bits are taken randomly and encoder-decoder module is fully known about frozen bit information. Deletion bits are taken randomly with the fixed deletion probability as high as 0.3 and deletion positions are not known to decoder. CRC polynomial is selected according to [92 -93]. More specifically, CRC-8, CRC-12, CRC-16, CRC-23 and CRC-32 are used with codelengths N_1=128, N_2=256, N_3=512, N_4=1024 and N_5=2048, respectively.

In Figure 3.2, the relation between Codelength N and overall BLER is demonstrated for the proposed CA-SCL polar code of different information rates $R_{info} \in$ {3/7, 1/2, 5/9, 3/5, 2/3} with CRC length $\in$ {8, 12, 16, 23, 32} and List size |L| ≥1. As expected, polar codes perform better with lower codelengths N and show poor performance at high information rate. In Figure 3.3, the relation between polar code rate R_{polar} and overall error probabilities is illustrated for the proposed CA-SCL polar code of different code-lengths or Block-sizes N $\in$ {128, 256, 512, 1024, 2048} with CRC length $\in$ {8, 12, 16, 23, 32} and List size |L| ≥1. Results show that polar codes perform better with lower code-rates applying different CRC-sizes and able to retrieve original messages accurately. At lower block size (or codelength), polar codes exhibit high error probability even if in low code rates.

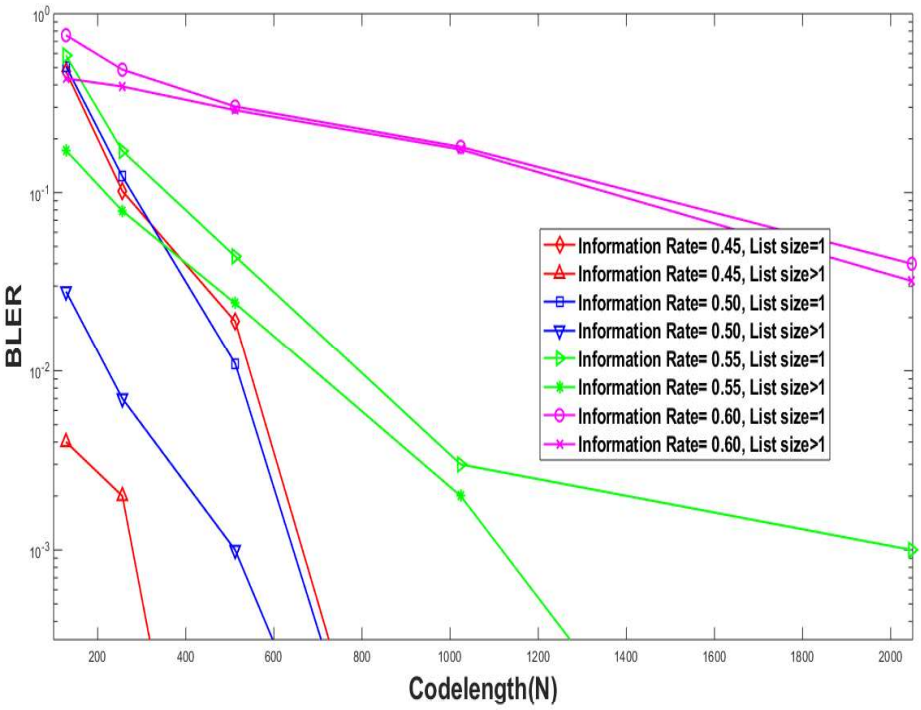


Figure 3.2 Plot of BLER against codelength (N) with fixed deletion probability and variable CRC-length.

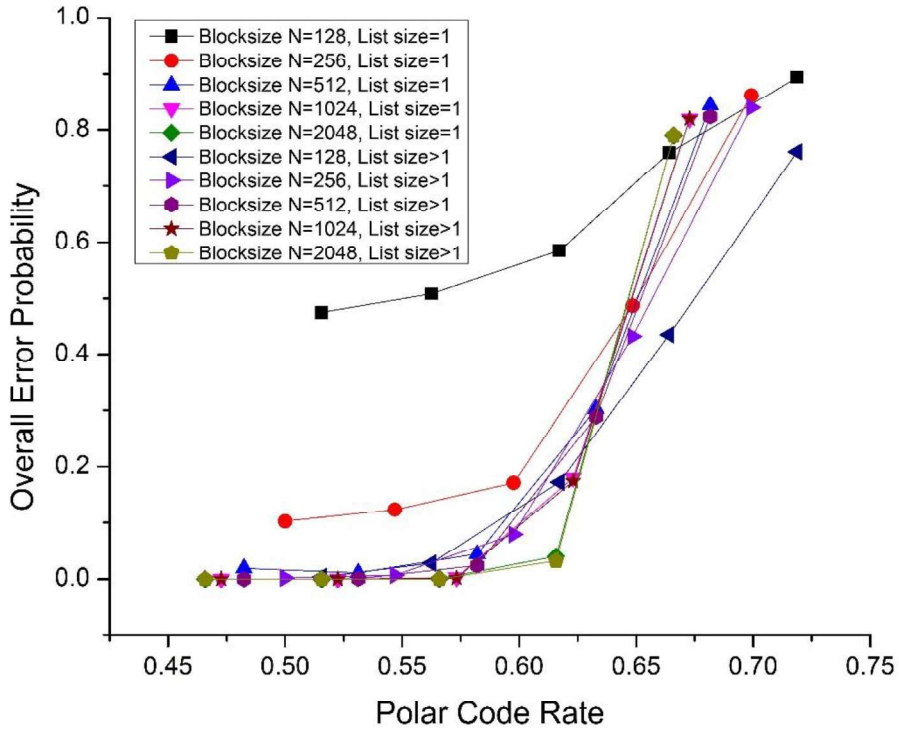


Figure 3.3 Plot of overall error probability against polar code rate with fixed deletion probability and variable CRC-length.

The decoding complexity of the proposed design is compared with the existing works in Table 3.2 and it outperform in terms of level of complexity.

Table 3.2 Comparison of decoding complexity.

Scheme	ECC used	Decoding Complexity	Level of complexity
Thomas et al. [90]	Polar	$O(N^{d+1} \log N)$ for d-deletion symbols	Very High
Hanna et al. [91]	Reed-Solomon (RS)	$O(k^{d+2} / \log^{d} k)$, where information-length is k.	High
Tian et al. [92]	Polar	$O(d^2 N \log N)$	Moderate
Proposed	Polar (with CA-SCL decoding)	O(dLN log N)	Low

3.3 Design and Implementation of polar codes in Solid State Drives (SSD) channel

A pictorial example of the polar code combination implementation in Flash cells for polar code (8, 6) is presented in Figure 3.4. Here, U signifies the bit-channels which are resulting from the code structure consistent with the input data, whereas X and Y indicate the encoded bits and the noisy bits respectively. Y-bits could be altered/corrupted after X-bits are stored into flash cells in the presence of noise and polar code fails on decoding/correcting some noisy bits. Due to the polarization, each bit-channel may be connected to the several physical / flash cells. Based upon the code construction aimed to achieve high error correction ability, the bit-channels are rated. U3 to U8 bit channels are utilized for information bits as they have lesser raw BER than U2 and U1 [94]. The error correction performance would be satisfactory if the corresponding flash cells of the information bits are more reliable than the adjacent frozen bits.

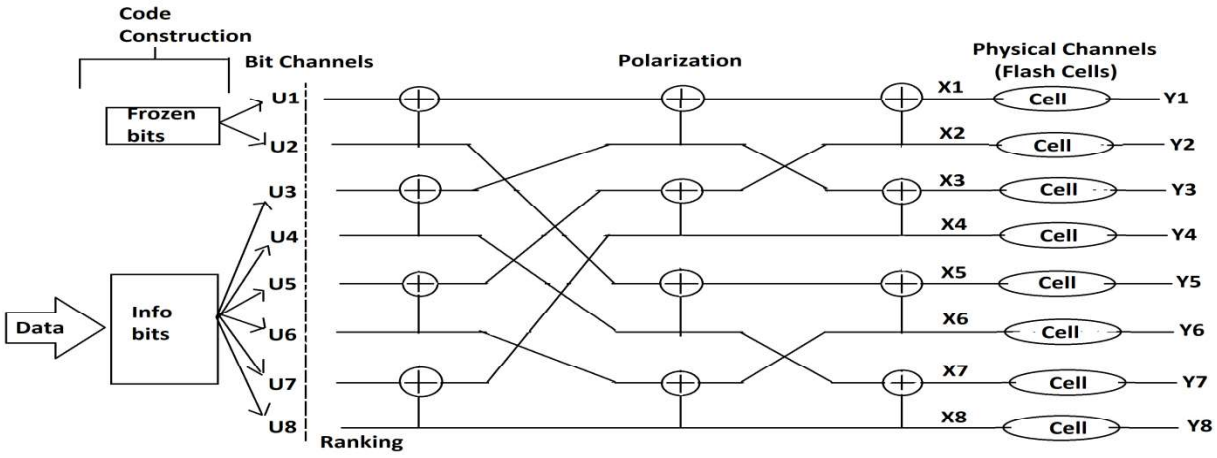


Figure 3.4 A pictorial example of polar code (8, 6) implementation in flash cells.

3.3.1 Related existing works

The existing works on the reliability in SSD is presented in concise manner in Table 3.3 below. The existing works show the ability of good error correction at the expense of high E_b/N_o implementing polar codes with SCL decoding. None of the previous works had used CA-SCL decoding and achieved target BLER at low E_b/N_o.

Table 3.3 Existing works on reliability in SSD.

Existing work	ECC employed	Channel used	Performance	Merits & Demerits
Hsu et al. [94]	Polar code	NAND flash memory	proactive channel adjustment design modifies the quality of the critical flash cells to maintain the correctness of the code construction	good error correction capability, but higher space overhead
Song et al. [101]	Polar code	Multi Level Cell (MLC) flash memory	multi-strategy ECC scheme using pre-check mechanism based on polar code is proposed	hard decoder can correct the majority of erroneous codewords with increasing system complexity

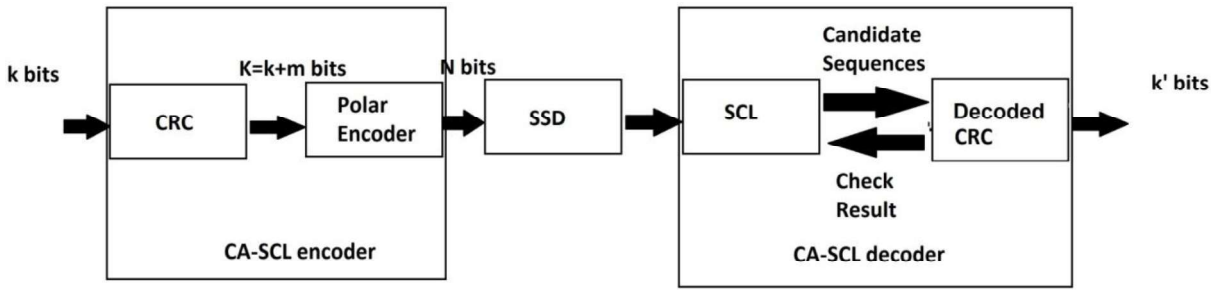


Figure 3.5 Block diagram of CRC concatenated polar encoding and list decoding model for SSD.

Figure 3.5 displays the block diagram for the realization of the polar codes in flash cells. To practically implement this codes in SSD, first k number of information bits have to be encoded with the appending of certain number of CRC bits, then encoding is done for the total number of N bits as in polar coding, which is to be stored in the SSD. When the data is to be retrieved, the CA-SCL decoding comes into play in which SCL decoding takes place first and then CRC checking is performed to detect errors. This decoding process repeats several times according to the given list size and then the final decoded bits come out with low error probability.

For mathematical modelling of Multi Level Cell (MLC) flash the voltage distribution given by Intel [110] is generally considered as a Gaussian distribution with probability density function (PDF) as

$$P_0(X) = \frac{1}{\sigma_0\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2{\sigma_0}^2}} \tag{3.2}$$

where the standard deviation is σ_0 and the expectation is μ. As the distribution of the voltage is given by the Gaussian distribution, thus the implementation of CRC concatenated polar code in SSD by AWGN channel comes under consideration in this work.

3.3.2 Result analysis

In this section, the proposed concatenated polar codes are investigated for AWGN channel, considering the characteristics of flash cell channels. Additionally, the decoding performance is compared with existing work [101]. The m number of CRC bits are added with the k number of information and all the $K = k + m$ bits are fed into the polar encoders becoming the corresponding code rate $R = K/N$. For all coding schemes, R=1/2 is taken and CRC generator polynomial are randomly selected to identify best error correction performance in terms of BLER.

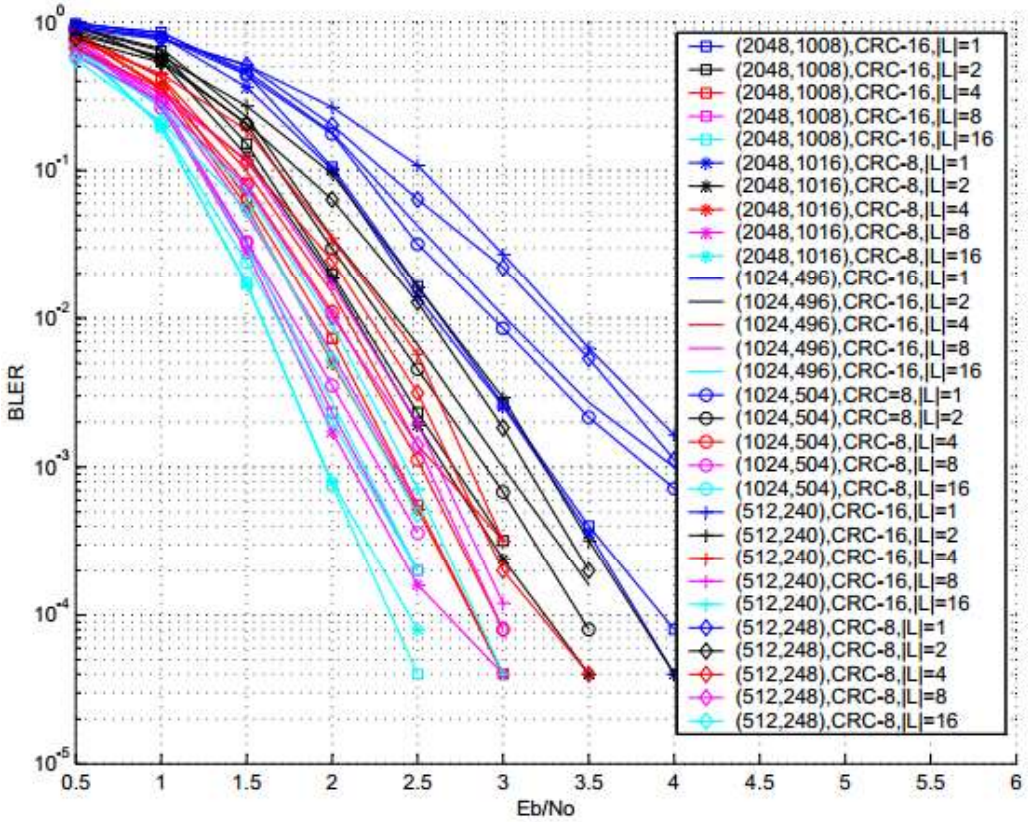


Figure 3.6 Plot of BLER performance results for CRC concatenated polar codes with fixed code rate R = ½ and varying codelength, CRC and list size.

In Figure 3.6, the BLER performance for proposed concatenated polar codes with different block length N, CRC size and list size L are examined for fixed code rate R. The error correction performance of proposed codes increases rapidly with increasing E_b/N_o value for a constant CRC size, list size and encoded bits. Again, for a constant number of encoded bits, CRC size and E_b/N_o value, the performance of correcting errors increases smoothly as the list size increases greater than 1. From Figure 3.6, it can be witnessed that the performance of correcting errors decreases with decreasing L or increasing CRC length after a certain limit or decreasing E_b/N_o value. Thus, to increase the performance of decoders in SSD, relatively low value of CRC length and comparatively high value of L should be used during SCL decoding.

Though the value of L should not be increased above 16 because of excessive rise in latency that gets introduced in high value of L during the decoding process. Also the CRC length is taken 8 and 16, as low CRC length lead to loss of effective correction of the error bits. Figure 3.7 shows basically the graphical representation of Table 3.5 results where codelength N = 1024 and K = 512 with CRC = {8, 16}, list size = {1, 2, 4, 8} and varying code rate R={0.508, 0.516} respectively, though L = 16 is not considered since the simulation result for L = 8 and L = 16 are close enough.

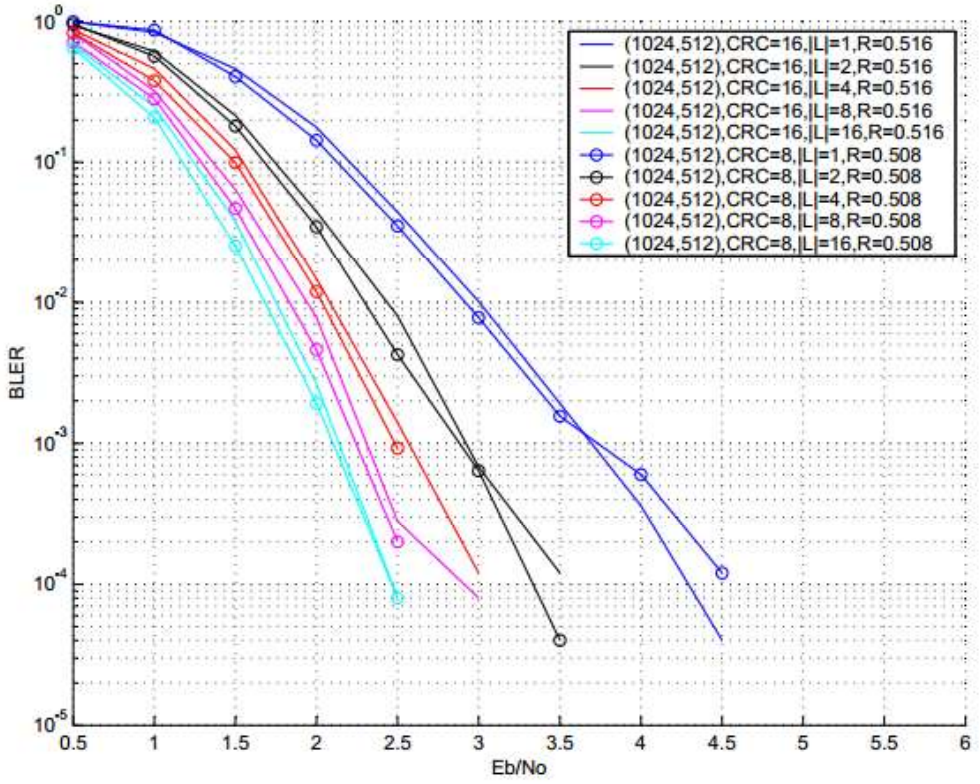


Figure 3.7 Plot of BLER performance results for CRC concatenated polar codes with fixed codelength N = 1024, CRC= {16, 8}, list size= {1, 2, 4, 8, 16} and varying code rate.

Table 3.4 Decoding performance of polar codes (with SCL decoding) for SSD channel as presented in previous work [101]

BLER Performance	**E_b/N_o for P(1024, 512) with no CRC and R=0.500**	
	Hard decoding	**Quantized-soft decoding**
10^{-1}	1.72	0.80
10^{-2}	2.92	2.06
10^{-3}	4.12	3.12
10^{-4}	5.10	4.12

Table 3.5 Decoding performance of CRC aided polar codes (CA-SCL decoding) for SSD channel as proposed in this work.

BLER Performance	**E_b/N_o for P(1024, 512) with CRC=8 and R=0.508 [Hard decoding]**			
	L=1	L=2	L=4	L=8
10^{-1}	2.15	1.6	1.5	1.3
10^{-2}	2.9	2.3	2.0	1.8
10^{-3}	3.7	2.9	2.5	2.25
10^{-4}	-	3.35	-	-

From Tables 3.4 and 3.5, it can be observed that by keeping fixed code rate as R=0.5 and varying the information bits with respect to fixed CRC bits would increase the performance of SCL decoder very much, though the information bits cannot be decreased too much because the net bit rate becomes very low compared to gross bit rate and thus rendering the purpose of SSD of having the capacity of storing large amount of data. Also, if the CRC length is decreased too much, then the performance of the cyclic redundancy check to detect and correct the error bits reduces. Comparing the values of Table 3.4 and Table 3.5, it is shown in Figure 3.8 that by taking N=1024 and K=512 for proposed CRC concatenated polar codes with different lists also give much better BLER performance than the decoders (hard and quantized-soft decoder) mentioned in existing work [101] exclusively for NAND-Flash cells of SSD. The E_b/N_o improvement in the proposed design is compared with the existing works in Table 3.6 and it outperforms in terms of E_b/N_o to achieve target BLER.

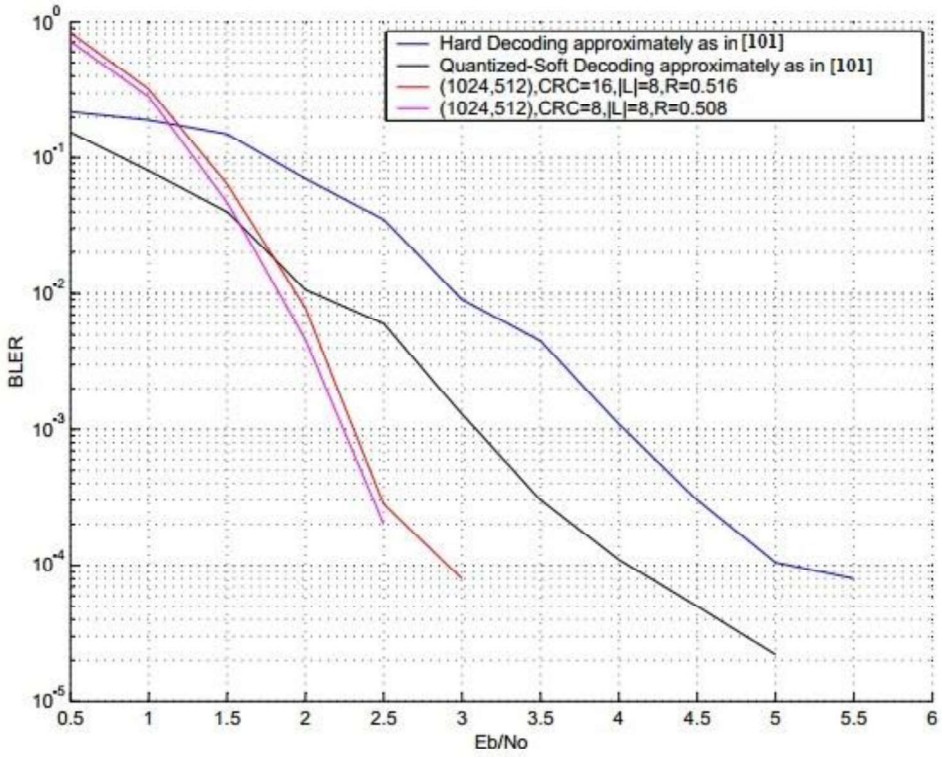


Figure 3.8 Plot of BLER performance comparison between the proposed methods with existing works [101].

Table 3.6 E_b/N_o improvement in the proposed design.

BLER	Proposed work (CA-SCL decoding)	SCL decoding [101]	E_b/N_o Improv. (%) in Proposed work
	E_b/N_o for P(1024, 512) with CRC=8 and R=0.508	E_b/N_o for P(1024, 512) with no CRC and R=0.500	
	Hard decoding [L=2]	Hard decoding [L=2]	
10^{-1}	1.60	1.72	6.97
10^{-2}	2.31	2.92	21.23
10^{-3}	2.92	4.12	29.61
10^{-4}	3.35	5.10	34.31

3.4 Summary

The error correction ability of CRC-aided polar codes is investigated in this chapter for the Binary Deletion Channel and SSD Channel. A compromising relationship between the BLER performance and CRC codelength is observed when the polar code rate is same. If the CRC length is higher than essential for a specified BLER, the dreadful performance is anticipated as a result of the comparative increment in the polar code rate. Also, larger list size and codelength tends to increase computation time much longer. The precise performance analysis of CA-SCL decoding is still challenging because of correlation between codeword bits. Also, the proposed CRC concatenated polar codes with SCL decoding are shown effectively improve error correction ability in flash data storage in SSD. The investigational outcomes confirm that the decoding performance of the proposed method approaches the preferred error floor behaviour for error correction in flash storage devices. The performance evaluation of polar codes across different channel models demonstrates their versatility and effectiveness in error correction. The integration of CRC and SCL decoding techniques enhances the reliability of polar codes, making them suitable for applications in modern communication systems. The next chapter presents Genetic Algorithm based efficient polar codes for Additive White Gaussian Noise and Rayleigh Fading Channels which are designed with the aim of reducing both the BLER and computation time.

Chapter 4

Construction of efficient polar codes using Genetic Algorithm

4.1 Introduction

Wireless communication experiences corruption of data due to fading and noise in the transmission channel. When simulating a non-line-of-sight wireless transmission, the RFC helps to represent a situation in which a radio signal is scattered by multiple objects in the surrounding area before it reaches the receiver. Whereas the AWGN channel is used to mimic the scenario in which surrounding noise affects the communication system. The primary approach for safeguarding data during transmission is the implementation of ECC between the sender and receiver. Consequently, the crucial challenge for researchers in recent decades has been to focus on developing suitable ECC for communication channels. In order to improve error performance, several decoding algorithms and diverse code parameters have been put forth for the same ECC. In recent years, the application of Genetic Algorithms (GA) has gained popularity in optimizing the design of polar codes. GA is inspired by the process of natural selection and is particularly effective in solving complex optimization problems.

This chapter explores the construction of efficient polar codes using GAs, focusing on their application in both the AWGN channel and the RFC. The remaining sections of this chapter are organized as follows: Existing works are reviewed and discussed briefly in Section 4.2. Section 4.3 offers a detailed explanation of the polar code construction employing GA. In Section 4.4, we present and elucidate the outcomes of our research. Finally, Section 4.5 summarizes the key findings and presents the summary of the study.

4.2 Related existing works

The practical application of polar codes is currently limited because of their high complexity, sequential decoding as well as significant delay for longer block length. There are a number of different polar decoding techniques, including Belief Propagation (BP), Belief Propagation List (BPL) [83], and CRC-Aided BPL (CA-BPL) decoding [84]. Nevertheless, these schemes exhibit inferior performance in comparison to CRC-Aided SCL (CA-SCL) decoding [111], while they have merits in terms of latency and parallelism. Hence, a well-designed polar code is crucial for achieving either lower error rate or complexity for a specific decoding method. To address the polar code design challenge, Tal et al. [95] suggest a useful approximation approach. In several published literatures, distinct concatenated polar coding techniques are presented using various decoders, enhancing finite-length performance. However, custom designs of BP or SCL decoders tend to not be compact owing to the manifold dependencies in the decoding tree.

GA offer a suitable solution for the polar code design issue in contrast to other optimization techniques, particularly due to the challenging decisions regarding whether bits are frozen or non-frozen in the bit-channel [112]. Later, the authors in [113] presented a polar code employing CRC relaxed BP decoding, which provides a performance boost with a high

latency. In order to reduce computational complexity, a polar coding system based on the Genetic algorithm with Redefined Crossover (RC-GenAlg) is presented in [114] employing a hash table and CA-SCL decoding. But it does not outperform in error correction compared to [112]. There is still room for improvement in BLER by choosing the right population size ('S') using GA to quickly find optimal solutions. Additionally, there is a need for reduced computation time by minimizing decoding latency and employing less complex decoding schemes, in contrast to popular polar decoding methods.

The primary importance of polar code design lies in the identification of the most likely locations for the k bits and the ranking of the bit-channels based on how reliable they are. The goal of this procedure, which may be observed as an optimization issue, is to determine the optimal group of k trustworthy positions inside an information set of indices {1,..., N}. The degree of reliability, which can be measured through metrics like BLER (Block Error Rate) or BER (Bit Error Rate), is a fundamental distinction among various code construction algorithms. It's important to note that polar codes are not all-inclusive, as the choice of A-vector is contingent on specific channel parameters, including the channel's Eb/No. The authors in [115] introduce adaptive polar codes where channel-specific parameters are used to design the polar codes, like the design Signal-to-Noise Ratio (design-SNR), which corresponds to the SNR of the channel in question. Furthermore, the code's minimum distance (d_{min}) relies upon the architecture of the polar code (or A), highlighting the code's sensitivity to the chosen parameterization. In the context of the AWGN channel, various design approaches have been developed. However, it's worth noting that effective polar code design is primarily achievable in a BEC (Binary Erasure Channel) situation. In reference [116], an analysis was conducted to understand the effect of the design-SNR and the specific code structure employed, offering an error-rate analysis of Successive Cancellation decoding in the AWGN channel. Furthermore, polar codes have been developed in various previous findings with a specific decoding arrangement in mind, like SCL or BP decoding. For the RFC, the polar code construction issue is addressed in [112, 117-120]. The design in [117], incorporates dynamic frozen bits, the polar code efficiency in RFC has been improved. The results presented in [118] demonstrate that long polar codes in RFC come close to the theoretical limit. In [119], the polar code design for RFC is examined using a Gaussian approximation approach. Each of these construction techniques has notable complexity and falls under the category of channel-dependent construction. Therefore, the channel parameters and state indirectly influence the error probability. These studies contribute to the understanding and optimization of polar codes for different channel conditions.

In traditional polar code construction employing Successive Cancellation (SC) decoding, Bhattacharyya parameters are commonly selected because of their close relationship to a well-defined limit on the probability of block errors. However, it is important to note that polar code designs based on mutual information or Bhattacharyya parameters are specifically tailored for SC decoding (hard-output) and may not perform optimally with SCL or BP decoding (soft-output). The polar and Reed-Muller (RM) concatenation codes are common approaches, but it can reduce the minimum distance of the code while improving error rates in SCL decoding. In [121 - 122], a code family is proposed that offers a continuum between polar and RM codes, provides flexibility in code design. All of these code design strategies aim to improve BLER in SCL as well as BP decoding. Systematic optimization for the design of polar codes is not typically obtainable, and the characteristics of list and iterative decoding are often not applied when designing the information set vector 'A'. As a result, the choice of the decoding scheme during polar code design remains an open problem.

4.3 Architecture of GA based polar code design

GAs are optimization techniques that mimic the process of natural evolution. They work by maintaining a population of potential solutions and use techniques like selection, crossover, and mutation to gradually evolve the population toward improved solutions. In the context of polar code design, GAs can be employed to optimize the selection of frozen and non-frozen bit positions, thereby enhancing the overall performance of the code. GA have undergone numerous modifications by various researchers and have found applications in various research domains. GA simulates biological evolution processes and can provide effective solutions to a wide range of problems, including search and optimization. A well-designed GA can swiftly identify the optimal local minima and efficiently tackle complex optimization problems that may lack a robust theoretical basis for solving them. In the realm of channel coding research, GAs are employed in various capacities, such as searching for maximal distance codes, decoding linear block codes, and designing decoders for convolutional and LDPC codes. GA is a versatile and powerful tool for solving complex problems in the disciplines of coding theory and telecommunications.

GA simulates the step-by-step evolution of natural organisms. It begins with a starting pool of potential candidates, where the candidates who are most fit endure and have progeny, representing the new population. Crossover involves selecting segments of chromosomes from each parent and merging them to create offspring. Mutation then randomly modifies certain chromosomes to promote exploration during the evolutionary process. These offspring continue to evolve and become the fittest. The fitness criteria are a crucial aspect of GA, and it ultimately produces an optimal solution when the population size and evolution steps are adequate enough. Viewing the optimization issue as the design of polar codes, the goal is to find the best 'A' vector using the smallest cost function, and this can be accomplished employing GA. The BLER serves as the cost function for optimization in this finding. Furthermore, to achieve a desired BLER, complexity can be taken as a cost function, e.g., minimizing the BP iteration numbers or L value of SCL decoding. In this work, a fitness function is established to evaluate the quality of newly generated offspring. Here, fitness is determined by the performance of the code.

A brief overview of applying a genetic algorithm to code design is as follows:

(i) Initially, a set of code parameters is randomly generated, forming the initial population.
(ii) A subset of promising code parameters is chosen as parents.
(iii) Offspring are created through crossover among these selected parents.
(iv) Random mutations are applied to offspring to acquaint with unique features.
(v) Good offspring replace less promising ones in the population, and this cycle continues iteratively.

The implementation of GA for polar code design involves several key steps:

(i) Initialization: A population of candidate solutions ('A'-vectors) is generated, where each candidate represents a potential configuration of frozen and non-frozen bits. An initial population is essential for GA to quickly find an optimal solution. It is drawn from [78] based on Bhattacharyya construction and SCL decoding to enhance the quality of the solutions. Additionally, the population can be sourced to expedite the solution using GA. The population (P) consists of 'A' vectors, denoted as 'A_i', for i ranging from 1 to 'S'. The population size, denoted as 'S', represents binary 'A' vectors in the search space, each with an individual fitness function, or BLER.

(ii) Fitness Evaluation: The fitness of each candidate is evaluated based on its performance in terms of BLER at a fixed design-SNR. The fitness function is typically the inverse of the BLER (where BLER is chosen as the cost function), allowing the selection of candidates with lower error rates. The fitness function evaluates and ranks different '*A*' vectors, identifying the best one as the vector that results in the lowest BLER. The outcome is further improved through error-rate simulations. It's worth noting that other practical metrics suitable for usage as fitness functions can offer an open avenue for further exploration.

(iii) Selection, Crossover and Mutation: The fittest candidates are selected to form a new population. This selection process ensures that the best-performing candidates are retained for further evolution. Crossover is applied to pairs of selected candidates to create offspring, while mutation introduces variability by randomly altering certain bits in the '*A*'-vectors. The fittest $\binom{T}{2}$ '*A*' vectors are chosen, subjected to mutation and crossover, and generate a new population. This step ensures that the optimal value is achieved while maintaining the pattern of the cost function over population evolution. Successive bit flipping is applied to the resulting vector to maintain a fixed number of 1's in k while keeping the code rate (R) constant. To generate new $\binom{T}{4}$ offspring, every pair of the fittest $\binom{T}{2}$ '*A*' vectors undergoes a crossover. At specific bit positions, mutation prevents premature convergence and adds diversity. It involves a random bit flip in '*A*', representing a switch from non-frozen to frozen or vice versa. However, $R = \frac{k}{N}$ is ensured since the non-frozen bit location numbers remain k. Mutations are given to the most fit $\binom{T}{2}$ '*A*' vectors, generating new $\binom{T}{2}$ mutated offspring.

(iv) Iteration: The process of evaluation, selection, crossover, and mutation is repeated for a predetermined number of generations or until convergence is achieved. The evaluation of fitness and the check of Population P_{i+1} are iterated until (i+1) equals S. This process continues as long as i is less than the maximum number of populations. When this condition is not satisfied, the algorithm returns the most fit '*A*' and concludes the search procedure. In this finding, the new S is determined using following expression:

$$S = \binom{T}{2}_{Fittest} + \binom{T}{4}_{Crossover} + \binom{T}{2}_{Mutation} \tag{4.1}$$

The proposed design scheme for polar codes using GA is described in Algorithm 4.1 and also is depicted in Figure 4.1.

Algorithm 4.1 GA based polar code design

// **Input parameters:**

N	% code length
R	% code rate
design-SNR	% design-SNR ($\frac{E_b}{N_o}$) of GA
maxPop	% maximum population
T	% truncated parents number
S	% population size
P_{input}	% input population of '*A'* vector

// **Outputs:**

P_{out} % new population after GA transformations

A_{best} % optimum '***A***' vector

BLERs % optimum BLER

1: S ⟵ $({}^{T}C_2+{}^{T}C_4+{}^{T}C_2)$ % if T = 5, S = $({}^{T}C_2+{}^{T}C_4+{}^{T}C_2) = 25$

// **Initialization of population:**

2: design-SNR ⟵ {0, 1, 2, 3, 4, 5} dB

3: start with an empty population P_{init}

4: for every SNR in design-SNR do

5: A ⟵ Bhattacharyya construction (N, R, SNR)

6: add A to P_{init}

7: end for

8: P_{init} ⟵ select fittest S '*A*'-vector in P_{init}

9: for i = 1, 2,…,maxPop do

// **Update population:**

10: initialize empty population P_{out}

11: newlyParents ⟵ fittest T '*A*'-vectors from P_{input}

12: add newlyParents to P_{out}

13: for every A in newlyParents do

14: m ⟵ mutation (A)

15: add m to P_{out}

16: end for

17: for every pair A_1, A_2 in newlyParents do

18: c ⟵ crossover (A_1, A_2)

19: add c to P_{out}

20: end for

// **Compute BLERs & A_{best}**

21: start with an empty vector BLERs

22: for every A in P_{input} do

23: BLER ⟵ PolarDecode (A, design-SNR) % simulate A vector across a given channel (e.g., AWGN, RFC) with a chosen decoder (e.g., BP, SCL etc.) for specific design-SNR

24: add BLER to BLER**s**

25: end for

26: end for

27: BLERs ⟵ select optimum BLER

28: A_{best} ⟵ select fittest '*A*' vector

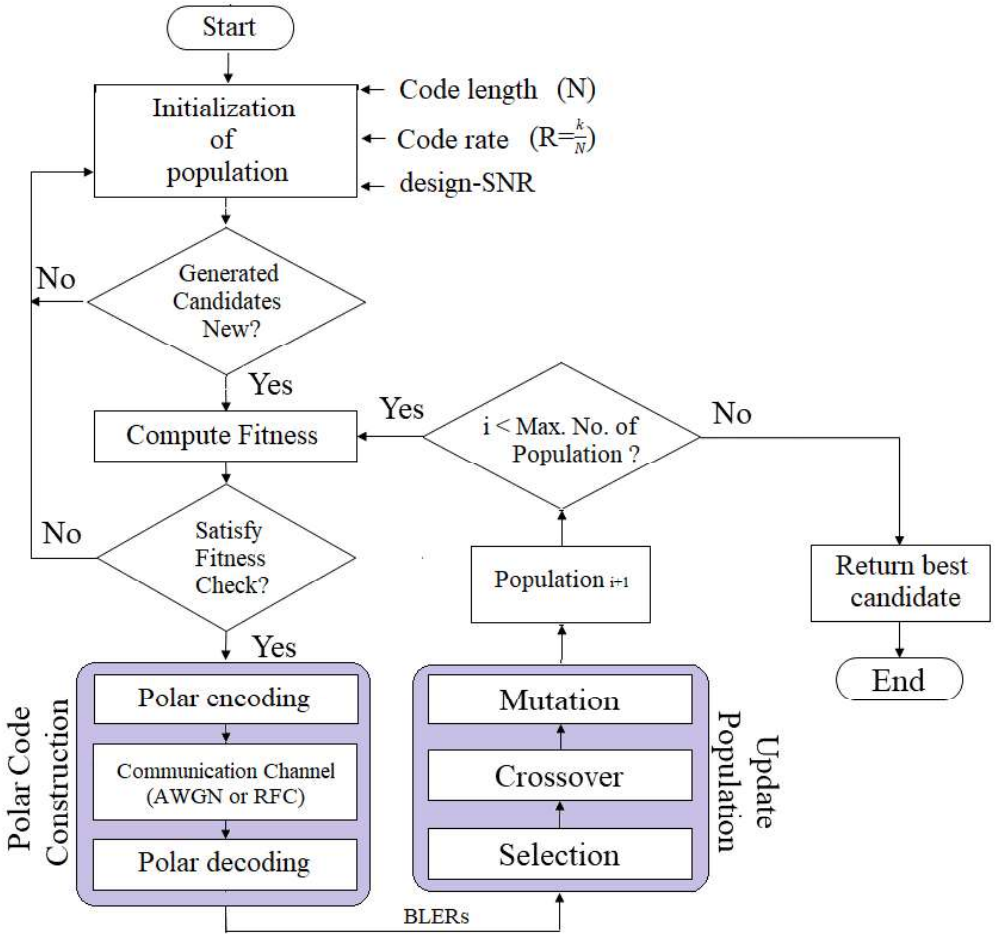


Figure 4.1 Diagram illustrating the GA-based proposed polar code design.

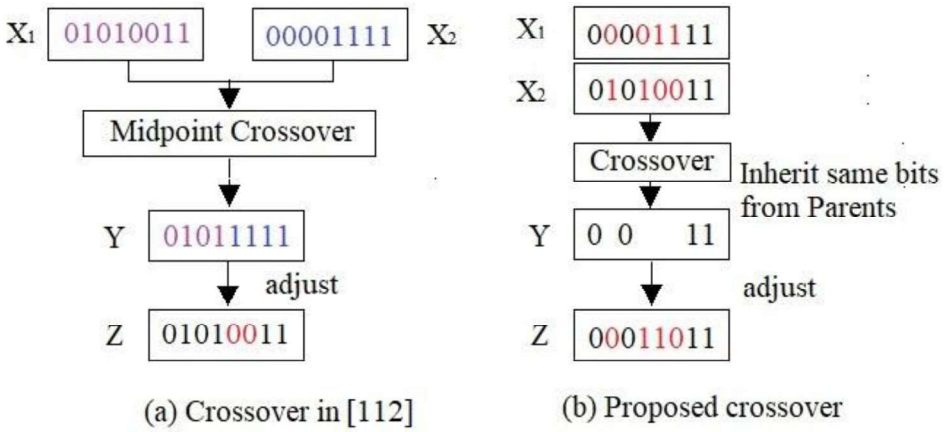


Figure 4.2 Diagram illustrating comparison between the crossover used in prior work [112] and the proposed crossover.

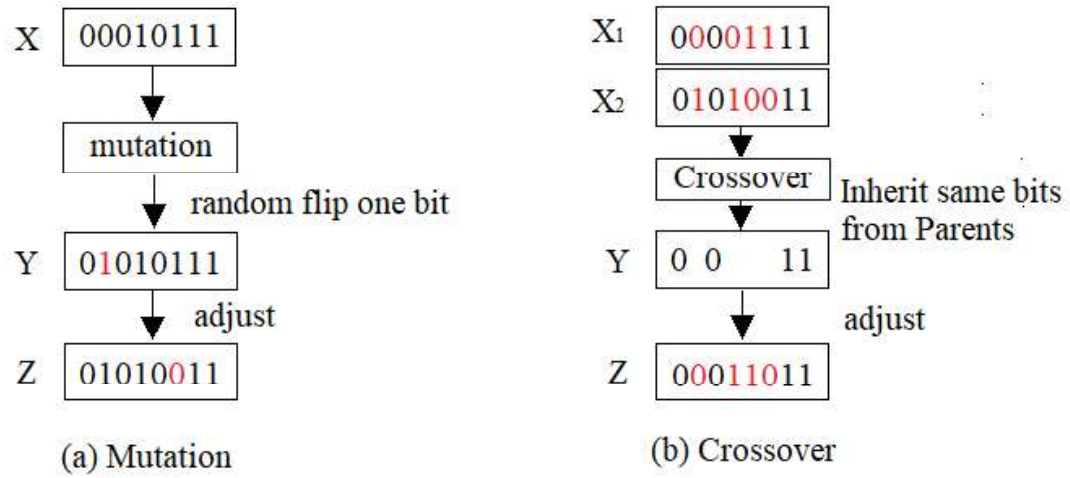


Figure 4.3 Diagram illustrating the examples of random mutation and crossover implemented in this work.

Two important issues are solved in our proposed design compared to the existing GA based polar code construction system. The issues are as follows:

(i) As new candidates are continually generated, recalculating the fitness value of a candidate that has appeared in previous populations becomes redundant and inefficient. So, repeatedly computing the fitness values of candidates unnecessarily increases computational complexity. In the proposed design, when a new candidate is generated, the scheme checks past candidates until a non-identical one is found, as shown in Figure 4.1.

(ii) The existing crossover rules are not optimal and may not be suitable for certain scenarios due to the significant randomness and limited hereditary influences. The crossover process may result in the loss of key characteristics inherited from the parents, potentially leading to poor performance in the offspring. Our proposed design ensures that the resulting crossover maintains the same values inherited from both parent candidates, thereby enhancing heritability. The effectiveness of the steps may depend on prior knowledge of the initial candidates.

The examples of the existing crossover [112] and proposed crossover are shown in Figure 4.2 (a) and (b), respectively. Also, the examples of the random mutation and proposed crossover are shown in Figure 4.3 (a) and (b), correspondingly.

Advantages of GA-Based polar code design: The use of GAs in polar code design offers several advantages:

(i) Optimization of Frozen Bits: GAs can effectively explore the search space for optimal frozen bit configurations, leading to improved error performance.

(ii) Reduced Complexity: By optimizing the design process, GAs can reduce the computational complexity associated with traditional polar code construction methods.

(iii) Flexibility: GAs can be adapted to various decoding schemes, such as SCL and BP, allowing for tailored solutions based on specific channel conditions.

4.4 Performance analysis of GA-based polar codes

The performance of the proposed GA-based polar code design is evaluated through simulations conducted over the AWGN and RFC channels. The simulation employs polar codes with specific parameters, where N (block length) is set to 1024, k (number of information bits) is 512, and R (code rate) is 0.5.The simulations compare the BLER performance of the GA-designed codes against existing polar code constructions. These parameters provide the basis for evaluating the performance and effectiveness of the designed polar codes in both channel environments.

4.4.1 Outcomes for AWGN channel

In Figure 4.4, a comparison of the error-rates of codes developed using various schemes, as described in references [83-84, 95, 112], and a code generated through a proposed GA-based approach employing BP decoding with a 200 iteration limit, is presented. At a BLER of 10^{-4}, the proposed architecture exhibits a coding gain of approximately 0.3dB to 1.2dB in comparison to all other design schemes [83-84, 95, 113], with the exception of the one outlined in [112]. This performance indicates that both our design and the one in [112] leverage GA to find an optimal solution over a wider range of Eb/No ratio. Additionally, our proposed design, employing GA and BP decoding, demonstrates superior performance in error-rates in contrast to other polar code designs [83-84, 95, 113] employing BPL or CA-BPL decoding.

Figure 4.5 compares the error rates of codes developed using our proposed design based on GA versus codes constructed as described in references [79, 104, 112, 114]. This time, the code is decoded using SCL decoding with L of 32. Compared to other design schemes mentioned in [79, 104, 112, 114], our method achieves a coding gain of approximately 0.2 dB to 1.8 dB at a BLER of 10^{-5}. Furthermore, our proposed design outperforms those in [112, 114] using CA-SCL decoding. Additionally, it outperforms in error-rates when compared to previous polar code designs [79, 112] employing SCL and the design [104] employing CA-SCL decoding.

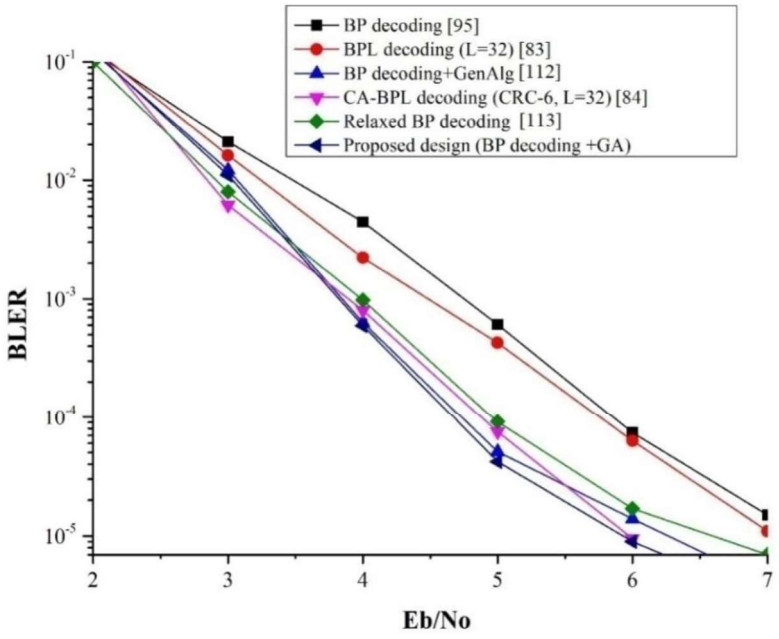


Figure 4.4 BLER vs. E_b/N_o plot for P(1024, 512) in AWGN channel with BP, BPL and CA-BPL decoding.

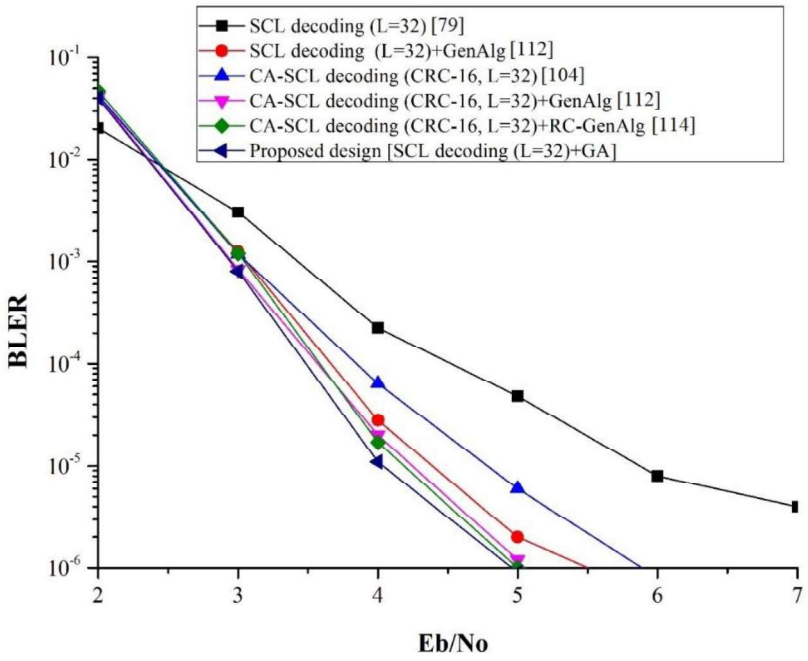


Figure 4.5 BLER vs. E_b/N_o plot for P(1024, 512) in AWGN channel with SCL and CA-SCL decoding.

Overall, our GA-based construction strategy, employing SCL decoding without cyclic redundancy check (CRC), outperforms the polar code using CA-SCL [104] and RC-GenAlg [114], with R=0.5 and L=32. Comparing the polar code employing CA-BPL [104] and Relaxed BP decoding [113] with R=0.5 and 200 iteration limit, the GA-based construction using BP decoding (with no list) performs brilliantly. Compared to other designs described in [104, 84, 113], our methods offer a coding gain of at least 0.5 dB at a BLER of 10^{-4}. These results demonstrate that the polar codes designed through our GA-based approach for the AWGN channel utilizing both BP and SCL decoding, outperform existing design schemes with respect to error-rates.

Table 4.1 $\frac{E_b}{N_o}$ comparison in polar code with BP and BPL decoding to achieve a BLER of 10^{-4} for P(1024, 512) in AWGN channel with a fixed design-SNR of 5dB during code design

Design scheme	**$\frac{E_b}{N_o}$ (BP or BPL decoding) at BLER of 10^{-4}**	**Improv. (%) of $\frac{E_b}{N_o}$ in Proposed design**
BP decoding [95]	5.85dB	20.17
BPL (L=32) [83]	5.80dB	19.48
BP + GenAlg [112]	4.75dB	01.68
CA-BPL (CRC-6, L=32) [84]	4.95dB	05.65
Relaxed BP decoding [113]	5.00dB	06.60
Proposed architecture (BP+GA)	**4.67dB**	--

Table 4.2 $\frac{E_b}{N_o}$ comparison in polar code with SCL and CA-SCL decoding to achieve a BLER of 10^{-5} for P(1024, 512) in AWGN channel with a fixed design-SNR of 5dB during code design

Design scheme	$\frac{E_b}{N_o}$ **(SCL or CA-SCL decoding) at BLER of 10^{-5}**	**Improv. (%) of $\frac{E_b}{N_o}$ in Proposed design**
SCL (L=32) [79]	5.81dB	30.98
SCL (L=32) + GenAlg [112]	4.39dB	08.65
CA-SCL (CRC-16, L=32) [104]	4.73dB	15.22
CA-SCL (CRC-16, L=32) + GenAlg [112]	4.31dB	06.96
CA-SCL (CRC-16, L=32) + RC-GenAlg [114]	4.22dB	04.97
Proposed construction [SCL (L=32) + GA]	**4.01dB**	--

Table 4.1 and Table 4.2 provide insights into the $\frac{E_b}{N_o}$ requirements during both coding and decoding processes to achieve a fixed BLER. Our proposed architecture, which utilizes BP decoding in conjunction with GA, is the most effective way to construct polar codes, as shown in Table 4.1. This method demands the lowest $\frac{E_b}{N_o}$ to reach a BLER of 10^{-4}. Similarly, Table 4.2 verifies that our proposed approach employing SCL decoding with GA, which offers the lowest $\frac{E_b}{N_o}$ to achieve a BLER of 10^{-5}. Therefore, our research showcases the most effective polar code construction strategy, capable of achieving a target BLER with the minimum required $\frac{E_b}{N_o}$.

4.4.2 Outcomes for RFC

Figure 4.6 presents a comparison of error-rates among the codes developed using our proposed methodology and the codes constructed using various schemes described in [78, 104, 112, 120]. With a 200 iteration limit, this code is decoded using BP decoding. At a BLER of 10^{-4}, our design demonstrates a coding gain of approximately 0.2 dB to 1 dB compared to all other design schemes [78, 104, 112, 120]. This performance suggests that both our design and the one in [112] leverage GA in order to generate the optimal solution over a broader range of Eb/No. Furthermore, our proposed design, employing GA, performs better at reducing error rates than other polar code designs [78, 104, 112, 120] that use BP decoding.

In Figure 4.7, we depict a comparison of error rates among the codes generated by our proposed approach and the codes constructed using the methods described in [78, 104, 112, 117, 119]. This time, the code is decoded using SCL decoding with L of 32. Our architecture achieves a coding gain of approximately 0.1 dB to 2 dB when compared to all other design schemes mentioned in [78, 104, 112, 117, 119] at a BLER of 10^{-4}. Additionally, our proposed design outperforms the one in [112], which employs CA-SCL decoding. It also demonstrates superior performance in terms of error rates in comparison to other polar code designs utilizing CA-SCL [104] and SCL decoding [78, 104, 112].

Overall, the proposed architecture, employing SCL decoding without a cyclic redundancy check (CRC), exhibits superior performance compared to [112], where both employ L of 32 and R of 0.5. Also, the code design based on GA and BP decoding works well in comparison to [112], where both utilize a maximum of 200 iterations and have an R of 0.5. Our designs provide a coding gain of atleast 0.1 dB to 0.2 dB compared to other design schemes outlined in [112] at a BLER of 10^{-4}. These results show the enhanced performance in error rates of the polar codes structured employing our methodology based on GA along with BP or SCL decoding, compared to existing design schemes for RFC.

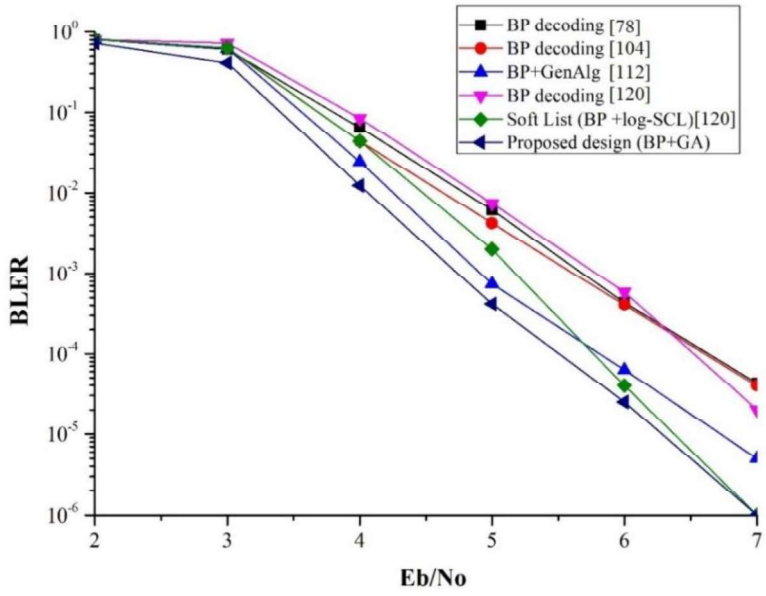


Figure 4.6 BLER vs. E_b/N_o plot for P(1024, 512) in Rayleigh fading channel with BP decoding.

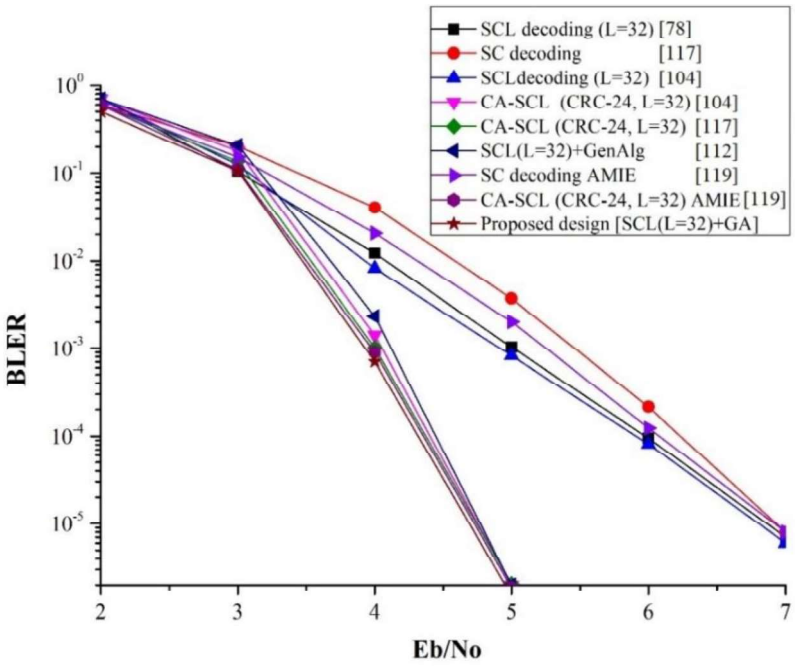


Figure 4.7 BLER vs. E_b/N_o plot for P(1024, 512) in Rayleigh fading channel with SC, SCL and CA-SCL decoding.

Table 4.3 and Table 4.4 provide insights into the $\frac{E_b}{N_o}$ requirements during both coding as well as decoding processes to achieve a desired BLER. Our suggested architecture, which utilizes BP decoding in conjunction with GA, is the most effective way to design polar codes with the lowest $\frac{E_b}{N_o}$to reach a BLER of 10^{-4} as shown in Table 4.3. Similarly, Table 4.4 validates the proposed methodology based on SCL decoding with GA, as the best designed polar code, as it necessitates the lowest $\frac{E_b}{N_o}$to achieve a BLER of 10^{-4}. Therefore, our research showcases the most effective design of polar codes, capable of achieving a target BLER with the minimum required $\frac{E_b}{N_o}$.

Table 4.3 $\frac{E_b}{N_o}$ comparison in polar code with BP decoding to achieve a BLER of 10^{-4} for P(1024, 512) in RFC with a fixed design-SNR of 5dB during code construction

Design scheme	$\frac{E_b}{N_o}$ **(Decoding: BP) at BLER of** 10^{-4}	**Improv. (%) of** $\frac{E_b}{N_o}$ **in Proposed design**
BP decoding [78]	6.71dB	17.28
BP decoding [104]	6.65dB	16.54
BP + GenAlg [112]	5.85dB	05.12
BP decoding [120]	6.60dB	15.90
Soft-List (BP + log-SCL) [120]	6.81dB	18.50
Proposed design (BP + GA)	**5.55dB**	--

Table 4.4 $\frac{E_b}{N_o}$ comparison in polar code with SCL or CA-SCL decoding to achieve a BLER of 10^{-4} for P(1024, 512) in RFC with a fixed design-SNR of 5dB during code construction

Design scheme	$\frac{E_b}{N_o}$ **(Decoding: SC or SCL or CA-SCL) at BLER of** 10^{-4}	**Improv. (%) of** $\frac{E_b}{N_o}$ **in Proposed design**
SCL (L=32) [78]	6.11dB	27.86
SC decoding [117]	6.31dB	30.15
SCL (L=32) [104]	6.02dB	26.74
CA-SCL (CRC-24, L=32) [104]	4.55dB	03.07
SCL (L=32) + GenAlg [112]	4.62dB	04.54
CA-SCL (CRC-24, L=32) [117]	4.51dB	02.21
SC decoding AMIE [119]	6.22dB	29.09
CA-SCL (CRC-24, L=32) AMIE [119]	4.45dB	00.89
Proposed structure [SCL (L=32) + GA]	**4.41dB**	--

4.4.3 Complexity analysis and computation time reduction

The GA-based approach also shows a reduction in computational complexity and decoding latency. The proposed designs minimize the number of iterations required for decoding, thereby enhancing the overall efficiency of the communication system. Our suggested design concept for the polar code, leveraging GA, possesses the capability to reach a target BLER with the minimum required Eb/No, all without the need for additional CRC. While it's true that GA introduces some design complexity, the proposed designs significantly reduce computational complexity and computation time in contrast to the present design concepts for polar codes described in references [78, 83 - 84,104,112,117, 119]. On the i-th iteration, the comparison of computational complexity in [112, 114] and the proposed approaches are shown in Table 4.5.

Table 4.5 On the i-th iteration, the comparison of computational complexities of designs in [112, 114] and proposed design.

Scheme	Computational Complexity	Level of Complexity
SCL + GenAlg [112]	O (C_rLNlog$_2$N) where C_r= candidates repeatedly selected from the 1st to the i-th populations, L = list size and N = Encoded Block size.	High
CA-SCL+ RC-GenAlg [114]	O (i(S-T)N) where (S-T) = fresh contenders for every population (apart from the initial population)	Low
Proposed design (SCL + GA)	O (C_nLNlog$_2$N) where C_n = non-identical candidates selected from populationi+1	**Medium**

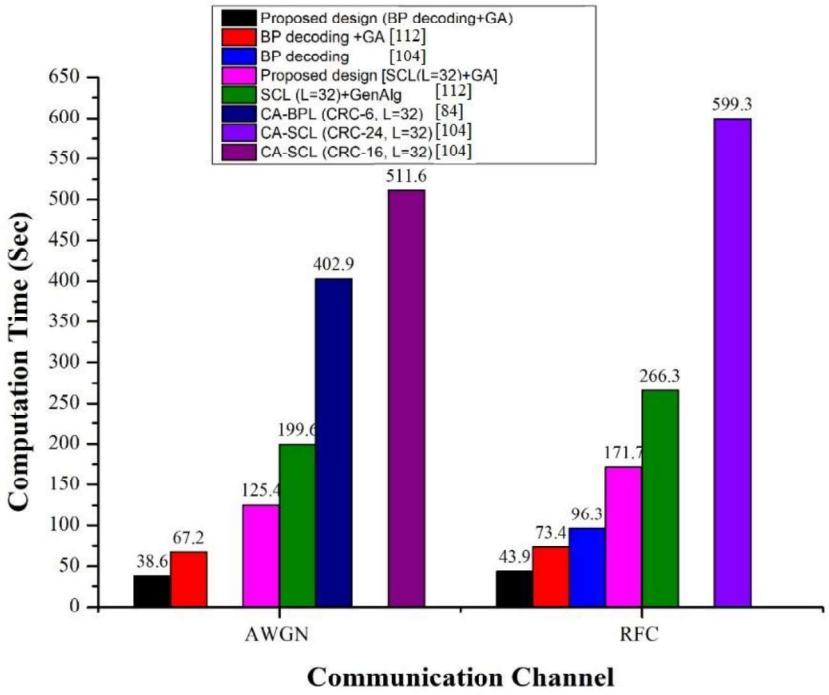


Figure 4.8 Comparison of computation time for proposed and existing polar codes.

Furthermore, a graphical comparison of computation time among different polar codes, including existing ones, is presented in Figure 4.8. GA accelerates the search process to

identify the best result, thereby reducing computation time and minimizing decoding latency. In comparison to existing ECC, our proposed codes employing GA strike a better balance between computation time and performance.

4.5 Summary

Finite length polar codes may not perform as effectively as other well-established coding techniques. To reduce the error-rate in practical decoding approaches, e.g., SCL and BP, we suggest a polar code design method that prioritizes the selection of the channel and decoding strategies. GA is employed to get an optimal result by adjusting the population size and the number of evolution steps. Our design yields improved outcomes by automatically adapting the A-vector, emphasizing the importance of appropriate polar code construction for both AWGN channel and RFC. In this research, we demonstrate that lower error rates can be achieved using proposed GA-based BP decoding (without CRC or list) and SCL decoding (without CRC) in contrast to existing techniques such as CA-SCL as well as CA-BPL decoding correspondingly. Furthermore, GA is employed along with polar code construction to minimize the gap between traditional SCL and iterative BP decoding performances, while also reducing computation time and complexity. Thus, GA optimization enables the attainment of a target error rate with lesser BP iterations or a reduced list size.

The construction of efficient polar codes using GA presents a promising avenue for enhancing error correction in modern communication systems. By optimizing the selection of frozen bits and leveraging the evolutionary principles of GAs, the proposed designs achieve superior performance in terms of error rates and computational efficiency. The design of ECC schemes with higher error correction capabilities like Golay and polar codes have been reported in the next chapter for Physical Layer Security (PLS) application.

Chapter 5

Design and Performance Evaluation of Concatenated polar coding schemes for WSNs and 5G NR control channels

5.1 Introduction

The use of portable wireless devices with better QoS (Quality of Service) and lower latency web connectivity have increased massively in recent times. The rising need of high speed data, video and message traffic accelerates the advancement and conversion of 4G to 5G NR (New Radio) mobile technology. It is expected that future wireless systems will achieve remarkable progress in range of usage states, traffic capacity and connectivity. This remarkable progress also draws a key challenge of viable advancement of enhanced system capabilities like spectral energy, system operative and cost efficiencies. Although channel coding is used for error detection and correction in the presence of noise, fading, and interference in wireless systems, it is still challenging to guarantee perfectly reliable communication under all channel conditions. Channel coding of user data and control information, both can be measured distinctly by 3GPP NR standard. Instead of turbo and TBCC (Tail-Bit Convolutional Codes) of existing 4G technology, polar and LDPC codes are carefully chosen for NR control and data channels correspondingly. Polar codes have been integrated into the 5G NR for controlling channels in both the Enhanced Mobile Broadband (eMBB) and the Ultra-Reliable Low-Latency Communication (URLLC) use-cases. Additionally, they have been recognized as prospective options for the data and control channels within the massive Machine Type Communication (mMTC) use-cases. It is expected that capacity attaining channel coding scheme like polar codes are capable of finest error correction along with low FAR and low power consumption at short block lengths for control information.

5G technology is poised to offer impressive performance metrics, including a peak data rate exceeding 10 Gbps, cell-edge rates up to 1 Gbps, and latency as low as 1 ms. Achieving such ambitious goals requires significant advancements in various aspects of the physical layer, with channel coding being a critical focus area. Several coding schemes have been thoroughly examined from both theoretical and practical standpoints to meet these demanding requirements. Ultimately, polar coding emerged as the chosen channel coding scheme for 5G eMBB control channels. Despite its attractive characteristics, polar coding necessitates meticulous design to ensure alignment with 5G NR specifications. One approach to address complexity and latency concerns involves early termination of the decoding process if the user equipment detects that it's not the intended recipient of the code block under scrutiny. Moreover, studies have shown that the utilization of tree-pruning techniques in decoding can improve polar code performances in terms of BLER when compared to conventional methods employing CRC-aided list decoding. Consequently, there is a strong demand for a design that can support both early termination and tree-pruning decoding

approaches, while also aiming to reduce complexity to a minimum. The 5G polar code design criteria are presented in Figure 5.1.

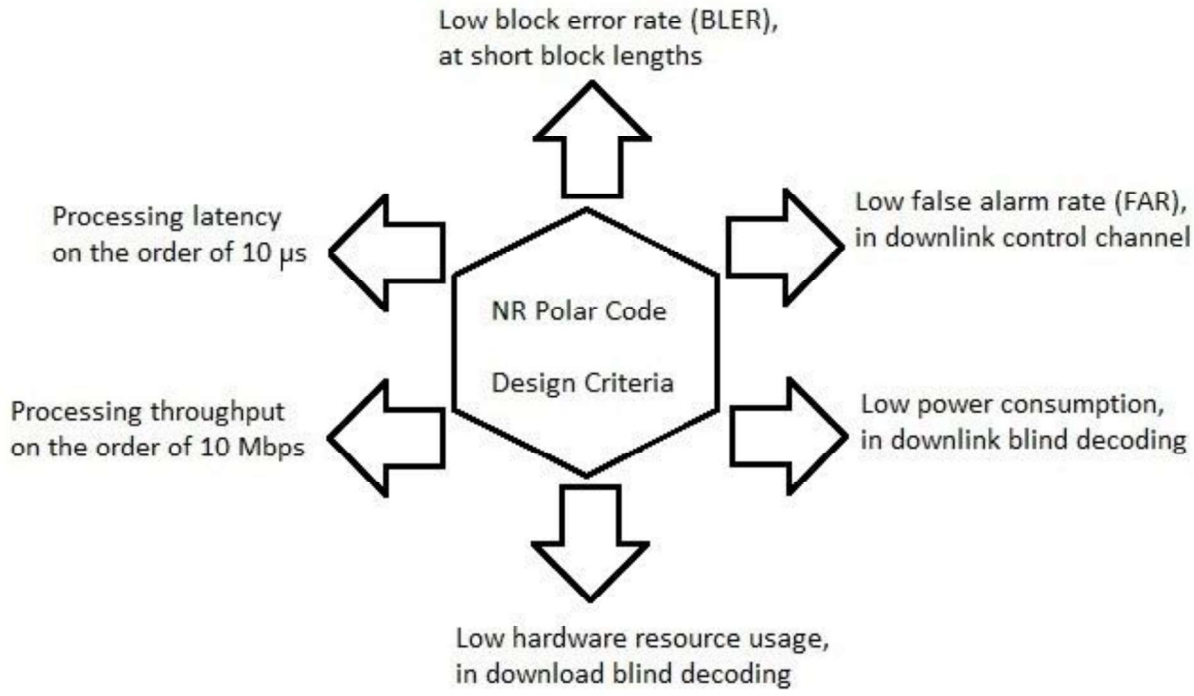


Figure 5.1 NR polar code design criteria.

The advent and commercialization of 5G wireless technology have made the Internet of Things (IoT) a popular research area in future communications. The energy-constrained networks such as WSNs play a big role in IoT. IoT uses a cloud server to communicate with actual physical objects using internet. A WSN is a collection of sensors used in wireless communication systems to cooperate, detect, interact, and process the desired information within its network coverage. As the sensor nodes often allow to consume low power, overcoming the energy constraints in sensor networks is a challenging issue. As a result, the networks are structured considering energy consumption and link reliability. In WSNs, Automatic Repeat Request helps to improve link reliability throughout the transmission if errors happen. Yet, sending the same information block over and over uses a lot of energy. Unfavorable situations arise on account of the poor channel quality, leading to repeated retransmissions and significant reduction of throughput. The categories of channels and noise have an impact on reliability of the received data. The Bit Error Rate (BER) of the received information can be reduced by using ECC, which also decreases the need for retransmissions in WSNs. In order to minimize the error of the received signal, various kinds of error detection and correction schemes are used to achieve reliable communication in WSNs.

Advantages of polar coding schemes for resource-constrained applications: Polar codes offer several advantages, particularly in resource-constrained environments:
(i) Capacity-Achieving: They can asymptotically achieve channel capacity, making them suitable for high-performance applications.
(ii) Low Complexity: The encoding and decoding processes can be implemented with low computational complexity, which is crucial for devices with limited processing power.
(iii) Flexibility: Polar codes can be adapted to various block lengths and rates, allowing for customization based on specific application requirements.

In the realm of modern communication systems, particularly in the context of the Internet of Things (IoT) and 5G networks, the demand for efficient and reliable data transmission has surged. Polar codes, a class of error-correcting codes, have emerged as a promising solution due to their capacity-achieving properties and relatively low complexity. This chapter explores various polar coding schemes tailored for WSNs and 5G NR control channels, focusing on their design, performance, and practical implications.

5.2 Related existing works

The 3GPP NR technical report [104] delivers encoding specifications of LDPC and polar codes in detail whereas decoding processes are not clearly specified for 5G NR. Though the 5G polar codes are mentioned in [105] but polar coding components and decoding operations are not presented in detail. Further advances of NR polar codes are specified in [123 – 126]. Hash-polar codes are proposed in [124] for performance improvement over Parity Check (PC)-polar codes under specific FAR (False Alarm Rate). In [125], an enhanced blind detection structural design for downlink physical control channel is offered. A logarithmic stack polar decoding with low complexity has been recommended for 5G URLLC in [126]. The performance and development of the 5G polar codes are discussed briefly in [39]. Distributed CRC is used to improve both error performance of SCL decoding and decoding time by early termination as presented in [127]. None of these contributions offer systematic clarification of the components and decoding of polar codes for specific NR control channels.

Previous works on ECC for WSNs mostly concentrated on codes like convolutional [44], BCH [44] and RS codes [128]. ECC techniques can considerably increase energy efficiency in WSNs, despite of the energy consumption by the encoding and decoding methods. ECC's iterative decoding algorithm is capable of both resolving the hot-spot issue as well as extending the network lifetime. The iterative nature of decoding algorithms, combined with their ability to exchange soft information regarding reliability, collaborate between decoding stages to collectively address errors, adapt under varying conditions, and perform multiple iterations, enables them to effectively address the hot-spot issue. Application of the various coding methods can improve link reliability while decreasing transmitting energy. However, not all coding techniques are appropriate for WSNs. Consequently, it is desirable to have a coding method that can boost link reliability while using a convenient amount of decoding power.

Historically, Golay codes have demonstrated robust error correction capabilities, while polar codes are renowned for their theoretical optimality in attaining channel capacity. These characteristics make polar codes widely applicable in contemporary communication systems, valued for their efficiency and reliability. Arikan's polar codes can theoretically attain the Shannon limit with relatively lower encoder and decoder complexity [78]. The construction of polar codes with finite length for practical applications is presently the key research area even if polar codes can attain channel capacity theoretically as the first coding method. Polar code is now a coding technique used for 5G control channels. The extensive use of polar codes in 5G wireless technology has taken centre stage to implement as a favourable ECC in IoT system design. IoT systems suffer from burst and random errors. However, polar codes have several limitations, for instance, poor error performance when the code lengths are finite (small to moderate). The following two methods can be used to resolve the aforementioned issues. One option is to construct a decoder that is more sophisticated than the SC decoder, or to alter the polar code design, such as by creating concatenated polar codes. For polar decoding, the SCL decoder using a high list size outperforms the SC decoder in terms of BER performance. Nonetheless, the complexity of decoding rises quickly as the list size rises [79].

The existing research on polar codes explores advanced decoding techniques to enhance performance under various conditions, particularly focusing on computational efficiency and error correction capabilities. A novel polar coding scheme for higher-order modulation that employs sign-bit shaping to reduce shaping loss in AWGN channel, is proposed in [129]. Despite the advancements, the literature lacks comprehensive studies on the long-term performance of the code under varying channel conditions and modulation orders, indicating a need for further research in these areas. The authors in [130] present a hybrid early stopping criterion that combines parity-check and G-matrix methods to intelligently terminate the belief propagation (BP) decoding process, achieving a substantial reduction in computational complexity while maintaining error correction performance. Overall, these works contribute to the ongoing development of polar codes, highlighting the need for innovative approaches to enhance their practical applicability in varying channel conditions. Further research could explore the scalability of concatenated polar codes and its performance in modern communication systems. The authors in [131] show that concatenated LDPC-Polar codes are built using outer LDPC codes and they outperformed concatenated RS-polar codes. In [132], concatenated turbo polar convolutional codes utilize systematic polar codes along with recursive systematic convolutional codes and the proposed model outperforms existing turbo polar codes, especially in low SNR regions. Further research could focus on mitigating the error floor problem in high SNR region, possibly through concatenated approaches that combine different coding strategies.

It is found that interleaving has the ability to split up burst errors into separate random errors. The outer binary BCH codes and convolutional codes are concatenated individually with inner polar code in an interleaved fashion, which are presented in [44]. According to [44], polar codes using SC and BP decoding perform noticeably worse than concatenated polar codes using outer convolutional codes. A refined RS-polar concatenated approach has been introduced in [128]. However, the performance improvement is limited due to the non-binary nature of the outer RS codes used. In comparison to RS-polar codes and long polar codes, BCH-polar [133] and convolutional-polar codes perform better. In [134], an interleaved LDPC-polar code is proposed and shown that it outperforms RS-polar codes. Authors in [135] propose concatenated polar codes and BER performance is assessed using various decoding algorithms and interleaving techniques. In [136], concatenated polar codes for visible light communication (VLC) systems are proposed using interleaving and performance under burst errors is verified compared with concatenated convolutional-polar codes. According to earlier studies, short-length concatenated polar codes are incomparable to existing cutting-edge codes. It is expected that using lower dimension code length, performance will be poor compared to long code length ECC. In IoT applications, almost no significant study has been done on the use of concatenated polar codes till so far.

5.3 Design and Implementation of Distributed CRC-Aided polar codes (DCA-polar) in 5G NR control channels

CRC codes can be integrated with polar codes to enhance their error-correction capabilities. This combination, known as CRC-Aided polar Codes (CA-polar), improves the performance of SC decoding by providing additional error detection. The use of CRC bits allows for early termination of the decoding process, reducing latency and energy consumption, which is particularly beneficial for mobile applications. An advanced variant of CA-polar codes is the Distributed CRC-Aided polar (DCA-polar) codes. This scheme distributes CRC bits throughout the information bits, allowing for more efficient error detection and correction. The DCA-polar codes have shown significant performance improvements in terms of false alarm rates (FAR) and BLER in uplink and downlink control channels of 5G networks.

5.3.1 Selection of CRC polynomials

CRC is significant in identifying the accurate candidate during SCL decoding of polar codes, and it's crucial to choose a CRC length that optimizes error-correction performance. To evaluate the impact of CRC on error correction performance within SCL decoding of polar codes, we utilize list sizes $L \in \{8, 16\}$ and CRC polynomials as per 5G standard. The CRC polynomials adopted in our work are the following:

$$z_{11}(g) = [g^{11} + g^{10} + g^{9} + g^{5} + 1] \quad (5.1)$$

$$z_{24}(g) = [g^{24}+g^{23}+g^{21}+g^{20}+g^{17}+g^{15}+g^{13}+g^{12}+g^{8}+g^{4}+g^{2}+g+1] \quad (5.2)$$

The polynomial $z_{24}(g)$ is used for the payload in PBCH and DCI in the PDCCH, whereas $z_{11}(g)$ is used for UCI in the PUCCH. The different parameters used to construct DCA-polar codes are presented in Table 5.1.

For uplink in 5G polar codes, feasible code lengths are 2^n and range is $5 \leq n \leq 10$, whereas for downlink, the applicable range is $7 \leq n \leq 9$.

Table 5.1 The different parameters used to construct DCA-polar codes.

NR Physical channels	Information block lengths	Encoded block lengths	CRC bits	CRC generator polynomials	ECC
PUCCH	$A \in [16, 1706]$	$E \in [A+11, 8192]$ or $E \in \{108, 216, 432, 864\}$	r = 11 bits	$z_{11}(g) = [g^{11} + g^{10} + g^{9} + g^{5} + 1]$	DCA-polar
PBCH	$A = 32$	$E = 864$	r = 24 bits	$z_{24}(g) = [g^{24}+g^{23}+g^{21}+g^{20}+g^{17}+g^{15}+g^{13}+g^{12}+g^{8}+g^{4}+g^{2}+g+1]$	
PDCCH	$A \in [16, 140]$	$E \in [A+24, 8192]$ or $E \in \{108, 216, 432, 864\}$			

5.3.2 Distributed CRC

To mitigate the considerable latency associated with SC decoding, 5G polar codes have introduced distributed CRC (D-CRC) as a solution. Unlike conventional methods where CRC bits are appended at the end of the information bits, D-CRC integrates CRC bits within the information bits. Importantly, each CRC bit solely relies on preceding information bits. Taking a CRC polynomial $z(g) = g^4 + g + 1$ as an example, its generator matrix is represented by the notation [I P], where P is given below for information length =10.

$$P = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \quad (5.3)$$

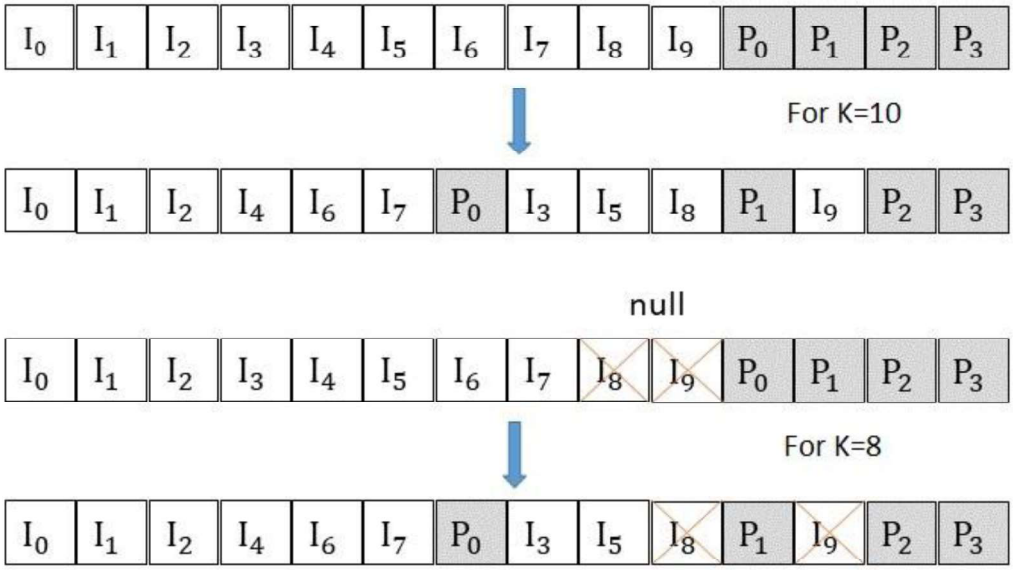


Figure 5.2 Illustration of D-CRC for K = 10, K = 8 and $z(g) = g^4 + g + 1$

The *i*-th CRC bit's dependence on the information bits is determined by the nonzero rows within the *i*-th column of *P*. For instance, the first CRC bit depends on 6 information bits. By introducing interleaving of the information bits, we can ensure that these 6 bits appear first, as depicted in Figure 5.2. This enables the distribution of CRC bits from any CRC polynomial. It's worth noting that the interleaving patterns may vary depending on the information length. However, if the pattern is designed for the longest information length, patterns for shorter lengths can be derived by considering null elements from the longer pattern. This strategy has been adopted by 5G polar codes, where the null elements depicted in Figure 5.2 adhere to this approach.

Let's consider how D-CRC can be used to decrease the latency of SC decoding. If any CRC check bit fails, the entire code block is considered faulty, allowing the decoder to stop decoding once such a failure occurs. This approach can be particularly useful for processing the physical downlink control channel (PDCCH) using polar codes, where there are many blind decoding attempts within a search space, with only a few candidates usually being valid. It's worth noting that early termination is possible only if all candidates in the list fail during SCL decoding.

The approach involves distributing CRC bits throughout the entire input information block instead of attaching them at the end. This can lead to better BLER performance, especially for shorter blocks. Additionally, this design enables early termination of decoding. By checking CRC bits distributed throughout, the polar decoder can halt early if all paths in the list fail. This capability has the potential to reduce decoding latency and energy consumption in hardware, crucial for mobile applications like 5G.

5.3.3 Distributed CRC-Aided polar (DCA-polar) codes

CRC distributed polar coding offers a versatile design, allowing for various decoding strategies. Its adaptability primarily arises from the flexibility to choose CRC bits for tree pruning. When prioritizing BLER performance, more CRC bits are employed for this purpose. Conversely, for a preference towards low undetected error probability (UEP), fewer

or no CRC bits are utilized for pruning. In such cases, the proposed scheme essentially resembles conventional CRC-aided list decoding, albeit with different CRC bit positions. Moreover, early termination can be seamlessly combined with either tree pruning or error detection to conserve power and reduce decoding delay.

Suppose $u = (u_1, \ldots, u_k)$ represents the information block to be transmitted. For CRC-aided list decoders, we append r CRC bits to u. Thus, the input to the encoder becomes $v = (u_1, \ldots, u_k, p_1, \ldots, p_r)$, where $p = (p_1, \ldots, p_r)$ represents the generated CRC bits. At the receiver, L decoding paths operate concurrently, and ideally, the surviving path would be the one passing the CRC check with the best path metric. In contrast to appending CRC bits at the end of the information block, in our new approach, the information and check bits are mixed together as $v = (v_1, \ldots, v_{k+r}) = (u_a, \ldots, p_i, \ldots, u_b, \ldots, p_j, \ldots, u_c)$. The distribution of CRC bits serves two purposes. Firstly, as demonstrated in the simulation results, it consistently enhances the BLER performance through tree pruning assistance. Secondly, it facilitates early termination.

Initially, let's employ a straightforward example to demonstrate the concept of CRC distribution. Assuming the CRC polynomial is $g^4 + g^2 + g + 1$, the corresponding generator matrix for the CRC code is *G*.

$$G = [\mathrm{I}|\mathrm{P}] = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix} \tag{5.4}$$

The information block along with CRC bits is represented as $uG = (u_1, u_2, u_3, u_4, u_5, p_1, p_2, p_3, p_4)$. It's important to note that the CRC bits depend on almost the entire input vector. As a result, conducting early tree pruning or termination isn't feasible because a partial CRC check can't be performed at the beginning of the information block. By rearranging rows and columns of G, we can modify the generator matrix as such:

$$G' = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.5}$$

With the modified generator matrix, the input to the polar encoder shifts from $(u_1, u_2, u_3, u_4, u_5, p_1, p_2, p_3, p_4)$ to $(u_4, u_3, p_1, u_5, p_2, u_2, p_4, u_1, p_3)$, allowing for early tree pruning or termination. Thus, adjustments are needed in both encoding and decoding processes. The updated encoding process involves these steps:

A. Generate and distribute CRC bits based on G'.

B. Provide the resulting vector v to the polar encoder.

For tree-pruning style list decoding, further modifications are required:

1. Recursively compute LLRs for all paths in the list.

2. If the current bit in v is a frozen or information bit, conduct normal list decoding.

3. When encountering a distributed CRC bit, compute its value for each path using previously decoded bits and the CRC polynomial.

4. Calculate the path metric, update the list, and proceed to the next bit in v.

5. Repeat step 1 until the last bit in v is reached; then, conduct a CRC check and select the surviving path.

It's important to note that step 2 penalizes paths with incorrect CRC bits, and step 4 prunes paths whose calculated CRC bit does not match the LLR value.

This example effectively demonstrates the concept of CRC distributed polar codes, yet it faces a significant challenge: the distribution pattern's dependence on specific values of k. In 5G NR, supporting a wide range of k values with fine granularity becomes impractical due to the need to store numerous distribution patterns in hardware. To tackle this issue, we propose a method that necessitates storing only a single pattern.

Polar code is chosen as the preferred 5G-NR coding scheme for UCI and DCI (uplink and downlink control information) [104]. For PDCCH DCIs and PBCH payloads, polar code aided with distributed CRC and input bit interleaver is facilitated. By bit interleaving between the CRC encoder and the polar encoder, distributed CRC bits are acquired. Moreover, after the last bit necessary for calculation, the CRC bit is placed [137]. Therefore, by early terminating the decoding process, decoding complexity is declined when improper check is met [125, 138], or else by trimming the decoding tree, error-correction is enriched as PC (Parity-Check) polar codes do [139]. The interleaver equally distributes the CRC bits within the information bits even though CRC remainder bits are fall upon after appropriate information bits during the decoding. By using this idea, DCA-polar decoder eases the decoding complexity implementing early termination of the decoding when each path met improper check. The decoder's performance is improved by pruning the SCL decoding tree using distributed CRC bits [139]. Figure 5.3 illustrates the DCA polar design process for 5G-NR. As previously mentioned, combining CRC distribution with tree-pruning style decoding enhances performance. Ideally, using a single interleaving/deinterleaving pattern for all supported block sizes would greatly simplify hardware. This is made feasible by the properties of CRC generator matrices, as shown in Figure 5.3, forming the basis of our proposed coding chain.

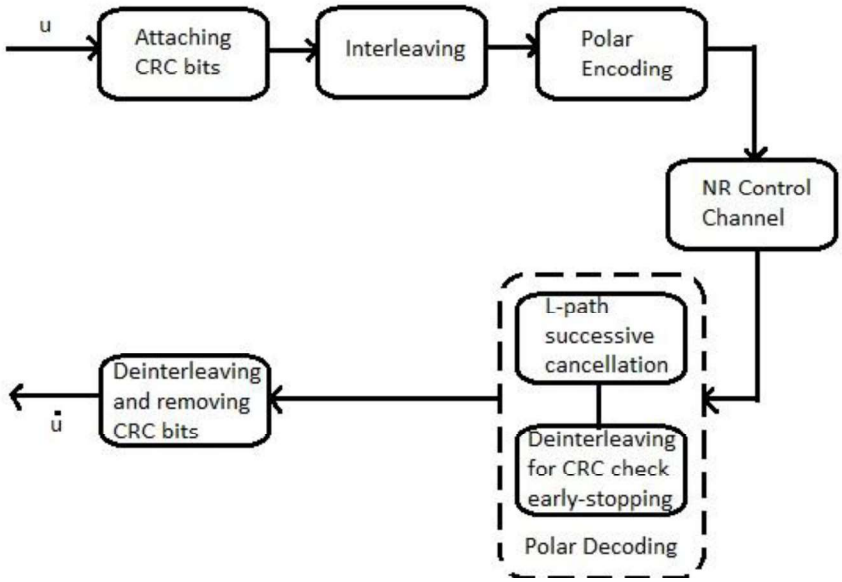


Figure 5.3 DCA polar design process for 5G-NR control channel.

Due to the successive decoding property of polar codes, it becomes feasible to halt the decoding process prematurely with distributed CRC bits, as some check bits are positioned at the beginning of the information block. During decoding, if all paths in the L-list fail the

CRC check, we can terminate the process and declare a decoding error. This is because the complete decoding process would ultimately result in failure. This process is referred to as early termination. Early termination can occur either at the first distributed CRC bit or after encountering multiple CRC bits.

For the first CRC bit, decoding halts and an error is declared immediately if all paths fail the CRC check. In the case of multi-bit early termination, all available distributed CRC bits are utilized for checking. Decoding continues only if at least one path satisfies the CRC check. If all paths contain at least one CRC bit error, the decoding process can be terminated. The entire flow chart is depicted in Figure 5.4. Early termination and tree pruning are not mutually exclusive; in fact, early termination can be performed concurrently with tree pruning if necessary. In such cases, if not all paths fail at a CRC bit, the decoder can proceed with tree pruning using the same CRC bit.

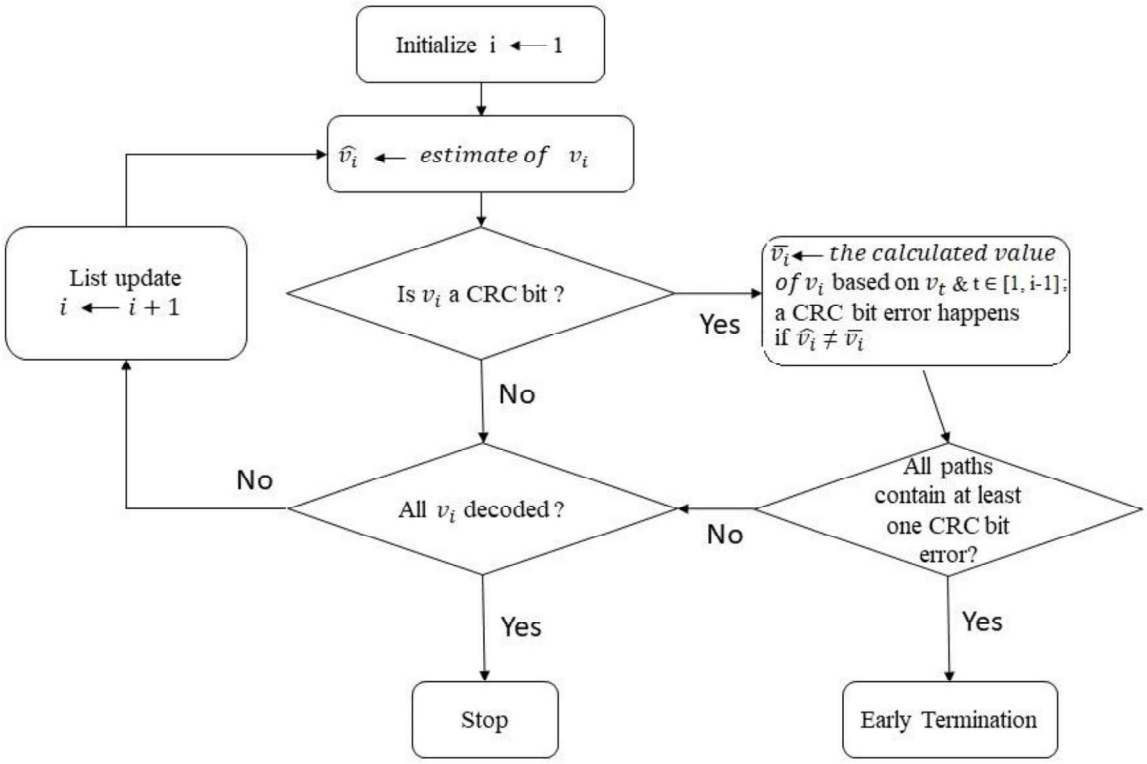


Figure 5.4 Flow chart of early termination.

The operation of polar coding components in 5G NR physical channels is presented in this section. In PUCCH channel, code block segmentation (CBS) is applied for long blocks during encoding and decoding. As the block lengths of information are restricted to $A \in [12, 140]$ for PDCCH channel and $A = 32$ for PBCH channel, CBS is prohibited to apply in both cases. CBS permits the polar code length restricted to $N = 1024$ bits, even though the largest required $A = 1706$ bits. The reduction in polar code complexity increases proportionally with $N \log N$. Code block segmentation also permits up to E= 2048 bits for $N = 1024$, deprived of depend on repetition, which worsens performance. During decoding, the decoded information blocks are concatenated and padding bits of the encoding process are removed to complement CBS.

To achieve error detection and correction in the decoder, individual information block is added with redundant CRC bits in the course of polar encoding. Polar decoder does CRC check deciding codeword is error free or not and pick a codeword from a list of decoding

candidates. The amounts of CRC bits necessary for error detection is mentioned as [-log_2(FAR)]. The list size L = 8 or number of CRC bits $log_2(L)$ = 3 is used to support the CA-SCL decoding with acceptable error decoding and correction complexity. To complete a CRC check in polar decoding, CRC generator collects decoded information bits resulting r CRC bits which is matched with the r decoded CRC bits. No errors are sensed and the CRC check is successful if the both are equal. Otherwise, the decoded information sequence along with CRC bits is inputted in the CRC generator attaining r-bit syndrome. No errors are identified and the CRC check is successful when the syndrome has r zero-valued bits.

Early termination is implemented in polar decoding for the PDCCH channel, and CRC bits are separated from the information bits through interleaving to reduce decoding complexity and extend the battery life of UE. Instead of retrieving CRC bits only at the completion of the decoding, the CRC bits are recovered by DCA-polar decoding during decoding and decoding is stopped as quickly as if the checking of CRC bits will fail [39]. For downlink channels, DCA-polar displays better performance in FAR compared to CA-polar. This achievement is reached by the interleaver and CRC bits are moved to further suitable locations.

CRC interleaving is applied by rearranging the order of bits after CRC attachment in PBCH encoding and after CRC scrambling in PDCCH encoding. NR polar code implements redundancy as PC and frozen bits in the information sequence and CRC bits to facilitate forward error correction. PC and frozen bits are implanted to rise input sequence length from K bits to N = {32, 64, 128, 256, 512, 1024} block length.

The rate matching process is decomposed in sub-block interleaving, bit selection and channel interleaving. The capability of error correction is provided mainly by the polar coding, whereas it is improved by the sub-block interleaving and the burst errors are dispersed by the channel interleaver. But, the channel interleaving is applied in PUCCH channel only, whereas avoided in PBCH and PDCCH channels. Figure 5.5 illustrates the DCA polar design process for 5G-NR.

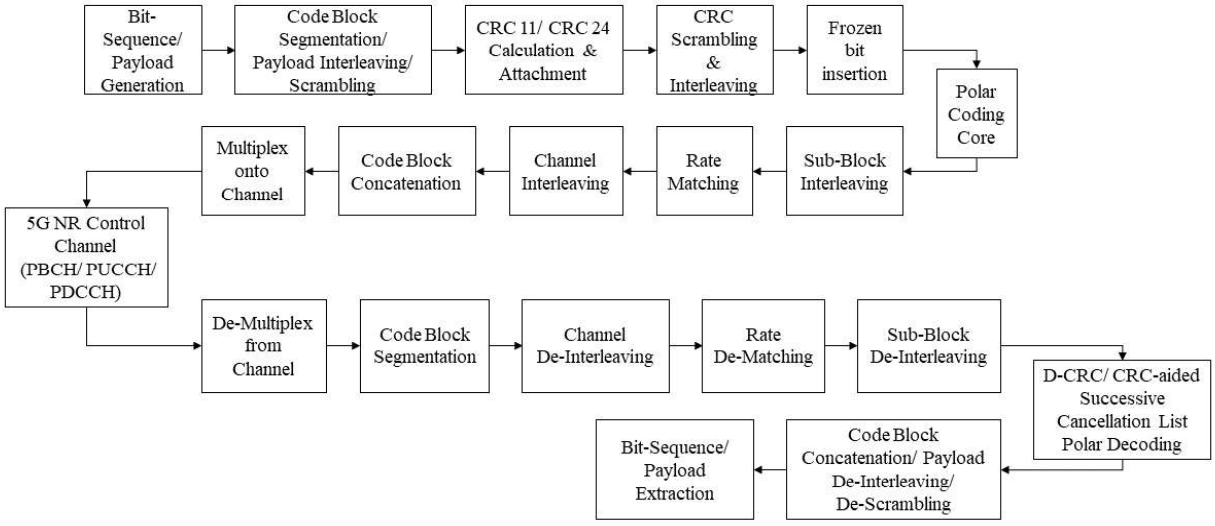


Figure 5.5 DCA polar encoding and decoding process in NR control channels.

5.4 Design and Implementation of Concatenated polar codes for WSNs

Concatenated polar codes involve combining polar codes with other coding schemes, such as BCH or LDPC codes, to enhance performance while maintaining low complexity. This approach allows

for the leveraging of the strengths of different coding techniques, resulting in improved error correction capabilities.

The existing works on the concatenated polar codes are presented in concise manner in Table 5.2. The existing works shows the ability of good error correction at the cost of high decoding complexity. None of the previous works had used suitable interleaving techniques along with conventional polar decoding to get better BER performance with moderate decoding complexity.

Table 5.2 Existing works on concatenated polar codes.

Existing work	Concatenated Codes	Decoding	Interleaving	Performance
Yu et al. [134]	LDPC-polar & RS-polar	SC & SCL	Not used	LDPC-polar codes can outperform RS-polar codes
Wang et al. [135]	BCH-polar, BCH-LDPC & BCH-turbo	SC & CA-SCL	used	BCH-polar codes outperform BCH-LDPC codes and perform worse than BCH-turbo codes. CA-SCL decoding complexity increases rapidly with larger list size.
Liu et al. [136]	BCH-polar & Convolutional-polar	SC	used	Convolutional-polar schemes offer better robustness than BCH-polar schemes and both effectively correct random errors and burst errors in VLC systems.

5.4.1 Interleaving Techniques

Interleaving is a crucial technique used in concatenated polar codes to mitigate burst errors. Two primary interleaving methods are employed:

(i) Interleaving Randomly (I_R): This method rearranges the bits in a pseudo-random manner, which helps to distribute errors more evenly across the codeword. Random interleaving is a process in which elements of a data sequence are rearranged in a pseudo-random manner to minimize the effect of errors in communication systems. The key characteristic of random interleaving is that it does not follow a fixed, deterministic pattern but instead uses a random permutation to rearrange the sequence. Let's denote the original sequence as:

$$x = [x_0, x_1, x_2, \dots\dots, x_{N-1}]$$

The random interleaved sequence y is generated by a random permutation π_R, which randomly reorders the indices of the sequence. Mathematically, this can be expressed as:

$$y = \left[x_{\pi_R(0)}, x_{\pi_R(1)}, x_{\pi_R(2)}, \dots\dots, x_{\pi_R(N-1)}\right] \tag{5.6}$$

where π_R is generated randomly, often using a pseudo-random number generator.

(ii) Interleaving Blindly (I_B): In this approach, a fixed interleaving pattern is used, which does not require synchronization between the transmitter and receiver. Blind interleaving is a technique often used in communication systems where the interleaving pattern is fixed or pre-determined without the receiver having explicit knowledge of the interleaving pattern during

decoding. This means that the data is interleaved blindly, without any adaptive adjustments based on the data content or error conditions. Given an input sequence:

$$x = [x_0, x_1, x_2, \dots\dots, x_{N-1}]$$

The blind interleaved sequence y is:

$$y = [x_{\pi_B(0)}, x_{\pi_B(1)}, x_{\pi_B(2)}, \dots\dots, x_{\pi_B(N-1)}] \quad (5.7)$$

where π_B is a predefined fixed permutation pattern that rearranges the elements of x and is known by both the transmitter and receiver before communication.

Both methods have been shown to significantly enhance the performance of concatenated polar codes, particularly in terms of BER.

5.4.2 Concatenated polar codes utilizing I_R Scheme

First, the input information bits are organized into M_b blocks of K_b bits per group at the transmitter. Subsequently, the BCH (outer-code) encoder separately encodes the K_b bits and produces a block of N_b bits. Next, BCH encoded block bits are restructured into a row vector U and interleaved randomly using random permutation with a specific state value to generate the interleaved vector V. Interleaved bits of V are divided into M_p blocks of K_p bits per group and the polar encoder (inner-code) encodes K_p bits to produce encoded vector W. The reverse process is followed in the receiver. The inner polar decoder decodes the received bits. Subsequently, the de-interleaving is performed by inverting a random permutation with a specific state value, opposite of the interleaving. The outer BCH decoder decodes the grouped de-interleaved bits. Finally, the output bits are acquired. The schematic diagram of proposed I_R scheme for concatenated codes is given in Figure 5.6 and it is described in Algorithm 5.1.

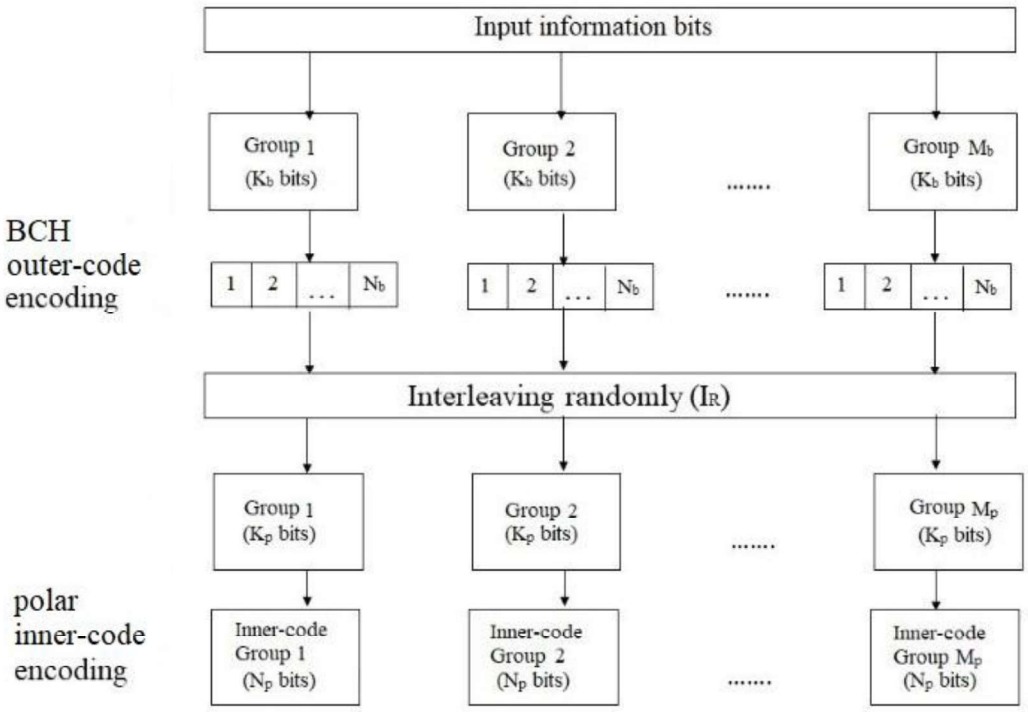


Figure 5.6 Schematic diagram of proposed I_R scheme for concatenated codes.

Algorithm 5.1 Encoding & Decoding for Random Interleaving (I_R) method

//**Encoding:**

//**Input:** u % message vector

//**Output:** C_{polar} % encoded polar codeword

1: C_{BCH} ⟵ BCH_encode(u) % encode the input message u using (31, 16) BCH Encoder

2: Generate π % generate a pseudorandom permutation sequence π.

3: C_{int} ⟵ π (C_{BCH}) % apply the permutation π to the BCH codeword $C_{BCH.}$

4: C_{polar}⟵ polar_encode(C_{int}) % encode the interleaved sequence C_{int} using the (32, 16) polar encoder

//**Decoding:**

//**Input:** r % received vector

//**Output:** $u^{'}$ % decoded message

1: $C_{int}^{'}$ ⟵ polar_decode(r) % decode the received vector r using the (32, 16) polar decoder

2: Generate π^{-1} % generate inverse pseudorandom permutation sequence π^{-1}

3: $C_{BCH}^{'}$⟵ $\pi^{-1}(C_{int}^{'})$ % apply the inverse permutation π^{-1} to the decoded sequence $C_{int}^{'}$

4: $u^{'}$ ⟵ BCH_decode($C_{BCH}^{'}$) % decode the deinterleaved sequence $C_{BCH}^{'}$ using (31, 16) BCH decoder

5.4.3 Concatenated polar Codes utilizing I_B Scheme

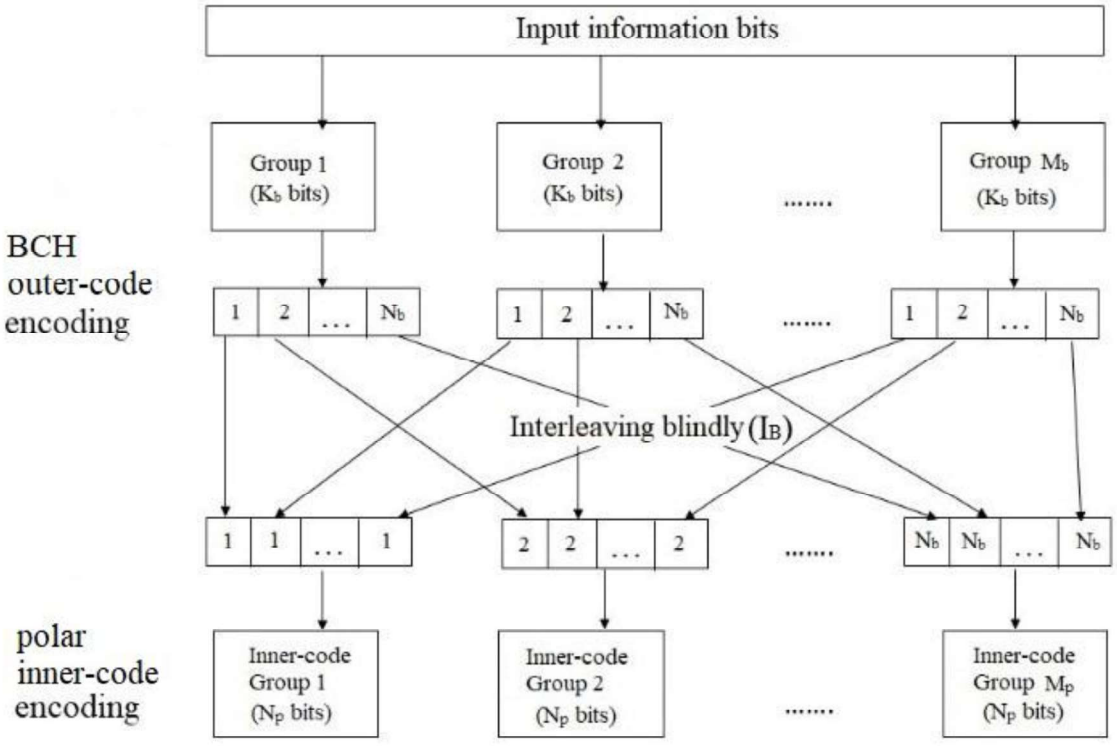


Figure 5.7 Schematic diagram of proposed I_B scheme for concatenated codes.

First, the input information bits are organized into M_b blocks of K_b bits per group at the transmitter, and each block is encoded by an outer BCH encoder into a block comprising N_b bits. Next, BCH encoded block bits are restructured into a row vector U. Then, a new interleaved vector V will be created by blindly interleaving the bits that are in the same place across various BCH coded blocks. Interleaved bits of V are divided into M_p blocks of K_p bits per group each and the inner polar encoder encodes K_p bits to produce codeword W. The key concept of I_B technique is to distribute entire bits of a BCH code block into various blocks before polar encoding. Thus, distinct polar code blocks may receive errors distributed by each BCH code blocks. The receiver has an entirely reverse process in compared to the transmitter. Inner polar decoding is followed by blind de-interleaving, which reorders the decoded bits back in the same order as the input bits. The outer BCH decoder then decodes the de-interleaved bits. Finally, corrected output bits are collected. The schematic diagram of proposed I_B scheme for concatenated codes is given in Figure 5.7 and it is described in Algorithm 5.2.

Algorithm 5.2 Encoding & Decoding for Blind Interleaving (I_B) method

//**Encoding:**

//**Input:** u % message vector

//**Output:** C_{polar} % encoded polar codeword

1: C_{BCH} ⟵ BCH_encode(u) % encode the input message u using (31, 16) BCH encoder

2: C_{int} ⟵ Blind_interleave(C_{BCH}) % rearrange the BCH codeword C_{BCH} according to a fixed blind pattern without requiring synchronization

3: C_{polar} ⟵ polar_encode(C_{int}) % encode the interleaved sequence C_{int} using the (32, 16) polar encoder

//**Decoding:**

//**Input:** r % received vector

//**Output:** u' % decoded message

1: C'_{int} ⟵ polar_decode(r) % decode the received vector r using the (32, 16) polar decoder

2: C'_{BCH} ⟵ $Blind\_deinterleave(C'_{int})$ % rearrange the decoded sequence C'_{int} according to the fixed blind pattern used during encoding

3: u' ⟵ BCH_decode(C'_{BCH}) % decode the deinterleaved sequence C'_{BCH} using (31, 16) BCH decoder

Figures 5.6 and 5.7 show a specific example where $M_b = M_p$; $N_b = 31$, $N_p = 32$ and $K_b = 16$, $K_p = 16$ wherever N_b and N_p stand for the code length, K_p and K_b stand for the information length of inner polar and outer BCH codes, correspondingly. So, the outer BCH code has a code rate of $R_{outer} = \frac{K_b}{N_b} = \frac{1}{2}$ and the inner polar code has a code rate of $R_{inner} = \frac{K_p}{N_p} = \frac{1}{2}$

Thus, the code rate of the proposed concatenated BCH-Polar code is

$$R_{concatenated} = R_{outer} \times R_{inner} = \frac{1}{4}$$

Systematic polar codes are employed in this work because they perform better than non-systematic variant. The power-efficient coding construction for distributed wireless sensing system is implemented in this work as presented in [140]. Despite the additional power and decoding complexity added by coding schemes, 28% energy savings can be attained in a WSN with 1000 sensor nodes and 10 super nodes. I_R and I_B schemes with an inner polar code and an outer BCH code are employed to design the concatenated BCH-polar codes. Concatenated BCH-turbo and BCH-LDPC codes can be constructed similarly replacing turbo or LDPC codes instead of the polar codes. The basic distinction between I_R and I_B methods is that the former uses a random permutation to randomly reorder the input block bits, whilst the later does it in accordance with predetermined principles. The schematic diagram of concatenated BCH-polar codes transmitter and receiver for IoT applications is shown in Figure 5.8.

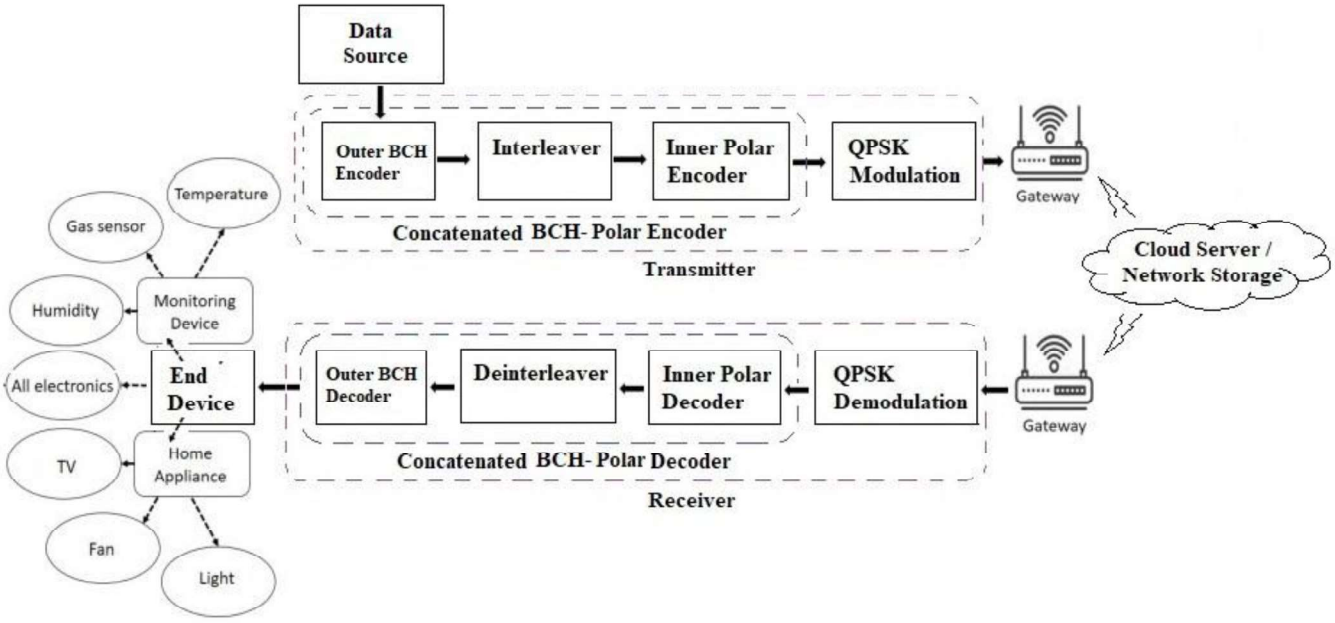


Figure 5.8 Schematic diagram of concatenated codes based transmitter and receiver for IoT applications.

5.5 Performance analysis

The performance of various polar coding schemes is evaluated using metrics such as SNR, BLER, and FAR. Simulation results indicate that DCA-polar codes outperform traditional CA-polar codes in terms of error correction capabilities, especially in scenarios with short block lengths. Additionally, concatenated polar codes utilizing I_B have demonstrated superior performance compared to those using I_R, particularly in low SNR conditions.

5.5.1 Performance evaluation of DCA-polar codes for 5G NR control channels

This section estimates the performances of DCA-polar codes to examine error detection and correction in 5G control channels such as PBCH, PUCCH and PDCCH framework, respectively. BLER performances are attained as a function of E_b/N_0 by considering QPSK modulation and AWGN. In our work, we have used 1000 maximum number of block errors which adhere to the Monte-Carlo approach of calculating BLER.

Role of designSNR on Performance: The effectiveness of polar codes is directly dictated by their construction method, where three critical parameters—code rate (R), block length (N),

and designSNR—play pivotal roles. Achieving an optimal construction is challenging because all three parameters need to be optimized within a recursive structural model. Figure 5.9 shows the BLER vs. Eb/No plot for DCA-polar coded system with A=32 and E=108 considering different designSNR of {1, 2, 3, 4, 5} dB. It can be seen that the BLER decreases for a particular designSNR when Eb/No increases. Especially for a designSNR of 5 dB, the associated BLER values exhibit the maximum decrease, resulting in improved system performance. Hence, the choice of designSNR significantly influences the achievement of better performance in a polar-coded system. The performance comparison of DCA-polar and CA-polar coded system with A=32 in AWGN channel is presented in Figure 5.10. DCA-polar codes outperform CA-polar in terms of BLER for the same E_b/N_0.

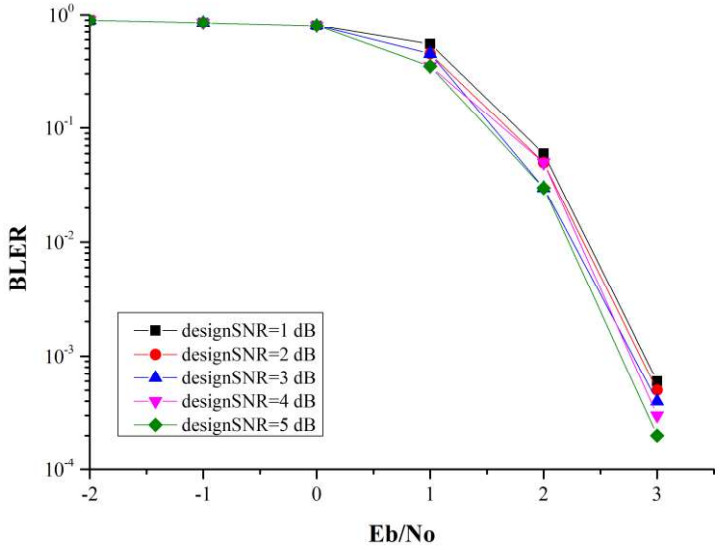


Figure 5.9 Graph of BLER vs. Eb/No for DCA-polar coded system with A = 32, E = 108 and different designSNR in AWGN channel.

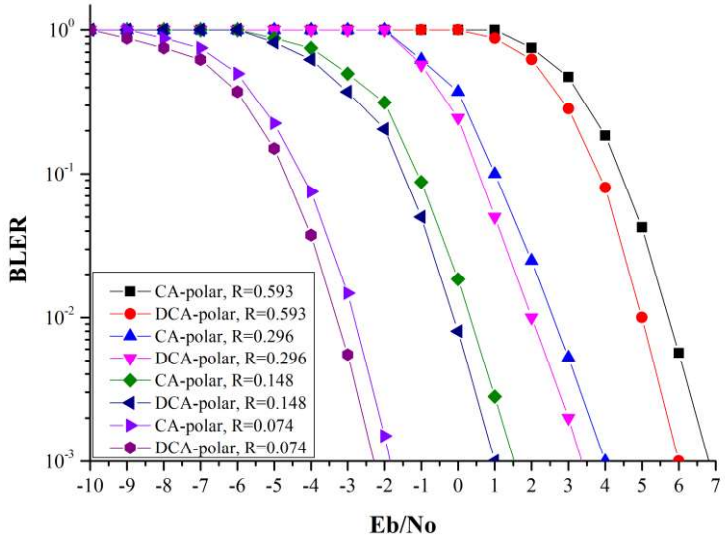


Figure 5.10 Graph of BLER vs. Eb/No for performance comparison of DCA-polar and CA-polar coded system with A = 32 in AWGN channel.

BLER performances of DCA-polar codes in PBCH: In Figure 5.11 for PBCH channel, BLER vs. E_b/N_o is characterized for various list size L of the decoder. It also illustrates the effect of list size over BLER in PBCH channel using fixed A = 32 and E = 864. BLER approaches the capacity bound as the list size increases, with diminishing returns observed between list sizes of 8 and 32. Benefit of NR polar codes at short block lengths can be seen clearly from the figure as BLER approaches near the capacity bound.

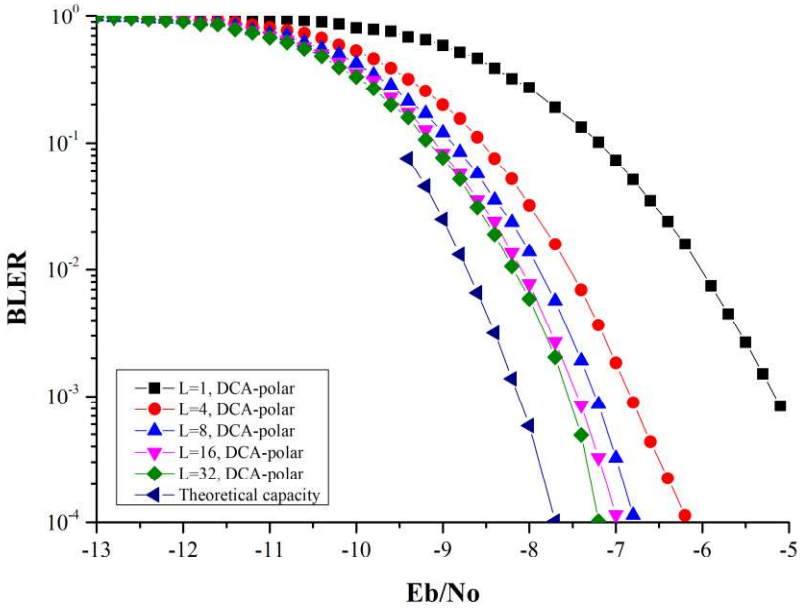


Figure 5.11 Graph of BLER vs. E_b/N_o for DCA-polar code in PBCH with A = 32 and E = 864.

Eb/No performances of DCA-polar codes in PUCCH and PDCCH: To attain a BLER of 10^{-4}, the E_b/N_o necessary in PUCCH and PDCCH channels is depicted in Figures 5.12 and 5.13, respectively. However keeping the coding rate A/E unchanged, BLER advances upon incrementing '*A'*. Similarly keeping 'A' unchanged, the BLER increases on improving 'E', whereas falling returns are detected when count on repetition. Once '*A'* and 'E' are small, BLER of the polar codes approaches near the capacity bound. Once the coding rate is less, NR polar decoding in PUCCH and PDCCH channels achieve error correction requirements of BLER = 10^{-4} at lower SNRs as shown in Figures 5.12 and 5.13.

Figure 5.14 and 5.15 shows the Eb/N_0 required by the 5G polar code to achieve a BLER of 10^{-4} as a function of the message length 'A' for different encoded block lengths. QPSK modulation over AWGN channel has been used for the simulations. Polar codes are decoded through SCL decoding with a list size of 8, as proposed by 3GPP for baseline. Figure 5.14 considers the PUCCH case, where a noticeable performance improvement observed by the sudden jump in the BLER curves at A>24. Figure 5.15 presents the PDCCH case, where early termination is enabled and the SCL decoder maintains failing paths in the list. The BLER performance is degraded compared to PUCCH due to the larger CRC inserted, however with a better FAR mitigation and a more efficient early termination mechanism. For both the PUCCH and PDCCH cases, our proposed codes outperform the published works.

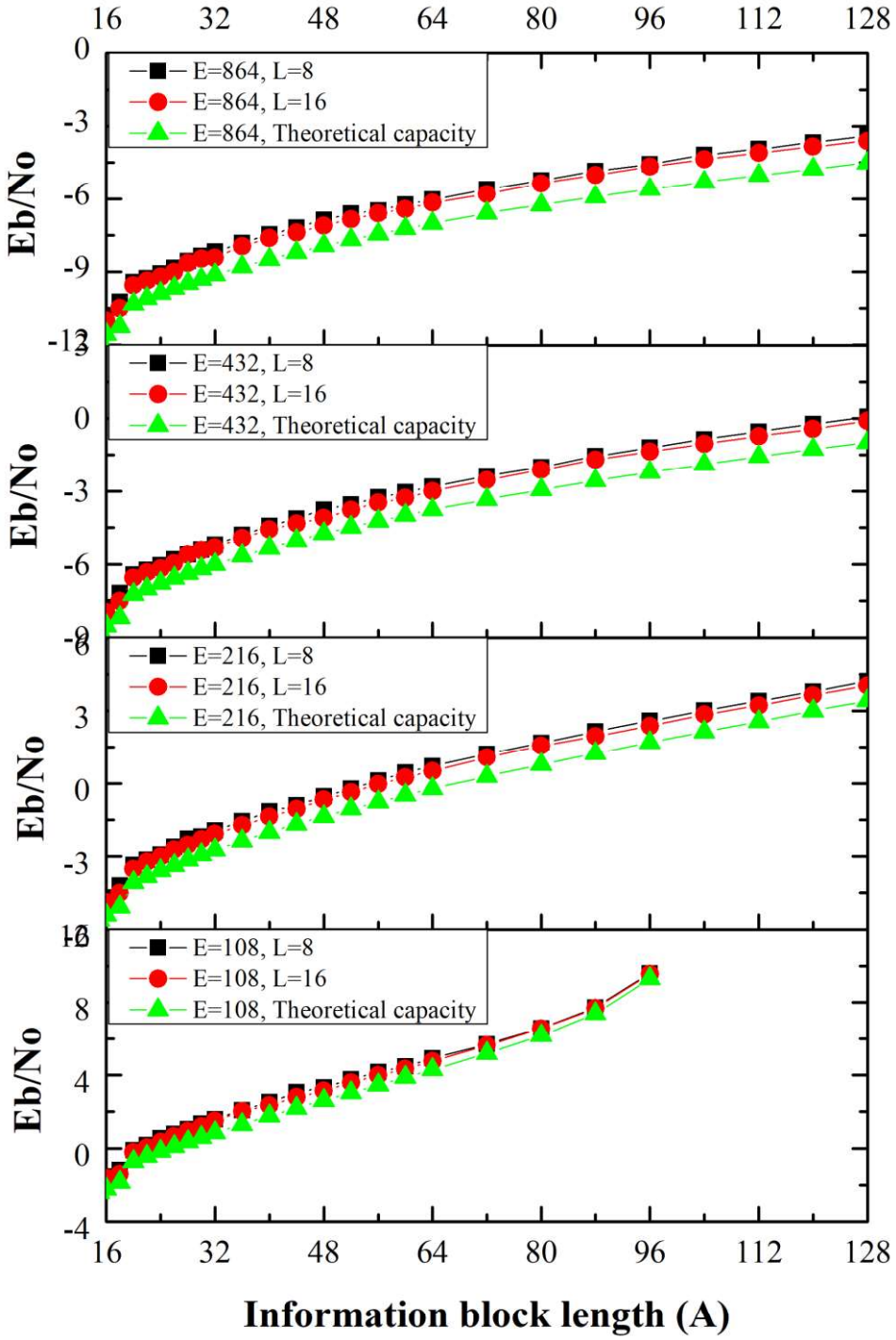


Figure 5.12 Graph of E_b/N_o vs. A for DCA-polar code in PUCCH with BLER of 10^{-4}

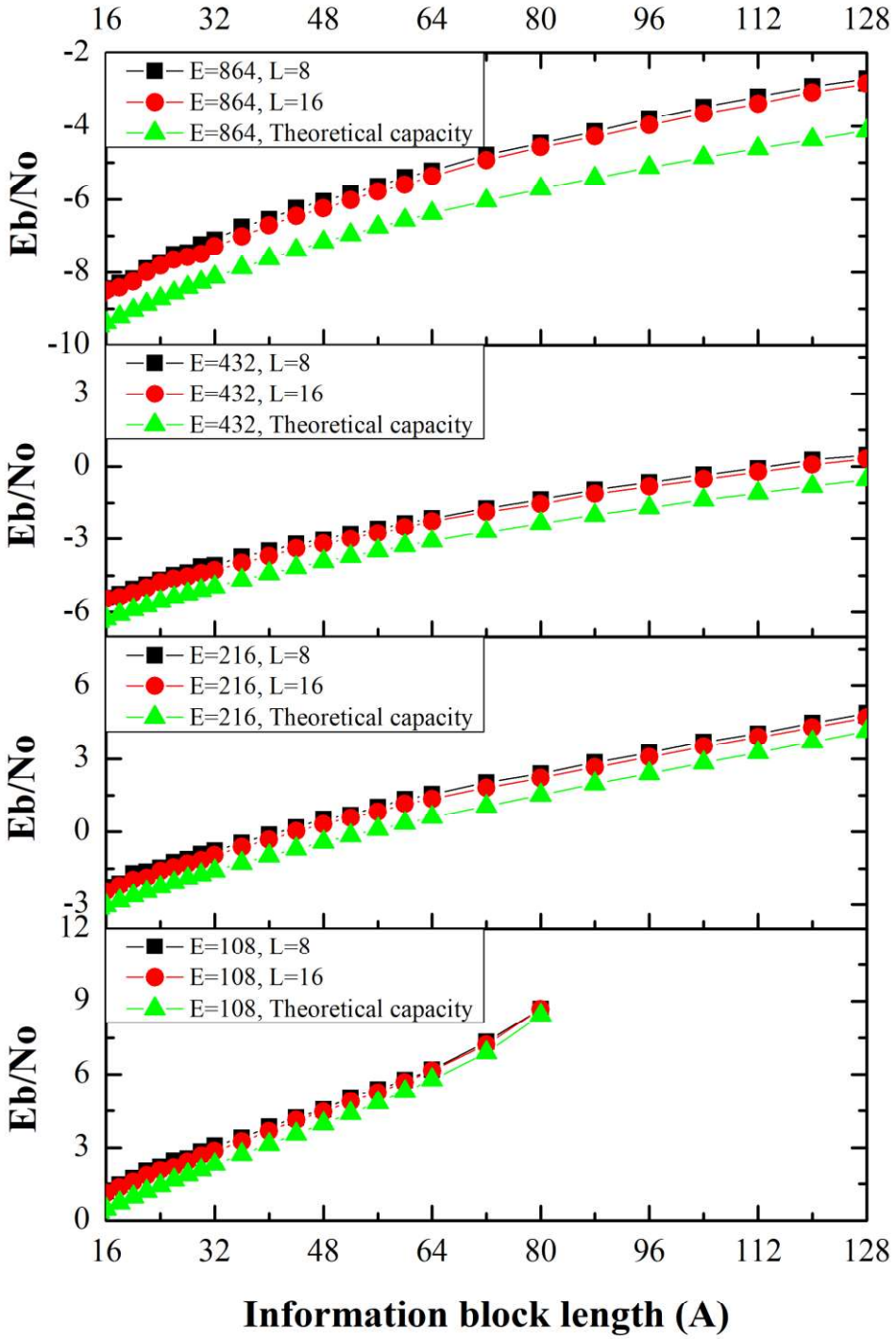


Figure 5.13 Graph of E_b/N_o vs. A for DCA-polar code in PDCCH with BLER of 10^{-4}

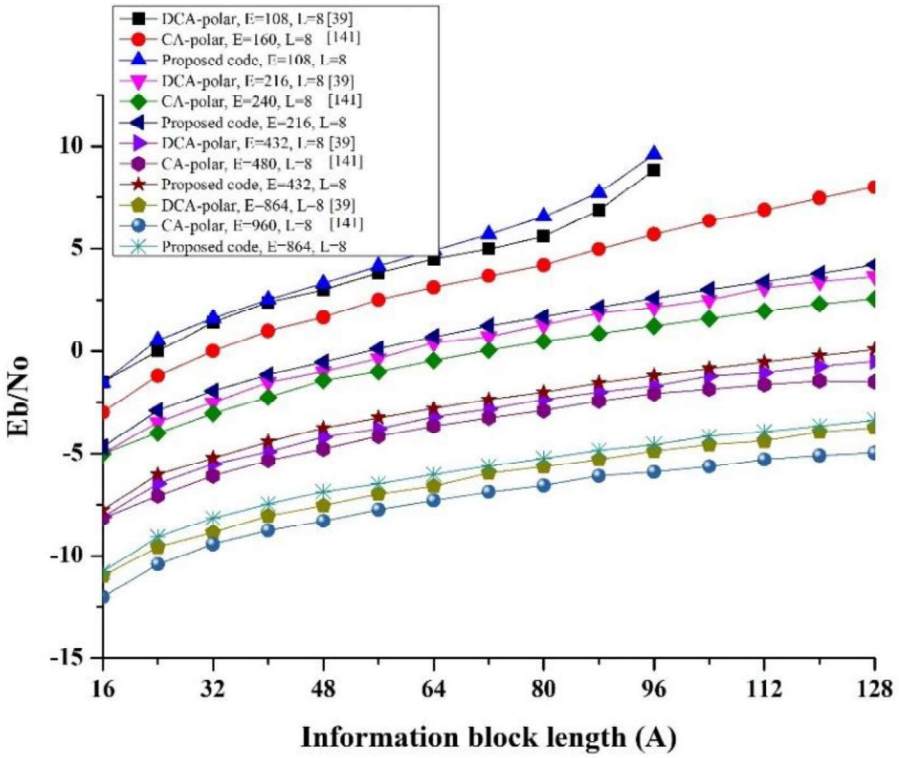


Figure 5.14 Performance comparisons of polar codes in PUCCH at BLER of 10^{-4} with prior works.

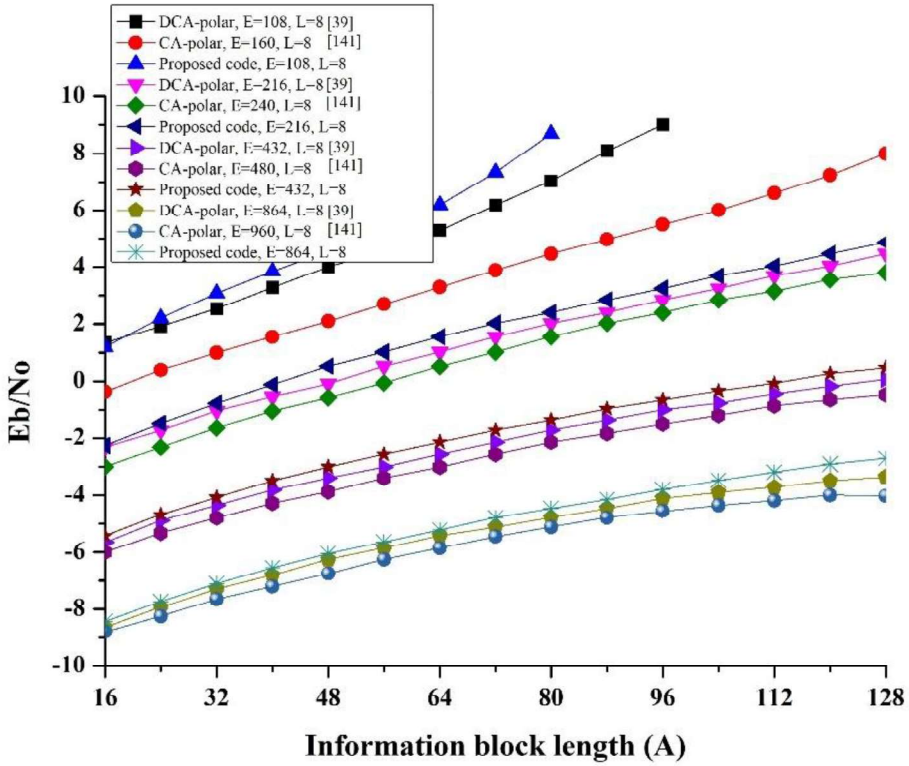


Figure 5.15 Performance comparisons of polar codes in PDCCH at BLER of 10^{-4} with prior works.

Power efficiency: In wireless communication system design, transmitter power is often constrained, necessitating power-efficient transmission to meet specific BLER requirements. A key metric for assessing power efficiency is the ratio of energy per information bit to the noise power per unit bandwidth, known as $\frac{E_b}{N_o}$. The relationship between $\frac{E_b}{N_o}$ and the SNR can be articulated as

$$SNR = \frac{E_s}{\sigma^2} = \frac{R \times E_b}{\frac{N_o}{2}} = 2R \times \frac{E_b}{N_o} \quad (5.8)$$

$$\frac{E_b}{N_o} = \frac{SNR}{2R} \quad (5.9)$$

$$\frac{E_b}{N_o}(dB) = SNR(dB) - 10\,|log_{10}(2R)| \quad (5.10)$$

where E_s= signal power, σ^2= power spectral density, E_b= bit energy and N_o= noise power.

To find such a code rate for polar codes, we first fix a target BLER of 10^{-4} and design polar codes of encoded block lengths $E \in \{108, 216, 432, 864\}$ optimized for fixed design SNR = 3.5 dB. We then find the $\frac{E_b}{N_o}$ value which results in the target BLER for the information blocklengths $A \in \{16, 32, 64, 128\}$ and list sizes L ϵ 8. Table 5.3 shows the $\frac{E_b}{N_o}$ requirements for different code rates of SCL decoding of DCA-polar codes at BLER = 10^{-4} and fixed design SNR = 5 dB. It's evident that power efficiency hinges on the code rate. Thus, there exists a particular code rate that yields maximum power efficiency. When the rate is excessively low, power efficiency diminishes because the allocated power is spread across fewer information bits. Furthermore, as the list size expands, the rate at which power efficiency is optimized, decreases.

Table 5.3: $\frac{E_b}{N_o}$ requirements for different code rates of DCA-polar with list size (L = 8), design SNR = 5 dB and target BLER = 10^{-4}.

Information blocklength (A)	**Encoded blocklength (E)**	**Code rate (R)**	$\frac{\mathbf{E_b}}{\mathbf{N_o}}$ **(dB)**
16	108	0.148	0.287
	216	0.074	3.297
	432	0.037	6.308
	864	0.018	9.437
32	108	0.296	2.723
	216	0.148	0.287
	432	0.074	3.297
	864	0.037	6.308
64	108	0.593	4.259
	216	0.296	2.276
	432	0.148	0.287
	864	0.074	3.297
128	216	0.593	4.259
	432	0.296	2.723
	864	0.148	0.287

False Alarm Rate (FAR) performances: Due to the adoption of blind detection in 3GPP and 5G standards, coding schemes over the control channels necessitate low False Alarm Rate (FAR) performance. The FAR is defined as the ratio of incorrectly decoded blocks passing the CRC (E_{crc}) to the total number of decoded blocks (E_{total}). In our proposed DCA-polar

codes for 5G NR control channels, we aim for low FAR in the uplink and both low FAR and early termination (ET) in the downlink. The theoretical FAR performance is determined as $2^{-(r-3)}$, where (r-3) bits out of r are utilized for error detection, and the remaining 3 CRC bits are employed to enhance error correction.

Table 5.4 Observations of different parameters of DCA-polar codes for different control channels.

Channel	CRC length (r)	Information Blocklength (A)	Encoded Blocklength (E)	List size (L)	Simulated FAR	Theoretical FAR	Computational complexity O(LN logN)
PUCCH	11	32	108	8	3.51×10^{-5}	3.91×10^{-5}	2941
		32	216	8	3.62×10^{-5}		5883
		32	108	16	4.33×10^{-5}		5553
		32	216	16	3.81×10^{-5}		11107
PBCH	24	32	864	8	4.51×10^{-7}	4.77×10^{-7}	23531
		32	864	16	4.36×10^{-7}		44427
PDCCH	24	32	108	8	4.61×10^{-7}	4.77×10^{-7}	2941
		32	216	8	4.84×10^{-7}		5883
		32	108	16	4.71×10^{-7}		5553
		32	216	16	4.73×10^{-7}		11107

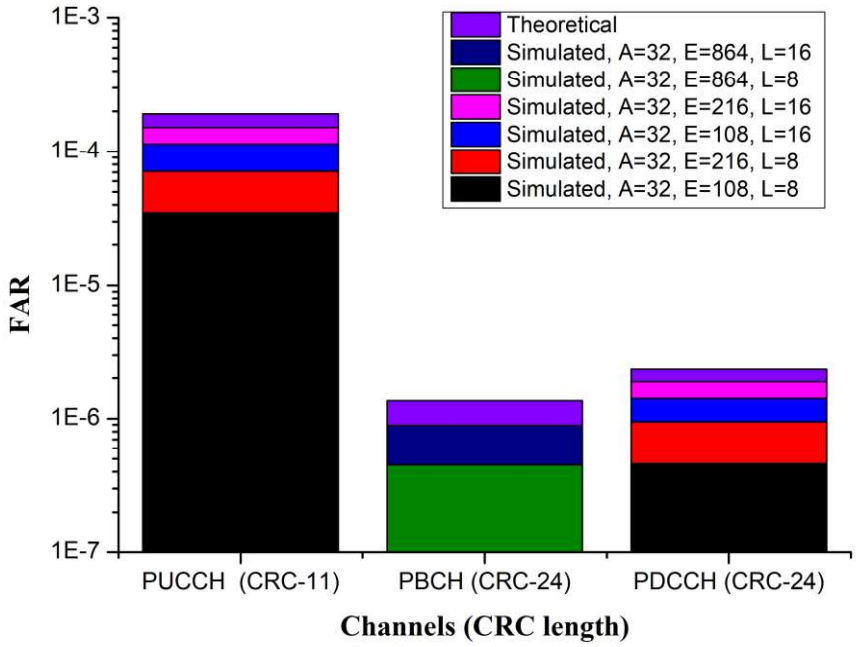


Figure 5.16 Comparison of simulated and theoretical FAR for different 5G control channels and CRC length using DCA-polar code.

The details of theoretical and simulated FAR of DCA-polar codes for different control channels are presented in Table 5.4 and Figure 5.16. Also, computational complexity of DCA-polar codes for different control channels are presented in Table 5.4. In Figure 5.16, the graph of FAR is attained by decoding random Gaussian distributed LLRs whereas simulation is performed till 1000 false alarms or block errors are detected. FAR progresses on incrementing the CRC length and list sizes of DCA-polar codes for different control channels. Through extensive analysis of FAR performance, we have determined that the proposed DCA-polar codes, equipped with carefully designed CRC bits, are capable of meeting the FAR targets set for 5G standards.

5.5.2 Performance evaluation of Concatenated polar codes with different interleaving and decoding optimization for WSN

This section presents the simulation outcomes to validate the performance of concatenated codes utilizing various interleaving and decoding strategies. At first, a long polar code is employed to the system model along with the concatenated polar codes (with and without interleaving). The code rates and decoding scheme of the four ECC are alike as shown in Table 5.5, making it suitable for assessing these ECC's performance in reducing errors. Also, Table 5.5 mentions the outer and inner codes' information and code lengths. For BCH and Polar decoding, BD (Bounded Distance) and SC algorithms are employed respectively.

Table 5.5 Parameters of different polar codes.

Coding Scheme	Interleaver	Outer code	Inner code	Code rate	Decoding Scheme
Concatenated LDPC-polar code [134]	without	(32, 28) LDPC	(256, 96) Polar	1/3	SC
Concatenated RS-polar code [134]	without	(15, 11) RS	(512, 232) Polar		
(512, 128) Polar code	without	-	-	1/4	
Proposed Concatenated code	without	(31, 16) BCH	(32, 16) Polar		
Proposed Concatenated code	I_R				
Proposed Concatenated code	I_B				

Figure 5.17 shows that the concatenated polar code using I_B which provides the finest performances. At BER=10^{-5}, concatenated BCH-polar codes with I_R scheme requires approx. 0.5 dB less and concatenated BCH-polar codes with I_B scheme requires approx. 0.9 dB less SNR to attain this BER in comparison to long polar code. Further, the proposed concatenated BCH-polar codes outperform interleaved concatenated LDPC-polar [134] and RS-polar codes [134]. At BER=10^{-4}, concatenated BCH-polar codes with I_B scheme requires approx. 1 dB and 1.5 dB less SNR to attain this BER in comparison to concatenated LDPC-polar and RS-polar code respectively. In contrast, concatenated BCH-polar code without interleaving shows worst performance than long polar code due to lower code dimensions as two short length ECC are cascaded to design concatenated BCH-polar code compared to long length polar code where code length is approx. 16 times more than the concatenated BCH-polar code.

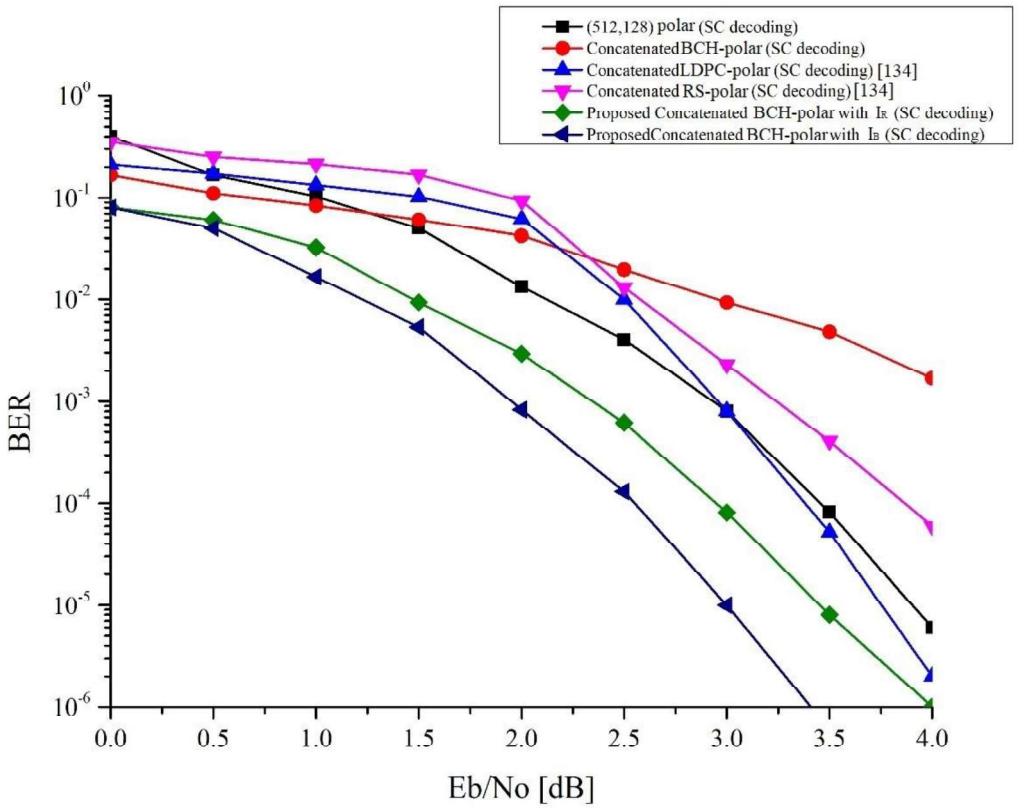


Figure 5.17 Comparing the performance of various concatenated codes in an AWGN channel.

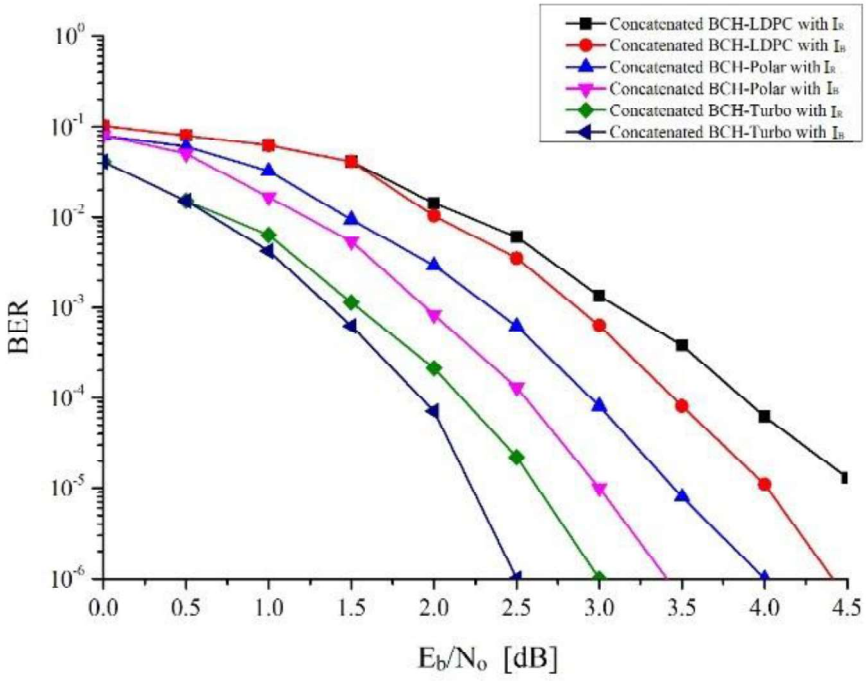


Figure 5.18 Comparing the performance of concatenated codes using distinct interleavers in an AWGN channel.

Table 5.6 Parameters of different concatenated codes using I_R or I_B scheme.

Coding Scheme	Interleaver	Outer code	Inner code	Code rate	Decoder
Concatenated BCH-LDPC code	I_R	(31, 16) BCH	(32, 16) LDPC	1/4	SC
	I_B				
Concatenated BCH-polar code	I_R		(32, 16) polar		
	I_B				
Concatenated BCH-turbo code	I_R		(32, 16) turbo		
	I_B				

Figure 5.18 compares the performance of six distinct concatenated codes using I_R or I_B scheme. Concatenated codes with I_B perform better compared to concatenated codes with I_R. Concatenated BCH-turbo codes with I_R perform the finest of all concatenated codes using I_R, whereas concatenated BCH-LDPC codes with I_R perform the poorest. Precisely, compared to concatenated BCH-polar codes with I_R, concatenated BCH-LDPC codes with I_R require approximately 1.2 dB more SNR and concatenated BCH-turbo codes with I_R require approximately 0.7 dB less SNR to attain the BER= 10^{-5}. Moreover, concatenated BCH-turbo codes with I_B perform the finest of all concatenated codes using I_B, whereas concatenated BCH-LDPC codes with I_B perform the poorest. Precisely, compared to concatenated BCH-polar codes with I_B, concatenated BCH-turbo codes with I_B require approximately 0.75 dB less SNR and concatenated BCH-LDPC codes with I_B require approximately 1.1 dB more SNR to attain the BER= 10^{-5}. The code rates of the six ECC are 0.25 as shown in Table 5.6, making it suitable for assessing the performance of these ECC.

Table 5.7 Parameters of different concatenated polar codes with SC decoding scheme.

Coding Scheme	Interleaver	Outer code	Inner code	Code rate	Decoder
Concatenated code [136]	RI	(63,30) BCH	(128,64) polar	1/4	SC
Concatenated code [136]	BI				
Concatenated code [135]	RI	(31,16) BCH	(32,16) polar		
Concatenated code [135]	BI				
Proposed Concatenated code	I_R	(31,16) BCH	(32,16) polar		
Proposed Concatenated code	I_B				

The proposed concatenated BCH-polar codes' BER performance is compared against existing published works [135 - 136]. The code rates and decoding scheme of the four ECC are alike as shown in Table 5.7. Also, Table 5.7 mentions the dimensions of outer and inner codes and code lengths. Figure 5.19 compares the BER performance of proposed concatenated polar codes against existing published works [135 - 136] exploiting different interleaving strategies. As expected, short length codes show poor performance compared to the code mentioned in [136, 142 - 143]. Nevertheless, contrary to that, our proposed concatenated code outperforms the code mentioned in [136] although having a shorter dimension code length. At BER=10^{-6}, the proposed concatenated BCH-polar codes can attain a noticeable gain of 0.5 dB as compared to the code mentioned in [136]. Further, our proposed codes outperform the codes mentioned in [135]. At BER=10^{-6}, the proposed concatenated BCH-polar codes can attain a noticeable gain of 0.2 dB as compared to the code mentioned in [135]. Simulation outcomes show that the proposed concatenated polar codes exhibit lower error rate compared to existing related concatenated codes for the said application. The SC decoder of polar codes tends to produce bursty or correlated errors, especially in low-SNR

conditions. Proposed I_B maintains a predictable reordering of bits, which can better align the error patterns between the inner polar and outer BCH codes. This allows the outer BCH decoder to correct errors more effectively, since the errors are more likely to be spread out and uncorrelated. On the other hand, I_R may fail to adequately decorrelate these bursts.

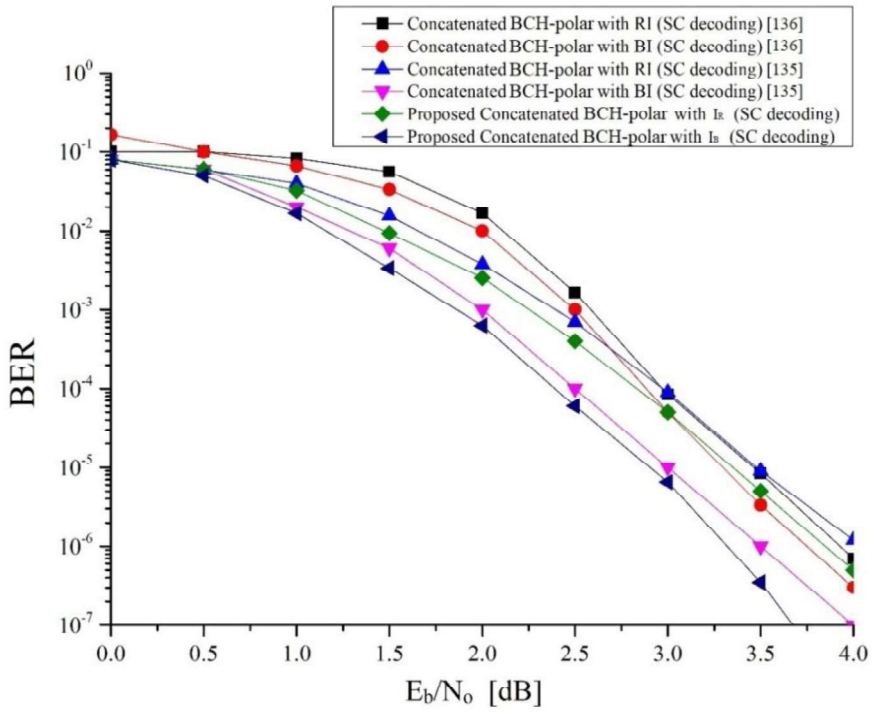


Figure 5.19 Comparing the performance of proposed concatenated codes with prior works.

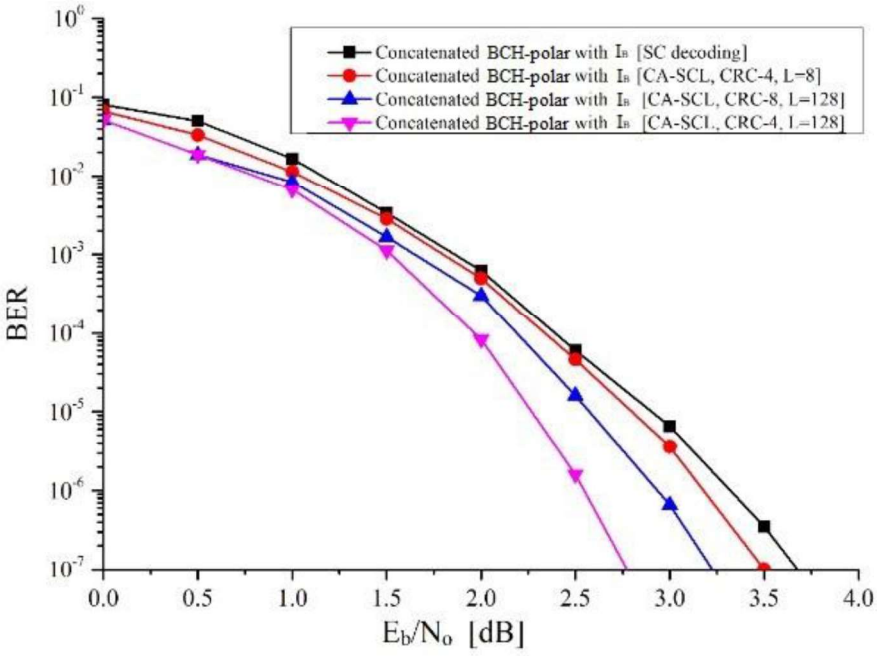


Figure 5.20 Performance comparison of various concatenated codes utilizing I_B scheme.

Table 5.8 Parameters of various concatenated polar codes utilising I_B scheme.

Coding Scheme	Interleaver	Outer code	Inner code	Code rate	Decoding Scheme
Proposed Concatenated code	I_B	(31,16) BCH	(32,16) Polar	1/4	SC
					CA-SCL [CRC-4, L=8]
					CA-SCL [CRC-4, L=128]
					CA-SCL [CRC-8, L=128]

The comparison of four distinct concatenated codes in terms of their performance and simulated results is presented in Figure 5.20. Concatenated BCH-polar codes using I_B and CA-SCL outperform the concatenated BCH-polar codes using I_B and SC decoding. At BER=10^{-5}, concatenated BCH-polar codes utilizing I_B can achieve a noticeable gain of 0.8 dB when utilizing CA-SCL (L = 128 and *CRC-4*) decoding as opposed to those codes utilizing SC decoding. Using same CRC length, CA-SCL decoding performance becomes better by incrementing list size L compared to all concatenated BCH-polar codes using I_B. At BER= 10^{-6}, concatenated BCH-polar codes utilizing I_B and CA-SCL (L = 128 and *CRC*-4) decoding require approx. 0.7 dB less than concatenated BCH-polar codes utilizing I_B and CA-SCL (L = 8 and *CRC*-4) decoding. However, the concatenated BCH-polar codes utilizing I_B and CA-SCL (L = 128 and *CRC*-4) decoding performs better than concatenated BCH-polar codes utilizing I_B and CA-SCL (L = 128 and *CRC*-8) decoding. Although the polar decoding scheme is different, the code rates of the four concatenated polar codes are alike as shown in Table 5.8, making it suitable for assessing their performance.

The decoding complexity of the proposed codes is also examined. In Table 5.9, various concatenated coding arrangements are compared in terms of their decoding complexity. Concatenated codes employing I_R or I_B method can significantly reduce decoding complexity compared to existing ECC attaining same BER. While increase in CA-SCL's list size can enhance the BER performance, it also can cause a rapid increase in decoding complexity. However SC decoding offers the lowest complexity, but its BER performance is relatively poor.

Table 5.9 Decoder complexity of various concatenated polar codes.

Coding Scheme	Interleaver	Outer code	Inner code	Decoding Scheme	Decoder complexity
(32,16) polar	-	-	-	SC	O(N logN)
(512,128) polar					
Proposed Concatenated code		(31,16) BCH	(32,16) polar		O(Nb) + O(N logN)
	I_R				
	I_B				
				CA-SCL [CRC-4, L = 8]	O(Nb) + O(LN logN)
				CA-SCL [CRC-4, L =128]	
				CA-SCL [CRC-8, L =128]	

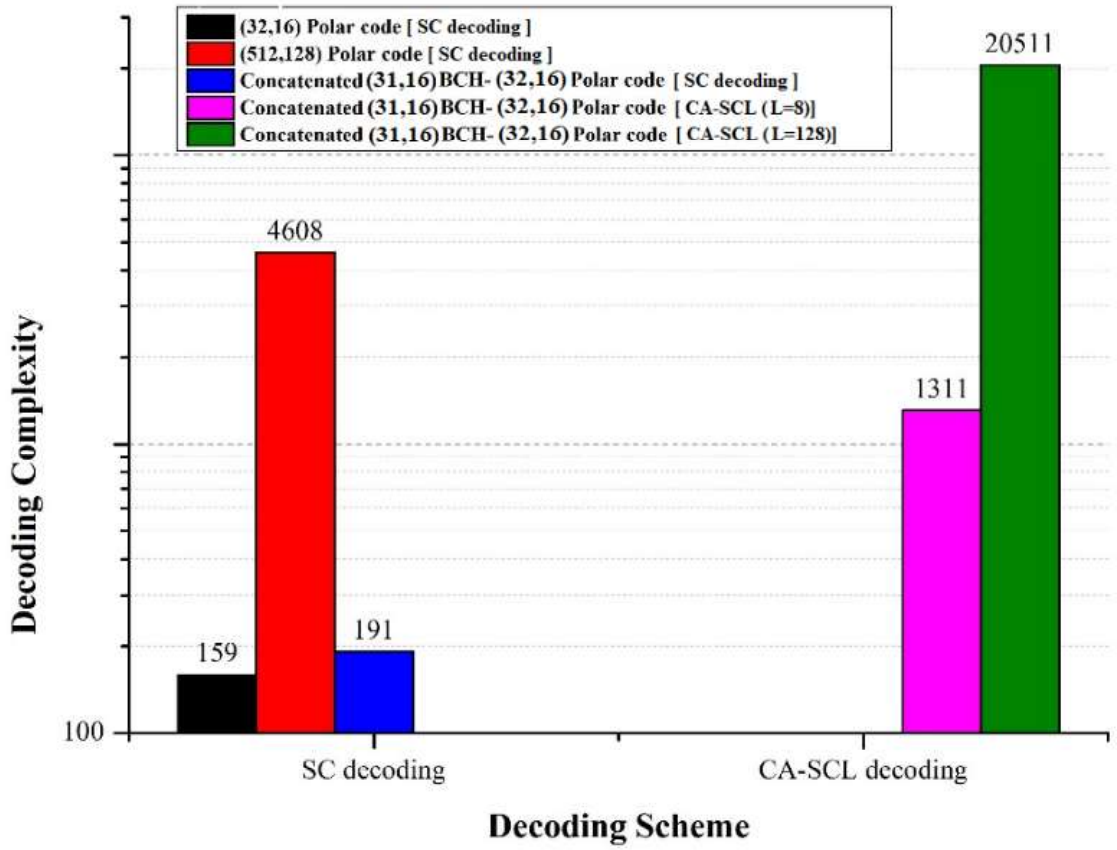


Figure 5.21 Comparison of proposed concatenated polar codes' decoding complexity with existing codes attaining same target BER.

Finally, the graphical comparison of decoding complexity (in terms of N and L) of different concatenated polar codes along with traditional polar codes is done as depicted in Figure 5.21. The graphical analysis indicates that optimizing the design of concatenated codes can reduce decoding complexity in terms of computation time while maintaining a target BER of 10^{-5}. Comparing to existing finite length ECC, our proposed concatenated codes (*blue & purple colour bar in Fig.5.21*) show a more favourable trade-off between complexity and performance.

The integration of polar codes in resource-constrained applications, such as WSNs (Wireless Sensor Networks) and IoT devices, presents several practical implications:

(i) Energy Efficiency: The low complexity of polar codes allows for reduced energy consumption, which is critical for battery-operated devices.

(ii) Enhanced Reliability: The combination of polar codes with CRC and interleaving techniques improves the reliability of data transmission in noisy environments.

5.6 Summary

Polar coding schemes, particularly when enhanced with CRC and concatenation techniques, offer a robust solution for resource-constrained applications in modern communication systems. Their capacity-achieving properties, low complexity, and adaptability make them ideal candidates for deployment in IoT and 5G networks, where efficiency and reliability are paramount. Continued research and development in this field will further enhance the applicability of polar codes, paving the way for innovative solutions in wireless communication.

This chapter presents an analysis of polar codes chosen for 5G control channels, with a specific focus on short lengths and low code rates. Our investigation reveals that for each code length, there exists an optimal rate at which polar codes attain maximum power efficiency. Consequently, we propose a framework for polar code design in 5G control channels that accounts for error-correction performance, decoder efficiency, and flexibility across various design parameters. The core of this approach involves distributing CRC bits throughout the entire information block, with a unified interleaving/deinterleaving pattern established for CRC distribution implementation. Simulation results demonstrate a notable improvement in BLER performance compared to conventional CRC-aided list decoders. In conclusion, CRC distributed polar coding emerges as a flexible design capable of supporting various decoding paradigms to accommodate diverse requirements. As per the 5G NR requirement, DCA-polar codes are proposed in this chapter to provide significant error correction performance using short block lengths and to permit early termination during decoding. By increasing the minimum Hamming distance between valid codewords, CRC bits improve the performance of SCL decoding in CA-SCL. Also, early termination and path trimming is allowed by distributed CRC bits. Low energy consumption as well as low decoding latency of hardware is very crucial for 5G applications and this work is able to achieve these criteria. For low code rates and short block lengths, this work reflects the error performance analysis of polar codes in uplink, downlink and broadcast NR control channels (PUCCH, PDCCH and PBCH). The error detection and correction of the polar codes are systematically examined using BLER, SNR and FAR analysis as the performance measures. Given the conflicting metrics—such as coding complexity, coding delay, code length, and code rate—and the associated design trade-offs, the need to formulate optimal polar codes becomes increasingly important. It is extremely promising and challenging research interest is to design the optimum polar codes, subject to the specific practical application.

The most difficult challenge during energy-constrained WSNs design for IoT systems is to decrease the energy consumption whereas safeguarding link reliability. Although networks using ECC can lower energy usage while maintaining connection reliability, it is crucial to pick a WSN-appropriate coding scheme. The viability of using polar codes in WSNs for IoT system is also assessed in this chapter. Concatenated codes with BCH as outer code and short polar as inner code are considered to enhance BER with less decoding complexity compared to long polar codes. Moreover, two distinct interleaving arrangements such as I_R and I_B, are proposed and experimental findings indicate that I_B method outperforms I_R method. Alternatively, concatenated polar code using I_B has better BER and shows very low decoding complexity in comparison to the long polar code. Further, the performance is intended to be improved by concatenating polar codes using various decoding and interleaving arrangements. According to the simulation findings, our proposed concatenated code with CA-SCL decoding performs better compared to SC decoding. Further, expanding the list size can enhance CA-SCL's decoding performance keeping the CRC length constant. Nevertheless, CA-SCL decoding complexity will increase rapidly as the size of list rises. Instead, the loss in the code rate is what causes the unanticipated decrease in decoding ability of CA-SCL if code length of CRC is increased while keeping the list size constant. The proposed concatenated polar code using I_B method is the best ECC choice in WSNs for IoT system considering performance and decoding complexity. These codes are ideal for IoT applications, including smart cities, industrial IoT, and remote sensing, where devices often operate under noisy conditions with constrained power and bandwidth. In the next chapter, ECC based two different security schemes have been presented and effectiveness of their design is highlighted.

Chapter 6

Design of Error Correcting Code based Security Schemes

6.1 Introduction

Wireless communication systems are widely utilized in both military and civilian applications, necessitating a robust security measures to protect sensitive information. The need for secure transmission of information has become increasingly critical due to the proliferation of eavesdropping technologies and the vulnerabilities inherent in traditional cryptographic methods. Traditional cryptographic methods often rely on complex algorithms that may not be suitable for low-power, portable devices. As a result, there is a growing interest in PLS, which leverages the physical properties of the communication channel to secure data transmission without relying solely on computational complexity. PLS techniques exploit the inherent randomness of wireless channels, allowing legitimate users to communicate securely while preventing unauthorized access by eavesdroppers. This chapter explores the intersection of ECC and physical layer security (PLS), focusing on how ECC can enhance the security of wireless communications. We investigate various coding schemes, particularly polar codes and Golay codes, and their applications in PLS, highlighting their effectiveness in ensuring confidentiality and integrity in data transmission.

The rest of this chapter is organized in the given fashion. Overview of existing works is briefly introduced in Section 6.2. Basics of PLS and ECC are presented in Section 6.3. Design and performance analysis of Golay encoder and decoder for modified McEliece cryptosystem is shown in Section 6.4. The performance evaluation of AN-aided secure polar coding for wireless networks is provided in Section 6.5. In Section 6.6, summary is presented.

6.2 Related existing works

It is anticipated that as technology develops, various existing public-key cryptosystems could be compromised by the advancement of a quantum computer. The integrity and confidentiality of communications will be negotiated. Public key cryptosystems, which include the RSA (Rivest-Shamir-Adleman), Diffie-Hellman, Elliptic curve cryptosystems and DSA (digital signature algorithm), have grown to be an essential part of cyber security throughout the years. Shor's algorithm is well-known in the field of cryptography due to its potential to break a variety of cryptosystems, including the RSA and elliptic curve cryptography [144]. Using Shor's method, any of these public key cryptosystems can be broken. Other significant categories of cryptosystems include those that use codes, lattices, hashes, multivariate quadratic equations, and secret-key. Security and effectiveness are traded off in the use of code-based cryptography. The objective of post-quantum cryptography is to create cryptosystems which are compatible with existing networks and communications protocols as well as secure against any attacks. The current status of various cryptosystems is shown in Table 6.1.

Table 6.1 Current status of various cryptosystems [145].

Cryptographic Algorithm	Year of introduction	Impact of Quantum computing
Diffie-Hellman	1976	Cracked
RSA (Rivest-Shamir-Adleman)	1978	
McEliece	1978	Not Cracked
Niederreiter	1986	Cracked
Elliptic curve	1987	
Buchmann-Williams	1988	
Sidelnikov	1994	
NTRU (N-th degree Truncated polynomial Ring Units)	1998	Not Cracked
Lattice-based	1998	

Golay codes [62], which were developed in 1949, have seen extraordinary growth in recent years and implemented as a linear error-correcting code in communication links. Bit interleaving technology can be used with Golay codes to fix burst errors. McEliece [146] suggested an asymmetric encryption cryptosystem using Goppa codes in 1978, and it is still in use today due to its uncrackable security criteria. By employing alternate error-correcting codes, such as convolutional, LDPC, or AGC (algebraic geometric codes) in place of Goppa codes, many scholars developed improved McEliece cryptosystems. All of these plans have turned out to be vulnerable later. The comparison of different codes used in cryptosystem is shown in Table 6.2.

Table 6.2 Comparison of different codes used in cryptosystem.

Codes	Application	Merits	Observations
Hamming	Security	2-bit error detection and 1-bit error correction.	fail to detect uncorrected errors.
Reed-Solomon		multiple symbol errors detection and correction.	non-binary cyclic code.
Reed-Muller		multiple bit errors detection and correction.	low transmission rate.
Convolutional		appropriate for large data.	computational and decoding complexity is high.
Turbo		perform well at low SNR and uses interleaver to reduce burst errors.	decoding complexity and latency is high.
LDPC		perform well and attains near Shannon capacity.	high complexity.
Polar		perform well for short blocklengths and attains near Shannon capacity.	high complexity.
Golay		7-bit error detection and 3-bit error correction.	low complexity.
Goppa		correct multiple errors based on the Goppa polynomial.	high complexity.

There are many encoding-decoding techniques available. Although [63 - 65] deal with different algorithms, hardware implementation is not possible with these due to higher complexity. For hardware implementation, LFSR (linear feedback shift register)-based [66 - 69] approaches are suitable, although they are not acceptable due to high latency and less

throughput. CRC-based hardware implementation is suggested in [70]. Several hardware architectures [71 - 75] have been developed based on various Golay code decoding methods. A modified McEliece cryptosystem using extended Golay code is implemented in this chapter.

Coding for secrecy do not have a formal proof of security, but in reality the data are secure. The trade-offs in design rate are similar to traditional error-correcting codes, and secrecy coding structures are similar in style to linear block channel codes. Various schemes of secrecy coding are introduced in earlier publications offered transmission of private information reliably between appropriate users by preventing interception of same information by the eavesdropper. Wyner [147] offered a coding system without need of the eavesdropper's computational capacity and introduced the wiretap channel model. Later, researchers have demonstrated some design of secure coding techniques in [148 - 150]. In recent times, the development of wireless communication has made stronger attention in PLS implementation. The coding technique which reaches a secrecy capacity, the eavesdropper can be mixed up by random information at a rate almost its channel capacity in a discrete memoryless wiretap channel, resulting in transmission rate loss. Golay codes historically have strong error correction capabilities, whereas polar codes offer theoretical optimality in achieving channel capacity and find applications in modern communication systems due to their efficiency and reliability. For control channels in 5G-NR, the 3GPP group has chosen short block length polar codes as their preferred coding scheme [104].

Polar codes introduced by Arikan [78], are appropriate practical codes for PLS with ability to promise a favourable secrecy capacity in weak security constraints [151]. In [152], polar coding scheme ensuring strong security and attaining the secrecy capacity of various wiretap channels are presented in strong and weak security situation. Only the bit channels that are good for Bob but bad for Eve can be used to transmit information bits, and the secrecy capacity of the wiretap channel is therefore limited. Multi-block polar code is introduced [153] to guarantee reliability with strong security. But, the schemes in [151 - 153] are restricted assuming that the channel of Eve is degraded compared to Bob. Several practical situations prove this assumption unfulfilled and low transmission efficiency as well as weak secrecy capacity is noticed. To attain the secrecy capacity in a wiretap channel, polar coding structure was described in [152] and bit channel nature can be of three types as shown in Figure 6.1.

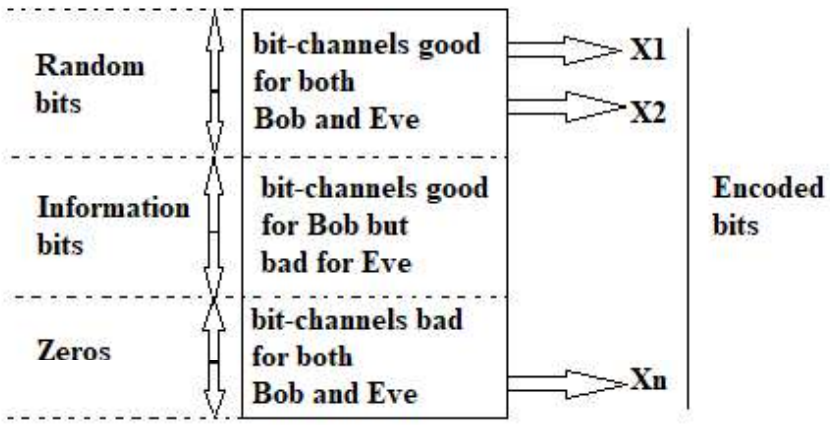


Figure 6.1 Polar coding structure in bit-channels.

Introducing AN, secrecy capacity can be improved by puzzling Eve's channel [154 - 156]. Specifically, an AN-aided polar code [154] offers increasing secrecy rate, where AN is

inserted in the existing codeword before modulation. The authors in [155] propose two AN power allocation schemes, both of which involve inserting AN after the modulation stage. As the transmitter is designed considering power constraint, proper power allocation between desired signals as well as AN requires to increase the secrecy capacity. Although methodologies to secrecy over the Gaussian wiretap channel exist, there are many open and important problems in this area.

The existing works on security in wiretap channel is presented in concise manner in Table 6.3. The existing works shows the improvement in secrecy rate and secrecy capacity using AN-aided polar codes. Neither the previous works had used security gap or BLER as a performance metric to evaluate security performance in wiretap channel using AN-aided polar codes, nor had they compared wiretap coding performance with the conventional AN-aided beamforming techniques.

Table 6.3 Existing works on security implementing AN-aided polar codes in wiretap channel.

Existing work	**Performance**	**Findings**
Bai et al. [154]	Optimization problem of jamming position selection is investigated and a suboptimal solution based on the greedy algorithm is provided.	**Secrecy rate** is improved.
Zhang et al. [155]	Two AN power allocation arrangements are offered considering power constraint environment	proper power allocation between suitable signals as well as AN requires to increase the **secrecy capacity**

6.3 Physical Layer Security and Error Correction Codes

In cryptography, no assumptions are made on the channel wiretapped by the eavesdropper which is computationally bounded. But in information theoretic security, no assumptions are considered for the computational power of eavesdropper and the wiretapped channel experiences more noise. In 1949, Shannon initiated the idea of 'information theoretic perfect secrecy' and secure transmission is done when two legitimate users (Alice and Bob) have identical secret key [157]. Any potential eavesdropper (Eve) is not capable to extract the private information without accessing the secret key. A generic version of Shannon's cryptosystem is displayed in Figure 6.2.

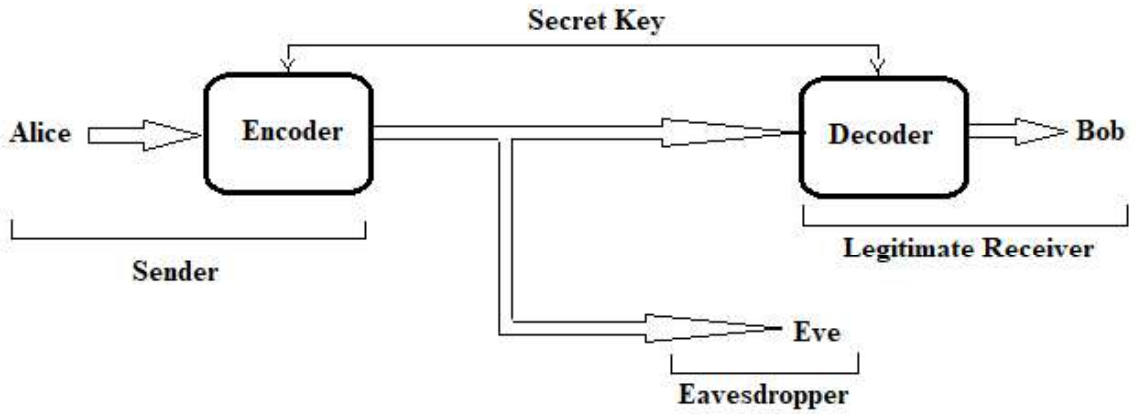


Figure 6.2 Shannon's cryptosystem.

The convention of identical secret keys is often unrealistic due to the inherently stochastic nature of communication channels. Various cryptographic algorithms were developed using shorter secret keys, though they depend on limited computational resources of Eve and unproved mathematical assumptions for secrecy. Using coding in physical layer, information theoretic secure transmission is probable if Eve's channel is not as good as Bob. The conventional metric for information theoretic security is equivocation which is problematic to estimate and analyze of noisy coded sequences. Presently practical coding scheme using finite block lengths almost not exist for secure transmission. Also, the existing codes cannot be applied directly to continuous channels such as Gaussian wiretap channel. Our work utilizes PLS comprehensively, rather than focusing solely on information-theoretic security, and employs BLER as an alternative performance metric instead of 'equivocation'.

In 1975, Wyner established 'degraded wiretap channel model' based on Shannon's work and showed that secure transmission of private information is possible without using a secret key [146]. The wiretap channel was used in Wyner's original work, whereas all types of communication channels including wireless links for the eavesdropper are considered at present time. The maximum communication limit of a system over degraded wiretap channel is characterized as secrecy capacity to measure the utmost competence of secret and reliable message transmission. When the secrecy capacity exceeds the secrecy rate, ECC can be designed to enable both reliable and secure communication. These ECC schemes, also known as wiretap coding, have been proven effective in preserving secrecy over the Wyner wiretap channel. No specific attack algorithm for Eve are considered in perfect secrecy condition and Eve has infinite time and computing power attempting to compromise the system supposing a worst case situation. Information-theoretic security constraints are considered to be the strongest security requirements on a system for this reason. Wyner's approach assumed Eve receives a noisy version of the transmitted codeword and relaxed the requirement for zero information leakage, introducing an asymptotic security constraint instead. While perfect secrecy is unattainable in practice, systems can still achieve weak and strong secrecy in a practical sense [158 -160]. This new model opened the way for innovative contributions in coding for PLS.

Figure 6.3 illustrates the most common performance metrics employed to evaluate the performance of key less PLS schemes. BLER signifies a more practical metric compared to secrecy capacity at present time as concerns the coding scheme. Security gap is also an important metric based on the slope of the code's BLER curvature as well as the SNRs of the main and the adversarial channel. The security gap needs to be potentially small to promise security of the system. A generic version of Wyner wiretap channel model is displayed in Figure 6.4. Alice encodes the message using channel coding scheme and send it to Bob through main channel. Eve eavesdrops this on a noisy wiretap channel and attempts to recover the information.

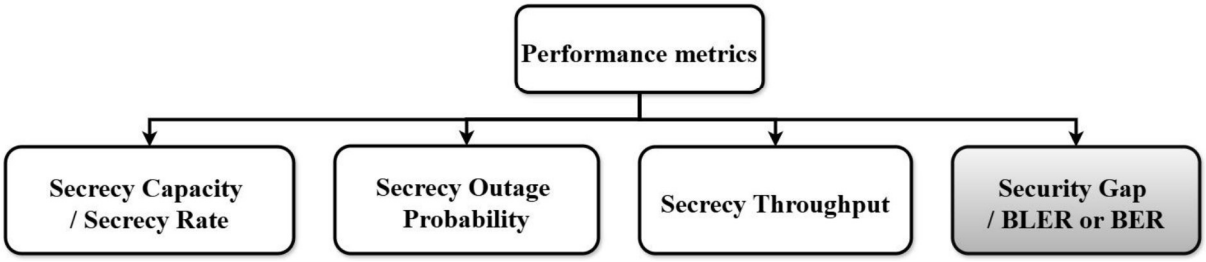


Figure 6.3 Performance metrics to evaluate keyless PLS.

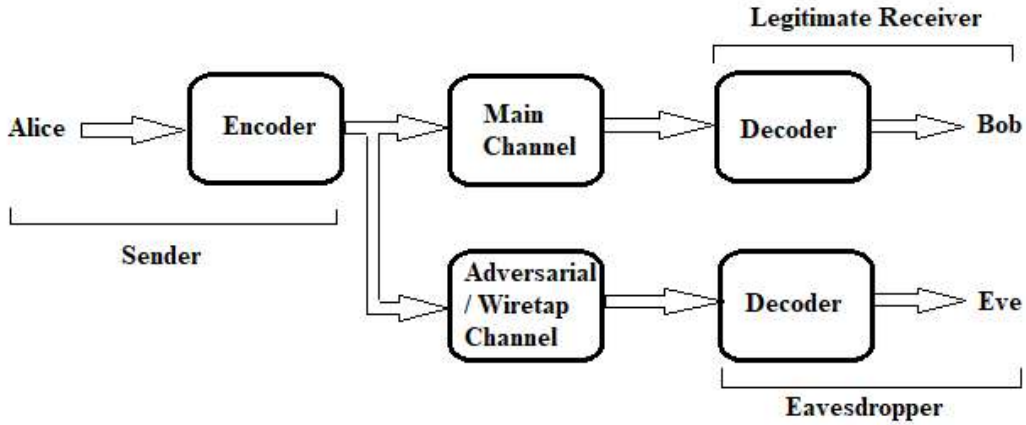


Figure 6.4 Wyner's wiretap channel model.

Let P_e^E and P_e^B are an average BLER on Eve's and Bob's estimate respectively. Also consider $SNR_{B,min}$ and $SNR_{E,max}$ are the reliability and the security threshold respectively. P_e^B should be adequately small to guarantee reliability and P_e^E be high. Hence, for fixed $P_{e,max}^B$ and $P_{e,min}^E$, $P_e^B \leq P_{e,max}^B$ (reliability) and $P_e^E \geq P_{e,min}^E$ (security). The security gap size ($SNR_{B,min}$ / $SNR_{E,max}$) in dB as displayed in Figure 6.5 shows the smallest variance between Bob and Eve's SNRs. Secure coding, when evaluated using the security gap, should exhibit a steep BLER curve and maintain a minimal security gap.

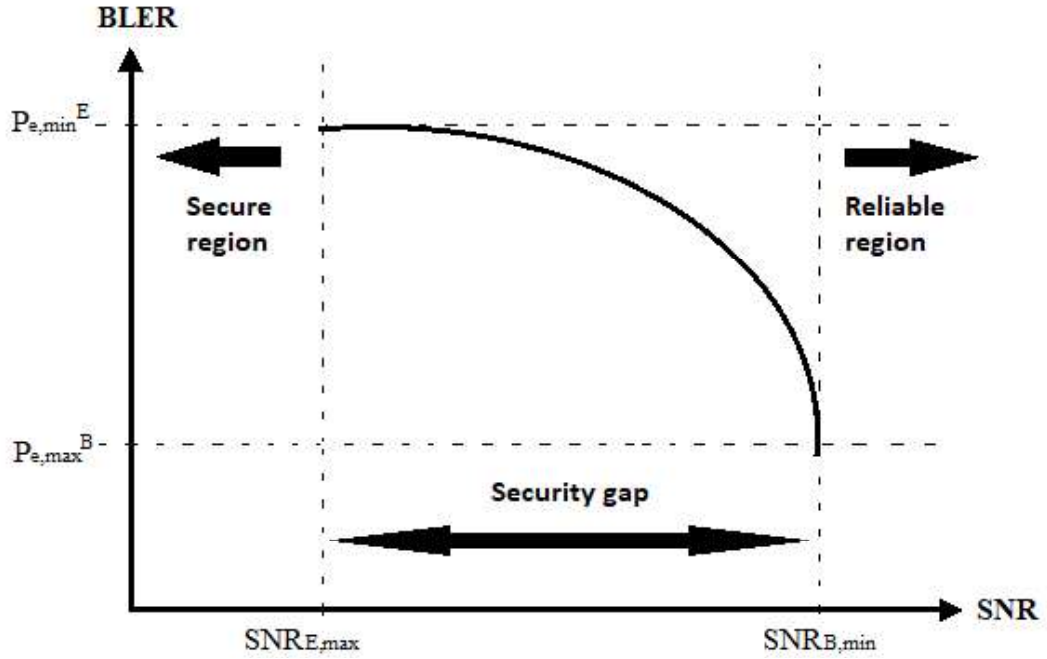


Figure 6.5 Security gap measurement using BLER and SNR.

6.3.1 Overview of Physical Layer Security

Wireless network security is becoming a growing concern among researchers and test field operatives. In traditional communication systems, security is considered in the application layer of the protocol stack and cryptography offers the solutions. Today, security is a multi-layer problem with solutions at every layer in the stack and additional layers are capable of providing further security benefits. Earlier the physical layer has been unnoticed for security implementations in general communication systems, but PLS solutions at present time can provide a significant gain in security for wireless links when eavesdropping is a rising

alarm. PLS solutions can co-exist with higher layer solutions and add an extra layer of security to prevent eavesdroppers from attaining clean signals of the transmitted messages. Encryption based methodologies involve real-world complications for future wireless networks and so PLS develops as a novel and powerful substitute of it. Different Security techniques of wireless networks are mentioned in Figure 6.6. In our work, any two of the three PLS techniques—wiretap coding, artificial noise, and beamforming—are combined and implemented together (as indicated in Figure 6.6 with a different colour box). Also, comparison of these security techniques in wireless networks is illustrated in Table 6.4.

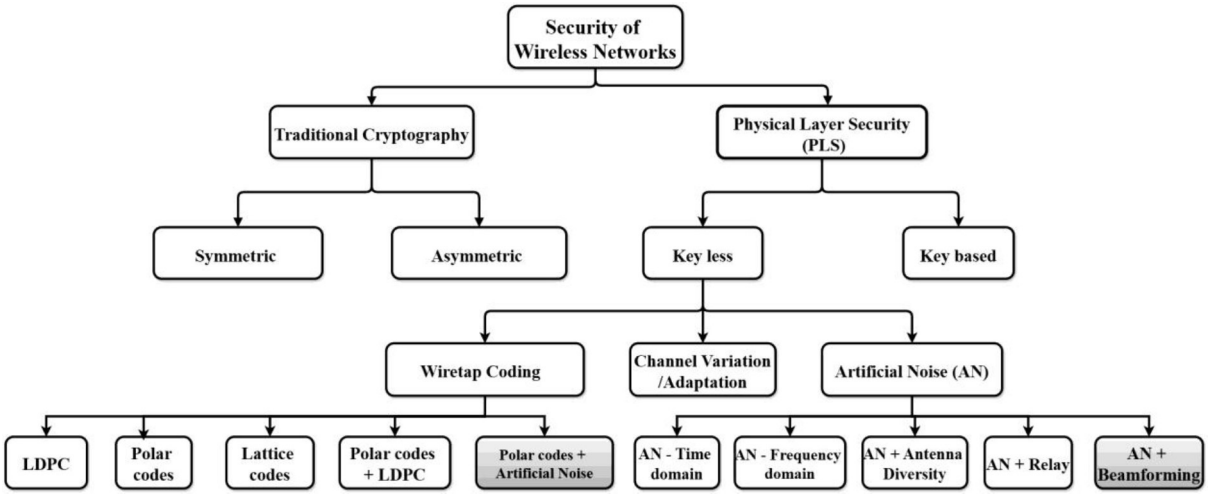


Figure 6.6 Different security schemes of wireless networks.

Table 6.4 Comparison of PLS techniques in wireless networks.

Security technique	**Type**	**Technical characteristics**	**Ability to defend against eavesdropping attacks**	**Ability to defend against interference attacks**	**System complexity**
Wiretap coding	Time domain	Powerful error correction capability	-	High	Low
Artificial noise	Time domain	Increased channel diversity	High	-	High
Beamforming	Spatial domain	Superimposed multiantenna signal	Medium	Low	High

PLS is a promising approach that provides security at the physical layer of communication systems. Unlike traditional cryptography, which assumes a bounded computational power for eavesdroppers, PLS does not make such assumptions. Instead, it focuses on the characteristics of the communication channel to ensure secure transmission. PLS can be considered as a collection of several security methods that exploits the random nature of wireless channels to provide secure communications. PLS methods do not depend on computational complexity. Security and reliability in communication can be attained although the

eavesdroppers (unauthorized smart devices) are armed with adequate computational power. 5G wireless networks have decentralized structures and devices can arbitrarily join in or leave the system. The devices which are linked to the nodes in the network have not the same power and computational capacity at all times. Cryptographic security arrangement comes to be difficult task for this case. PLS produces collectively a well-integrated security solution in 5G wireless network and is applied as extra protection layer above the present security systems using vigilant management and implementation approach. The primary benefits of MIMO multi-antenna schemes depend on improved data rates and improved SNR at distant receivers by directing the signal wirelessly in the path of the preferred receiver, i.e. beamforming (BF) and dropping the signal level spread in all other directions.

In the context of wiretap channels, polar codes can be employed to ensure that the legitimate receiver (Bob) can decode the transmitted message while the eavesdropper (Eve) is left with minimal information. In the practice of AN (Artificial Noise), Alice (sender) adds artificial noise with transmitted data and sends it in all directions except to Bob (designated receiver). If Alice is aware of Eve's CSI (channel state information), then the effect of the AN on Eve can be enhanced by Alice considering Eve is not a passive eavesdropper. Application of AN worsens the channel of potential eavesdroppers but the quality of the channel of designated receiver is not effected at the same time. The secrecy capacity of the channel can be enhanced by carefully allocating power between the message signal and artificial noise (AN).

6.3.2 Role of ECC in PLS

ECC play a crucial role in enhancing the reliability of data transmission over noisy channels. By correcting errors that may occur during transmission, ECC can improve the overall performance of PLS systems. The extended Golay code (24, 12, 8), can detect up to 7-bit errors or correct up to 3-bit errors, making it a robust choice for secure communications. By employing Golay codes in conjunction with cryptosystem, it is possible to achieve secure transmission even in the presence of noise and interference. The performance of Golay codes in cryptosystem can be analyzed through various metrics, including throughput, latency, and resource utilization. Recent studies have shown that architectures based on Golay codes can outperform existing solutions in terms of efficiency and effectiveness. Polar codes are a class of linear block codes that achieve the capacity of symmetric binary-input memoryless channels. They are constructed using a process called channel polarization, which transforms a set of identical channels into a set of channels with varying capacities. The performance of polar codes in PLS applications can be evaluated using metrics such as BLER and security gap. These metrics help to assess the effectiveness of the coding scheme in maintaining low error rates to the legitimate receiver and high error rates to the eavesdropper.

6.4 Modified McEliece cryptosystem employing Extended Golay code

The McEliece cryptosystem generates public and private keys using a linear error-correcting code. The error-correcting code in the McEliece cryptosystem is binary Goppa code [4]. The different alternative codes can be used to determine the secret key. Several variants of the McEliece cryptosystem were put forth employing different secret codes. McEliece cryptosystem using extended Golay code functions similarly to the standard McEliece cryptosystem using Goppa code, however it creates the secret matrix 'G' in a different fashion and employs different decoding methods. The second row of the cyclic Golay code matrix 'B_{24}' is created by shifting the first component to the last position (as shown in equation 2.2). Similar to this, every row of matrix 'B_{24}' may be created by right-shifting the row before it, with the exception of the final row. The matrix 'B_{24}' is a component of the extended Golay code's generator and parity check matrices, and its decoding is fairly

straightforward. As long as the selected sparse random matrix ‘S’ is a non-singular invertible matrix, the constructed system is efficient and secure. Each codeword can correct up to 3 bit errors and is effective due to sparse matrices. The random permutation matrix ‘P’ and non-singular invertible matrix ‘S’ is presented in equation 6.1 and 6.2 respectively. The permutation matrix ‘P’ has exactly single ‘1’ in each row & column and other entries are ‘0’.

$$P = \begin{bmatrix}
0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0\\
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0\\
0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0\\
0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0
\end{bmatrix} \tag{6.1}$$

$$S = \begin{bmatrix}
1&1&0&0&0&0&0&0&1&1&0&1\\
0&1&0&1&1&1&1&1&1&0&1&1\\
1&0&0&1&0&1&0&0&1&0&1&1\\
1&1&1&0&0&0&1&1&0&1&1&1\\
1&1&0&0&0&0&1&1&0&1&1&0\\
1&0&0&0&0&0&1&1&0&1&0&1\\
0&1&0&1&1&1&1&1&1&0&1&0\\
0&1&1&0&1&1&1&0&1&0&1&0\\
1&0&0&1&1&0&0&1&1&1&0&0\\
1&1&1&1&1&0&1&1&0&1&0&0\\
1&0&1&1&0&0&0&1&1&0&0&1\\
1&0&1&0&0&1&0&0&1&0&0&0
\end{bmatrix} \tag{6.2}$$

Additionally, a finite state machine is developed for the decoding part and used extended Golay code to design the modified McEliece cryptosystem. The proposed cryptosystem using extended binary Golay code is tested by MATLAB program and the description of the extended Golay code-based McEliece cryptosystem's detailed algorithm is provided in Algorithm 6.1 - 6.4. Figure 6.7 depicts the proposed McEliece cryptosystem based on extended Golay code.

Algorithm 6.1 Key generation.

Parameter:

F = family of t-error correcting (n, k, d) codes with t << n;

S = Non-singular invertible matrix;

Public key: G_m ; Private key: (S, G_n)

Input:

Take G_{24} = extended Golay code (24, 12, 8) which encodes message length k = 12 bit in a codeword length n = 24 and any t = 3 bit errors can be corrected.

Output:

$G \leftarrow [I, B]$ % Generate (k × n) generator matrix G

Generate P % Generate (n × n) permutation matrix P, having exactly 1 in each row & column; with other entries as 0.

$G_1 \leftarrow GP$ % Compute G_1

$G_n \leftarrow G_1$ % Arrange G_1 in systematic format of generator matrix

$S \in F^{k \times k}$ % Generate S which is a random, sparse and invertible binary matrix

$G_m \leftarrow SG_n$ % Compute G_m = Public key

Return G_m & *(S, G_n,)* % Public key: G_m ; Private key: (S, G_n)

Algorithm 6.2 Encoding.

Parameter:

G_m = Public key; m = Message; e = Error vector; c = Cipher text.

Input:

$G_m \in F^{k \times n}$ % G_m = Public key

$m \in F^k$ % m = Message

$e \in F^n$ % e = Error vector

Output:

$y \leftarrow mG_m$ % Compute codeword y

$c \leftarrow y + e$ % Compute c = Cipher text

Return c

Algorithm 6.3 Decoding.

Parameter:

Private key = (S, G_n); m = Message; e = Error vector; c = Cipher text.

Input:

(S, G_n) % Private key = (S, G_n)

$c \in F^n$ % c = Cipher text

Output:

e ←D (c, G_n) % Calculate error vector e by calling subroutine D (c, G_n)

$y_1 \leftarrow c + e$ % Compute encoded message y_1

$y_1 \leftarrow mSG_n + e$ % Update y_1

$mS \leftarrow [G_n^t \mid (mSG_n)^t]$ % Compute mS by row reducing $[G_n^t \mid (mSG_n)^t]$

$m \leftarrow (mS)S^{-1}$ % Compute m = Message

Return m

Algorithm 6.4 Subroutine D (c, G_n)

Parameter:

Generator matrix (Private) = G_n; e = Error vector; c = Cipher text;

l_i = 12 length codeword with 1 in the i^{th} position and 0 elsewhere in I_{12} identity matrix.

Input:

G_n % Compute generator matrix (Private) = G_n

$c \in F^n$ % Compute c = Cipher text

Output:

$S_1 \leftarrow cG_n$ % Compute first syndrome S_1

If weight(S_1) ≤ 3,

Return $e \leftarrow [S_1, 000000000000]$

Else If weight(S_1+B_i) ≤ 2,

Return $e \leftarrow [S_1 + B_i, l_i]$

Else $S_2 \leftarrow S_1 B$ % Compute second syndrome S_2

If weight(S_2) ≤ 3,

Return $e \leftarrow [\,000000000000, S_2]$

Else If weight(S_2+B_i) ≤ 2,

Return $e \leftarrow [l_i,\ S_2 + B_i,]$

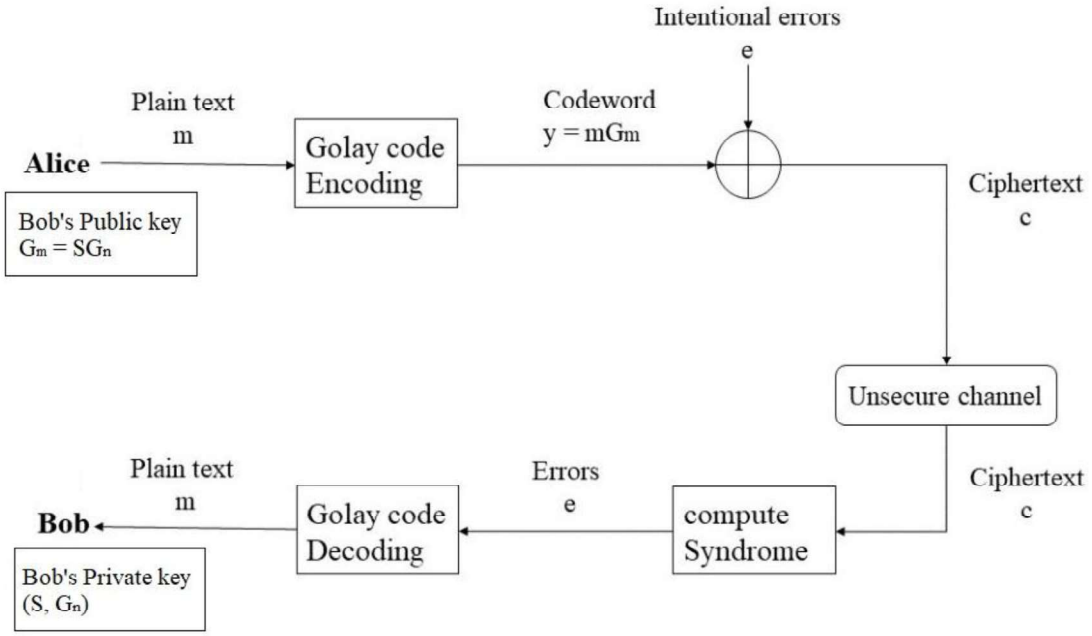


Figure 6.7 Proposed McEliece cryptosystem based on extended Golay code.

6.4.1 Proposed Encoder and Decoder design for Extended Golay code

Encoder design: Figures 6.8, 6.9 and 6.10 depict optimised structures for encoders. The entire structure is split into three sections displaying the generation of extended Golay code generator matrix G_{24} from binary Golay code generator matrix G_{23}. As shown in Figure 6.8, input message and resultant polynomial are saved in registers R8 and R2, respectively. A 2:1 multiplexer selects the data from R8. Initially, the control signal 'Rst' is set to 'HIGH', but it remains 'LOW' through out the polynomial division process. During each step of polynomial division, binary XOR operation takes place for modulo-2 subtraction. A 12:4 priority encoder is used to efficiently detect how many zeros present before first '1' in R3. A condition block then checks if the priority encoder's output is greater than the content of R5. If the condition satisfies then the residual result is circularly left-shifted by the amount in R5; otherwise it is shifted by the priority encoder's output value O[3:0], and the result is stored in R4. Since 'Rst' is 'LOW', R1 gets updated with the value of R4, and this process repeats until the final iteration. Another 2:1 multiplexer is controlled by the output of R7, which becomes 'HIGH' when the priority encoder output O[3:0] exceeds the value of R5.

Figure 6.9 shows iteration control unit, which consists of a 2:1 multiplexer and a subtractor. At the beginning, when the control signal 'Rst' is 'HIGH', register R5 holds the value 11, representing the weight of the polynomial. During the division process, as 'Rst' remains 'LOW', R5 is updated with the value from R7 after each iteration based on the multiplexer's selection. The output of the priority encoder is the other input to the subtractor. At the end of the final iteration, the subtractor's result becomes zero and is stored in R7. At this point, register R6 is loaded with the 23-bit Golay (23, 12) codeword formed by combining the contents of R8 and R9. This loading happens only when the control signal 'Cs' is 'HIGH', indicating the completion of the division process.

Figure 6.10 presents the architecture used to add an extra parity bit to a 23-bit binary Golay codeword (G23), creating the extended 24-bit Golay codeword (G24). The 23-bit binary Golay codeword, stored in register R6, is generated by the architecture shown earlier in

Figure 6.8. The most significant (MSB) 12 bits of R6 are stored in R12, whereas R13 contains least significant (LSB) 11 bits of R6. The total weight of the codeword is calculated by adding the weight 1 (weight of R12) and weight 2 (weight of R13), and this result is saved in R10. The R6' contains the data of R6 appended with '0' and R6" contains the data of R6 appended with '1'. A 2:1 multiplexer selects between R6' and R6" based on the LSB of R10. The final 24-bit extended Golay codeword is then saved in register R11.

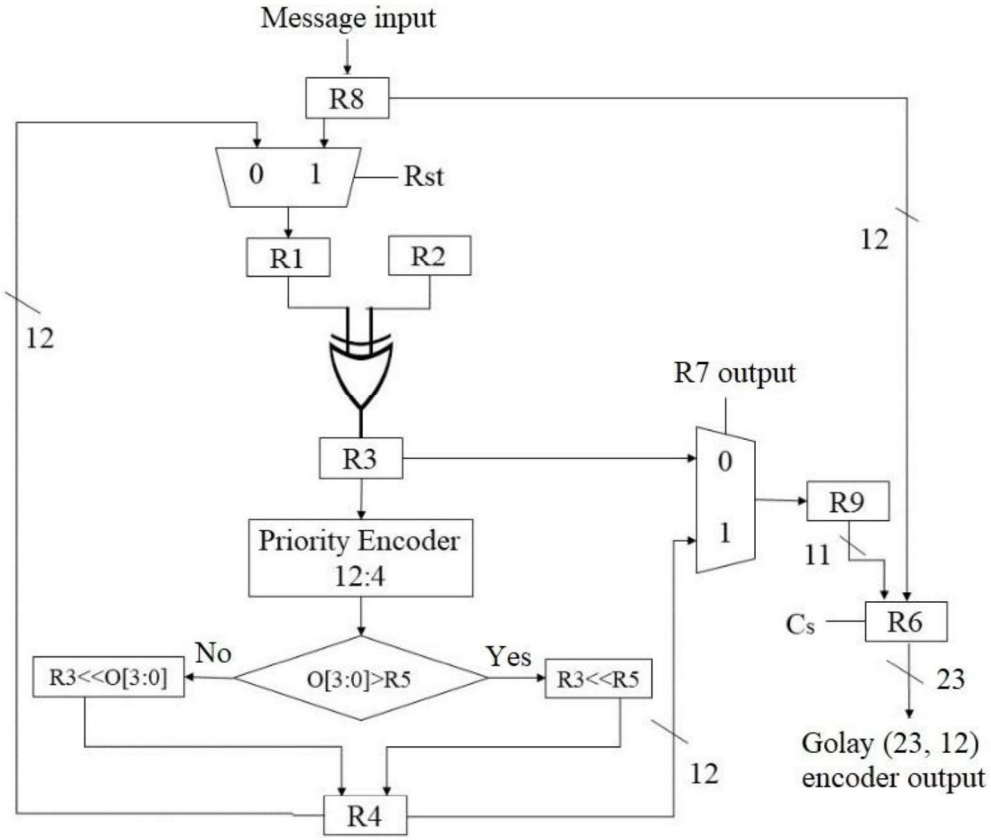


Figure 6.8 Proposed structure for producing binary Golay code.

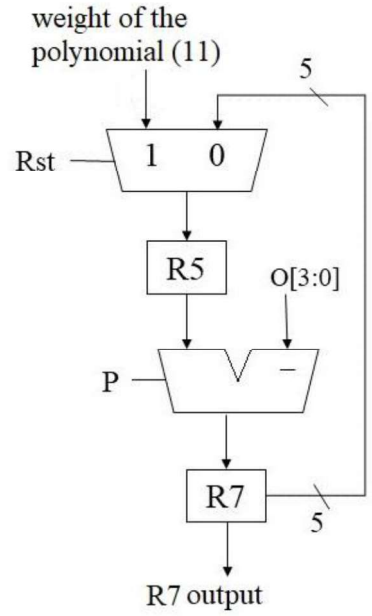


Figure 6.9 Proposed structure for iteration control unit.

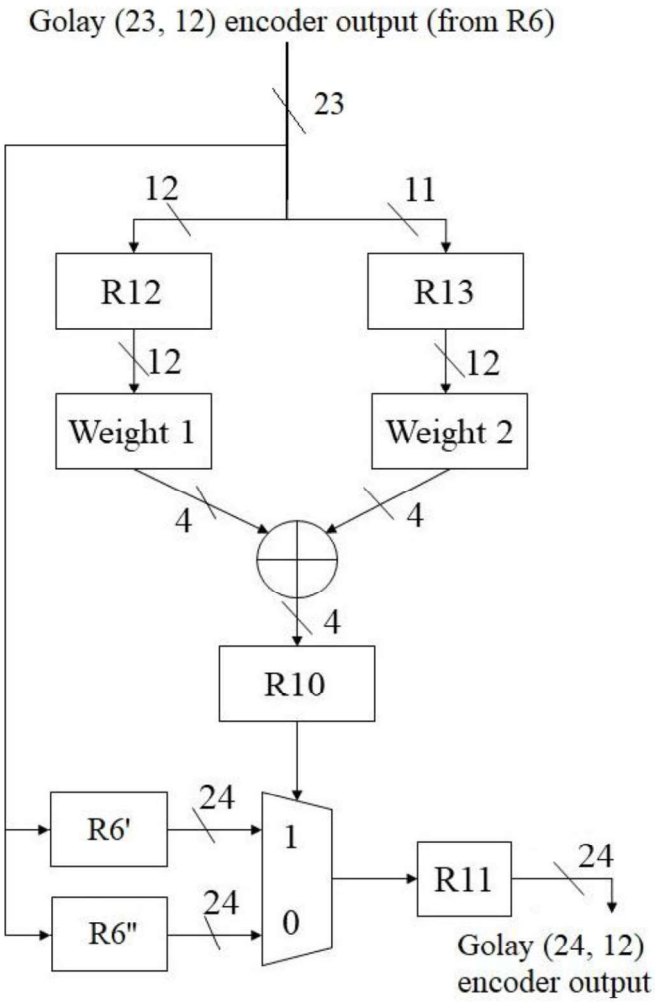


Figure 6.10 Proposed structure for extended Golay code generation from binary Golay code.

Decoder design: Assume that e represents the error pattern, b_i is the ith row of B_{24}, and y is the received codeword.

(i) Syndrome computation unit: By multiplying the received codeword r with the parity check matrix H, syndrome S is assessed. As a result, the following is the logical statement for computing the MSB bit of syndrome vector:

$$S[11] = r[23] \oplus r[11] \oplus r[10] \oplus r[8] \oplus r[7] \oplus r[6] \oplus r[2] \oplus r[0] \quad (6.3)$$

Figure 6.11 depicts the delay optimization construction for the S[11] bit of the syndrome vector. The logical equations for computing other 11 syndrome bits are as follows:

$$S[10] = r[20] \oplus r[19] \oplus r[18] \oplus r[17] \oplus r[16] \oplus r[14] \oplus r[13] \oplus r[10] \quad (6.4)$$

$$S[9] = r[20] \oplus r[19] \oplus r[18] \oplus r[17] \oplus r[15] \oplus r[14] \oplus r[12] \oplus r[9] \quad (6.5)$$

$$S[8] = r[21] \oplus r[20] \oplus r[19] \oplus r[18] \oplus r[16] \oplus r[15] \oplus r[13] \oplus r[8] \quad (6.6)$$

$$S[7] = r[22] \oplus r[21] \oplus r[20] \oplus r[19] \oplus r[17] \oplus r[16] \oplus r[12] \oplus r[7] \quad (6.7)$$

$$S[6] = r[22] \oplus r[21] \oplus r[20] \oplus r[18] \oplus r[17] \oplus r[13] \oplus r[12] \oplus r[6] \quad (6.8)$$

$$S[5] = r[22] \oplus r[21] \oplus r[19] \oplus r[18] \oplus r[14] \oplus r[13] \oplus r[12] \oplus r[5] \quad (6.9)$$

$$S[4] = r[22] \oplus r[20] \oplus r[19] \oplus r[15] \oplus r[14] \oplus r[13] \oplus r[12] \oplus r[4] \quad (6.10)$$

$$S[3] = r[22] \oplus r[21] \oplus r[20] \oplus r[18] \oplus r[14] \oplus r[13] \oplus r[12] \oplus r[3] \quad (6.11)$$

$$S[2] = r[23] \oplus r[22] \oplus r[21] \oplus r[19] \oplus r[15] \oplus r[14] \oplus r[13] \oplus r[2] \quad (6.12)$$

$$S[1] = r[23] \oplus r[22] \oplus r[20] \oplus r[16] \oplus r[15] \oplus r[14] \oplus r[12] \oplus r[1] \quad (6.13)$$

$$S[0] = r[23] \oplus r[21] \oplus r[17] \oplus r[16] \oplus r[15] \oplus r[13] \oplus r[12] \oplus r[0] \quad (6.14)$$

These equations give the full 12-bit syndrome vector S = {S[0], S[1], ..., S[11]}, and each bit indicates parity-check failure locations if errors exist.

(ii) (s + b_i) and (SB + b_i) calculation: By flipping part of the bits in registers S and SB or leaving them unaltered, this unit computes (S + b_i) and (SB + b_i), where $1 \leq i \leq 12$.

(S + b_1)'s logical contents are given by

$$(S + b_1) = \{\sim S[11], \sim S[10], S[9], \sim S[8], \sim S[7], \sim S[6], S[5], S[4], S[3], \sim S[2], S[1], \sim S[0]\}$$

where symbol ~ stands for logical NOT.

S is switched out for SB in each expression to calculate (SB + b_i). Since there are two possible formats for the parity check matrix - either [I|B] or [B|I], it is possible to get two different syndromes, namely S and SB.

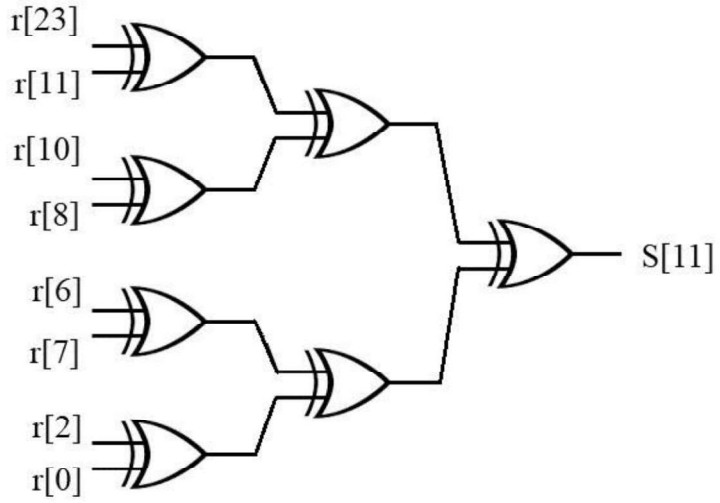


Figure 6.11 Delay optimization construction for S[11] of the syndrome vector.

6.4.2 Synthesis results of extended Golay Encoder and Decoder

Utilizing the Xilinx ISE tool, the structure of the extended Golay encoder and decoder has been built on the FPGA (Field Programmable Gate Array) based virtex-4 platform. Contrary to [66], Table 6.5 shows that the suggested encoder architecture does not have any additional clocking mechanisms, which minimises both area and power consumption. Additionally, as opposed to the 23 clock cycles shown in [66], the proposed architecture's latency is 12 clock cycles. Comparison of LUT, Slices or Area and Synthesized frequency of proposed encoder architechture with existing is shown in Table 6.6. The designed encoder architecture can handle MSB messages with '0' and '1' inputs, which is crucial for error-correcting codes. The implementation results, which came at a acceptable area and delay time overhead, confirmed

the accuracy of the designed architecture. Table 6.7 compares the throughput and latency metrics. The proposed decoder design is promising for systems with high data rates. Comparison of LUT, Slices or Area and Synthesized frequency of proposed decoder architechture with existing is shown in Table 6.8.

Table 6.5 Comparison of different extended Golay encoder structure's clocking and latency mechanisms.

Scheme	Latency (clock cycles)	Clocking mechanism
[66]	23	System clock + clock doubler
[70]	12	System clock
[74]	12	
Proposed	12	

Table 6.6 Comparison of different extended Golay encoder structure's LUT, slices and frequency of operation.

Scheme	Number of LUT	Number of Slices (Area)	Frequency (MHz) of operation
[70]	187	103	238.575
[74]	149	85	380.967
[75]	191	102	162.425
Proposed	119	73	344.827

Table 6.7 Latency and throughput comparison of different extended Golay decoder structure.

Scheme	Latency (clock cycles)	Throughput
[72]	576	1 output/ 144 clock cycle
[73]	48	1 output/ 24 clock cycle
[70]	27	1 output/ clock cycle
[74]	24	
Proposed	24	

Table 6.8 Comparison of LUT, slices (area) and frequncy of different extended Golay decoder structure.

Scheme	Number of LUT	Number of Slices (Area)	Frequency (MHz) of operation
[70]	785	230	195.08
[74]	113	60	220.6
Proposed	121	56	318.47

Table 6.9 and 6.10 show the Area-Delay improvement in the proposed design compared to other existing works in terms of Golay encoder and decoder construction respectively. Also, comparison of Area-Delay Product (ADP) is presented in both the tables.

Table 6.9 Comparison of different parameters of the extended Golay encoder design.

Scheme	Frequency (MHz) of operation	Number of LUT (Area)	Latency (clock cycles)	Delay (ns)	Area-Delay Product (ADP)	Improv. (%) in Proposed design	
						Area	Delay
[70]	238.575	187	12	50.29	9404.23	36.36	30.80
[74]	380.967	149	12	31.49	4692.01	20.13	10.51
[75]	162.425	191	12	73.88	14111.08	37.69	52.89
Proposed	344.827	119	12	34.80	4141.20	-	-

Table 6.10 Comparison of different parameters of the extended Golay decoder design.

Scheme	Frequency (MHz) of operation	Number of LUT (Area)	Latency (clock cycles)	Delay (ns)	Area-Delay Product (ADP)	Improv. (%) in Proposed design	
						Area	Delay
[70]	195.08	785	27	138.40	108644	84.58	45.54
[74]	220.60	113	24	108.79	12293.27	07.07	30.72
Proposed	318.47	121	24	75.36	9118.56	-	-

6.5 Artificial Noise-aided secure polar coding for Wireless Networks

The AN-aided security permits sender to produce interfering random signals or artificial noise to affect badly the eavesdropper only whereas the desired receiver gets clean message signal. Thus wiretap channel's capacity is reduced and secrecy capacity is increased although main channel's capacity remained unchanged. Some fraction of sender's transmit power is allotted to produce AN by deteriorating only the wiretap channel whereas desired wireless transmission from sender to legitimate receiver in the main channel remains unaffected. Thus, a significant security enhancement is done by using AN. The AN-aided security is capable to promise secure wireless transmission by wasting power as a fraction of transmit power is allotted for generating AN. Security oriented beamforming permits sender to transmit message signal in a specific direction to the appropriate receiver and the eavesdropper receives destructive interference and weak signal. So, the legitimate receiver's signal strength becomes much higher than the eavesdropper leading to a favourable secrecy capacity improvement. Security and reliability were noticeably improved in combining the artificial noise and beamforming. Secure wireless transmission using AN-aided beamforming is shown in Figure 6.12 and implemented in our proposed work to compare performance analysis with AN-aided polar coding as shown in Figure 6.13. Both CA-polar and DCA-polar coding methods are incorporated with artificial noise generation in our proposed work. ECC can be used to secure messages from eavesdroppers over noisy wiretap channels to attain reliable and secure communications.

Polar codes are selected as a preferred 5G-NR coding scheme and aided with distributed CRC as well as input bit interleaver is facilitated. By bit interleaving between the CRC encoder and the polar encoder, distributed CRC bits are acquired. Moreover, after the last bit necessary for calculation, the CRC bit is placed. Thus, the decoding complexity is diminished by early terminating the decoding as soon as improper check is encountered, or else error-correction is enhanced by trimming the decoding tree. The interleaver equally

distributes the CRC bits within the information bits even though CRC remainder bits are fall upon after appropriate information bits during the decoding. By using this idea, DCA-polar decoder eases the decoding complexity implementing early termination of the decoding when each path met improper check. The performance of the decoder is enhanced trimming SCL decoding tree by the distributed CRC bits. Figure 6.14 depicts the DCA polar code design scheme and CRC generator polynomials of polar codes are given in Table 6.11.

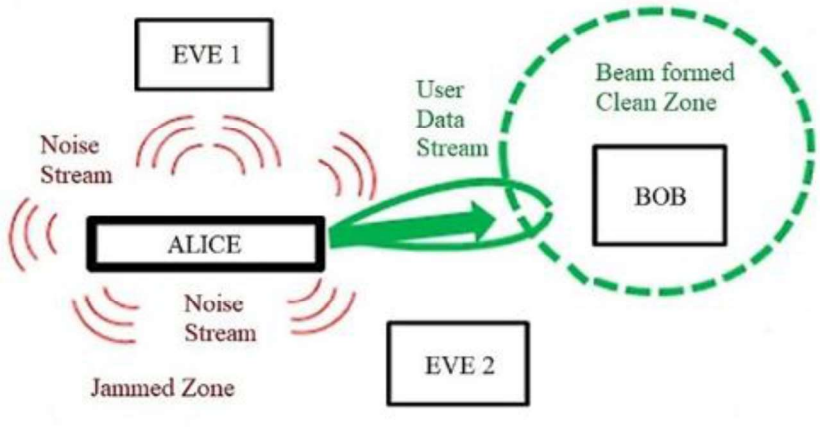


Figure 6.12 AN-aided beamforming scheme.

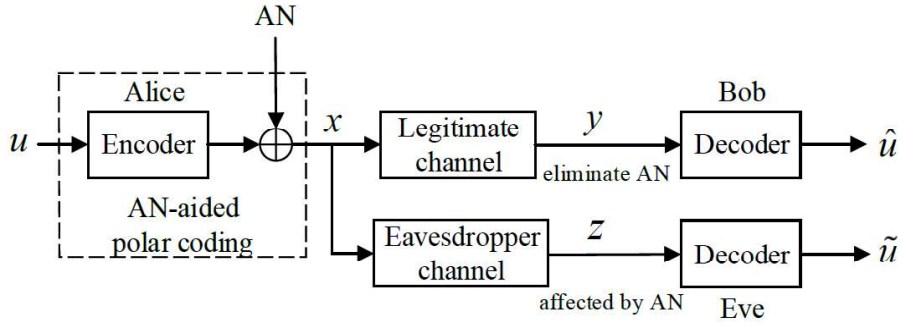


Figure 6.13 AN-aided polar coding scheme.

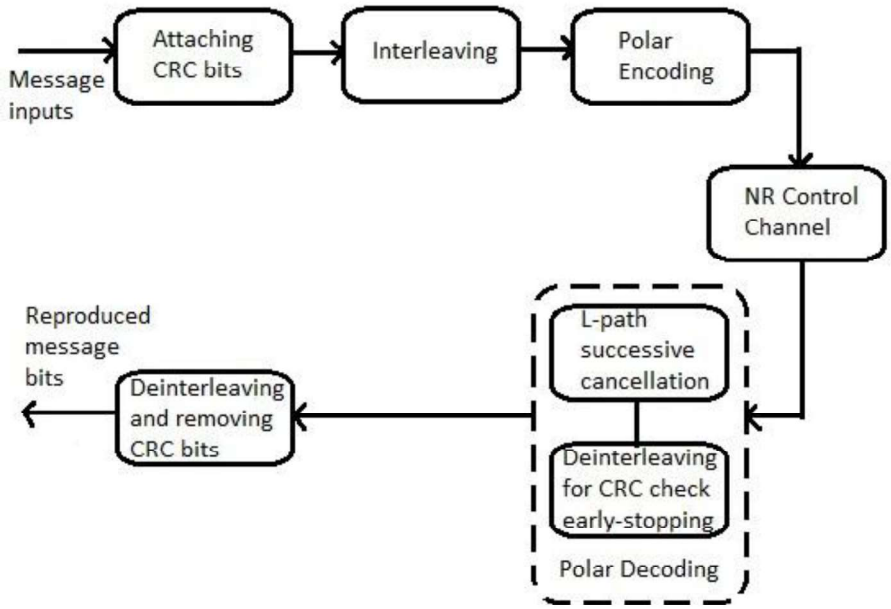


Figure 6.14 DCA-polar coding scheme.

Table 6.11 CRC generator polynomials of polar codes.

Information block lengths	CRC bits (P)	CRC generator polynomials	ECC
$A = 32$	11 bits	$g_{CRC11}(x) = [x^{11} + x^{10} + x^{9} + x^{5} + 1]$	CA-polar
$A = 32$	24 bits	$g_{CRC24}(x) = [x^{24}+x^{23}+x^{21}+x^{20}+x^{17}+x^{15}+x^{13}+x^{12}+x^{8}+x^{4}+x^{2}+x+1]$	DCA-polar

Error performances analysis using proposed methods:

This section evaluates BLER and security gap to examine secrecy performance metrics of the intended recipient, Bob and the eavesdropper, Eve in a secure MISO communication system using both AN-aided beamforming and AN-aided polar coding scheme.

Figure 6.15 shows the total transmit power percentage allotted to AN generation is fixed at 0%, 30%, 60% and 90%. The eavesdropper, Eve's BLER is significantly raised since a 30% of the power usage for AN generation in comparison to her BLER in absence of AN. Though Eve's BLER was rising greater than the legitimate receiver Bob's BLER for SNR above -5 dB because of the Transmit Beamforming (Tx BF) by the sender Alice, the complimentary effects of the AN's presence is noticed. For higher SNR values in Eve's case, increment in the SNR for the message signal increases the SNR of the AN leaving Eve with no advantage in BLER in the higher SNR. In contrast, Bob's BLER is marginally affected because of the AN being produced depending on Bob's CSI because the portion of total transmit power allotted to the message signal has declined to 70% from 100% in absence of AN. When AN power allocation is increased to 60% and 90%, it impact Eve's BLER greatly in comparison to Bob's BLER too. It is visible in the figure that Bob's BLER is always lower than Eve's BLER and this is highly desirable for secure communication. The simulation results in this figure showed usage of AN-aided beamforming significantly improve the secrecy level of transmitted information among a sender and an appropriate receiver incrementing the interception difficulty of an eavesdropper. While the addition of AN marginally worsen the appropriate receiver's performance, the impression over the eavesdropper is considerably larger creating AN-aided beamforming as a significant technique of a PLS scheme.

The simulation results of AN-aided polar coding techniques using both CA-polar and DCA-polar scheme for secure MISO communication system are shown in Figure 6.16. DCA-polar scheme is found to be superior in terms of small security gap in comparison with CA-polar scheme for AN-aided polar coding technique. It can be identified seeing the result that Bob's BLER is always lower than Eve's BLER with respect to same SNR and this is highly desirable for secure and reliable communication. The comparative study of all the simulation results using both AN-aided beamforming and AN-aided polar coding schemes is shown in Figure 6.17. DCA-polar coding based AN-aided polar coding scheme demonstrates best performance metrics in comparison to attempts using other techniques including CA-polar coding and AN-aided beamforming with AN power allocation of 0% to 90%. The optimized AN intended to worsen Eve's channel and code rate rises with higher SNR. Thus, Eve's BLER performance worsens with growing SNR. The power requirement for AN generation in the AN-aided polar coding schemes are lesser than the AN-aided beamforming techniques. So AN-aided polar coding is more efficient in terms of precious power resource saving and useful for applications in power constraint portable electronic

devices. With the offered performance metric, a longer gap between the BLERs of Bob and Eve can be noticed. This longer gap indicates higher secrecy capacity attained and confirms the effectiveness of our work.

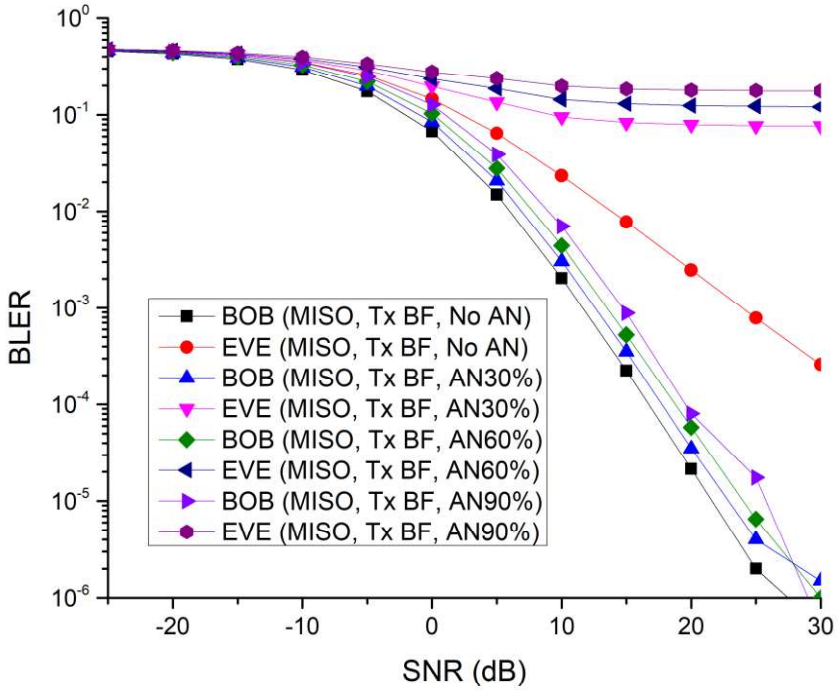


Figure 6.15 Graph of BLER vs. SNR using AN-aided beamforming.

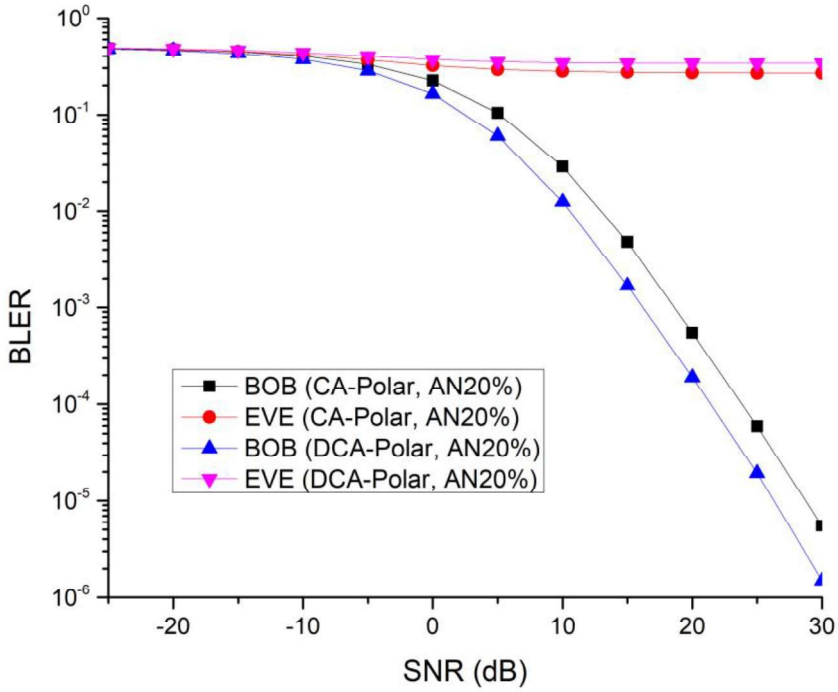


Figure 6.16 Graph of BLER vs. SNR using AN-aided polar codes.

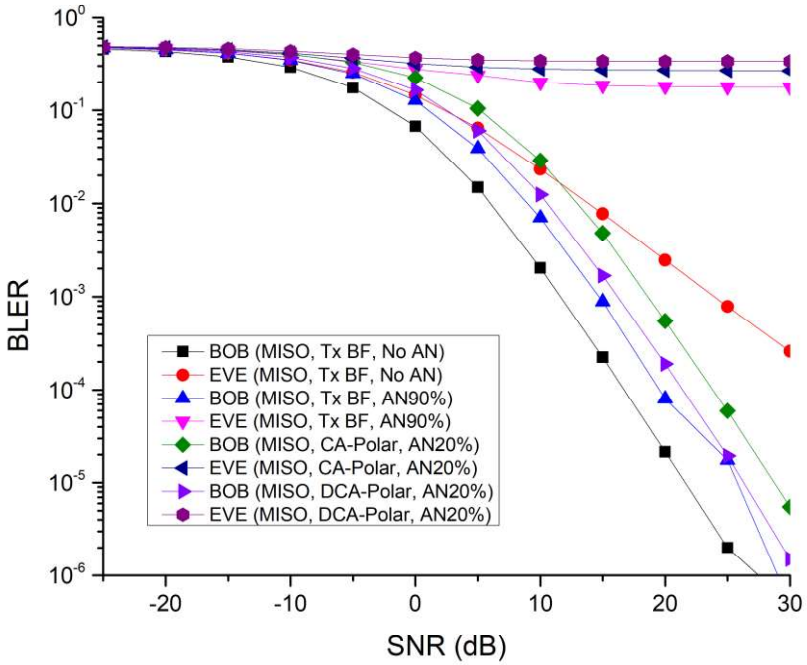


Figure 6.17 Comparative graph of BLER vs. SNR using AN-aided beamforming and AN-aided polar codes.

6.6 Summary

Efficient hardware structure for extended binary Golay code have been developed and employed for the proposed modified McEliece cryptosystem. The shortcomings of the existing architectures are omitted by the proposed structures and outperforms in terms of synthesis results considering various performance metrics. High throughput, minimal latency and low area are achieved for both hardware modules. For secure high-speed communications, the proposed Golay encoder and decoder is a promising option in the McEliece cryptosystem. This chapter also deals with both the aspect of secrecy performance analysis and the operation of the appropriate receiver and the eavesdropper for reliable and secure transmission in a MISO wireless communication system using both AN-aided beamforming and AN-aided polar coding schemes. Rather than equivocation, BLER over SNR is chosen as the security metric along with security gap to measure secrecy capacity. It is illustrated that BLER of the eavesdropper is always higher than BLER of the appropriate receiver with increasing SNR and this is highly desirable as PLS solutions. The proposed AN-aided polar codes are successfully employed attaining higher BLER at the eavesdropper. More practical coding construction approaches are implemented for Gaussian wiretap channels to satisfy both reliability and security constraints of the wireless network.

The integration of ECC, particularly polar and Golay codes, into physical layer security schemes presents a promising avenue for enhancing the security and reliability of wireless communications. By leveraging the unique properties of these codes, it is possible to develop robust PLS solutions that can withstand eavesdropping attempts while ensuring the integrity of transmitted data. This chapter discusses how ECC, particularly polar codes and Golay codes, can be integrated into PLS frameworks to enhance security and reliability. Future research should focus on optimizing these coding schemes for specific applications and exploring new methods to further enhance their performance in real-world scenarios.

Chapter 7

Conclusions and Future Scope

This thesis focused on the design, implementation, and evaluation of two advanced error correction coding schemes, namely polar codes and Golay codes, to address some challenges in communication, security, and networks. The research explored various aspects of ECC such as performance analysis, optimization against different channel models, and integration with security mechanisms. This chapter highlights the key contributions and also provides a few suggestions for extending the current work in future.

7.1. Concluding Remarks

This thesis has advanced the understanding and application of ECC in modern communication systems, with a particular focus on enhancing reliability and security in diverse environments. The proposed coding schemes and optimization strategies demonstrate the potential of ECC to meet the challenges of current and future communication networks. By addressing both theoretical and practical aspects, this research provides a foundation for further exploration and innovation in the field of error correction coding. The ongoing evolution of communication technology will continue to demand more efficient and secure data transmission methods. The insights and findings from this thesis will help to develop next-generation ECC solutions that are adaptable, robust, and capable of supporting the ever-expanding landscape of global connectivity. In this thesis, several significant contributions are presented in the field of error correction coding and its applications in communication, security, and networks.

Key contributions:

(i) Performance Evaluation of polar codes in Diverse Channels: The thesis provides a comprehensive performance evaluation of polar codes for BDC and SSD channels, highlighting their versatility and robustness.

(ii) GA-based polar Code Construction Techniques: The thesis presents an improved method for constructing polar codes using GA to enhance performance against both AWGN and Rayleigh fading channels.

(iii) Design of Concatenated codes for WSNs and 5G NR control channels: Implementation of concatenated polar codes tailored for WSNs and 5G NR control channels is provided in this thesis.

(iv) Integration of ECC for Reliability and Security: The thesis proposes two ECC-based security schemes, including Golay codes for the modified McEliece cryptosystem and AN-aided polar codes for keyless physical layer security, demonstrating the dual benefits of reliability and security.

7.2. Future Scope of Research

While this thesis has addressed many critical aspects of ECC, there are several avenues for future research that can further expand the field:

(i) Future research could explore more sophisticated optimization techniques, such as deep learning-based polar code construction, to further improve error correction performance under dynamic channel conditions. Machine learning algorithms could be used to predict optimal code configurations in real-time, enhancing adaptability and robustness.

(ii) With the evolution of communication standards such as 6G, there is a need to investigate the role of polar codes and other advanced ECC in ultra-reliable low-latency communications and massive machine-type communications. Future studies could focus on optimizing code designs for these next-generation applications, ensuring that they meet the stringent performance and resource constraints of future networks/applications.

(iii) Developing low-complexity decoding algorithms with better performance is crucial for expanding the applicability of polar codes. Future work could explore alternative decoding approaches, such as neural-assisted decoders or approximate computing techniques, to further reduce the computational burden while maintaining error correction capabilities.

Bibliography

[1] C. E. Shannon, “A mathematical theory of communication,” The Bell system Technical Journal, vol. 27, no. 3, pp. 379–423, Jul. 1948.

[2] G. C. C. Jr and J. B. Cain, Error-correction coding for digital communications. Springer Science & Business Media, 2013.

[3] K. D. Rao, *Channel Coding Techniques for Wireless Communications*. Singapore: Springer Singapore, 2019. doi: https://doi.org/10.1007/978-981-15-0561-4.

[4] R. E. Blahut, Theory and practice of error control codes. Addison-Wesley Reading, 1983.

[5] S. Lin and D. J. Costello, Error control coding: fundamentals and applications. Prentice-Hall, 1983.

[6] R. W. Hamming, “Error detecting and error correcting codes,” The Bell system technical journal, vol. 29, no. 2, pp. 147–160, Apr. 1950.

[7] M. Y. Rhee, Error-correcting coding theory. McGraw-Hill, Inc., 1989.

[8] V. Pless, Introduction to the theory of error-correcting codes. John Wiley & Sons, 1998.

[9] S. Tong, D. Lin, A. Kavcic, B. Bai, and L. Ping, “On short forward error correcting codes for wireless communication systems,” 2007 16th International Conference on Computer Communications and Networks. IEEE, Aug. 2007, pp. 391–396, doi: https://doi.org/10.1109/ICCCN.2007.4317850.

[10] B. P. Smith, Error-correcting codes for fibre-optic communication systems. University of Toronto, 2011.

[11] T. R. Rao and E. Fujiwara, Error-control coding for computer systems. Prentice-Hall, Inc., 1989.

[12] S. A. Alabady, M. F. Salleh, and F. Al-Turjman, “A novel approach of error detection and correction for efficient energy in wireless networks,” *Multimedia Tools and Applications*, vol. 78, pp. 1345–1373, Jan. 2019, doi: https://doi.org/10.1007/s11042-018-6282-0.

[13] G. A. Hussain and L. Audah, “RS codes for downlink LTE system over LTE-MIMOchannel,” TELKOMNIKA (Telecommunication Computing Electronics and Control), vol. 16, no. 6, pp. 2563–2569, Dec. 2018, doi: https://doi.org/10.12928/telkomnika.v16i6.9177.

[14] M. Shirvanimoghaddam *et al.*, “Short Block-Length Codes for Ultra-Reliable Low Latency Communications,” *IEEE Communications Magazine*, vol. 57, no. 2, pp. 130–137, Feb. 2019, doi: https://doi.org/10.1109/mcom.2018.1800181.

[15] T. H. Liew and L. Hanzo, “Space-time codes and concatenated channel codes for wireless communications,” Proceedings of the IEEE, vol. 90, no. 2, pp. 187–219, Feb, 2002, doi: https://doi.org/10.1109/5.989869.

[16] C. R. Berger, S. Zhou, Y. Wen, P. Willett, and K. Pattipati, “Optimizing joint erasure- and error-correction coding for wireless packet transmissions,” *IEEE Transactions on*

Wireless Communications, vol. 7, no. 11, pp. 4586–4595, Nov. 2008, doi: https://doi.org/10.1109/t-wc.2008.070581.

[17] R. M. Islam, "Error correction codes in wireless sensor network: An energy aware approach," International Journal of Electronics and Communication Engineering, vol. 4, no. 1, pp. 26–31, Jan. 2010.

[18] N. A. Alrajeh, U. Marwat, B. Shams, and S. S. Shah, "Error correcting codes in wireless sensor networks: an energy perspective," *Applied Mathematics & Information Sciences*, vol. 09, no. 2, pp. 809–818, Jan. 2015.

[19] W. Wu, D. Haccoun, R. Peile, and Y. Hirata, "Coding for satellite communication,"*IEEE Journal on Selected Areas in Communications*, vol. 5, no. 4, pp. 724–748, May 1987, doi: https://doi.org/10.1109/jsac.1987.1146583.

[20] J. Cantillo, J. Lacan, and I. Buret, "A CRC usefulness assessment for adaptation layers in satellite systems," 24th AIAA International Communications Satellite Systems Conference, Jun. 2006, p. 5358, doi: https://doi.org/10.2514/6.2006-5358.

[21] Y. Liu, Y. Guan, J. Zhang, G. Wang, and Y. Zhang, "Reed-Solomon codes for satellite communications," 2009 IITA International Conference on Control, Automation and Systems Engineering (CASE 2009), Jul. 2009, pp. 246–249, doi: : https://doi.org/10.1109/CASE.2009.30.

[22] E. W. Ryan et al., "An introduction to LDPC codes," CRC Handbook for Coding and Signal Processing for Recording Systems, vol. 5, no. 2, pp. 1–23, Sep. 2004.

[23] C. Zhang, X. Mu, J. Yuan, H. Li, and B. Bai, "Construction of multi-rate quasi-cyclic LDPC codes for satellite communications," *IEEE Transactions on Communications*, vol. 69, no. 11, pp. 7154–7166, Nov. 2021, doi: https://doi.org/10.1109/TCOMM.2021.3107578.

[24] B. Wang, P. Chen, Y. Fang, and F. C. Lau, "The design of vertical RS-CRC and LDPC code for ship-based satellite communications on-the-move," *IEEE Access*, vol. 7, pp. 44977–44986, Jan. 2019, doi: https://doi.org/10.1109/access.2019.2895746.

[25] K. Liu, M. Zhu, M. Jiang, C. Zhao, and X. Zhang, "Efficient decoding of parity check concatenated RS codes," 2022 IEEE 8th International Conference on Computer and Communications (ICCC). IEEE, Dec. 2022, pp. 1676–1681, doi: https://doi.org/10.1109/iccc56324.2022.10065615.

[26] P. M. Shah, P. D. Vyavahare, and A. Jain, "Modern error correcting codes for 4G and beyond: Turbo codes and LDPC codes," Radio and Antenna Days of the Indian Ocean (RADIO), Sep. 2015, pp. 1–2, doi: https://doi.org/10.1109/RADIO.2015.7323369.

[27] R. F. Ormondroyd and J. J. Maxey, "Performance of low-rate orthogonal convolutional codes in DS-CDMA applications," *IEEE Transactions on Vehicular Technology*, vol. 46, no. 2, pp. 320–328, May 1997, doi: https://doi.org/10.1109/25.580770.

[28] D. R. Cideciyan, E. Eleftheriou, and M. Rupf, "Concatenated Reed-Solomon/convolutional coding for data transmission in CDMA-based cellular systems," *IEEE Transactions on Communications*, vol. 45, no. 10, pp. 1291–1303, Oct. 1997, doi: https://doi.org/10.1109/26.634693.

[29] P. K. Frenger, P. Orten, T. Ottosson, and A. B. Svensson, "Rate-compatible convolutional codes for multirate DS-CDMA systems," *IEEE Transactions on*

Communications, vol. 47, no. 6, pp. 828–836, Jun. 1999, doi: https://doi.org/10.1109/26.771338.

[30] D. J. Shah, V. K. Patel, and H. A. Patel, “Performance analysis of Turbo code for CDMA 2000 with convolutional coded IS-95 system in wireless communication system,” 2nd International Conference on Electronic Computer Technology (ICECT), May. 2010, pp. 42–45, doi: https://doi.org/10.1109/ICECTECH.2010.5479994.

[31] P. Luukkanen and P. Zhang, "Comparison of optimum and sub-optimum turbo decoding schemes in 3rd generation cdma2000 mobile system," *WCNC. 1999 IEEE Wireless Communications and Networking Conference (Cat. No.99TH8466)*, May 1999, pp. 437-441 vol.1, doi: https://doi.org/10.1109/WCNC.1999.797863.

[32] Q. Li and N. S. Ramesh, “Channel coding performance in cdma2000 systems,” IEEE Emerging Technologies Symposium on Broadband, Wireless Internet Access. Digest of Papers (Cat. No. 00EX414), Apr. 2000, p. 5, doi: https://doi.org/10.1109/ETS.2000.916513.

[33] S. H. Gupta and B. Virmani, “LDPC for Wi-Fi and WiMAX technologies,” 2009 international conference on emerging trends in electronic and photonic devices & systems, Dec. 2009, pp. 262–265, doi: https://doi.org/10.1109/ELECTRO.2009.5441120.

[34] 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,” 2007 Design, Automation & Test in Europe Conference & Exhibition, Apr. 2007, pp. 1–6, doi: https://doi.org/10.1145/1266366.1266437.

[35] I. Tsatsaragkos and V. Paliouras, “A reconfigurable LDPC decoder optimized for 802.11 n/ac applications,” IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 26, no. 1, pp. 182–195, Sep. 2017, doi: https://doi.org/10.1109/tvlsi.2017.2752086.

[36] R. J. McEliece and L. Swanson, “Reed-Solomon codes and the exploration of the solar system,” 1994.

[37] E. Arikan, “Systematic polar coding,” IEEE Communications Letters, vol. 15, no. 8, pp. 860–862, Jun. 2011, doi: https://doi.org/10.1109/lcomm.2011.061611.110862.

[38] E. Arikan, “A performance comparison of polar codes and Reed-Muller codes,” IEEE Communications Letters, vol. 12, no. 6, pp. 447–449, Jun. 2008, doi: https://doi.org/10.1109/lcomm.2008.080017.

[39] Z. B. K. Egilmez, L. Xiang, R. G. Maunder, and L. Hanzo, “The development,operation and performance of the 5G polar codes,” IEEE Communications Surveys & Tutorials, vol. 22, no. 1, pp. 96–122, Jan. 2020, doi: https://doi.org/10.1109/comst.2019.2960746.

[40] S. Huilgol, “Channel coding techniques for 5G using polar codes,” Ph.D. dissertation, 2017.

[41] O. Mouhoubi, C. A. Nour, and A. Baghdadi, “Latency and Complexity Analysis of Flexible Semi-Parallel Decoding Architectures for 5G NR Polar Codes,” *IEEE Access*, vol. 10, pp. 113980–113994, 2022, doi: https://doi.org/10.1109/access.2022.3216292.

[42] H. Mahdavifar, M. El-Khamy, J. Lee, and I. Kang, “Performance Limits and Practical Decoding of Interleaved Reed-Solomon Polar Concatenated Codes,” *IEEE transactions on communications*, vol. 62, no. 5, pp. 1406–1417, May 2014, doi: https://doi.org/10.1109/tcomm.2014.050714.130602.

[43] J. Zhao, W. Zhang, Y. Liu, J. Gao, and R. Zhang, “A Rate-Matching Concatenation Scheme of Polar Codes With Outer Reed-Solomon Codes,” *IEEE Wireless Communications Letters*, vol. 10, no. 3, pp. 459–463, Mar. 2021, doi: https://doi.org/10.1109/lwc.2020.3033850.

[44] Y. Wang, K. R. Narayanan, and Y.-C. Huang, “Interleaved Concatenations of Polar Codes With BCH and Convolutional Codes,” *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 2, pp. 267–277, Feb. 2016, doi: https://doi.org/10.1109/jsac.2015.2504320.

[45] X. Li, Q. Yu, Z. Shi, Y. Li, and Q. Yan, “Concatenations of polar codes with outer nonbinary LDPC codes,” in 2017 IEEE 17th International Conference on Communication Technology (ICCT). IEEE, Oct. 2017, pp. 117–121, doi: https://doi.org/10.1109/icct.2017.8359615.

[46] S. Zengyou and L. Huanhuan, “Application of concatenated codes in the IEEE802. 11b channel coding,” Proceedings of the 3rd International Conference on Computer Science and Service System,pp. 386–389, Jun. 2014, Atlantis Press, doi: https://doi.org/10.2991/csss-14.2014.91

[47] P. Hershey, A. Ephremides, and R. Khatri, “Performance of RS-BCH Concatenated Codes and BCH Single-Stage Codes on an Interference Satellite Channel,” *IEEE Transactions on Communications*, vol. 35, no. 5, pp. 550–556, 1987, doi: https://doi.org/10.1109/tcom.1987.1096813.

[48] J. Ke, X. Lu, X. Wang, X. Chen, and S. Tang, “Concatenated Coding for GNSS Signals in Urban Environments,” *Applied Sciences*, vol. 10, no. 18, pp. 6397, Sep. 2020, doi: https://doi.org/10.3390/app10186397.

[49] G. Ricciutelli, T. Jerkovits, M. Baldi, F. Chiaraluce, and G. Liva, “Analysis of the Block Error Probability of Concatenated Polar Code Ensembles,” *IEEE Transactions on Communications*, vol. 67, no. 9, pp. 5953–5962, Jun. 2019, doi: https://doi.org/10.1109/tcomm.2019.2924897.

[50] P. Chen, B. Bai, Z. Ren, J. Wang, and S. Sun, “Hash-Polar Codes with Application to 5G,” *IEEE Access*, vol. 7, pp. 12441–12455, Jan. 2019, doi: https://doi.org/10.1109/access.2019.2892969.

[51] J. King, A. Kwon, H. Yang, W. Ryan, and R. D. Wesel, “CRC-Aided List Decoding of Convolutional and Polar Codes for Short Messages in 5G,” *ICC 2022 - IEEE International Conference on Communications*, pp. 92–97, May 2022, doi: https://doi.org/10.1109/icc45855.2022.9838726.

[52] D. Bhattacharjee, S. Kumar, and A. Kumari, “Implementation of Polar Codes Over Multipath Rayleigh Fading Channel Using Channel Transformation,” *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, pp. 1–6, Jul. 2023, doi: https://doi.org/10.1109/icccnt56998.2023.10308183.

[53] M. Ren, B. Wu, and K. Niu, “Polar Codes for Joint Energy and Information Transfer,” *2024 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1–6, Apr. 2024, doi: https://doi.org/10.1109/wcnc57260.2024.10571238.

[54] I. C. Wong, B. K. Ng, and C.-T. Lam, “Genetic-Algorithm-Based Polar Code Construction for OFDM Systems with Imperfect Channel Estimation,” *2022 IEEE 8th*

International Conference on Computer and Communications (ICCC), pp. 22–25, Dec. 2022, doi: https://doi.org/10.1109/iccc56324.2022.10066037.

[55] H. Wang, X. Tao, H. Wu, N. Li, and J. Xu, "Secure polar coding for a joint source-channel model," *Science China Information Sciences*, vol. 64, no. 11, Oct. 2021, doi: https://doi.org/10.1007/s11432-020-3119-3.

[56] J. Del Olmo Alòs, "Polar coding for the wiretap broadcast channel," 2019. doi: 10.5821/dissertation-2117-183239.

[57] V. Bioglio and I. Land, "Polar-Code Construction of Golay Codes," *IEEE Communications Letters*, vol. 22, no. 3, pp. 466–469, Mar. 2018, doi: https://doi.org/10.1109/lcomm.2018.2793273.

[58] C. Xu, L. Wu, W. Liu, and H. Xu, "Block Code Design Based on Golay Code for Future Communication Systems," *2024 IEEE 99th Vehicular Technology Conference: (VTC2024-Spring)*, pp. 1–6, Jun. 2024, doi: https://doi.org/10.1109/vtc2024-spring62846.2024.10683230.

[59] B. Srikanth, D. S. Pranathi, P. S. Kumar Raju and V. Sarika, "Design and Implementation of Low Power Golay Encoder Architecture," *2024 5th International Conference for Emerging Technology (INCET)*, pp. 1-6, May 2024, doi: 10.1109/INCET61516.2024.10593302.

[60] J. Ding, C. Lin, and S. Mesnager, "Secret sharing schemes based on the dual of Golay codes," *Cryptography and Communications*, vol. 13, no. 6, pp. 1025–1041, Nov. 2021, doi: https://doi.org/10.1007/s12095-021-00531-w.

[61] Y. Li, L. Wang, and T.-K. Truong, "Soft decoding of the (23, 12, 7) Golay-code up to five errors," *IET Communications*, vol. 5, no. 15, pp. 2206–2211, Oct. 2011, doi: https://doi.org/10.1049/iet-com.2011.0318.

[62] M. J. E. Golay,"Notes on digital coding,"*Proc. IRE*, vol. 37, p. 657, Jan. 1949.

[63] X.-H. Peng and P. G. Farrell, "On Construction of the (24, 12, 8) Golay Codes," *IEEE Transactions on Information Theory*, vol. 52, no. 8, pp. 3669–3675, Jul. 2006, doi: https://doi.org/10.1109/tit.2006.876247.

[64] B. Honary and G. Markarian, "New simple encoder and trellis decoder for Golay codes," *Electronics Letters*, vol. 29, no. 25, pp. 2170–2171, Dec. 1993, doi: https://doi.org/10.1049/el:19931456.

[65] B. K.Classon,"Method, system, apparatus, and phone for error control of Golay encoded data signals," U.S.Patent 6 199 189, 2001.

[66] M.I.Weng andL.N.Lee,"Weighted erasure codec for the (24, 12) extended Golay code," U.S.Patent 4 397 022, 1983.

[67] M. Sprachmann, "Automatic generation of parallel CRC circuits," *IEEE Design & Test of Computers*, vol. 18, no. 3, pp. 108–114, May 2001, doi: https://doi.org/10.1109/54.922807.

[68] G. Campobello, G. Patane, and M. Russo, "Parallel CRC realization," *IEEE Transactions on Computers*, vol. 52, no. 10, pp. 1312–1319, Oct. 2003, doi: https://doi.org/10.1109/tc.2003.1234528.

[69] R. Nair, G. Ryan and F. Farzaneh, "A symbol based algorithm for hardware implementation of cyclic redundancy check (CRC)," *Proceedings VHDL International Users' Forum. Fall Conference*, 1997, pp. 82-87, doi: https://doi.org/10.1109/viuf.1997.623934.

[70] S. Sarangi and S. Banerjee, "Efficient Hardware Implementation of Encoder and Decoder for Golay Code," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 23, no. 9, pp. 1965–1968, Sep. 2015, doi: https://doi.org/10.1109/tvlsi.2014.2346712.

[71] A. Alimohammad and S. F. Fard, "FPGA-Based Bit Error Rate Performance Measurement of Wireless Systems," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 22, no. 7, pp. 1583–1592, Jul. 2014, doi: https://doi.org/10.1109/tvlsi.2013.2276010.

[72] A. D. Abbaszadeh and C. K. Rushforth, "VLSI implementation of a maximum-likelihood decoder for the Golay (24, 12) code," *IEEE Journal on Selected Areas in Communications*, vol. 6, no. 3, pp. 558–565, Apr. 1988, doi: https://doi.org/10.1109/49.1924.

[73] P. Adde and R. L. Bidan, "A low-complexity soft-decision decoding architecture for the binary extended Golay code," *2012 19th IEEE International Conference on Electronics, Circuits, and Systems (ICECS 2012)*, pp. 705–708, Dec. 2012, doi: https://doi.org/10.1109/icecs.2012.6463628.

[74] P. Bhoyar, "Design of encoder and decoder for Golay code," *2016 International Conference on Communication and Signal Processing (ICCSP)*, pp. 1491-1495, Apr. 2016, doi: https://doi.org/10.1109/iccsp.2016.7754406.

[75] M. Nazeri, A. Rezai and H. Azis, "An Efficient Architecture for Golay Code Encoder," *2018 2nd East Indonesia Conference on Computer and Information Technology (EIConCIT)*, pp. 114–117, Nov. 2018, doi: https://doi.org/10.1109/eiconcit.2018.8878513.

[76] N. Goela, S. B. Korada and M. Gastpar, "On LP decoding of polar codes," *2010 IEEE Information Theory Workshop*, pp. 1–5, Aug. 2010, doi: https://doi.org/10.1109/cig.2010.5592698.

[77] 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, Dec. 2012, doi: https://doi.org/10.1109/lcomm.2012.111612.121898.

[78] E. Arikan, "Channel Polarization: A Method for Constructing Capacity-Achieving Codes for Symmetric Binary-Input Memoryless Channels," *IEEE Transactions on Information Theory*, vol. 55, no. 7, pp. 3051–3073, Jul. 2009, doi: https://doi.org/10.1109/TIT.2009.2021379.

[79] I. Tal and A. Vardy, "List Decoding of Polar Codes," *IEEE Transactions on Information Theory*, vol. 61, no. 5, pp. 2213–2226, May 2015, doi: https://doi.org/10.1109/tit.2015.2410251.

[80] K. Niu and K. Chen, "CRC-Aided Decoding of Polar Codes," *IEEE Communications Letters*, vol. 16, no. 10, pp. 1668–1671, Oct. 2012, doi: https://doi.org/10.1109/lcomm.2012.090312.121501.

[81] T. Murata and H. Ochiai, "On design of CRC codes for polar codes with successive cancellation list decoding," *2017 IEEE International Symposium on Information Theory (ISIT)*, pp. 1868-1872, Jun. 2017, doi: https://doi.org/10.1109/isit.2017.8006853.

[82] Q. Zhang, A. Liu, X. Pan, and K. Pan, “CRC Code Design for List Decoding of Polar Codes,” *IEEE Communications Letters*, vol. 21, no. 6, pp. 1229–1232, Jun. 2017, doi: https://doi.org/10.1109/lcomm.2017.2672539.

[83] A. Elkelesh, M. Ebada, S. Cammerer, and S. ten Brink, “Belief Propagation List Decoding of Polar Codes,” *IEEE Communications Letters*, vol. 22, no. 8, pp. 1536–1539, Aug. 2018, doi: https://doi.org/10.1109/lcomm.2018.2850772.

[84] M. Geiselhart, A. Elkelesh, M. Ebada, S. Cammerer, and S. ten Brink, “CRC-Aided Belief Propagation List Decoding of Polar Codes,” *2020 IEEE International Symposium on Information Theory (ISIT)*, Jun. 2020, pp. 395-400, doi: 10.1109/ISIT44484.2020.9174249.

[85] M. Mitzenmacher, “A survey of results for deletion channels and related synchronization channels,” *Probability Surveys*, vol. 6, pp. 1–33, 2009, doi: https://doi.org/10.1214/08-ps141.

[86] R. Venkataramanan, S. Tatikonda, and K. Ramchandran, “Achievable Rates for Channels With Deletions and Insertions,” *IEEE Transactions on Information Theory*, vol. 59, no. 11, pp. 6990–7013, Nov. 2013, doi: https://doi.org/10.1109/tit.2013.2278181.

[87] R. Wang, J. Honda, H. Yamamoto, R. Liu and Y. Hou, "Construction of polar codes for channels with memory," *2015 IEEE Information Theory Workshop (ITW)*, pp. 187–191, Oct. 2015, doi: https://doi.org/10.1109/itwf.2015.7360760.

[88] E. Şaşoğlu and I. Tal, “Polar Coding for Processes With Memory,” *IEEE Transactions on Information Theory*, vol. 65, no. 4, pp. 1994–2003, Apr. 2019, doi: https://doi.org/10.1109/tit.2018.2885797.

[89] B. Shuval and I. Tal, “Fast Polarization for Processes With Memory,” *IEEE Transactions on Information Theory*, vol. 65, no. 4, pp. 2004–2020, Apr. 2019, doi: https://doi.org/10.1109/tit.2018.2878575.

[90] E. K. Thomas, V. Y. F. Tan, A. Vardy, and M. Motani, “Polar Coding for the Binary Erasure Channel With Deletions,” *IEEE Communications Letters*, vol. 21, no. 4, pp. 710–713, Apr. 2017, doi: https://doi.org/10.1109/lcomm.2017.2650918.

[91] S. K. Hanna and S. E. Rouayheb, “Guess & Check Codes for Deletions, Insertions, and Synchronization,” *IEEE Transactions on Information Theory*, vol. 65, no. 1, pp. 3–15, Jan. 2019, doi: https://doi.org/10.1109/tit.2018.2841936.

[92] K. Tian, A. Fazeli, and A. Vardy, “Polar Coding for Channels With Deletions,” *IEEE Transactions on Information Theory*, vol. 67, no. 11, pp. 7081–7095, Nov. 2021, doi: https://doi.org/10.1109/tit.2021.3083785.

[93] L. Dolecek and V. Anantharam, "Using Reed–Muller RM(1,m) Codes Over Channels With Synchronization and Substitution Errors," in *IEEE Transactions on Information Theory*, vol. 53, no. 4, pp. 1430-1443, Apr. 2007, doi: https://doi.org/10.1109/tit.2007.892776.

[94] K.-C. Hsu, C.-W. Tsao, Y.-H. Chang, T.-W. Kuo, and Y.-M. Huang, “Proactive Channel Adjustment to Improve Polar Code Capability for Flash Storage Devices,” *2018 55th ACM/ESDA/IEEE Design Automation Conference (DAC)*, pp. 1–6, Jun. 2018, doi: https://doi.org/10.1109/dac.2018.8465820.

[95] I. Tal and A. Vardy, “How to Construct Polar Codes,” *IEEE Transactions on Information Theory*, vol. 59, no. 10, pp. 6562–6582, Oct. 2013, doi: https://doi.org/10.1109/tit.2013.2272694.

[96] S. H. Hassani and R. Urbanke, “Universal polar codes,” *2014 IEEE International Symposium on Information Theory (ISIT)*, pp. 1451–1455, Jun. 2014, doi: https://doi.org/10.1109/isit.2014.6875073.

[97] Q. Li, A. Jiang, and E. F. Haratsch, “Noise modeling and capacity analysis for NAND flash memories,” *2014 IEEE International Symposium on Information Theory (ISIT)*, pp. 2262–2266, Jun. 2014, doi: https://doi.org/10.1109/isit.2014.6875236.

[98] M. Darnell, “Error Control Coding: Fundamentals and Applications,” *IEE Proceedings F (Communications, Radar and Signal Processing)*, vol. 132, no. 1, p. 68, Feb. 1985, doi: https://doi.org/10.1049/ip-f-1.1985.0011.

[99] C. Zhang and K. K. Parhi, “Low-Latency Sequential and Overlapped Architectures for Successive Cancellation Polar Decoder,” *IEEE Transactions on Signal Processing*, vol. 61, no. 10, pp. 2429–2441, May 2013, doi: https://doi.org/10.1109/tsp.2013.2251339.

[100] Y. Cai, E. F. Haratsch, O. Mutlu, and K. Mai, “Error patterns in MLC NAND flash memory: measurement, characterization, and analysis,” *Design, Automation, and Test in Europe*, pp. 521–526, Mar. 2012, doi: https://doi.org/10.5555/2492708.2492838.

[101] H. Song, C. Zhang, S. Zhang, and X. You, “Polar code-based error correction code scheme for NAND flash memory applications,” *2016 8th International Conference on Wireless Communications & Signal Processing (WCSP)*, Oct. 2016, doi: https://doi.org/10.1109/wcsp.2016.7752700.

[102] Gerrar, N.K.; Zhao, S.; Kong, L. Error correction in data storage systems using polar codes. *IET Commun.* 2021, *15*, 1859–1868.

[103] K. Chen, K. Niu, and J. Lin, “A Hybrid ARQ Scheme Based on Polar Codes,” *IEEE Communications Letters*, vol. 17, no. 10, pp. 1996–1999, Oct. 2013, doi: https://doi.org/10.1109/lcomm.2013.090213.131670.

[104] 3GPP TS 38.212, 3rd Generation Partnership Project: Technical Specification Group Radio Access Network: NR: Multiplexing and Channel Coding (Release 16) V16.3.0 Technical Specification (TS). 2020. Web-link: https://www.3gpp.org/ftp//Specs/archive/38_series/38.212/38212-g30.zip

[105] D. Hui, S. Sandberg, Y. Blankenship, M. Andersson, and L. Grosjean, “Channel Coding in 5G New Radio: A Tutorial Overview and Performance Comparison with 4G LTE,” *IEEE Vehicular Technology Magazine*, vol. 13, no. 4, pp. 60–69, Dec. 2018, doi: https://doi.org/10.1109/mvt.2018.2867640.

[106] E. Abbe and A. Barron, “Polar coding schemes for the AWGN channel,” *2011 IEEE International Symposium on Information Theory (ISIT)Proceedings*, Jul. 2011, doi: https://doi.org/10.1109/isit.2011.6033892.

[107] E. Sasoglu,“Polar coding theorems for discrete systems,” Ecole Polytechnique Federale de Lausanne. Thesis no. 5219, 2011

[108] H. Vangala, Y. Hong, and E. Viterbo, “Efficient Algorithms for Systematic Polar Encoding,” *IEEE Communications Letters*, vol. 20, no. 1, pp. 17–20, Jan. 2016, doi: https://doi.org/10.1109/lcomm.2015.2497220.

[109] P. Trifonov, “Efficient Design and Decoding of Polar Codes,” *IEEE Transactions on Communications*, vol. 60, no. 11, pp. 3221–3227, Nov. 2012, doi: https://doi.org/10.1109/tcomm.2012.081512.110872.

[110] G. Atwood, A. Fazio, D. Mills, and B. Reaves, "Intel StrataFlash™ Memory Technology Overview," *Intel Technol. J.* vol. 1, no. 2, 1997.

[111] K. Niu, K. Chen, J. Lin, and Q. T. Zhang, “Polar codes: Primary concepts and practical decoding algorithms,” *IEEE Communications Magazine*, vol. 52, no. 7, pp. 192–203, Jul. 2014, doi: https://doi.org/10.1109/mcom.2014.6852102.

[112] A. Elkelesh, M. Ebada, S. Cammerer, and S. ten Brink, “Decoder-Tailored Polar Code Design Using the Genetic Algorithm,” *IEEE Transactions on Communications*, vol. 67, no. 7, pp. 4521–4534, Jul. 2019, doi: https://doi.org/10.1109/tcomm.2019.2908870.

[113] S. Hong and J. Chung, “Improved CRC aided BP decoding for polar codes,” *Electronics Letters*, vol. 57, no. 13, pp. 526–528, Apr. 2021, doi: https://doi.org/10.1049/ell2.12175.

[114] H. Zhou, W. J. Gross, Z. Zhang, X. You, and C. Zhang, “Low-Complexity Construction of Polar Codes Based on Genetic Algorithm,” *IEEE Communications Letters*, vol. 25, no. 10, pp. 3175–3179, Oct. 2021, doi: https://doi.org/10.1109/lcomm.2021.3104092.

[115] B. Yuan and K. K. Parhi, “Early Stopping Criteria for Energy-Efficient Low-Latency Belief-Propagation Polar Code Decoders,” *IEEE Transactions on Signal Processing*, vol. 62, no. 24, pp. 6496–6506, Dec. 2014, doi: https://doi.org/10.1109/tsp.2014.2366712.

[116] H. Vangala, E. Viterbo, and Y. Hong, “A Comparative Study of Polar Code Constructions for the AWGN Channel,” *arXiv (Cornell University)*, Jan. 2015, doi: https://doi.org/10.48550/arxiv.1501.02473.

[117] P. Trifonov, “Design of polar codes for Rayleigh fading channel,” *2015 International Symposium on Wireless Communication Systems (ISWCS)*, Aug. 2015, doi: https://doi.org/10.1109/iswcs.2015.7454357.

[118] A. Bravo-Santos, “Polar Codes for the Rayleigh Fading Channel,” *IEEE Communications Letters*, vol. 17, no. 12, pp. 2352–2355, Dec. 2013, doi: https://doi.org/10.1109/lcomm.2013.111113.132103.

[119] D. Zhou, K. Niu, and C. Dong, “Construction of Polar Codes in Rayleigh Fading Channel,” *IEEE Communications Letters*, vol. 23, no. 3, pp. 402–405, Mar. 2019, doi: https://doi.org/10.1109/lcomm.2019.2892453.

[120] L. Xiang, Y. Liu, Z. B. K. Egilmez, R. G. Maunder, L. -L. Yang and L. Hanzo, “Soft List Decoding of Polar Codes,” *IEEE Transactions on Vehicular Technology*, vol. 69, no. 11, pp. 13921–13926, Nov. 2020, doi: https://doi.org/10.1109/tvt.2020.3021258.

[121] M. Mondelli, S. Hamed Hassani, and R. L. Urbanke, “From Polar to Reed-Muller Codes: A Technique to Improve the Finite-Length Performance,” *IEEE Transactions on Communications*, vol. 62, no. 9, pp. 3084–3091, Sep. 2014, doi: https://doi.org/10.1109/tcomm.2014.2345069.

[122] M. Mondelli, S. H. Hassani, and R. L. Urbanke, “Construction of Polar Codes With Sublinear Complexity,” *IEEE Transactions on Information Theory*, vol. 65, no. 5, pp. 2782–2791, May. 2019, doi: https://doi.org/10.1109/tit.2018.2889667.

[123] O. İşcan, R. Böhnke and W. Xu, “Probabilistic Shaping Using 5G New Radio Polar Codes,” *IEEE Access*, vol. 7, pp. 22579–22587, Feb. 2019, doi: https://doi.org/10.1109/access.2019.2898103.

[124] P. Chen, B. Bai, Z. Ren, J. Wang, and S. Sun, “Hash-Polar Codes with Application to 5G,” *IEEE Access*, vol. 7, pp. 12441–12455, Jan. 2019, doi: https://doi.org/10.1109/access.2019.2892969.

[125] C. Condo, S. A. Hashemi, A. Ardakani, F. Ercan, and W. J. Gross, “Design and Implementation of a Polar Codes Blind Detection Scheme,” *IEEE Transactions on Circuits & Systems II Express Briefs*, vol. 66, no. 6, pp. 943–947, Jun. 2019, doi: https://doi.org/10.1109/tcsii.2018.2872653.

[126] L. Xiang, Z. B. Kaykac Egilmez, R. G. Maunder and L. Hanzo, “CRC-Aided Logarithmic Stack Decoding of Polar Codes for Ultra Reliable Low Latency Communication in 3GPP New Radio,” *IEEE Access*, vol. 7, pp. 28559–28573, Jan. 2019, doi: https://doi.org/10.1109/access.2019.2901596.

[127] C. Pillet, V. Bioglio, and C. Condo, “On List Decoding of 5G-NR Polar Codes,” *2020 IEEE Wireless Communications and Networking Conference (WCNC)*, May 2020, doi: https://doi.org/10.1109/wcnc45663.2020.9120686.

[128] Y. Wang, W. Zhang, Y. Liu, L. Wang, and Y. Liang, “An Improved Concatenation Scheme of Polar Codes With Reed–Solomon Codes,” *IEEE Communications Letters*, vol. 21, no. 3, pp. 468–471, Mar. 2017, doi: https://doi.org/10.1109/lcomm.2016.2639482.

[129] O. İşcan, R. Böhnke, and W. Xu, “Sign-bit shaping using polar codes,” *Transactions on Emerging Telecommunications Technologies*, vol. 31, no. 10, Jul. 2020, doi: https://doi.org/10.1002/ett.4058.

[130] Q. Jan *et al.*, “Parity-check and G-matrix based intelligent early stopping criterion for belief propagation decoder for polar codes,” *Digital Communications and Networks*, vol. 9, no. 5, pp. 1148–1156, Oct. 2023, doi: https://doi.org/10.1016/j.dcan.2022.12.011.

[131] J. Guo, M. Qin, A. Guillen, and P. H. Siegel, “Enhanced belief propagation decoding of polar codes through concatenation,” *2014 IEEE International Symposium on Information Theory (ISIT)*, Jun. 2014, doi: https://doi.org/10.1109/isit.2014.6875382.

[132] W. Y. Alebady and A. A. Hamad, “Concatenated turbo polar-convolutional codes based on soft cancellation algorithm,” *Physical Communication*, vol. 58, p. 102010, Jun. 2023, doi: https://doi.org/10.1016/j.phycom.2023.102010.

[133] Y. Wang, L. Chen, C. Liu, and Z. Xing, “An Improved Concatenation Scheme of BCH-Polar Codes With Low-Latency Decoding Architecture,” *IEEE access*, vol. 7, pp. 95867–95877, Jan. 2019, doi: https://doi.org/10.1109/access.2019.2929188.

[134] Q. Yu, Z. Shi, X. Li, J. Du, J. Zhang, and K. M. Rabie, “On the Concatenations of Polar Codes and Non-Binary LDPC Codes,” *IEEE Access*, vol. 6, pp. 65088–65097, Jan. 2018, doi: https://doi.org/10.1109/access.2018.2877178.

[135] Q. Wang, P. Fu, and S. Zhang, “A Comparison of Concatenated Polar Codes with Different Interleaving and Decoding Schemes,” *2020 5th International Conference on Computer and Communication Systems (ICCCS)*, May 2020, doi: https://doi.org/10.1109/icccs49078.2020.9118473.

[136] W. Liu, X. Jin, X. Nie, and M. Wu, “Performance of Concatenated Polar Codes in VLC System,” *Proceedings of the 2020 4th International Conference on Electronic Information Technology and Computer Engineering (EITCE’20)*, pp. 163–167, Nov. 2020, doi: https://doi.org/10.1145/3443467.3443747.

[137] D. Hui, M. Breschel, and Y. Blankenship, “Interleaved CRC for Polar Codes,” *2018 IEEE 87th Vehicular Technology Conference: (VTC2018-Spring)*, Jun. 2018, doi: https://doi.org/10.1109/vtcspring.2018.8417497.

[138] J. Chen, Y. Chen, K. Jayasinghe, D. Du, and J. Tan, “Distributing CRC Bits to Aid Polar Decoding,”*2017 IEEE Global Communications Workshops (GLOBECOM)*, Dec. 2017, doi: https://doi.org/10.1109/glocomw.2017.8269177.

[139] T. Wang, D. Qu, and T. Jiang, “Parity-Check-Concatenated Polar Codes,” *IEEE Communications Letters*, vol. 20, no. 12, pp. 2342–2345, Dec. 2016, doi: https://doi.org/10.1109/lcomm.2016.2607169.

[140] A. Abedi, “Power-efficient-coded architecture for distributed wireless sensing,” *IET Wireless Sensor Systems*, vol. 1, no. 3, pp. 129–136, Sep. 2011, doi: https://doi.org/10.1049/iet-wss.2010.0077.

[141] V. Bioglio, C. Condo, and I. Land, “Design of polar codes in 5G new radio,” *IEEE Communications Surveys & Tutorials,* vol. *23*, no. 1, pp. 29–40, Jan. 2021, doi: https:// doi.org/10.1109/comst.2020.2967127

[142] W. Liu, W. Chen, and M. Wu, “Performance of interleaved CA-Polar concatenated codes in VLC System,” *Proceedings of the 2022 6th International Conference on Electronic Information Technology and Computer Engineering (EITCE’22)*, pp. 991–995, Mar. 2023, doi: https://doi.org/10.1145/3573428.3573607.

[143] A. Eslami and H. Pishro-Nik, “On Finite-Length Performance of Polar Codes: Stopping Sets, Error Floor, and Concatenated Design,” *IEEE Transactions on Communications*, vol. 61, no. 3, pp. 919–929, Mar. 2013, doi: https://doi.org/10.1109/tcomm.2013.012313.110692.

[144] L. Chen *et al.*, “Report on Post-Quantum Cryptography,” *Report on Post-Quantum Cryptography*, US Department of Commerce, National Institute of Standards and Technology, Apr. 2016, doi: https://doi.org/10.6028/nist.ir.8105.

[145] D. J. Bernstein, “Introduction to post-quantum cryptography,” *Post-Quantum Cryptography*, Springer, pp. 1–14, 2009, doi: https://doi.org/10.1007/978-3-540-88702-7_1.

[146] R. J. McEliece, “A Public-Key Cryptosystem Based On Algebraic Coding Theory,” *Deep Space Network Progress Report*, vol. 44, pp. 114–116, Jan. 1978.

[147] A. D. Wyner, “The Wire-Tap Channel,” *Bell System Technical Journal*, vol. 54, no. 8, pp. 1355–1387, Oct. 1975, doi: https://doi.org/10.1002/j.1538-7305.1975.tb02040.x.

[148] H. Wang, X. Tao, N. Li, and Z. Han, “Polar Coding for the Wiretap Channel With Shared Key,” *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 6, pp. 1351–1360, Jun. 2018, doi: https://doi.org/10.1109/tifs.2017.2774499.

[149] D. Chen *et al.*, “An LDPC Code Based Physical Layer Message Authentication Scheme With Prefect Security,” *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 4, pp. 748–761, Apr. 2018, doi: https://doi.org/10.1109/jsac.2018.2825079.

[150] Z. Liu and Q. Du, “Self-coupling Encryption via Polar Codes for Secure Wireless Transmission,” *2022 International Wireless Communications and Mobile Computing (IWCMC)*, pp. 384–388, May 2022, doi: https://doi.org/10.1109/iwcmc55113.2022.9824824.

[151] E. Hof and S. Shamai, "Secrecy-achieving polar-coding," *2010 IEEE Information Theory Workshop*, Aug. 2010, doi: https://doi.org/10.1109/cig.2010.5592878.

[152] H. Mahdavifar and A. Vardy, "Achieving the Secrecy Capacity of Wiretap Channels Using Polar Codes," *IEEE Transactions on Information Theory*, vol. 57, no. 10, pp. 6428–6443, Oct. 2011, doi: https://doi.org/10.1109/TIT.2011.2162275.

[153] E. Şaşoğlu and A. Vardy, "A new polar coding scheme for strong security on wiretap channels," *2013 IEEE International Symposium on Information Theory (ISIT)*, Jul. 2013, doi: https://doi.org/10.1109/isit.2013.6620400.

[154] H. Bai, L. Jin, and M. Yi, “Artificial noise aided polar codes for physical layer security,” *China Communications*, vol. 14, no. 12, pp. 15–24, Dec. 2017, doi: https://doi.org/10.1109/cc.2017.8246334.

[155] Y. Zhang, Z. Yang, A. Liu, and Y. Zou, “Secure transmission over the wiretap channel using polar codes and artificial noise,” *IET Communications*, vol. 11, no. 3, pp. 377–384, Feb. 2017, doi: https://doi.org/10.1049/iet-com.2016.0429.

[156] A. Arli and O. Gazi, “Noise-aided belief propagation list decoding of polar codes,” *IEEE Communications Letters,* vol. *23*, pp. 1285–1288, Aug. 2019, doi: https://doi.org/ 10.1109/lcomm.2019.2918535.

[157] C. E. Shannon, “Communication Theory of Secrecy Systems,” *Bell System Technical Journal*, vol. 28, no. 4, pp. 656–715, Oct. 1949, doi: https://doi.org/10.1002/j.1538-7305.1949.tb00928.x.

[158] W. K. Harrison, J. Almeida, M. R. Bloch, S. McLaughlin, and J. Barros, “Coding for Secrecy: An Overview of Error-Control Coding Techniques for Physical-Layer Security,” *IEEE Signal Processing Magazine,* vol. 30, no. 5, pp. 41–50, Sep. 2013, doi: https://doi.org/10.1109/msp.2013.2265141.

[159] M. R. Bloch, M. Hayashi, and A. Thangaraj, “Error-Control Coding for Physical-Layer Secrecy,” *Proceedings of the IEEE*, vol. 103, no. 10, pp. 1725–1746, Sep. 2015, doi: https://doi.org/10.1109/jproc.2015.2463678.

[160] M. Bloch, J. Barros, M. R. D. Rodrigues, and S. W. McLaughlin, “Wireless Information-Theoretic Security,” *IEEE Transactions on Information Theory*, vol. 54, no. 6, pp. 2515–2534, Jun. 2008, doi: https://doi.org/10.1109/tit.2008.921908.