The Development, Operation and Performance of the 5G Polar Codes

Zeynep B. Kaykac Egilmez, Luping Xiang, Robert G. Maunder, *Senior Member, IEEE*, and
Lajos Hanzo, *Fellow, IEEE*

***Abstract*—Since their inception in 2008, polar codes have been shown to offer near-capacity error correction performance across a wide range of block lengths and coding rates. Owing to this, polar codes have been selected to provide channel coding in the control channels of Third Generation Partnership Project (3GPP) New Radio (NR). The operation of the 3GPP NR polar codes is specified in the 3GPP standard TS 38.212, together with schemes for code block segmentation, Cyclic Redundancy Check (CRC) attachment, CRC scrambling, CRC interleaving, frozen and parity check bit insertion, sub-block interleaving, bit selection, channel interleaving and code block concatenation. The configuration of these components is different for the uplink, broadcast and downlink control channels. However, the lack of visualisations and diagrammatic explanations in TS 38.212 limits the accessibility of the standard to new readers. This motivates the aims of the paper, which provides detailed tutorials on the operation and motivation of the components of the 3GPP NR polar codes, as well as surveys of the 3GPP discussions that led to their specification. Furthermore, we comprehensively characterize the error correction and error detection performance of the 3GPP NR polar codes in the uplink, broadcast and downlink control channels.**

***Keywords*—5G, polar coding, forward error correction, New Radio, 5G wireless system standardization.**

I. Introduction

THE 3rd Generation Partnership Project (3GPP) represents a collaborative effort invested in the standardization of global network protocols, including the Universal Mobile Telecommunications System (UMTS) and Code Division Multiple Access-2000 (CDMA-2000), which were the first examples of developing technical specifications for the 3rd Generation (3G) of mobile telecommunication [1]–[3]. The 3G concept fuelled the smart-phone revolution and high speed Internet services, audio and video file transfers and other compelling services [1]–[3]. The increasing demand for data, video and messaging traffic at higher throughput was led to the 4th Generation (4G) 3GPP standard known as the Long Term Evolution (LTE), which is based on a Transmission Control Protocol/Internet Protocol (TCP/IP) model [1]–[3] for seamless integration with the Internet.

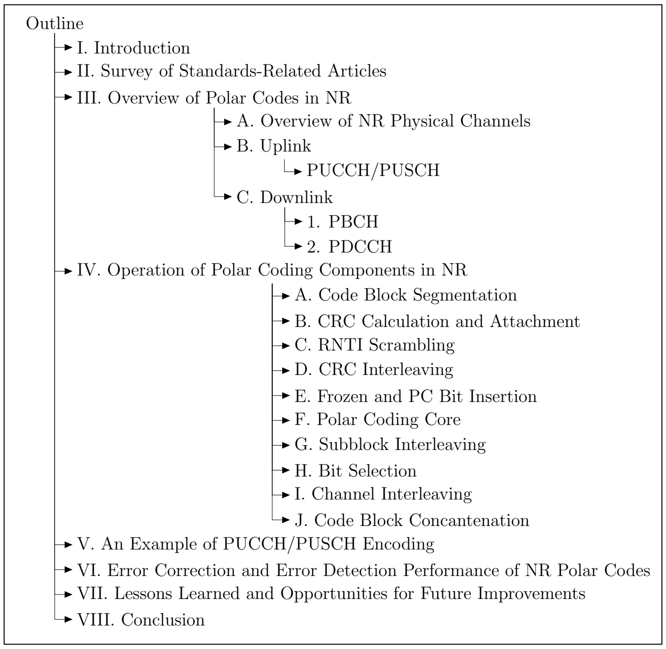

Outline
I. Introduction
II. Survey of Standards-Related Articles
III. Overview of Polar Codes in NR
A. Overview of NR Physical Channels
B. Uplink
PUCCH/PUSCH
C. Downlink
1. PBCH
2. PDCCH
IV. Operation of Polar Coding Components in NR
A. Code Block Segmentation
B. CRC Calculation and Attachment
C. RNTI Scrambling
D. CRC Interleaving
E. Frozen and PC Bit Insertion
F. Polar Coding Core
G. Subblock Interleaving
H. Bit Selection
I. Channel Interleaving
J. Code Block Concantenation
V. An Example of PUCCH/PUSCH Encoding
VI. Error Correction and Error Detection Performance of NR Polar Codes
VII. Lessons Learned and Opportunities for Future Improvements
VIII. Conclusion

Fig. 1: Skeleton structure of this paper.

Recent years have seen a rapidly growing population and increasing use of mobile devices with greater expectations of Quality of Service (QoS) [1], as well as the need for ultra-low latency and ultra-reliable web connectivity applications of Internet of Things (IoT) both in industry and in daily life [4]. This motivates the development of the 5th Generation (5G) of mobile communication systems.

The International Telecommunication Union Radiocommunications (ITU-R) has defined three 5G usage scenarios for 2020 and beyond, namely enhanced Mobile Broadband (eMBB) for high-capacity and ultra-fast mobile communication, Ultra-Reliable and Low Latency Communication (URLLC) for mission critical applications such as vehicle-to-vehicle (V2V) and massive Machine Type Communications (mMTC) for industrial and IoT applications [4].

Based on these development, it may be expected that during the 2020s mobile communication systems will face tremendous growth in the required connectivity, traffic volume and range of usage scenarios [5]. This demand presents a significant challenge, which requires a sustain-

The research data for this paper is available at https://github.com/robmaunder/polar-3gpp-matlab/.

L. Hanzo would like to acknowledge the financial support of the Engineering and Physical Sciences Research Council projects EP/Noo4558/1, EP/PO34284/1, COALESCE, of the Royal Society's Global Challenges Research Fund Grant as well as of the European Research Council's Advanced Fellow Grant QuantCom.

TABLE I: The milestones in mobile communication

Generation	Characteristics	Technology	Applications
1G ~ 1980 [1], [2]	Analog, 2.4-kbps throughput [1], [2]	Based on the Advanced Mobile Phone System (AMPS), multiplexing by Frequency Division Multiple Access (FDMA) [1], [2]	Analog phone calls only [1], [2]
2G ~ 1990 [1], [2]	Digital narrow-band, 9.6-kbps throughput [1], [2]	Based on Global System for Mobile (GSM), Time Division Multiple Access (TDMA) and Frequency Division Duplexing (FDD), Gaussian Minimum Shift Keying (G-MSK) modulated, convolutional coding, Viterbi equalization [1], [2]	Phone calls, Multimedia Message Services (MMS), Short Message Services (SMS) [1], [2]
3G ~ 2001 [1], [2]	Digital broadband, 5-MHz bandwidth, 3.9-Mbps throughput [1], [2]	Based on the Universal Mobile Telecommunication System (UMTS), multiplexing by Orthogonal Variable Spreading Factor (OVSF) CDMA sequences, Adaptive Quadrature amplitude modulation (AQAM), convolutional and turbo coding, 31 different code-rate and QAM modes [1], [2]	Constrained smart phones, web based applications, audio and video files transfer, high speed web, improved security, video conferencing, 3D gaming, mobile TV [1], [2]
4G ~ 2010 [1], [2]	20-MHz bandwidth, high throughput (1-Gbps) [1], [2]	Based on LTE with TCP/IP, Multiple Input Multiple Output (MIMO), multiplexing by Orthogonal-FDMA (OFDMA) in the DL and single carrier FDMA in the UL [1], [2]	Similar to 3G applications with higher throughput and quality of service [1], [2]
5G	Extreme broad-band, very-high throughput (20-Gbps), ultra-low latency [1], [2], [4]	Based on NR, massive MIMO communication in the milimeter wave (mmWave) frequency range, hybrid beamforming, new coding schemes such as LDPC for data and polar codes for control information, multiplexing by OFDM [1], [2], [4]	Expected applications: **Enhanced Mobile Broadband (eMBB)**: For high-capacity and ultra-fast mobile communications for phones and infrastructure, virtual and augmented reality, 3D and ultra-HD video, and haptic feedback; **Ultra-Reliable Low Latency Communication (URLLC)**: For vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communications, autonomous driving; **Massive Machine-Type Communications (mMTC)**: For consumer and industrial IoT, Industry 4.0, mission-critical machine-to-machine (MC-M2M) communication [4]

able development of improved systems efficiencies, such as spectral energy, operational and cost efficiencies [5], [6]. At the time of writing, the 3GPP is developing a set of New Radio (NR) standards, in order to address the 5G requirements [5]. While NR is not required to be backward compatible with LTE, the NR functionalities are required to be forward compatible and allow for smooth introduction of additional technology components on top of deployed LTE infrastructure, with support for new use cases [7]. A summary of the historical evaluation of mobile communication from 1st Generation (1G) to 5G is presented in Table I.

As in most standardised communication schemes, 3GPP NR uses channel coding to enable error correction and error detection in the presence of noise, fading and interference. During the specification of 3GPP NR, the channel coding of user data was considered separately from the channel coding of control information, such as channel state information (CSI) and scheduling information. Various channel coding techniques were compared [8] and in particular, turbo codes [9], Low Density Parity Check (LDPC) codes [10] and polar codes [11] were considered since they are all capacity approaching channel coding schemes [8], [12]–[14]. For these potential channel coding schemes, the RAN1-86b meeting of 3GPP captured comparisons [15] of the error correction capability, flexibility, Hybrid Automatic Repeat Request (HARQ) support, decoding latency, implementation complexity and other considerations. These observations are presented in Table II.

At the conclusion of a study item that spanned from April 2016 to November 2016, LDPC codes were selected for the data channels of NR, while polar codes were selected for the control channels, replacing the turbo and tail-biting convolutional codes (TBCC) of LTE, respectively. LDPC codes were selected for data channels because they can efficiently support multiple code rates, block lengths and HARQ, with better decoding latency, throughput and implementation complexity than other codes [16]. In the case of the control channels, polar codes were selected, because they offer the best error correction capability at the short information block lengths that are used for control information [17], [18].

Between February 2017 to April 2018, a 3GPP work item designed the operation of LDPC and polar codes for the data and control channels of NR, which has been specified in the 3GPP standard TS 38.212. However, the scope of TS 38.212 includes only the encoding operations performed in a transmitting basestation or user equipment (UE), without visualized diagrammatic explanation or examples and without discussion of the decoding operation performed within a receiver. Selected 3GPP meetings and their outcomes for 5G NR, and particularly for the NR polar codes are summarised in Table III.

Against this background, the contribution of this paper is to provide a survey of the operation and performance of the polar codes that have been specified for the NR control

TABLE II: Observations on potential channel coding schemes for NR captured during RAN1-86b meeting of 3GPP

Observations for	**LDPC**	**Polar codes**	**Turbo codes**
Error correction capability	• Further study was required to draw conclusions, owing to different views on the implementation complexities and possible enhancements for all potential channel coding shames.		
Flexibility	• All potential channel coding schemes were deemed to deliver acceptable flexibility.		
CC- and IR-HARQ support	• LDPC supports both CC-HARQ and IR-HARQ, but there are concerns on the complexity of IR-HARQ.	• Polar codes support both CC- HARQ and IR-HARQ, but the incremental freezing method of HARQ support is of concern.	• The ability of turbo codes to support both CC- and IR-HARQ is well known.
Latency	• Highly parallelised LDPC decoders can be used to reduce latency.	• Although polar codes are not highly parallelisable, there are other design techniques to reduce latency.	• Highly parallelised turbo decoders can be used to reduce latency.
Other considerations	• Well established. • Require effort in specification design, for meeting the NR requirements.	• Less well established, being the newest among the three. • Require more effort for meeting the NR requirements.	• Well established. • Require effort in specification design, for meeting the NR requirements.
Implementation complexity	• Widely implemented in commercial hardware supporting several Gbps throughput, attractive area and energy efficiency with some flexibility, but there are concerns for NR.	• Implementable, although there are currently no commercial implementations. However, there are some concerns for NR.	• Widely implemented in commercial hardware, supporting HARQ and flexibility similar to what is required for NR, but not at the high data rates or low latencies required for NR.

channels. More specifically, we provide schematics, flow charts, block diagrams and examples to detail the operation of both the polar encoding and decoding processes used in the NR control channels, including Cyclic Redundancy Check (CRC) operations, rate matching, interleaving and other peripheral operations. We discuss the motivations behind the design decisions that dictate these operations and we characterize the error correction and error detection performance of the NR polar codes.

The rest of this paper is organized as follows. Section II surveys the standards-related articles. Section III briefly reviews the uplink and downlink transport channels of 3GPP NR and provides encoding and decoding block diagrams for polar coding in the control channels. Section IV details the operation of each component in these block diagrams. An end-to-end example of polar encoding for the uplink control channel is provided in Section V. Section VI characterizes the error correction and error detection performance of the polar codes used in the NR control channels. Following this, Section VII provides a summary of lessons learned and possible improvements for future iterations of the NR polar code design, as well as for future applications of polar codes beyond 5G. Finally, Section VIII offers our conclusions. The skeleton structure of the paper is shown in Figure 1.

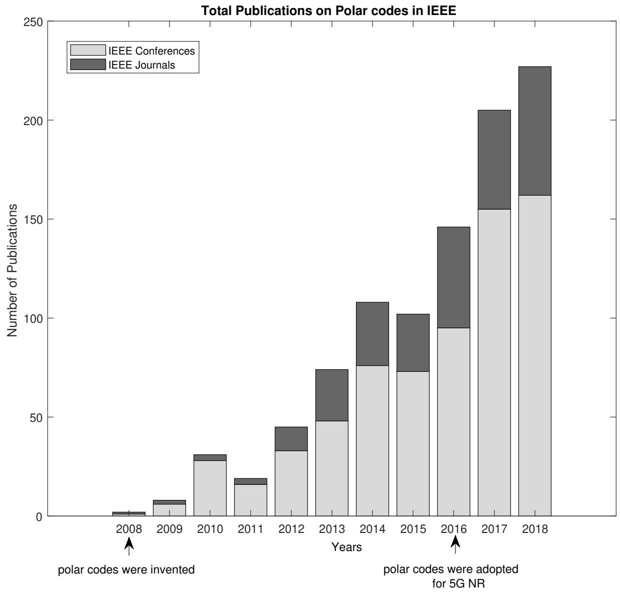


Fig. 2: The interest in polar codes related researches since its invention.

II. Survey of Standards-Related Articles

The NR technical specification TS 38.212 [37] produced by the 3GPP provides a complete specification of the LDPC and the polar encoding in the 5G NR data and control channels, respectively. However, the lack of visualisations and diagrammatic explanations as well as the reliance on numerous acronyms and mathematical descriptions in [37] limits the accessibility of the standard to new readers. Furthermore, [37] does not provide any discussion of the corresponding LDPC and polar decoding operations at the NR receiver.

The requirements of the 5G NR LDPC codes and a detailed NR LDPC code design philosophy are described in [38]. Furthermore, performance and complexity comparisons between the NR LDPC and LTE turbo codes are presented in [38]. Similarly, the key features of the 5G NR LDPC code design are described in [39]. Particularly, the

TABLE III: Selected 3GPP meetings and their outcomes for 5G NR polar codes

3GPP RAN1 meetings	Channel coding related outcomes
R1-84bApr. 2016AI: 8.1.6.1	• Channel coding requirements for 5G NR are identified [13], [19]. • LDPC, polar code, convolutional code and turbo code are identified as candidates for 5G NR [19], [20]. • Performance, implementation complexity, latency and flexibility chosen as selection criteria for appropriate channel coding scheme(s) [19], [20].
R1-85May. 2016AI: 7.1.5.1	• At least in AWGN channels, and for large information block lengths, all candidates show comparable link performance, hence further study is required to determine supported channel coding scheme(s) [21], [22]. • For the purpose of study and comparisons, quasi-cyclic like LDPC codes are defined [22], [23].
R1-86Aug. 2016AI: 8.1.4.1	• Agreed that coding schemes should support information block length and codeword length flexibility with rate matching [24]. • Channel coding technique(s) designed for data channels of NR should support both IR- HARQ (or similar) and CC- HARQ [24]. • Simulation assumption for control channel coding is agreed as AWGN channel, QPSK modulation with specific information lengths, by evaluating BLER vs. SNR and FAR vs. SNR [24].
R1-86bOct. 2016AI: 8.1.3.1	• Observations on LDPC, polar codes and turbo codes are captured, as summarised in Table II [16], [25]. • Adopt LDPC for large information block lengths as channel coding scheme for data channels [25].
R1-87Nov. 2016AI: 7.1.5.1	• As *working assumption* for UL data channels, adopt flexible LDPC as the single channel coding scheme for small block lengths [15]. • For DL data channels, • Adopt polar coding (except for very small block lengths) for uplink control channels [15]. • As *working assumption* adopt polar coding (except for very small block lengths) for downlink control channels [15].
R1-88Feb. 2017AI: 8.1.4	• Decided upon some code design targets for LDPC, such as at least supporting 20 Gbps decoder information throughput with code rate 8/9, while aiming for good throughput performance at lower code rate(s) [26]. • Agreed that different polar code designs will be evaluated, such as distributed CRC-basic polar and PC-Polar [26] • Maximum mother code length of polar code N is identified as $N_{max,DCI} = 512$ for DCI [26]. • As *working assumption*, $N_{max,UCI} = 1024$ is selected for UCI [26].
R1-88bApr. 2017AI: 8.1.4	• Agreed that polar code construction techniques will be studied to facilitate early termination without degrading BLER performance or latency [27]. • PC-CA polar is taken as the baseline for further designs and CRC bits can be used for error correction, with some PC bits inserted in reliable positions and some remaining frozen bits used for PC bits [28]. • Agreed that for UL and DL, the number of CRC bits may be different [28]. • If channel coding is applied for UCI at very short block lengths of $A = 1$ and $A = 2$, repetition code and simplex code are adopted, respectively. • Adopt LTE RM schemes for UCI with $3 \leq A \leq 11$ and note that if NR requires a codeword length N that is not supported by the LTE RM code, then the LTE RM code will be extended by repetition as in LTE [28]. • Adopt polar codes for UCI with $A \geq 12$ [28].
R1-89May. 2017AI: 7.1.4	• A single fixed polar code sequence should be used for information and frozen bit selection for each mother code length N [29]. • The set of fixed sequences for different mother code lengths N will be derived from a single sequence for a single reference mother code length [29]. • A competition was agreed for proposing a polar sequence design between 3GPP participants. • Puncturing refers to non-transmission of coded bits such that the non-transmitted bits are unknown at the receiver and the corresponding LLRs can be set to zero [29]. • Shortening involves setting input bits to a known value, and non-transmission of coded bits corresponding to those input bits, such that the corresponding LLRs can be set to a large value at the receiver [29]. • Maximum code length (N) of polar coding for UCI is confirmed as $N_{max,UCI} = 1024$ [29]. • Repetition is applied for encoded block length $E > N$, puncturing or shortening is applied when $E < N$; Puncturing is used for lower code rates, shortening for higher code rates [29]. • Polar coding is adopted for NR-PBCH using same polar code construction as for the control channel with $N_{max} = 512$ [29].
R1-90Aug. 2017AI: 6.1.4	• For UL code construction, CRC bits are generated by a single CRC polynomial and the CRC bits are attached as a block to the end of information bits [30]. • Huawei sequence from [31] is selected for polar code sequence design. • For evaluation purposes, assume $A_{max} = 140$ for DL [30]. • Proposed evaluation assumptions from [32] for polar code channel interleaver for DL is agreed. • As *working assumption*, use triangular interleaver for uplink channel interleaver (e.g. as in [33]). • For UL, the channel bit interleaving is a separate stage after rate matching [30].
R1-90bOct. 2017AI: 7.4	• RNTI is masked onto the last 16 CRC bits on the PDCCH [34]. • DL channel interleaver is not adopted [34]. • UCI segmentation into two segments with equal segment lengths (with a single zero-padding bit inserted at the beginning of the first segment if needed) is used for certain ranges of A and R [34]. • CRC appended to the first segment is calculated based on the first segment only, and CRC appended to the second segment uses the same CRC polynomial as for the first segment, and is calculated based on the second segment only [34]. • NR-PBCH adopts information block length length as 56 including 24 bit CRC [34]. • Same polar code construction of PDCCH with 24-bit D-CRC, is adopted for NR-PBCH for the control channel with $N_{max} = 512$ [34].

TABLE III- *continued from previous page*

3GPP RAN1 meetings	Channel coding related outcomes
R1-91Nov. 2017AI: 7.4	• No CRC added when RM codes are used in the uplink for $3 \leq A \leq 11$ bits [35]. • Polar code with PC bits are used for UCI transmission with $12 \leq A \leq 19$ and a CRC-6 polynomial is used [35]. • The value of the PC bits used for UCI is obtained from a length-5 cycle shift register [35]. • For UCI with $A > 19$, an 11-bit CRC polynomial is used [35]. • The agreed coding scheme for UCI is applicable for A_{max} up to at least (5/6)*(2048)=1706 [35]. • Segmentation is applied for UCI with $A \geq 360$ and $G \geq 1088$ [35]. • When segmentation is applied, channel interleaving is applied to each segment individually and channel interleaving is applied after rate matching [35]. • The maximum interleaver length per code block is 8192 [35]. • For polar encoding of DCI, $A_{max} = 140$ [35]. • There is no additional UE-specific scrambling of DCI motivated by channel coding [35]. • For DCI, initialize CRC shift register with 24 one-valued bits [35].
R1-92bApr. 2018AI: 7.1.4	• Polar code segmentation of UCI is applied if A ≥360 and G ≥ 1088 or A ≥1013 [36].

main contribution of [39] is the detailed analysis of the rate-matching algorithms conceived for the 5G NR LDPC. However, these papers do not provide a discussion on the NR polar code design.

The LDPC codes and polar codes adopted by the 5G NR standard are described in [40] with the objective of introducing the encoding operations associated with each key component in these codes. The main contribution of [40] is a performance comparison of these new codes to those of 4G LTE. However, this paper has not provided a comprehensive schematics and discussions of all polar coding components. In particular, it does not detail the decoding operations.

Since the standardization of polar codes in 3GPP NR, the interest in polar codes has increased significantly, as shown in Figure 2 and there has been further advances in the state-of-the-art of polar codes, as reflected in [41]–[44]. In particular, [41] has focused on the 5G NR polar code transmission chain by introducing a shaping polar encoder, which shows performance gains for higher-order modulations. Under the specific 5G False Alarm Rate (FAR) requirements, the authors of [42] have proposed hash-polar codes for attaining performance improvements over Parity Check based (PC)-polar codes. Furthermore, an improved blind detection architecture is proposed for Physical Downlink Control CHannel (PDCCH) in [43]. In [44], a low complexity logarithmic stack polar decoder has been proposed for 5G URLLC scenario. However, none of these papers provide detailed discussion of the NR polar coding components.

In contrast to these previous publications, this paper makes several contributions, which are detailed as follows. Firstly, the motivation for the introduction of NR and its use of polar codes is discussed in the context of a historic overview of the mobile communication. Secondly, this paper provides a detailed analysis of the timeline, discussions and evaluation of the polar codes specified during the 3GPP standardisation process. Furthermore, we provide the first set of comprehensive schematics and discussions of all polar coding components, including the decoder components used in the receiver. Most importantly, detailed discussions are provided of how and why 3GPP selected the various specific designs for the various components. In particular, we provide discussion of its advanced features, such as the early termination and CRC-aided polar decoding, which were integral to the design of the 3GPP polar code. Additionally, an example of generalised polar encoding and a complete end-to-end example of NR polar encoding in the uplink control channel is provided. Moreover, we have provided a discussion of the design criteria of the NR polar codes, which are summarised in Figure 3. Additionally, we have discussed the lessons learned and the opportunities for future improvements. Above all, we have characterised the error correction and error detection performance of the polar codes used in NR control channels and compared these to the NR LDPC code. A summary of the comparison with existing surveys on the topic is presented in Table IV.

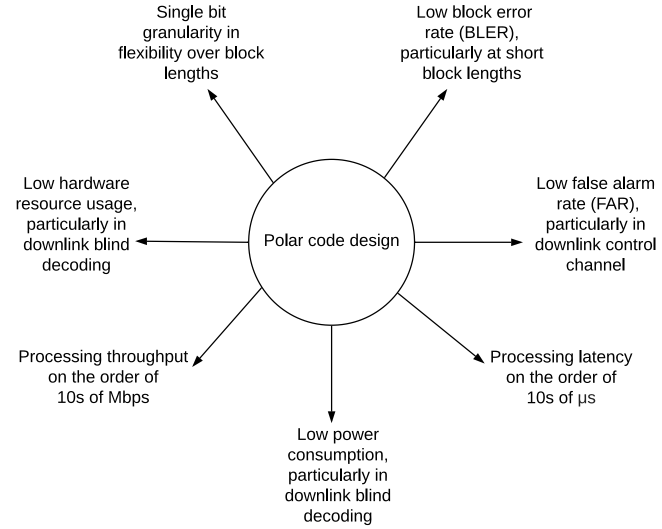

Single bit granularity in flexibility over block lengths
Low block error rate (BLER), particularly at short block lengths
Low hardware resource usage, particularly in downlink blind decoding
Polar code design
Low false alarm rate (FAR), particularly in downlink control channel
Processing throughput on the order of 10s of Mbps
Processing latency on the order of 10s of μs
Low power consumption, particularly in downlink blind decoding

Fig. 3: Conflicting design criteria of the NR polar codes.

III. Overview of Polar Codes in NR

This section provides an overview of channel coding in the physical channels of NR. In Section III-A, we will

TABLE IV: Comparison with existing surveys on the topic

Topics Covered	[40] (2018)	Our Paper
Development of mobile communication from 1G to 5G		✓
Timeline and discussions of 3GPP polar codes standardisation process		✓
An example of general polar encoding		✓
A complete start to finish example of the NR polar encoding process		✓
Coding chain of the NR polar codes for each NR control channel		✓
Overview of NR physical control channels	✓	✓✓
Schematics and discussions of polar coding components	✓	✓✓
Discussions of how and why 3GPP selected the various designs for the various components	✓	✓✓
Performance comparison of NR polar codes and other channel codes	✓	✓
Complexity comparison of NR polar codes and other channel codes	✓	✓
Lessons learned and opportunities for future Improvements	✓	✓

TABLE V: Selected physical channels for uplink and downlink in 3GPP NR

Physical Channels		
Uplink	**Description**	**Channel Coding**
Physical Uplink Control Channel (PUCCH)	The PUCCH is used to carry Uplink Control Information (UCI), which includes scheduling requests, Channel Quality Indicator (CQI) information and Hybrid Automatic Repeat Request (HARQ) information [45].	Polar code, PC- polar code or short block code
Physical Uplink Shared Channel (PUSCH)	The PUSCH is intended to carry application data, but also carries Radio Resource Control (RRC) signalling messages and UCI [45].	LDPC for application data and polar code, PC-polar code or short block code for UCI
Downlink	**Description**	**Channel Coding**
Physical Broadcast Channel (PBCH)	The PBCH carries part of the system information required for UE to connect to a NR basestation (gNodeB) [45].	Polar code
Physical Downlink Shared Channel (PDSCH)	The PDSCH carries application data and paging information [45].	LDPC
Physical Downlink Control Channel (PDCCH)	The PDCCH conveys Downlink Control Information (DCI), which includes scheduling decisions for PDSCH reception, and for scheduling grants for transmission on the PUSCH [45].	Polar code

introduce various NR physical channels, then in Section III-B and Section III-C, we will elaborate in particular on the uplink and downlink control channels, with the aid of block diagrams for their channel encoding and decoding.

Each component in the block diagrams of Figure 4 to 6 has a key feature, which is detailed in Section IV and summarised as follows. Code block segmentation splits long blocks in order to reduce the complexity, while the imposed CRC attachment facilitates error detection and aids error correction. Furthermore, the CRC interleaver disperses the CRC bits inserting them in between the information bits which in turn provides an early termination capability that reduces the complexity of PDCCH blind decoding if the indications are that this decoding operation is likely to fail, despite continuing the computations. Still referring to Figure 4 - 6 polar coding provides the main error correction capability, while sub-block interleaving increases the polar code's error correction capability and the channel interleaver has the role of dispersing burst errors.

A. Overview of NR Physical Channels

The NR physical layer is comprised of several physical channels, as described in Table V. The uplink physical channels convey information from the UE such as a mobile handset, to a basestation, which is referred to as a NR Basestation (gNodeB). Meanwhile, the gNodeB transmits to the UE in the downlink physical channels [46]. These physical channels correspond to particular sets of time-frequency resources [47], and may be classified as data channels for the transmission of user information, or control channels for the transmission of control information [45].

In NR, the Physical Uplink Control Channel is referred to as PUCCH, although some control information can also be carried in the Physical Uplink Shared Channel (PUSCH). Likewise, the downlink control channels are referred to as PDCCH and Physical Broadcast Channel (PBCH), where the later is used to broadcast control information to all UEs connected to the gNodeB. The information and encoded block lengths that are supported for these channels are summarised in Table VI.

TABLE VI: The information and encoded block lengths that are supported by polar coding in the NR physical channels

Physical channels for polar coding	Supported information block lengths	Supported encoded block lengths
PUCCH/PUSCH	$A \in [12, 1706]$	See Equation (1)
PBCH	$A = 32$	$G = 864$
PDCCH	$A \in [12, 140]$	$G \in [A{+}24, 8192]$, in practice $G \in \{108, 216, 432, 864, 1728\}$

The control information transmitted in the control channels is used to coordinate the many UEs connected to the gNodeB, in order to manage the transmission of data on the data channels and facilitate initial connections to the gNodeB [47]. More specifically, PUCCH and PUSCH delivers Uplink Control Information (UCI), which comprises Hybrid Automatic Repeat Request Acknowledgements (HARQ-ACK), Scheduling Requests (SRs), Radio Resource Control (RRC) signalling messages and CSI [48]. Meanwhile, PDCCH delivers Downlink Control Information (DCI) [7], which comprises transport format, resource allocation and HARQ information [48]. The PBCH conveys system information, which allows UEs to connect to the gNodeB [48].

B. Uplink

PUCCH/PUSCH: Figure 4 provides a block diagram for channel encoding and decoding in PUCCH/PUSCH. The encoding block diagram illustrates the conversion of an information block **a** into an encoded block **g**, including all intermediate operations defined in TS 38.212, namely code block segmentation, CRC calculation and attachment, CRC interleaving, frozen and PC bit insertion, polar coding core, sub-block interleaving, bit selection, channel interleaving and code block concatenation, as will be detailed in Section IV. The corresponding decoding operations are illustrated in the decoding block diagram of Figure 4. The operations enclosed in solid lines in Figure 4 are within the scope of this paper, while the operations enclosed in dashed lines are outside of the scope.

In PUCCH/PUSCH, channel coding for short information blocks comprising $A < 12$ bits is similar to that in LTE [49], where Reed-Muller (RM) [50] codes and other short block codes are used with no added CRC [51]. Meanwhile, a PC-polar code with concatenated 6-bit CRC is used for blocks comprising $A \in [12,19]$ bits and a conventional polar code with concatenated 11-bit CRC is used for $A \in [20,1706]$. This paper focuses on PC-polar and polar coding, where the range of supported encoded block lengths G is given by

$$G \in \begin{cases} [A+9, 8192] & \text{if } A \in [12,19] \\ [A+11, 8192] & \text{if } A \in [20,359] \\ [A+11, 16385] & \text{if } A \in [360,1012] \\ [2\lceil A/2 \rceil + 22, 16385] & \text{if } A \in [1013,1706]. \end{cases} \tag{1}$$

Note that PUCCH supports information block lengths of up to 1706 bits, which is significantly higher than the longest 140-bit information blocks supported in the PDCCH. This is because the UCI's Channel Quality Indicator (CQI) includes channel measurements, which requires more bits to convey than the signalling decisions carried by the DCI.

C. Downlink

1) PBCH: Figure 5 provides block diagrams for channel encoding and decoding in PBCH. The encoding block diagram illustrates the conversion of an information block **a** into an encoded block **g**, including payload generation, payload interleaving, payload scrambling, CRC calculation and attachment, CRC interleaving, frozen bit insertion, polar coding core, sub-block interleaving, and bit selection, as will be detailed in Section IV. The corresponding decoding operations are illustrated in the decoding block diagram of Figure 5. Similarly to Figure 4, the operations enclosed in the solid lines in Figure 5 are within the scope of this paper, while those enclosed in dashed lines are outside of the scope. In PBCH, only a single information block length of $A = 32$ bits is supported and a 24-bit Distributed-CRC (DCA-polar) is used with polar coding in order to yield a consistent encoded block length of $G = 864$ bits [52].

2) PDCCH: The channel encoding and decoding block diagrams of PDCCH are provided in Figure 6. Similarly to the other block diagrams, the operations enclosed in the solid lines in Figure 6 are within the scope of this paper, while those enclosed in dashed lines are beyond our scope. The encoding block diagram illustrates the conversion of an information block **a** to an encoded block **g**, including DCI bit sequence generation, CRC initializing, CRC scrambling, CRC calculation and attachment, CRC interleaving, frozen bit insertion, polar coding core, sub-block interleaving and bit selection, as defined will be detailed in Section IV. The corresponding decoding operations are illustrated in the decoding block diagram of Figure 6.

In order to extend the UE battery life, it is critical to minimise the polar decoding complexity during DCI blind decoding. For this reason, early termination [53] is supported for the PDCCH polar code, where the CRC bits are dispersed across the information bits using an interleaver. Rather than recovering the CRC bits only at the end of the polar decoding process, DCA-polar decoding recovers the CRC bits throughout the decoding process, which can be abandoned as soon as if is determined that the CRC will fail [54], [55]. Furthermore, it was observed in [53] that DCA-polar offers a slight FAR performance gain compared to CRC-aided polar (CA-polar) decoding in the downlink control channels. This gain is achieved by the CRC interleaver, which can move the CRC bits to more reliable positions.

Information block lengths in the range $A \in$ [12, 140] bits are supported in PDCCH, together with encoded block lengths in the range $G \in [A+24, 8192]$ [51], which are generated with the aid of a 24-bit distributed CRC (DCA-polar). Note however that encoded block lengths selected from the set $G \in$ {108, 216, 432, 864, 1728 } are used in practice.

The gNodeB multiplexes DCI intended for various connected UEs into the PDCCH. However, in order to reduce the control overhead, the gNodeB does not explicitly signal to the UEs where they can find their intended DCI within the time and frequency resources of the PDCCH. Instead, each UE performs a blind decoding process, in which it attempts the decoding of several hypothesised blocks having various combinations of information block length A, encoded block length G, DCI type and location within the PDCCH. As detailed in Section IV-C, the PDCCH polar code adopts an Radio Network Temporary Identifier (RNTI) scrambling process in order to ensure that an incorrect hypothesis will lead to a failing CRC with high probability, and that a passing CRC can only be obtained with high probability for a correct hypothesis.

In the next section, the operation of the blocks in Figures 4 to 6 will be explained in detail.

IV. Operations of Polar Coding Components in NR

This section details the operation of polar encoding and decoding in the PUCCH/PUSCH, PBCH and PDCCH schemes, introduced in Section III. Each of the following subsections details the operation of a different block in the schematics of Figures 4, 5 and 6. The operations is elaborated using schematics, flow charts, block diagrams, figures and examples.

A. Code Block Segmentation

Code block segmentation is employed in PUCCH/PUSCH when encoding and decoding long blocks, as defined in Subclause 5.2.1 of [37]. When code block segmentation is applied, the block is decomposed into $C = 2$ equal-length segments, which are encoded and decoded separately, but with identical configuration [56]–[58]. More specifically, $C =$ 2 block segments are used in PUCCH if the information block length is in the range $A \in$ [360, 1706] and the encoded block length is in the range $G \in$ [1088, 16385], or if $A \in$ [1013, 1066] and $G \in$ [1036, 1088]. Otherwise, $C = 1$ segment is used, and the code block remains undivided. There is no need to apply code block segmentation in PDCCH or PBCH, since the information block lengths are limited to A $\in$ [12, 140] for PDCCH and $A = 32$ for PBCH, as discussed in Section III.

There are two motivations for code block segmentation. The first is that it enables the polar code core length to be limited to $N = 1024$ bits, while still facilitating support for the largest required information block length of $A = 1706$ bits. This reduces the complexity of the polar code core, which increases with $Nlog(N)$. The second motivation is that code block segmentation allows encoded block lengths of up to $G = 2048$ bits to supported with $N = 1024$, without relying on repetition, which degrades the error correction performance of the polar code [56]–[63].

In the encoder, the input to code block segmentation is a block of A information bits $\mathbf{a} = [a_0, a_1, a_2, ..., a_{A-1}]$ and the output is either $C = 1$ or $C = 2$ code block segments, each of which comprises A' bits $\mathbf{a}' = [a'_0, a'_1, a'_2, ..., a'_{A'-1}]$, where $A' = \lceil A/C \rceil$.

Note that if A is odd and $C = 2$, then a single zero valued padding bit is inserted at the beginning of the information block [56]. When $C = 2$, the first set of A' bits in the (padded) information block becomes the first segment and the second set of A' bits becomes the second segment, as shown in Figure 7.

In the decoder, code block segmentation is reversed by concatenating the C number of decoded information block segments and removing the padding bit if one was used in the encoder.

B. Cyclic Redundancy Check (CRC) Calculation and Attachment

CRC calculation and attachment are defined in Subclause 5.1 of [37]. During polar encoding, a number of redundant CRC bits are appended to each information block segment $\mathbf{a}'$. These redundant CRC bits allow the polar decoder to perform error detection, as well as to aid its error correction [54]. More specifically, the polar decoder can perform CRC checks to determine if a codeword is free of errors, as well as to select a codeword from a list of decoding candidates, as it will be detailed in Section IV-F.

During the 3GPP NR polar codes standardization process, a particular error detection capability requirement was identified for each control channel, which was quantified by a corresponding target FAR. Following this, the numbers of CRC bits required for error detection was determined as $\lceil -log_2(\text{FAR}) \rceil$. Furthermore, it was agreed that CA-SCL (CRC-aided Successive Cancellation List) polar decoding with a list size of $L = 8$ offers an attractive trade-off between error correction performance and decoding complexity. Hence, it was agreed that a further $log_2(L) = 3$ CRC bits should be provided to aid the CA- SCL decoding algorithm. Then, a comprehensive evaluation of different CRC polynomials was performed in order to select those offering the best FAR. Finally, the 6-, 11- and 24-bit generator polynomials CRC6, CRC11 and CRC24 shown in Figures 8 to 11 were selected for the various physical control channels.

As described in Section IV-A, the A' information bits in each code block segment are denoted by $\mathbf{a}' = [a'_0, ..., a'_{A'-1}]$. During polar encoding, a sequence of P CRC parity bits $\mathbf{p} = [p_0, ..., p_{P-1}]$ are generated by a particular cyclic generator polynomial and attached to the end of the information bits, as illustrated in the examples of Figures 8 to 11 [64]. The resultant bit sequence comprises $A' + P$ bits and is denoted as $\mathbf{b} = [b_0, ..., b_{A'+P-1}]$.

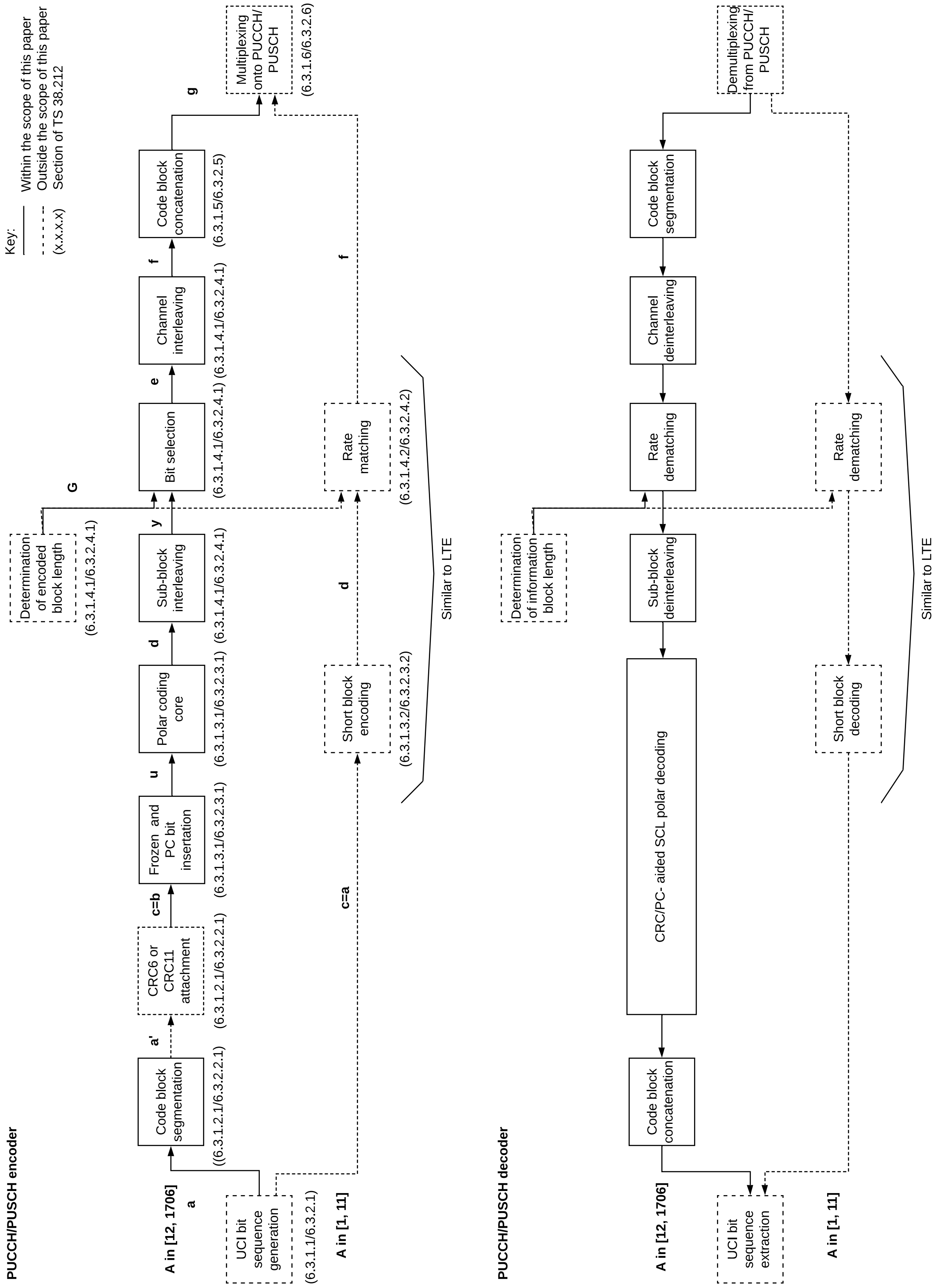


Fig. 4: Encoder and decoder block diagrams of channel coding for 3GPP NR PUCCH/PUSCH.

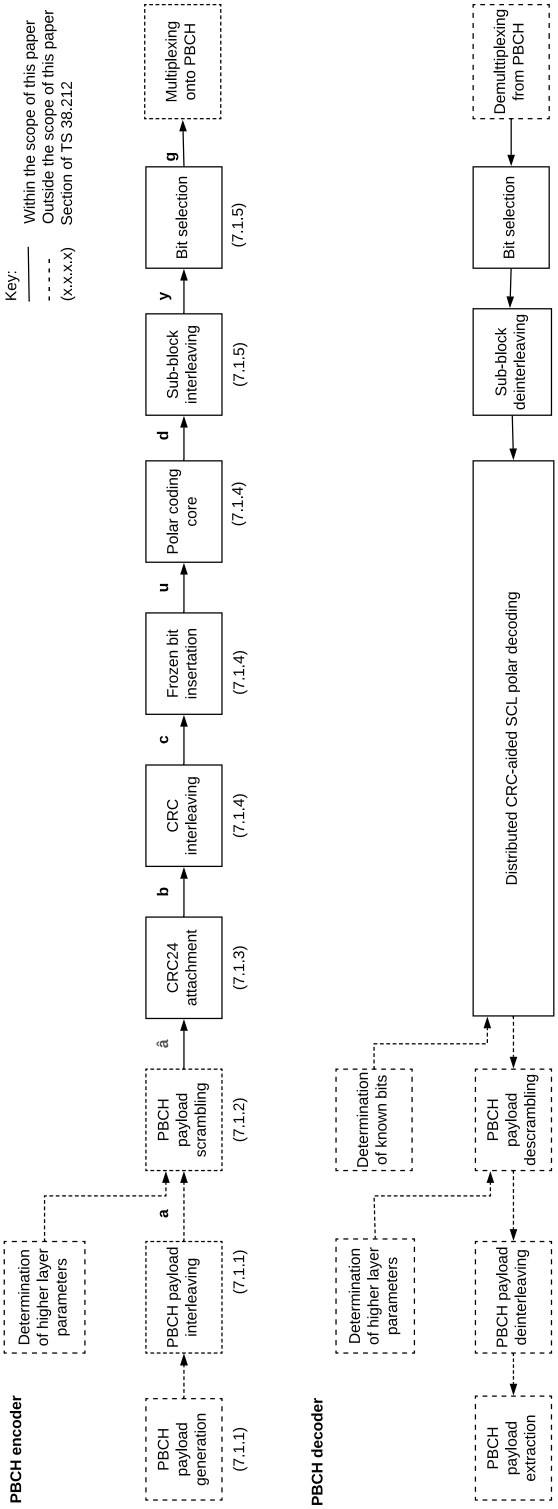


Fig. 5: Encoder and decoder block diagrams of channel coding for 3GPP NR PBCH.

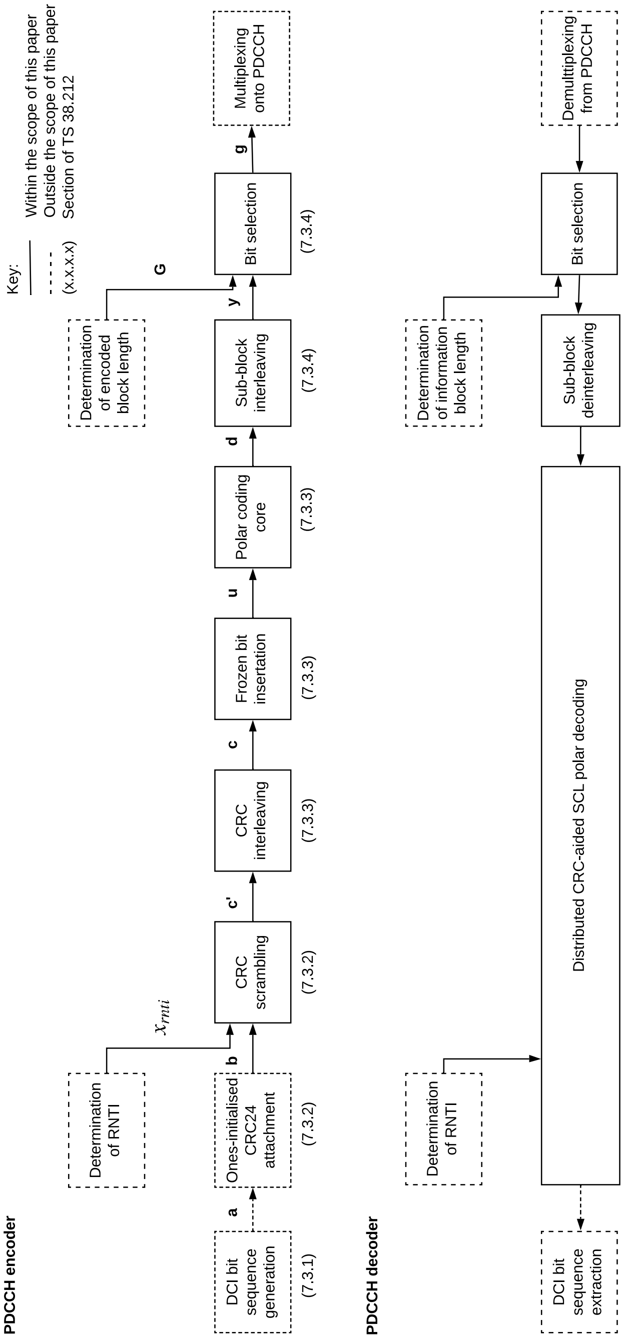

PDCCH encoder
DCI bit sequence generation
(7.3.1)
a
Ones-initialised CRC24 attachment
(7.3.2)
b
Determination of RNTI
x_{rnti}
CRC scrambling
(7.3.2)
c'
CRC interleaving
(7.3.3)
c
Frozen bit insertation
(7.3.3)
u
Polar coding core
(7.3.3)
d
Sub-block interleaving
(7.3.4)
y
Determination of encoded block length
G
Bit selection
(7.3.4)
g
Multiplexing onto PDCCH
Key:
Within the scope of this paper
Outside the scope of this paper
(x.x.x.x) Section of TS 38.212
PDCCH decoder
DCI bit sequence extraction
Determination of RNTI
Distributed CRC-aided SCL polar decoding
Sub-block deinterleaving
Determination of information block length
Bit selection
Demultiplexing from PDCCH

Fig. 6: Encoder and decoder block diagrams of channel coding for 3GPP NR PDCCH.

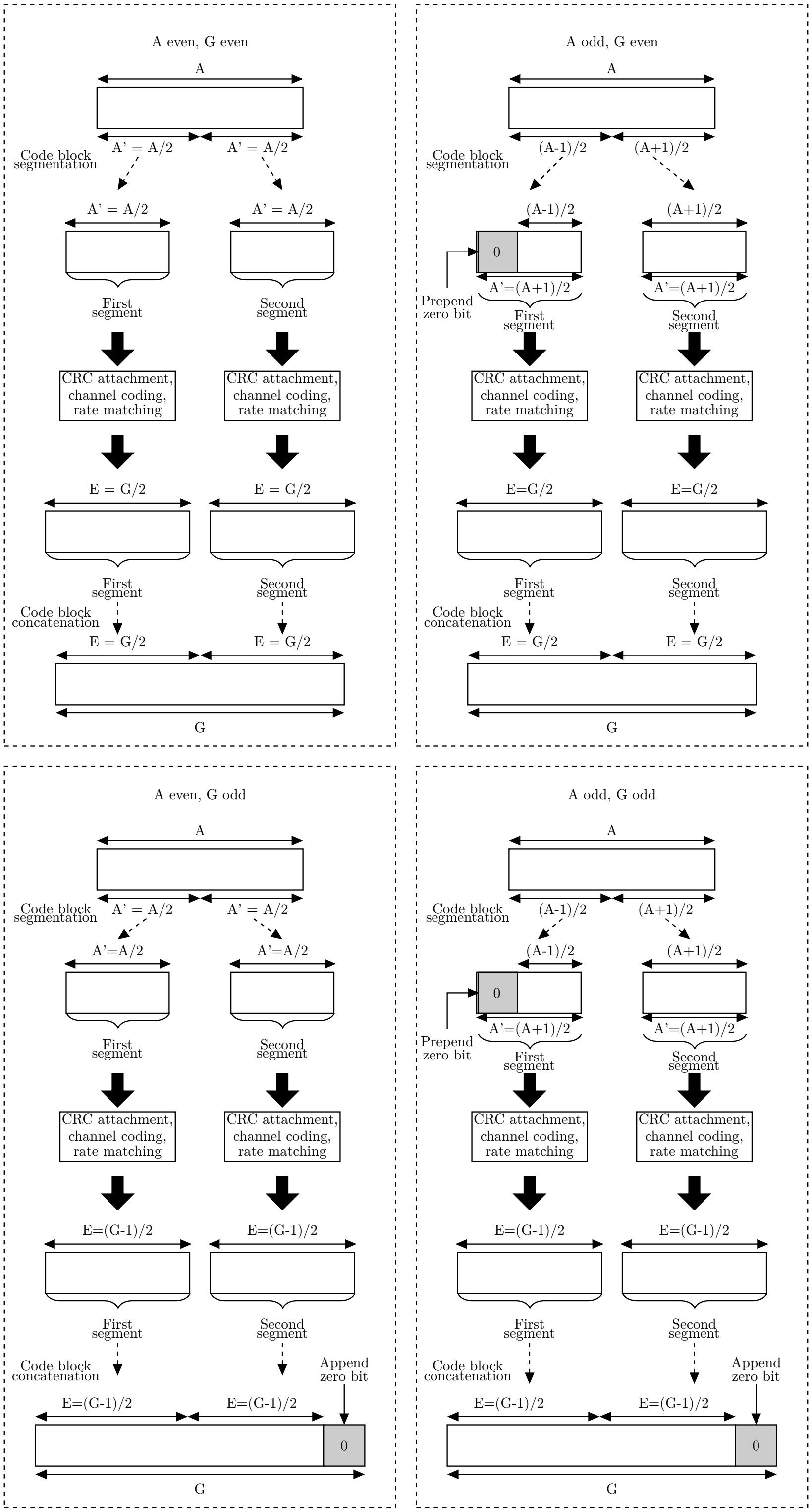


Fig. 7: Block diagram of code block segmentation and concatenation, considering various combinations of odd and even information block lengths (A) and encoded block lengths (G).

In the case of PUCCH with an information block length of $A \in [12,19]$, $P = 6$ CRC bits are used [51], which are obtained using the generator polynomial

$$g_{CRC6}(D) = [D^6 + D^5 + 1], \tag{2}$$

as exemplified in Figure 8.

If $A \in [20,1706]$ for PUCCH, then a $P = 11$ bit CRC is generated [51] using the generator polynomial

$$g_{CRC11}(D) = [D^{11} + D^{10} + D^9 + D^5 + 1], \tag{3}$$

as exemplified in Figure 9.

For all block lengths in PDCCH and PBCH, $P = 24$ CRC bits are used, which are obtained using the generator polynomial

$$\begin{aligned} g_{CRC24C}(D)^1 = [&D^{24} + D^{23} + D^{21} + D^{20} + D^{17} + D^{15} + D^{13} + D^{12} \\ &+ D^8 + D^4 + D^2 + D + 1], \end{aligned} \tag{4}$$

as exemplified in Figure 10 for PBCH.

In the case of PDCCH, the CRC generator is pre-loaded with sequence of $P = 24$ one-valued bits, before the information bits are input, as exemplified in Figure 11. This approach is employed for the sake of reducing the prevalence of false alarms during blind decoding [51], [65]. More specifically, this mechanism reduces the likelihood of erroneously having a successful CRC check when using the wrong encoded block lengths G during the blind decoding [65].

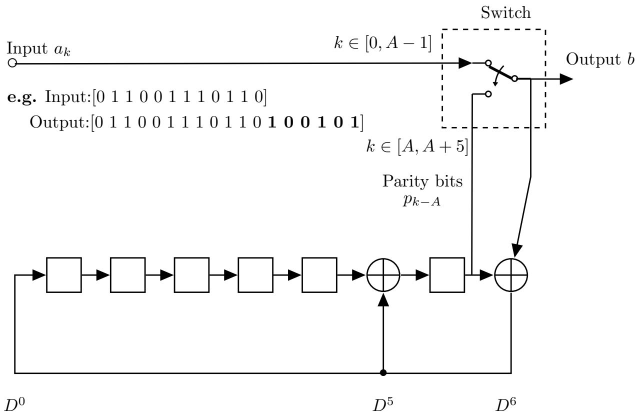


Fig. 8: Shift register block diagram of CRC-6, as used for PUCCH with an information block length of $A \in [12,19]$. Note that each shift register memory element is initialized to store a zero-valued bit before the CRC generation process is begun.

During polar decoding, a CRC check may be performed by inputting a sequence of A' decoded information bits into a CRC generator and comparing the resultant P CRC bits with the sequence of P decoded CRC bits. If the two P-bit sequences match, then the CRC check is successful and no errors are detected. Alternatively, the sequence of $A'+P$ decoded information and CRC bits may be input into the CRC generator, in order to obtain a P-bit syndrome. If the syndrome comprises P zero-valued bits, then the CRC check is successful and no errors are detected. Note that this process must be adjusted when using RNTI scrambling, CRC interleaving and CRC-aided SCL decoding, as discussed in Sections IV-C, D and F.

C. Radio Network Temporary Identifier (RNTI) Scrambling

RNTI scrambling for PDCCH is defined in Subclause 7.3.2 of [37]. Each UE in a cell is allocated a unique RNTI, which is used by the Medium Access Control (MAC) protocol to address particular downlink control information to particular UEs [46]. After the $P = 24$ bit CRC has been attached during PDCCH encoding in a basestation, the last 16 of these CRC parity bits are scrambled by performing XORs with the RNTI corresponding to the destination UE [64]–[66].

During CRC checking in a receiving UE, the last 16 bits of the received CRC are similarly scrambled using the UE's RNTI. In this way, only the intended UE will be able to pass through the CRC check. This avoids the requirement for the basestation to explicitly include 16-bit RNTI of the destination UE in the transmitted DCI.

D. CRC Interleaving

CRC interleaving for the NR polar codes is defined in Subclause 5.3.1.1 of [37]. It is employed in order to shuffle the order of the bits obtained following CRC attachment during PBCH encoding and following CRC scrambling during PDCCH encoding, as shown in Figures 5 and 6. Since the maximum of the information block lengths A supported in these channels is 140 bits and because the same $P = 24$-bit CRC generator polynomial is used in both channels, the maximum number of bits K that are interleaved by the CRC interleaver is 164. Hence, the CRC interleaver adopts a mother CRC interleaving pattern having a length of 164 bits [37], as specified in Table 5.3.1.1-1 of [37].

All interleaver patterns for lower numbers of bits K may be derived from this single mother pattern [67]. More specifically, a sequence of K bits may be prepended with $164 - K$ NULL-valued bits and then interleaved using the 164-bit interleaver pattern. The K-bit interleaver pattern may then be obtained by considering the interleaving operation that results from eliminating all NULL-valued bits in the bits sequences before and after interleaving. For example, Figures 12 and 13 illustrate the $K = 36$-bit and $K = 40$-bit interleaver patterns, which are used for $A = 12$-bit and $A = 16$-bit information blocks during PDCCH encoding, respectively. It may be observed that the $K = 36$-bit interleaver pattern of Figure 12 may be obtained by eliminating the four elements of the $K = 40$-bit interleaver

[1] Note that Subclause 5.1 of [37] provides three different cyclic generator polynomials for generating 24-bit CRCs, namely $g_{CRC24A}(D)$, $g_{CRC24B}(D)$ and $g_{CRC24C}(D)$. Of these, only g_{CRC24C} is used for polar codes. The two other cyclic generator polynomials g_{CRC24A} and g_{CRC24B} are used for generating 24-bit CRCs for the LDPC codes in the NR shared channels for data transmission.

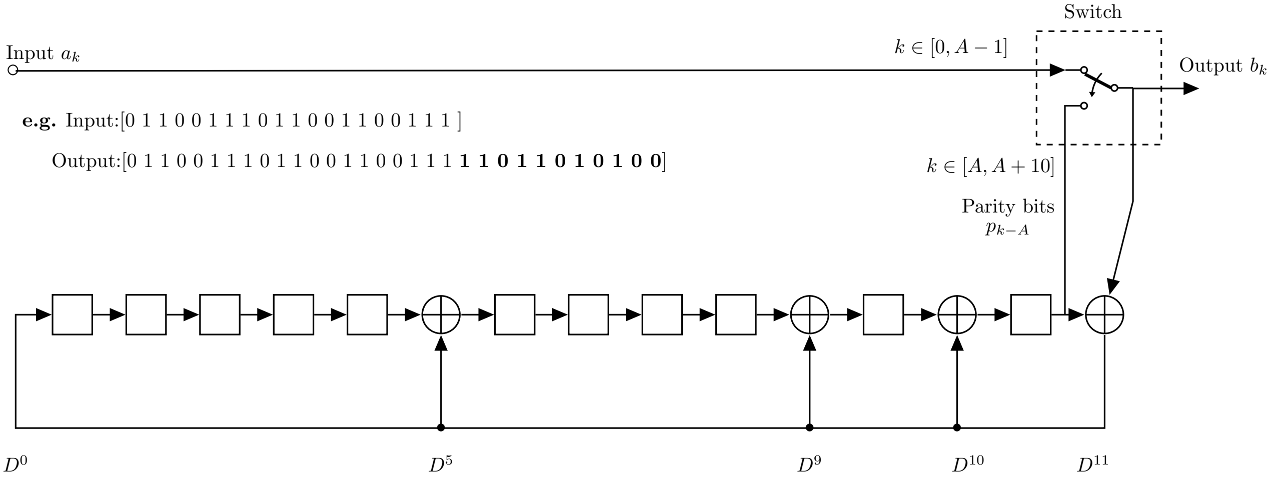

Switch
Input a_k
$k \in [0, A-1]$
Output b_k
e.g. Input:[0 1 1 0 0 1 1 1 0 1 1 0 0 1 1 0 0 1 1 1]
Output:[0 1 1 0 0 1 1 1 0 1 1 0 0 1 1 0 0 1 1 1 **1 1 0 1 1 0 1 0 1 0 0**]
$k \in [A, A+10]$
Parity bits
p_{k-A}
D^0
D^5
D^9
D^{10}
D^{11}

Fig. 9: Shift register block diagram of CRC-11, as used for PUCCH with an information block length of $A \in$[20,1706]. Note that each shift register memory element is initialized to store a zero-valued bit before the CRC generation process is begun.

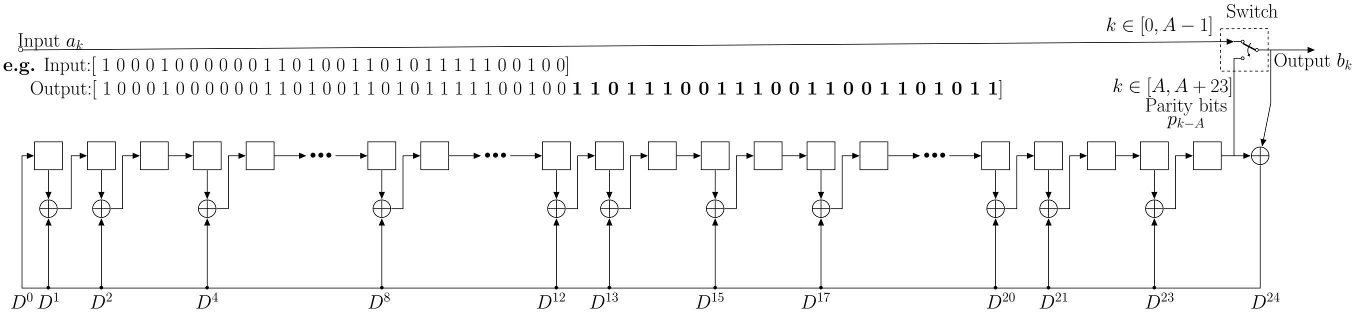

Switch
$k \in [0, A-1]$
Input a_k
e.g. Input:[1 0 0 0 1 0 0 0 0 0 0 1 1 0 1 0 0 1 1 0 1 0 1 1 1 1 1 0 0 1 0 0]
Output:[1 0 0 0 1 0 0 0 0 0 0 1 1 0 1 0 0 1 1 0 1 0 1 1 1 1 1 0 0 1 0 0 **1 1 0 1 1 1 0 0 1 1 1 0 0 1 1 0 0 1 1 0 1 0 1 1**]
Output b_k
$k \in [A, A+23]$
Parity bits
p_{k-A}
D^0
D^1
D^2
D^4
D^8
D^{12}
D^{13}
D^{15}
D^{17}
D^{20}
D^{21}
D^{23}
D^{24}

Fig. 10: Shift register block diagram of CRC-24, as used for PBCH with an information block length of $A = 32$. Note that each shift register memory element is initialized to store a zero-valued bit before the CRC generation process is begun.

pattern of Figure 13 that are indicated with bold dashed lines.

The CRC interleaver is designed for ensuring that each of the $P = 24$ CRC bits depends only on the information bits that precede it in the interleaved sequence [34]. For example, in Figure 12, the first CRC bit p_0 depends only on the information bits a_1, a_4, a_6, a_{10} and a_{11}. Likewise, the second CRC bit p_1 depends on some of those information bits and also depends on a_0, a_2, a_5 and a_7. This property may be exploited during a polar decoding process in which the information and CRC bits are obtained one at a time, in their interleaved order. More specifically, in the example of Figure 12, an expected value of the first CRC bit p_0 may be obtained as soon as the first five interleaved bits c_0 to c_4 have been decoded. This expected value can then be compared with the value of the interleaved CRC bit c_5. If these two bit values do not match, then it may be determined that a failing CRC will result from this polar decoding process. Hence, the decoding process may be early terminated before it is completed, in order to reduce the decoding complexity.

This is particularly useful during PDCCH blind decoding, where it may be expected that most decoding hypothesis will fail, as described in Section III. In situations where the CRC check is successful passes and early termination is not applied during polar decoding, the interleaving operation may be reversed by applying an inverse interleaving pattern, in order to restore the original order of the information bits and of their appended CRC bits.

E. Frozen and PC Bit Insertion

In order to enable forward error correction, the NR polar code introduces redundancy in the form of frozen and PC bits into the sequence of information and CRC bits, as defined in Subclause 5.3.1 of [37]. In the encoder, the input of this process is the information block segment along with (and interleaved in the case of PDCCH or PBCH) the CRC $\mathbf{c} = [c_0, ..., c_{K-1}]$ appended, while the output of frozen and

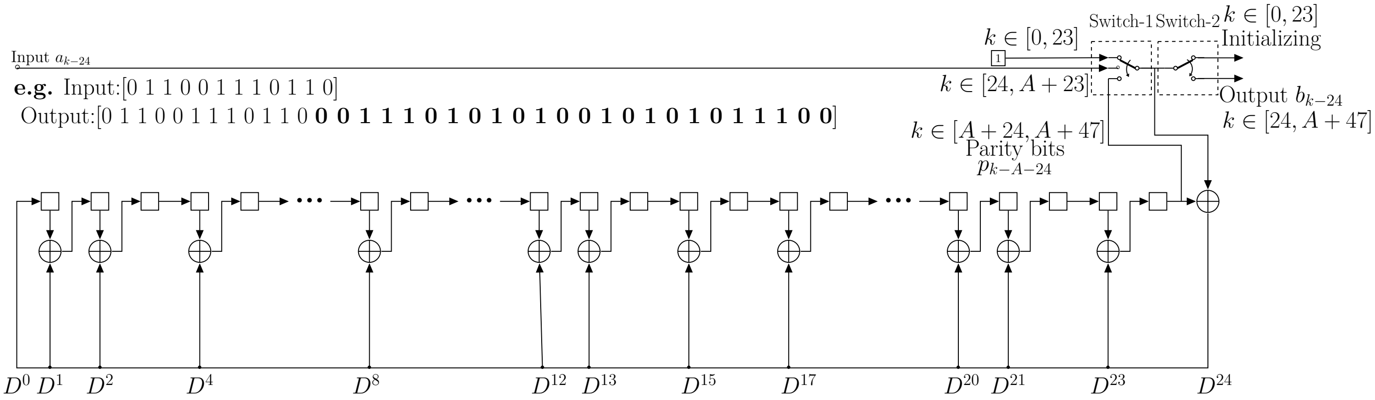

Input a_{k-24}
e.g. Input:[0 1 1 0 0 1 1 1 0 1 1 0]
Output:[0 1 1 0 0 1 1 1 0 1 1 0 0 0 1 1 1 0 1 0 1 0 1 0 0 1 0 1 0 1 0 1 1 1 0 0]
$k \in [0, 23]$
1
Switch-1
Switch-2
$k \in [0, 23]$
Initializing
$k \in [24, A+23]$
Output b_{k-24}
$k \in [24, A+47]$
$k \in [A+24, A+47]$
Parity bits
p_{k-A-24}
D^0 D^1 D^2 D^4 D^8 D^{12} D^{13} D^{15} D^{17} D^{20} D^{21} D^{23} D^{24}

Fig. 11: Shift register block diagram of CRC-24, as used for PDCCH with an information block length of $A \in [12, 140]$. However, the CRC process begins by clocking 24 one-valued bits into the CRC generator, in a process that re-initializes the shift register memory elements, from their initial values of zero.

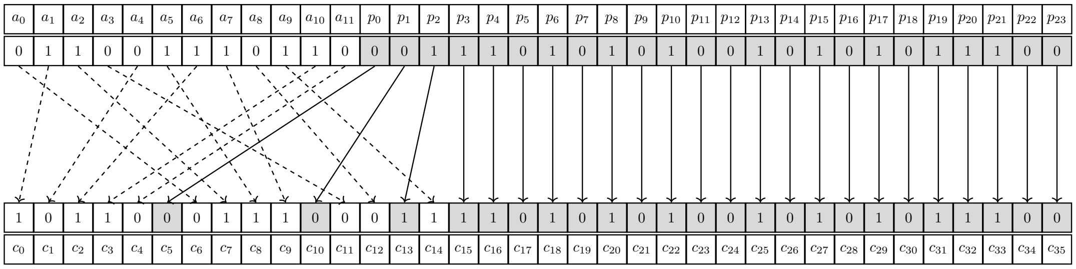

a_0 a_1 a_2 a_3 a_4 a_5 a_6 a_7 a_8 a_9 a_{10} a_{11} p_0 p_1 p_2 p_3 p_4 p_5 p_6 p_7 p_8 p_9 p_{10} p_{11} p_{12} p_{13} p_{14} p_{15} p_{16} p_{17} p_{18} p_{19} p_{20} p_{21} p_{22} p_{23}
0 1 1 0 0 1 1 1 0 1 1 0 0 0 1 1 1 0 1 0 1 0 1 0 0 1 0 1 0 1 0 1 1 1 0 0
1 0 1 1 0 0 0 1 1 1 0 0 0 1 1 1 1 0 1 0 1 0 1 0 0 1 0 1 0 1 0 1 1 1 0 0
c_0 c_1 c_2 c_3 c_4 c_5 c_6 c_7 c_8 c_9 c_{10} c_{11} c_{12} c_{13} c_{14} c_{15} c_{16} c_{17} c_{18} c_{19} c_{20} c_{21} c_{22} c_{23} c_{24} c_{25} c_{26} c_{27} c_{28} c_{29} c_{30} c_{31} c_{32} c_{33} c_{34} c_{35}

Fig. 12: An example of CRC-24 attachment and interleaving when using an information block length $A = 12$ for PDCCH.

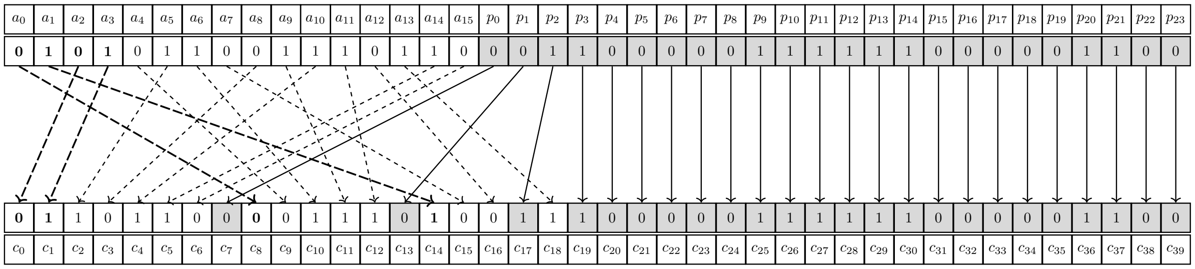

a_0 a_1 a_2 a_3 a_4 a_5 a_6 a_7 a_8 a_9 a_{10} a_{11} a_{12} a_{13} a_{14} a_{15} p_0 p_1 p_2 p_3 p_4 p_5 p_6 p_7 p_8 p_9 p_{10} p_{11} p_{12} p_{13} p_{14} p_{15} p_{16} p_{17} p_{18} p_{19} p_{20} p_{21} p_{22} p_{23}
0 1 0 1 0 1 1 0 0 1 1 1 0 1 1 0 0 0 1 1 0 0 0 0 0 1 1 1 1 1 1 0 0 0 0 0 1 1 0 0
0 1 1 0 1 1 0 0 0 0 1 1 1 0 1 0 0 1 1 1 0 0 0 0 0 1 1 1 1 1 1 0 0 0 0 0 1 1 0 0
c_0 c_1 c_2 c_3 c_4 c_5 c_6 c_7 c_8 c_9 c_{10} c_{11} c_{12} c_{13} c_{14} c_{15} c_{16} c_{17} c_{18} c_{19} c_{20} c_{21} c_{22} c_{23} c_{24} c_{25} c_{26} c_{27} c_{28} c_{29} c_{30} c_{31} c_{32} c_{33} c_{34} c_{35} c_{36} c_{37} c_{38} c_{39}

Fig. 13: An example of CRC-24 attachment and interleaving when using an information block length $A = 16$ for PDCCH.

PC bits insertion $\mathbf{u} = [u_0, ..., u_{N-1}]$ becomes the input of the polar encoder core. More specifically, frozen and PC bits are inserted in order to increase the length of the input bit sequence from K bits to the polar core block length N, which must be a power of 2, selected from the set $N = \{32, 64, 128, 256, 512, 1024\}$. The selection of which K of the N bits should be information or CRC bits, as well as of which $N - K$ bits should be frozen or PC bits is governed by a so-called sequence, which ranks the reliability of the N bits positions, as well as by the selected rate matching mode, as detailed in Section IV-H. The value of N is determined as a function of the information block length A and the encoded block length G, as detailed in Subclause 5.3.1 of [37], and illustrated in Figures 14 and 15 for PUCCH and PDCCH, respectively. In the case of PBCH, the polar code core block length is always $N = 512$ [68].

The NR polar code sequence Q_0^{N-1} corresponding to each supported value of N may be extracted as a subset from the mother polar sequence $Q_0^{N_{max}-1}$, which is specified in Table 5.3.1.2-1 of [37] for $N_{max} = 1024$ [37]. This process is illustrated in Figure 16, which shows an example to describe how the sequence for $N = 32$ may be obtained from the $N =$ 64-bit sequence. Here, all elements in the polar sequence Q_0^{63} having a value less than $N = 32$ are retained for the sequence Q_0^{31}, where they are concatenated in order.

After determining the value of N and the corresponding NR polar code sequence Q_0^{N-1}, the positions of the K information and CRC bits and the positions of the $N-K$ frozen and PC bits are determined, as detailed in Figure 17. In particular, bit positions that corresponding to puncturing or shortening during rate matching are used for frozen bits. In general, frozen bits are inserted into the remaining lower reliability positions, while information and CRC bits take the remaining higher reliability positions, as identified by the NR polar code sequence Q_0^{N-1}. This process is dependent on the rate matching mode that is used, as discussed on Section IV-H.

Owing to this, the frozen and PC bit insertion process requires knowledge of the various rate matching parameters, including the encoded block segment length E. Note that PC bits are only used in PUCCH/PUSCH and only when $A \in [12, 19]$ [69]. Here, $n_{PC} = 3$ bits are placed in the least reliable positions identified by the sequence Q_0^{N-1} unless $E - K + n_{PC} > 192$ in which case, one of the PC bits is placed in the most reliable bit position that remains after the positions of the information and CRC bits have been selected, as described in Figure 18.

In contrast to the frozen bits, which always adopt binary values of zero, the PC bits can adopt values of zero or one [70]. More specifically, the $n_{PC} = 3$ PC bits are obtained from a length-5 cyclic shift register. This is illustrated in Figure 19 [51], [55], [69], where the input is provided by the N-bit sequence $\mathbf{u} = [u_0, ..., u_{N-1}]$, comprising the K information and CRC bits in their correct positions, $N-K-n_{PC}$ zero-valued frozen bits and $n_{PC} = 3$ zero-valued place holders for the PC bits. This N-bit sequence is inserted into the cyclic shift register, which replaced the $n_{PC} = 3$ place-holders with the PC bit values.

In the decoder, the same process is used to determine the positions of the frozen and PC bits. The redundant bits are used to aid the polar decoding core, as described in Section IV-F. Following this, the frozen and PC bits are removed, in order to obtain the decoded information and CRC bits.

F. Polar Coding Core

After inserting frozen and PC bits among the information and CRC bits, polar encoding is performed by multiplying the row vector of N input bits $\mathbf{u} = [u_0, ..., u_{N-1}]$ by a Kronecker kernel matrix $\mathbf{G_N}$ to obtain a row vector of N encoded bits $\mathbf{d} = [d_0, ..., d_{N-1}]$ [55], according to $\mathbf{d} = \mathbf{u}\mathbf{G_N}$. Here, $\mathbf{G_N} = \mathbf{G_2}^{\oplus n}$ is the n^{th} Kronecker power of matrix $\mathbf{G_2}$, where $n = \log_2(N)$ and

$$\mathbf{G_2} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \qquad (5)$$

$$\mathbf{G_N} = \mathbf{G_2}^{\oplus n} = \begin{bmatrix} \mathbf{G_2}^{\oplus(n-1)} & 0 \\ \mathbf{G_2}^{\oplus(n-1)} & \mathbf{G_2}^{\oplus(n-1)} \end{bmatrix}. \qquad (6)$$

For example, the n = 2nd Kronecker power of matrix $\mathbf{G_2}$ is given by

$$\mathbf{G_4} = \mathbf{G_2}^{\oplus 2} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}. \qquad (7)$$

A polar encoding core operation can be represented by a graph, as exemplified in Figure 20 for the case of $N = 32$. More specifically, the graph comprises $n = \log_2 N$ horizontally-concatenated stages, each of which comprises $N/2$ vertically-aligned eXclusive-OR (XOR) operations [71], [72]. In this representation, the sequence of input bits $\mathbf{u} = [u_0, ..., u_{N-1}]$ is placed at the left-hand edge of the graph and the successive stages of XOR operations may be used to obtain the sequence of encoded bits $\mathbf{d} = [d_0, ..., d_{N-1}]$ on the right-hand edge.

In the receiver, polar encoding is complemented by polar decoding. The input to polar decoding is provided by soft information pertaining to the sequence of bits $\mathbf{d} = [d_0, ..., d_{N-1}]$. Here, the soft information is used because the demodulators are typically unable to gain absolute confidence about the value of the received bits [72], [73]. More specifically, the soft information expresses not only what the most likely value of each bit is, but also how likely that value is [72], [73]. The soft information $\hat{d}_k$ pertaining to each bit d_k is typically represented in the form of a Logarithmic Likelihood Ratio (LLR)

$$\hat{d}_k = \ln\left[\frac{\Pr(d_k = 0)}{\Pr(d_k = 1)}\right], \qquad (8)$$

where $\Pr(d_k = 0)$ represents the demodulator's confidence that the correct value of d_k is zero and $\Pr(d_k = 1)$ represents the demodulator's confidence that the correct value of d_k is one [73].

The sign of an LLR conveys the most likely value of the corresponding bit, where $\hat{d}_k > 0$ implies that a bit value of is zero more likely, while $\hat{d}_k < 0$ implies a bit value of one [73]. Additionally, the magnitude of an LLR conveys how likely that bit value is to be correct, where higher magnitudes correspond to higher likelihoods [73].

A number of decoding algorithms have been proposed for polar codes since their first introduction. Among these, Successive Cancellation (SC) decoding [11] extracts the successive bits of the bit sequence $\mathbf{u} = [u_0, ..., u_{N-1}]$ from the received LLRs $\hat{\mathbf{d}} = [\hat{d}_0, ..., \hat{d}_{N-1}]$ one at a time in order from $\hat{d}_0$ to $\hat{d}_{N-1}$. In this way, each successive decoded bit can then be eliminated from consideration during the decoding of the following bits. SC decoding benefits from a low complexity, but suffers from relatively weak error correction capability in wireless channels [74]–[76].

In order to improve the error correction capability of polar codes in wireless channels Successive Cancellation List

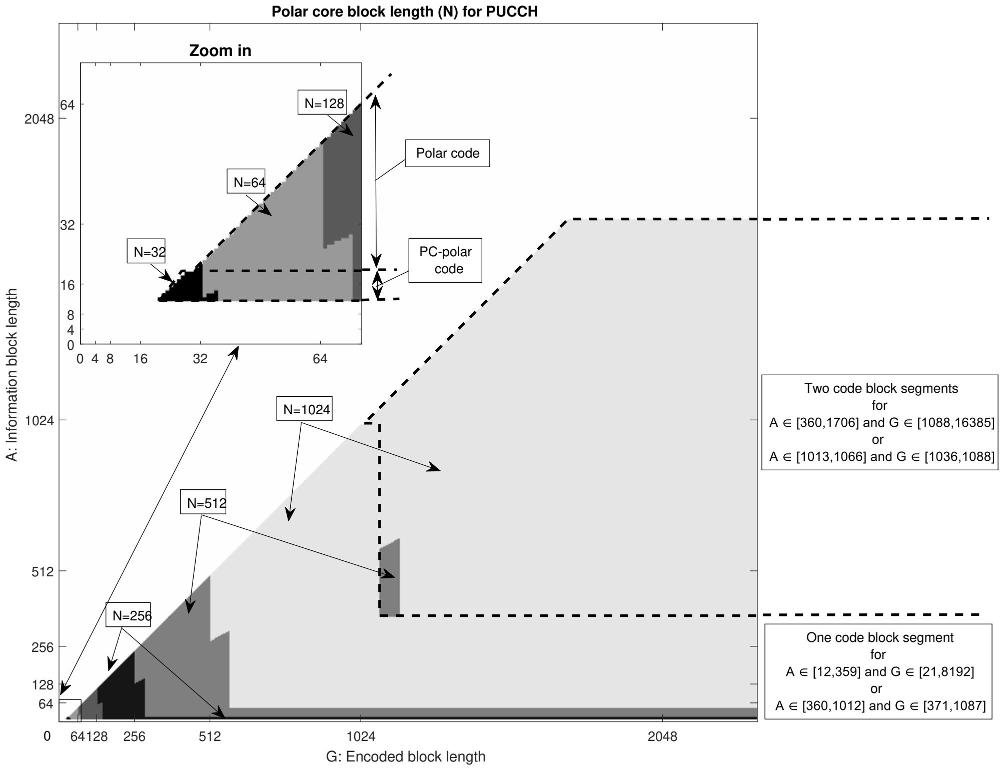


Fig. 14: Polar core block length N for each combination of the number of code block segments C, PC-polar mode, information block length A and encoded block length G for PUCCH.

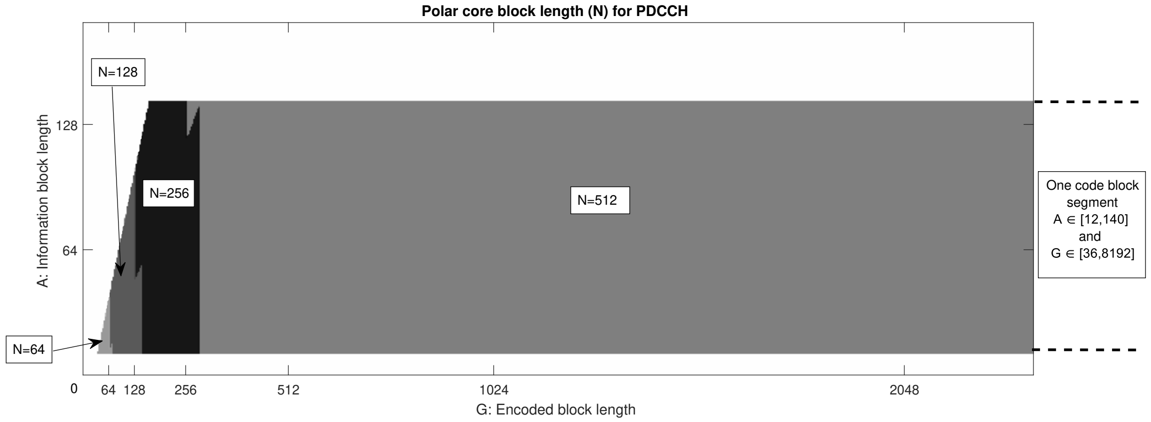


Fig. 15: Polar core block length N for each combination of information block length A and encoded block length G for PDCCH.

(SCL) decoding was proposed in [73]. Here, SCL decoding is parametrised by the list size L, which quantities the number of SC decoding process performed in parallel [73], [75]–[79]. More specifically, whenever one of these SC decoding processes reaches an information or CRC bit, it is branched into two replicas of that SC decoding process. In one of the replicas, referred to as a candidate decoding path [73], [75]–[79], a value of zero is assumed for the information

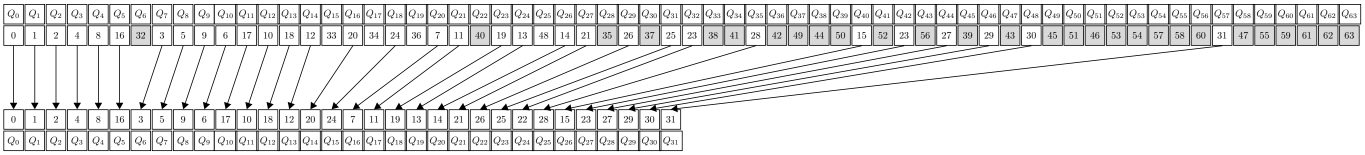

Q_0 Q_1 Q_2 Q_3 Q_4 Q_5 Q_6 Q_7 Q_8 Q_9 Q_{10} Q_{11} Q_{12} Q_{13} Q_{14} Q_{15} Q_{16} Q_{17} Q_{18} Q_{19} Q_{20} Q_{21} Q_{22} Q_{23} Q_{24} Q_{25} Q_{26} Q_{27} Q_{28} Q_{29} Q_{30} Q_{31} Q_{32} Q_{33} Q_{34} Q_{35} Q_{36} Q_{37} Q_{38} Q_{39} Q_{40} Q_{41} Q_{42} Q_{43} Q_{44} Q_{45} Q_{46} Q_{47} Q_{48} Q_{49} Q_{50} Q_{51} Q_{52} Q_{53} Q_{54} Q_{55} Q_{56} Q_{57} Q_{58} Q_{59} Q_{60} Q_{61} Q_{62} Q_{63}
0 1 2 4 8 16 32 3 5 9 6 17 10 18 12 33 20 34 24 36 7 11 40 19 13 48 14 21 35 26 37 25 23 38 41 28 42 49 44 50 15 52 23 56 27 39 29 43 30 45 51 46 53 54 57 58 60 31 47 55 59 61 62 63
0 1 2 4 8 16 3 5 9 6 17 10 18 12 20 24 7 11 19 13 14 21 26 25 22 28 15 23 27 29 30 31
Q_0 Q_1 Q_2 Q_3 Q_4 Q_5 Q_6 Q_7 Q_8 Q_9 Q_{10} Q_{11} Q_{12} Q_{13} Q_{14} Q_{15} Q_{16} Q_{17} Q_{18} Q_{19} Q_{20} Q_{21} Q_{22} Q_{23} Q_{24} Q_{25} Q_{26} Q_{27} Q_{28} Q_{29} Q_{30} Q_{31}

Fig. 16: Example of obtaining the $N = 32$-bit 3GPP polar sequence from the $N = 64$-bit sequence.

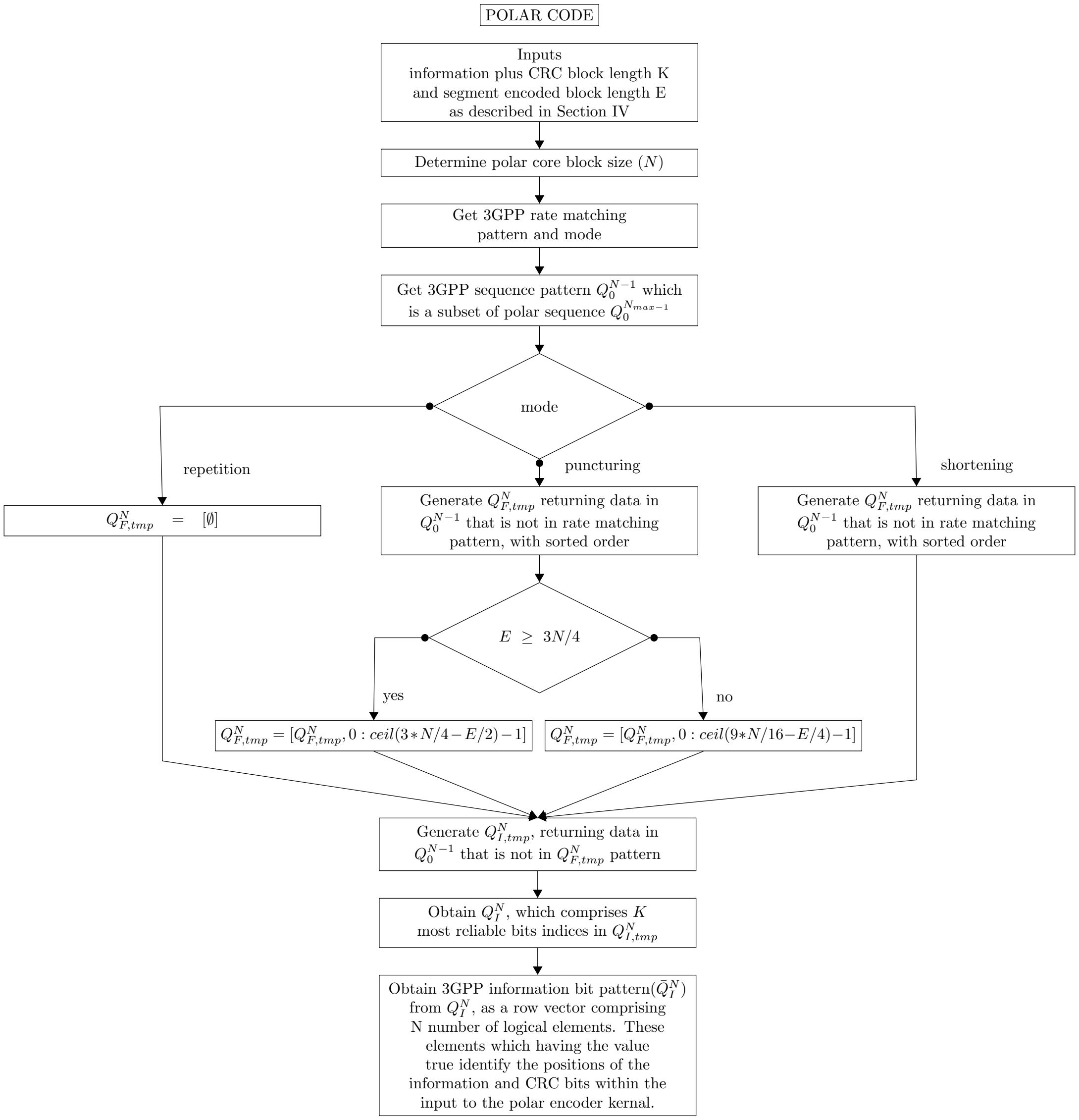

POLAR CODE
Inputs
information plus CRC block length K
and segment encoded block length E
as described in Section IV
Determine polar core block size (N)
Get 3GPP rate matching
pattern and mode
Get 3GPP sequence pattern Q_0^{N-1} which
is a subset of polar sequence $Q_0^{N_{max-1}}$
mode
repetition
puncturing
shortening
$Q_{F,tmp}^N = [\emptyset]$
Generate $Q_{F,tmp}^N$ returning data in
Q_0^{N-1} that is not in rate matching
pattern, with sorted order
Generate $Q_{F,tmp}^N$ returning data in
Q_0^{N-1} that is not in rate matching
pattern, with sorted order
$E \geq 3N/4$
yes
no
$Q_{F,tmp}^N = [Q_{F,tmp}^N, 0 : ceil(3*N/4-E/2)-1]$
$Q_{F,tmp}^N = [Q_{F,tmp}^N, 0 : ceil(9*N/16-E/4)-1]$
Generate $Q_{I,tmp}^N$, returning data in
Q_0^{N-1} that is not in $Q_{F,tmp}^N$ pattern
Obtain Q_I^N, which comprises K
most reliable bits indices in $Q_{I,tmp}^N$
Obtain 3GPP information bit pattern($\bar{Q}_I^N$)
from Q_I^N, as a row vector comprising
N number of logical elements. These
elements which having the value
true identify the positions of the
information and CRC bits within the
input to the polar encoder kernal.

Fig. 17: Flow chart for generating the information bit pattern $\bar{Q}_I^N$.

bit, while a value of one is assumed in the other. In this way, the number of decoding paths is doubled whenever an information or CRC bits is reached, until the list size L is exceeded [73], [75]–[79], whereupon all but the best L paths are discarded based on their path metrics (PMs), which may be calculated to quantify the likelihood of each decoding path being successful [72], [75]–[79].

Whenever a PC bit is encountered during the SCL de-

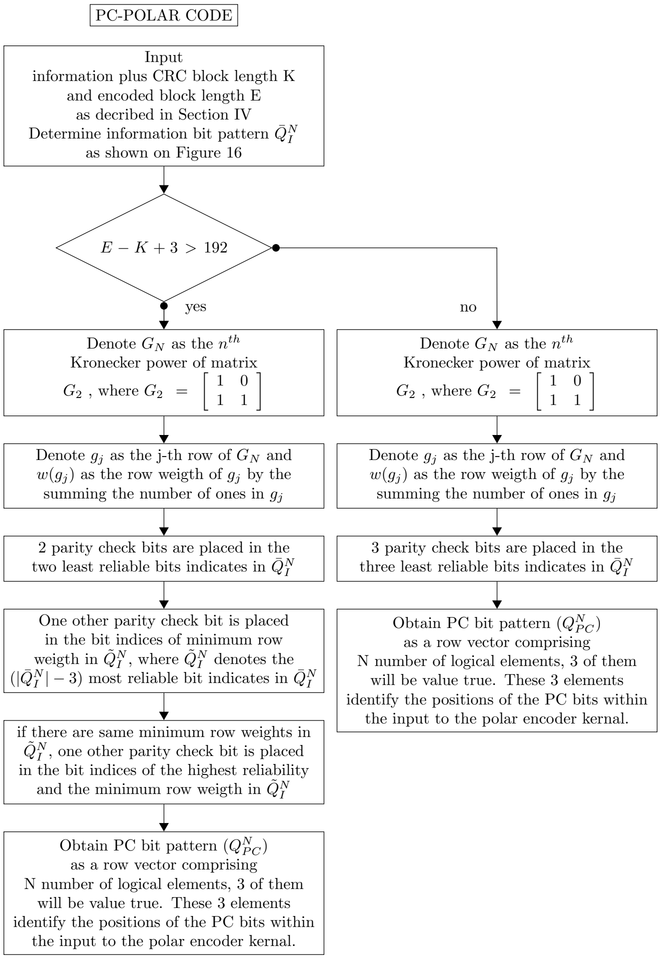

PC-POLAR CODE
Input
information plus CRC block length K
and encoded block length E
as decribed in Section IV
Determine information bit pattern $\bar{Q}_I^N$
as shown on Figure 16
$E - K + 3 > 192$
yes
no
Denote G_N as the n^{th}
Kronecker power of matrix
G_2 , where $G_2 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$
Denote g_j as the j-th row of G_N and
$w(g_j)$ as the row weigth of g_j by the
summing the number of ones in g_j
2 parity check bits are placed in the
two least reliable bits indicates in $\bar{Q}_I^N$
One other parity check bit is placed
in the bit indices of minimum row
weigth in $\tilde{Q}_I^N$, where $\tilde{Q}_I^N$ denotes the
$(|\bar{Q}_I^N| - 3)$ most reliable bit indicates in $\bar{Q}_I^N$
if there are same minimum row weights in
$\tilde{Q}_I^N$, one other parity check bit is placed
in the bit indices of the highest reliability
and the minimum row weigth in $\tilde{Q}_I^N$
Obtain PC bit pattern (Q_{PC}^N)
as a row vector comprising
N number of logical elements, 3 of them
will be value true. These 3 elements
identify the positions of the PC bits within
the input to the polar encoder kernal.
Denote G_N as the n^{th}
Kronecker power of matrix
G_2 , where $G_2 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$
Denote g_j as the j-th row of G_N and
$w(g_j)$ as the row weigth of g_j by the
summing the number of ones in g_j
3 parity check bits are placed in the
three least reliable bits indicates in $\bar{Q}_I^N$
Obtain PC bit pattern (Q_{PC}^N)
as a row vector comprising
N number of logical elements, 3 of them
will be value true. These 3 elements
identify the positions of the PC bits within
the input to the polar encoder kernal.

Fig. 18: Flow chart for generating the PC bit pattern.

coding of the PC-polar code used for PUCCH with $A \in [12, 19]$, the value of the PC bit within each particular decoding candidate path may be determined as a function of the preceding bits. Following this, the PC bit may be treated as a frozen bit having that particular value.

Once all N bits have been processed during SCL decoding, the winning decoding path must be selected from among the L candidates. This may be performed with consideration of the CRC bits in order to improve the error correction performance at the cost of degrading the error detection performance. However, in each of the NR control channels, the number of CRC bits used is three greater than is required to meet the error detection requirements of NR. This may be exploited by performing a CRC check for the $2^3 = 8$ decoding candidates having the best path matrices. If the CRC passes for any one of these eight decoding candidates, then it may be output as the winner.

As described in Section IV-D, a distributed CRC is adopted for PDCCH and PBCH in order to facilitate early termination [54], [55]. In the case of SCL decoding, early termination may be invoked as soon as it has been determined that all decoding candidates contain at least one failing CRC bit [54], [55].

G. Sub-block Interleaving

In 3GPP NR, the process of rate matching is decomposed into three subsections, the first of which is sub-block interleaving, as specified in Subclause 5.4.1.1 of [37]. The bits input into the sub-block interleaver are the sequence of N encoded bits $\mathbf{d} = [d_0, ..., d_{N-1}]$ generated by the polar encoder core. Meanwhile, the sequence of output bits is denoted by $\mathbf{y} = [y_0, ..., y_{N-1}]$ and also has a length N, which is guaranteed to be a multiple of 32. The purpose of sub-block interleaving is to reorder the N encoded bits for ensuring that the bits that are most important for the polar code's error correction capability are retained during bit selection [80], as discussed in Section IV-H. Sub-block interleaving is performed by decomposing the sequence of

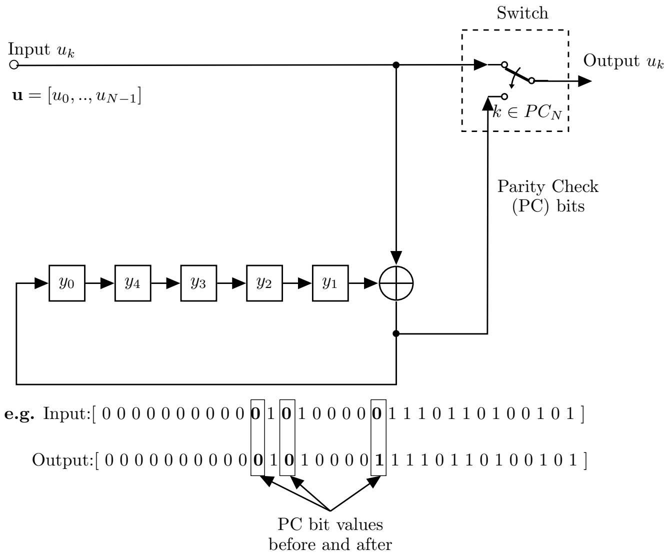

Switch
Input u_k
Output u_k
$\mathbf{u} = [u_0, .., u_{N-1}]$
$k \in PC_N$
Parity Check (PC) bits
y_0
y_4
y_3
y_2
y_1
e.g. Input:[0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0 0 0 1 1 1 0 1 1 0 1 0 0 1 0 1]
Output:[0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0 0 1 1 1 1 0 1 1 0 1 0 0 1 0 1]
PC bit values before and after

Fig. 19: Length-5 shift register block diagram of PC bit generation, as used for PUCCH with an information block length of $A \in [12,19]$. Note that each shift register memory element is initialized to store a zero-valued bit before the PC bits generation process is begun.

N bits $\mathbf{d} = [d_0, ..., d_{N-1}]$ into 32 sub-blocks each comprising a different set of $N/32$ consecutive bits. Following this, the order of the 32 sub-block is rearranged based on the sub-block interleaver pattern, which is given by Table 5.4.1.1-1 of [37]. Figures 21 and 22 illustrate the operation of the sub-block interleaver for $N = 32$ and $N = 64$, respectively.

In the decoder, sub-block deinterleaving is achieved by decomposing the sequence of N LLRs pertaining to $\mathbf{y} = [y_0, ..., y_{N-1}]$ into 32 sub-blocks and performing the inverse reordering operation.

H. Bit Selection

In the encoder, the second step in rate matching is bit selection, as specified in Subclause 5.4.1.2 of [37], which is employed to adjust the length of the sub-block interleaved bit sequence $\mathbf{y} = [y_0, ..., y_{N-1}]$ from N to E bits. Here, $E = \lfloor G/C \rfloor$, where G is the desired encoded block length and C is the number of coded block segments employed, which may be two in the case of long PUCCH blocks, as discussed in Section IV-A. This tunes the specific choice of a trade-off between the transmit time, frequency resource usage and error correction capability in order to match the prevailing channel conditions. Depending on how N compares to E, the encoded bits of $\mathbf{y} = [y_0, ..., y_{N-1}]$ are either repeated, shortened or punctured in order to generate the sequence of E bits $\mathbf{e} = [e_0, ..., e_{E-1}]$. More specifically, repetition is applied when $E \geq N$, and puncturing or shortening is applied when $E < N$ [81]. The bit selection mode adopted for all combination of information block length A and encoded block length G are illustrated in Figures 23 and 24 for PUCCH and PDCCH, respectively.

The bit selection process is commenced by writing the bit sequence $\mathbf{y} = [y_0, ..., y_{N-1}]$ into a circular buffer, as illustrated in Figure 25. Repetition and shortening are realized by reading successive bits starting from the beginning of the circular buffer and rotating around the buffer as many times as necessary to select a total E bits. In the case of repetition, more than one rotation will be required, while less than one rotation is required for shortening [81]. Puncturing is realized by selecting bits from the positions $(N - E)$ to $(N - 1)$ from the circular buffer [81]. In other words, puncturing and shortening results in some bits generated by the polar encoding core being excluded from the encoded bit sequence. In the case of puncturing, these excluded bits could have values of zero or one. However, in the case of shortening these excluded bits are guaranteed to have values of zero.

When repetition is used, the LLRs pertaining to the replicas of each bit in the sequence $\mathbf{y} = [y_0, ..., y_{N-1}]$ may be accumulated, in order to obtain a corresponding sequence of N LLRs. By contrast, puncturing refers to the non-transmission of coded bits, such that the values of these bits are unknown at the receiver. In this case, the corresponding LLRs can be set to zero [81] and prepended to the set of E received LLRs, in order to obtain the sequence of N LLRs pertaining to $\mathbf{y} = [y_0, ..., y_{N-1}]$. Meanwhile, shortening involves the non-transmission of coded bits that are guaranteed to have values of zero. In this case, the corresponding LLRs can be set to infinity and appended to the set of E received LLRs [81].

I. Channel Interleaving

The final step of polar code rate matching in NR is channel interleaving, as specified in Subclause 5.4.1.3 of [37]. However, channel interleaving is only used for PUCCH, while it is skipped for PDCCH and PBCH [82]. The motivation for applying channel interleaving is that it improves the polar code's error correction capability when employing higher-order modulation schemes for PUCCH transmission over fading channels [40], [82], [83].

In the encoder, the input of the channel interleaver is denoted by sequence of E rate matched bit $\mathbf{e} = [e_0, ..., e_{E-1}]$, while the output bit sequence is denoted by $\mathbf{f} = [f_0, ..., f_{E-1}]$. A triangle-based channel interleaver is adopted [84], [85], which is parametrised by the side length T of the triangle, which adopts the minimum value that satisfies $E \leq T(T+1)/2$ [83], [85]. Interleaving is performed by writing the bit sequence $\mathbf{e} = [e_0, ..., e_{E-1}]$ into the triangle row by row, followed by writing a sufficient number of NULL-valued bits in order to fill the triangle [85]. Following this, the output bit sequence $\mathbf{f} = [f_0, ..., f_{E-1}]$ is obtained by reading the bits from the triangular interleaver column by column and skipping the NULL-valued bits [85]. This operation is exemplified in Figure 26, where $E = 32$ and hence $T = 8$. The reverse operation is performed in the receiver, in order to deinterleave the sequence of E LLRs pertaining to $\mathbf{e} = [e_0, ..., e_{E-1}]$ [83].

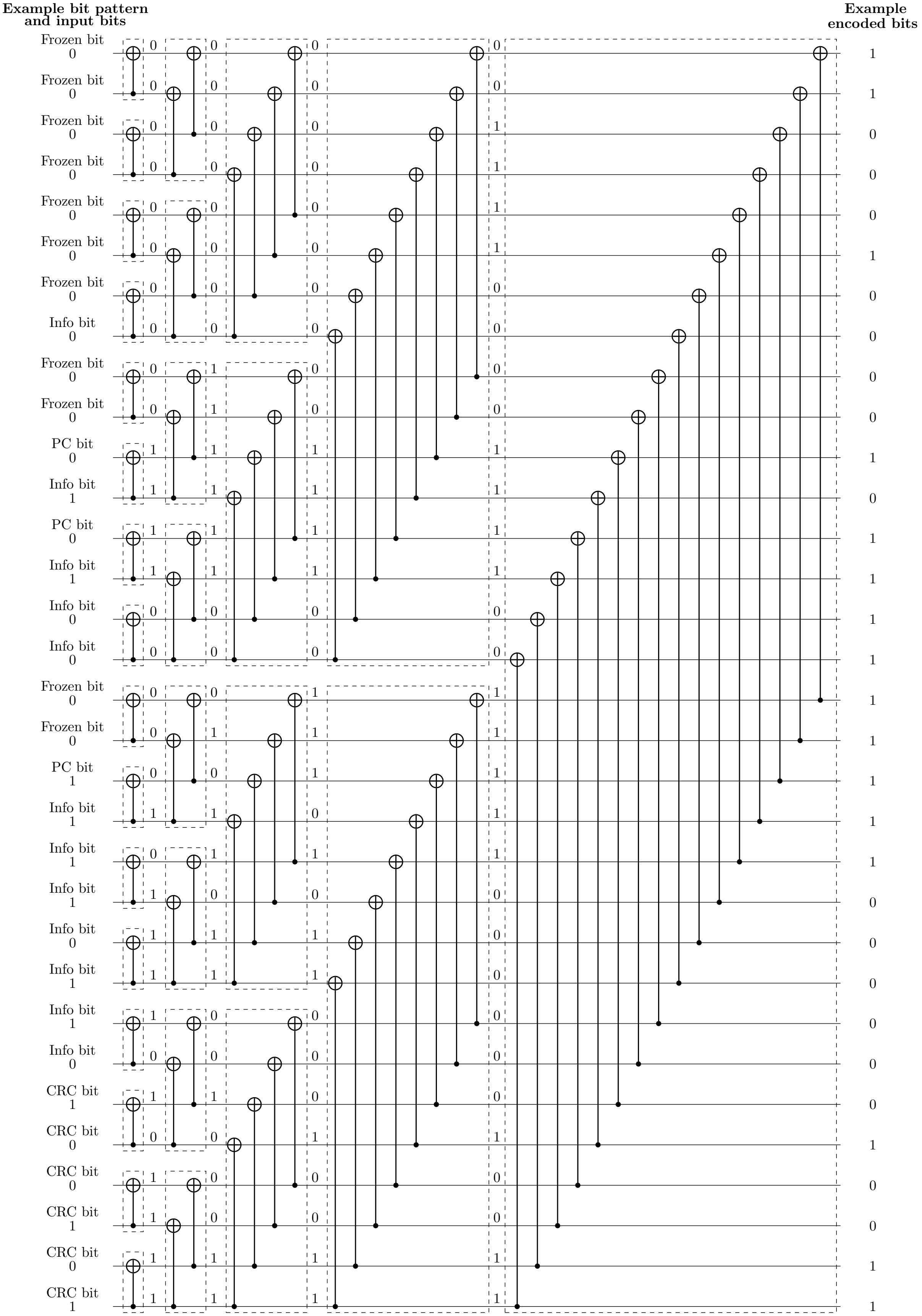


Fig. 20: Polar code graph representation for the example of converting $A' = 12$ information bits into $N = 32$ encoded bits, in the case of PUCCH/PUSCH encoding with $G = 34$.

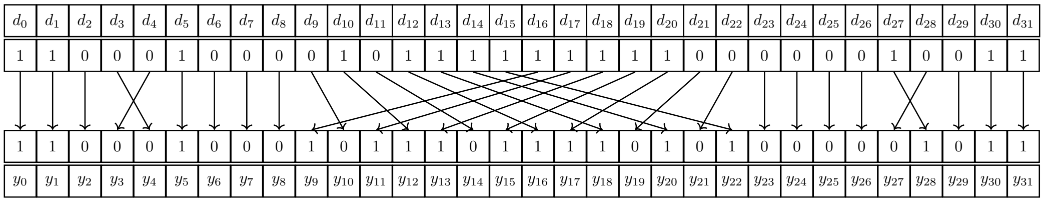

d_0 d_1 d_2 d_3 d_4 d_5 d_6 d_7 d_8 d_9 d_{10} d_{11} d_{12} d_{13} d_{14} d_{15} d_{16} d_{17} d_{18} d_{19} d_{20} d_{21} d_{22} d_{23} d_{24} d_{25} d_{26} d_{27} d_{28} d_{29} d_{30} d_{31}
1 1 0 0 0 1 0 0 0 0 1 0 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 1 0 0 1 1
1 1 0 0 0 1 0 0 0 1 0 1 1 1 0 1 1 1 1 0 1 0 1 0 0 0 0 0 1 0 1 1
y_0 y_1 y_2 y_3 y_4 y_5 y_6 y_7 y_8 y_9 y_{10} y_{11} y_{12} y_{13} y_{14} y_{15} y_{16} y_{17} y_{18} y_{19} y_{20} y_{21} y_{22} y_{23} y_{24} y_{25} y_{26} y_{27} y_{28} y_{29} y_{30} y_{31}

Fig. 21: Sub-block interleaving when $N = 32$.

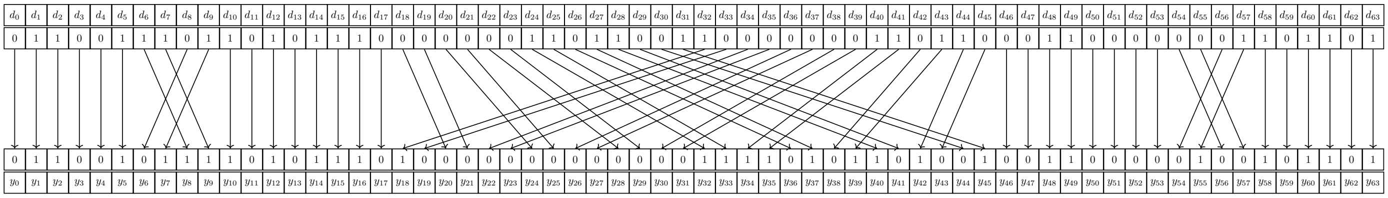

d_0 d_1 d_2 d_3 d_4 d_5 d_6 d_7 d_8 d_9 d_{10} d_{11} d_{12} d_{13} d_{14} d_{15} d_{16} d_{17} d_{18} d_{19} d_{20} d_{21} d_{22} d_{23} d_{24} d_{25} d_{26} d_{27} d_{28} d_{29} d_{30} d_{31} d_{32} d_{33} d_{34} d_{35} d_{36} d_{37} d_{38} d_{39} d_{40} d_{41} d_{42} d_{43} d_{44} d_{45} d_{46} d_{47} d_{48} d_{49} d_{50} d_{51} d_{52} d_{53} d_{54} d_{55} d_{56} d_{57} d_{58} d_{59} d_{60} d_{61} d_{62} d_{63}
0 1 1 0 0 1 1 1 0 1 1 0 1 0 1 1 1 0 0 0 0 0 0 0 1 1 0 1 1 0 0 1 1 0 0 0 0 0 0 0 1 1 0 1 1 0 0 0 1 1 0 0 0 0 0 0 0 1 1 0 1 1 0 1
0 1 1 0 0 1 0 1 1 1 1 0 1 0 1 1 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 0 1 0 1 1 0 1 0 0 1 0 0 1 1 0 0 0 0 0 1 0 0 1 0 1 1 0 1
y_0 y_1 y_2 y_3 y_4 y_5 y_6 y_7 y_8 y_9 y_{10} y_{11} y_{12} y_{13} y_{14} y_{15} y_{16} y_{17} y_{18} y_{19} y_{20} y_{21} y_{22} y_{23} y_{24} y_{25} y_{26} y_{27} y_{28} y_{29} y_{30} y_{31} y_{32} y_{33} y_{34} y_{35} y_{36} y_{37} y_{38} y_{39} y_{40} y_{41} y_{42} y_{43} y_{44} y_{45} y_{46} y_{47} y_{48} y_{49} y_{50} y_{51} y_{52} y_{53} y_{54} y_{55} y_{56} y_{57} y_{58} y_{59} y_{60} y_{61} y_{62} y_{63}

Fig. 22: Sub-block interleaving when $N = 64$.

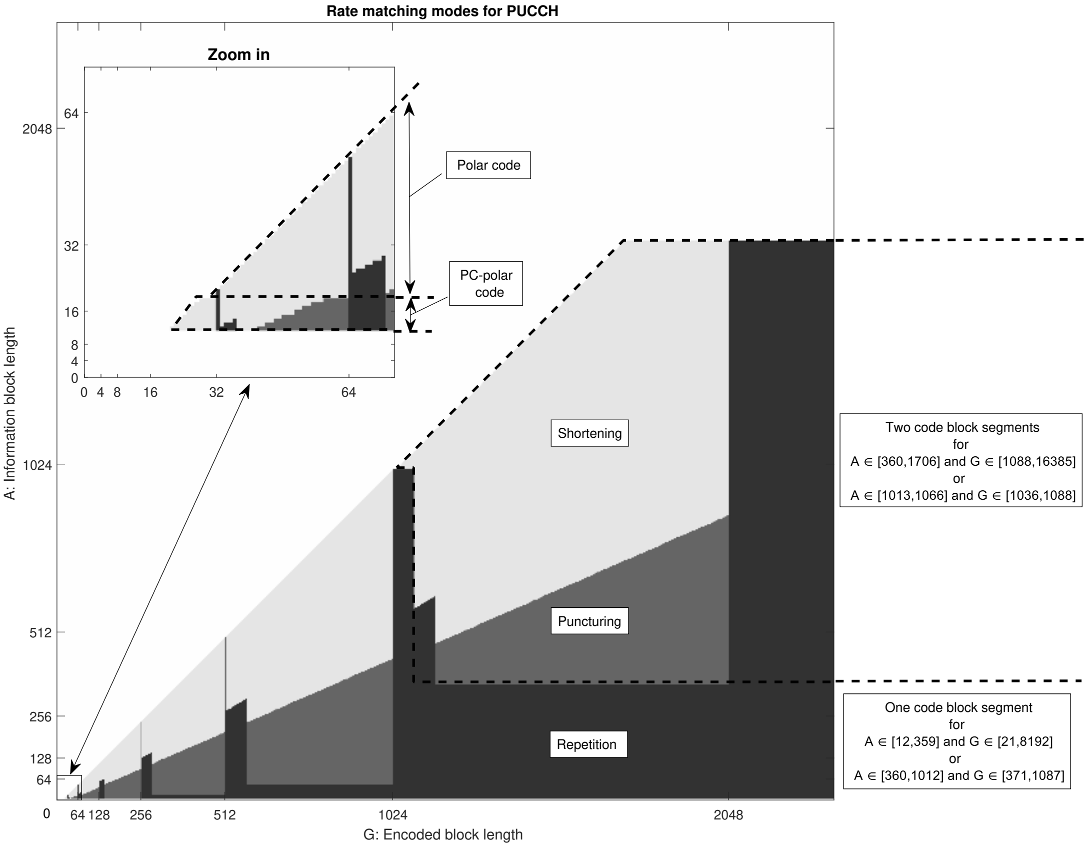


Fig. 23: Rate matching modes for each combination of information block length A and encoded block length G for PUCCH.

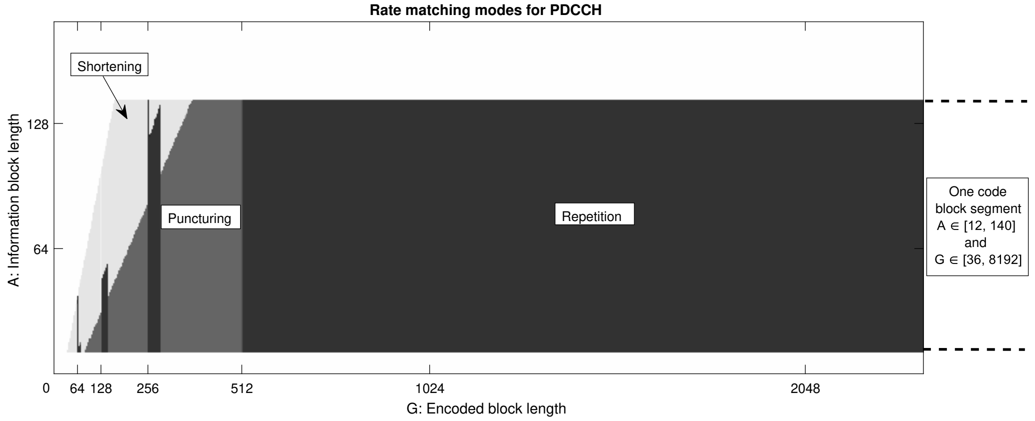


Fig. 24: Rate matching modes for each combination of information block length A and encoded block length G for PDCCH.

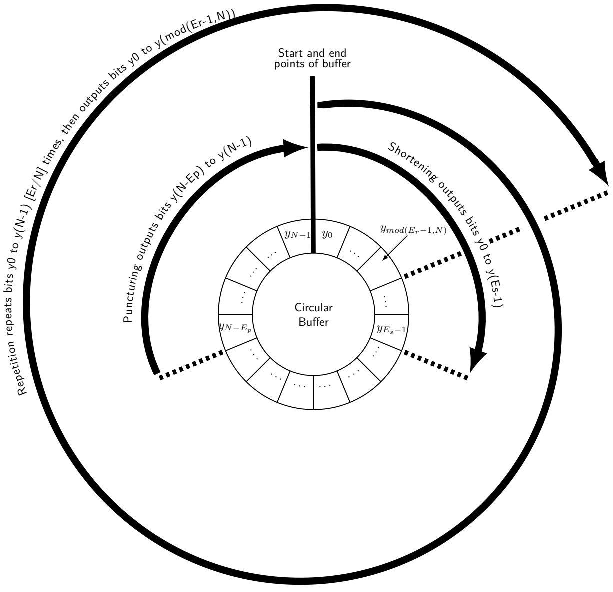

Repetition repeats bits y0 to y(N-1) [Er/N] times, then outputs bits y0 to y(mod(Er-1,N))
Start and end
points of buffer
Puncturing outputs bits y(N-Ep) to y(N-1)
Shortening outputs bits y0 to y(Es-1)
y_{N-1}
y_0
$y_{mod(E_r-1,N)}$
y_{N-E_p}
y_{E_s-1}
Circular
Buffer

Fig. 25: A representation of circular buffer and rate matching methods for repetition, puncturing and shortening.

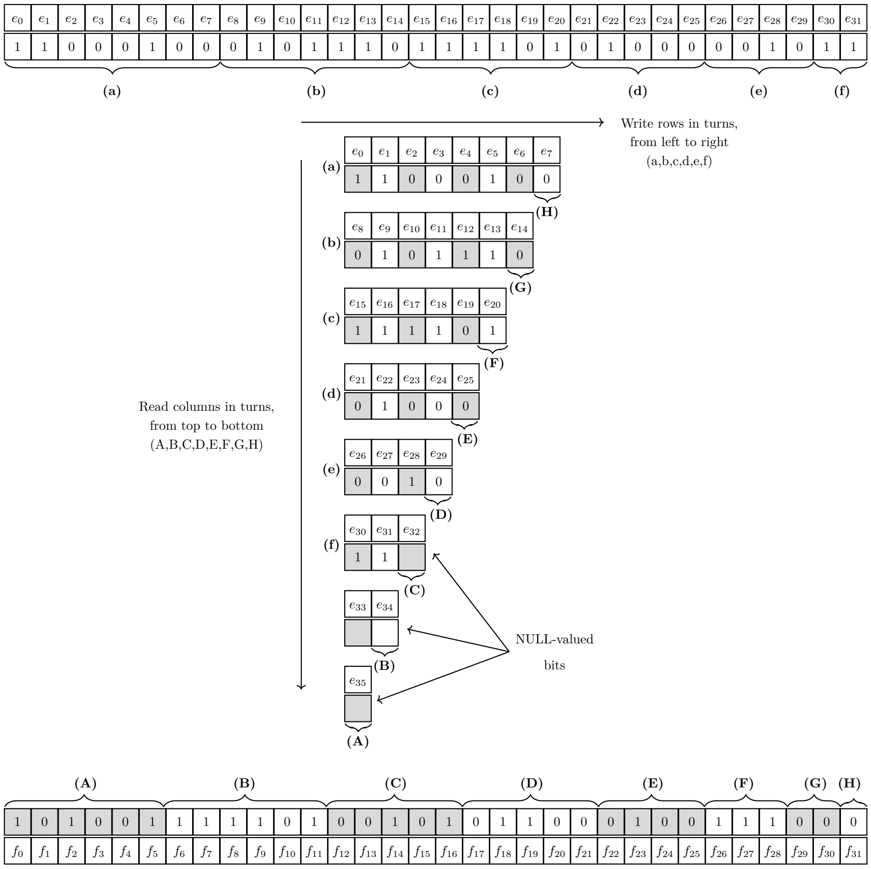
J. Code Block Concatenation

Code block concatenation is specified in Subclause 5.5 of [37] and it is only applied when $C = 2$ code block segments have been used in the PUCCH channels, namely when $A \geq 360$ and $G \geq 1088$ or $A \geq 1013$. In the encoder, the input of code block concatenation is a pair of bit sequences $\mathbf{f} = [f_0, \ldots\ldots, f_{E-1}]$, corresponding to the first and the second segments, each comprising E bits. The output bit sequence is $\mathbf{g} = [g_0, \ldots\ldots, g_{G-1}]$, which is obtained by appending the second segment onto the first. Furthermore, if G is odd, then a zero-valued padding bit is appended to the concatenated segments. In cases where only $C = 1$ code block segment is used, we have $E = G$, and the output bit sequence $\mathbf{g} = [g_0, \ldots\ldots, g_{G-1}]$ is set equal to the single input bit sequence $\mathbf{f} = [f_0, \ldots\ldots, f_{E-1}]$. In the decoder, the reverse operations are performed. More specifically, if $C = 2$ then the sequence of G input LLRs is decomposed into two equal length sequences of output LLRs, with the LLR corresponding to the padding bit being discarded if G is odd.

V. An Example of PUCCH/PUSCH Encoding

This section details the complete example of PUCCH/PUSCH polar encoding shown in Figure 27, which illustrates the step by step operation of the block diagram provided in Figure 4. For the sake of creating a simple example, the smallest supported information block length of $A = 12$ bits is chosen and the bit values of the information bit sequence $\mathbf{a} = [a_0, \ldots, a_{A-1}]$ are chosen randomly, as illustrated at the top of the Figure 27. In order to illustrate the repetition mode of the rate matching, an encoded block length of $G = 34$ bits is chosen.

The encoding process begins by determining how many code block segments to use, where $C = 1$ is selected in this case owing to the low values of A and G. Following this, the CRC is calculated and attached, where the CRC-6 is used since $A < 19$. Hence, the number of bits in the information block segment with concatenated CRC is given by $K = \lceil A/C \rceil + P = 12 + 6 = 18$, as shown in the second line of Figure 27. Next, the polar core block length is determined to be $N = 32$ according to Figure 14. Following this, $N - K - n_{PC} = 11$ frozen and $n_{PC} = 3$ PC bits are generated and inserted to the information and CRC bit sequence, according to the procedures shown in the flow charts of Figures 17 and 18. Here, all frozen bits adopt values of zero, while the values of the PC bits are generated according to Figure 19 as $\{0, 0, 1\}$.



e_0 e_1 e_2 e_3 e_4 e_5 e_6 e_7 e_8 e_9 e_{10} e_{11} e_{12} e_{13} e_{14} e_{15} e_{16} e_{17} e_{18} e_{19} e_{20} e_{21} e_{22} e_{23} e_{24} e_{25} e_{26} e_{27} e_{28} e_{29} e_{30} e_{31}
1 1 0 0 0 1 0 0 0 1 0 1 1 1 0 1 1 1 1 0 1 0 1 0 0 0 0 0 1 0 1 1
(a) (b) (c) (d) (e) (f)
Write rows in turns, from left to right (a,b,c,d,e,f)
Read columns in turns, from top to bottom (A,B,C,D,E,F,G,H)
(a) e_0 e_1 e_2 e_3 e_4 e_5 e_6 e_7 — 1 1 0 0 0 1 0 0 (H)
(b) e_8 e_9 e_{10} e_{11} e_{12} e_{13} e_{14} — 0 1 0 1 1 1 0 (G)
(c) e_{15} e_{16} e_{17} e_{18} e_{19} e_{20} — 1 1 1 1 0 1 (F)
(d) e_{21} e_{22} e_{23} e_{24} e_{25} — 0 1 0 0 0 (E)
(e) e_{26} e_{27} e_{28} e_{29} — 0 0 1 0 (D)
(f) e_{30} e_{31} e_{32} — 1 1 (C)
e_{33} e_{34} (B)
e_{35} (A)
NULL-valued bits
(A) (B) (C) (D) (E) (F) (G) (H)
1 0 1 0 0 1 1 1 1 1 0 1 0 0 1 0 1 0 1 1 0 0 0 1 0 0 1 1 1 0 0 0
f_0 f_1 f_2 f_3 f_4 f_5 f_6 f_7 f_8 f_9 f_{10} f_{11} f_{12} f_{13} f_{14} f_{15} f_{16} f_{17} f_{18} f_{19} f_{20} f_{21} f_{22} f_{23} f_{24} f_{25} f_{26} f_{27} f_{28} f_{29} f_{30} f_{31}

Fig. 26: Illustration of channel interleaving when $E = 32$ and hence $T = 8$.

The resulting sequence of $N = 32$ bits shown on the 3rd line of Figure 27 is input to the polar encoder core, in order to obtain the sequence of $N = 32$ encoded bits shown on the 4th line. Then sub-block interleaving is applied in the 5th line of Figure 27, according to the procedure shown in Figure 21. Next, the encoded bits are extracted from the output of the interleaver based on the bit selection mode, which is determined as repetition, since $E = G$ and $E > N$, as shown in Figures 23 and 25. As shown in the 6th line of Figure 27, bit selection results in the first two bits being repeated at the end of the the bit sequence, in order to give the desired encoded block length of $E = 32$. The final bit sequence shown on the 7th line of the Figure 27 is obtained using channel interleaving, according to the triangular interleaving process of Figure 26. Note that code block concatenation is not applied in this example, since we have only $C = 1$ code block segment.

VI. Error Correction and Error Detection Performance of NR Polar Codes

This section analyses the Block Error Rate (BLER) and FAR of the NR polar codes, in order to characterise their error correction and error detection performance, respectively. Here, BLER is defined as the fraction of transmitted blocks that are decoded erroneously [17], while FAR is defined as the fraction of erroneously decoded blocks that nonetheless have a passing CRC, in other words "missed detection of an error event" [2] [17], [86].

The BLER results are obtained using Gray coded Quadrature Phase Shift Keying (QPSK) for communication over an Additive White Gaussian Noise (AWGN) channel, as a function of the Signal to Noise Ratio (SNR) E_s/N_0. This facilitates direct comparisons with the results that were presented in the numerous 3GPP technical documents that were considered during the specification of the 3GPP NR polar code. Note that the only modulation schemes used by 3GPP NR for control information are BPSK and QPSK [37], [87]. In particular, BLER vs. E_s/N_0 is characterized for PBCH in Figure 28 for various SCL decoder with list size L. Furthermore, the SNR E_s/N_0 required for the PUCCH and PDCCH polar codes to achieve a BLER of 10^{-3} is characterized in Figures 29 and 30, as function of the information block length A, encoded block length G and the SCL decoder with list size L.

[2] In the standard this is referred to an False Alarm Rate (FAR).

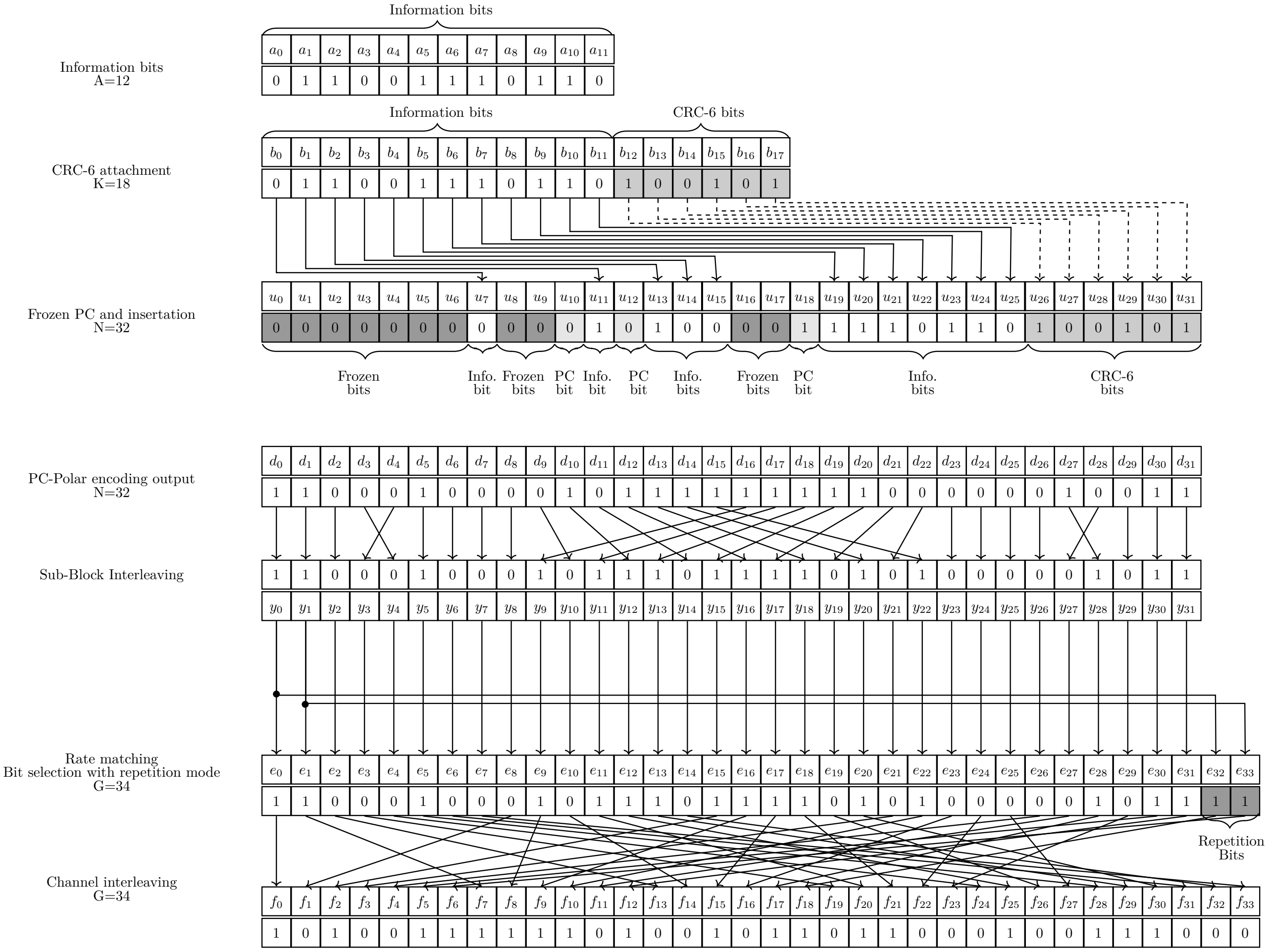

Information bits
Information bits
A=12
a0 a1 a2 a3 a4 a5 a6 a7 a8 a9 a10 a11
0 1 1 0 0 1 1 1 0 1 1 0
Information bits
CRC-6 bits
CRC-6 attachment
K=18
b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 b10 b11 b12 b13 b14 b15 b16 b17
0 1 1 0 0 1 1 1 0 1 1 0 1 0 0 1 0 1
Frozen PC and insertation
N=32
u0 u1 u2 u3 u4 u5 u6 u7 u8 u9 u10 u11 u12 u13 u14 u15 u16 u17 u18 u19 u20 u21 u22 u23 u24 u25 u26 u27 u28 u29 u30 u31
0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0 0 1 1 1 1 0 1 1 0 1 0 0 1 0 1
Frozen bits
Info. bit
Frozen bits
PC bit
Info. bit
PC bit
Info. bits
Frozen bits
PC bit
Info. bits
CRC-6 bits
PC-Polar encoding output
N=32
d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 d17 d18 d19 d20 d21 d22 d23 d24 d25 d26 d27 d28 d29 d30 d31
1 1 0 0 0 1 0 0 0 0 1 0 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 1 0 0 1 1
Sub-Block Interleaving
1 1 0 0 0 1 0 0 0 1 0 1 1 1 0 1 1 1 1 0 1 0 1 0 0 0 0 0 1 0 1 1
y0 y1 y2 y3 y4 y5 y6 y7 y8 y9 y10 y11 y12 y13 y14 y15 y16 y17 y18 y19 y20 y21 y22 y23 y24 y25 y26 y27 y28 y29 y30 y31
Rate matching
Bit selection with repetition mode
G=34
e0 e1 e2 e3 e4 e5 e6 e7 e8 e9 e10 e11 e12 e13 e14 e15 e16 e17 e18 e19 e20 e21 e22 e23 e24 e25 e26 e27 e28 e29 e30 e31 e32 e33
1 1 0 0 0 1 0 0 0 1 0 1 1 1 0 1 1 1 1 0 1 0 1 0 0 0 0 0 1 0 1 1 1 1
Repetition Bits
Channel interleaving
G=34
f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 f10 f11 f12 f13 f14 f15 f16 f17 f18 f19 f20 f21 f22 f23 f24 f25 f26 f27 f28 f29 f30 f31 f32 f33
1 0 1 0 0 1 1 1 1 1 1 0 1 0 0 1 0 1 1 0 1 1 0 0 0 1 0 0 1 1 1 0 0 0

Fig. 27: An example of each step in the PUCCH/PUSCH polar encoding process when $A = 12$ and $G = 34$.

Meanwhile, the presented FAR results were obtained by decoding random Gaussian distributed LLRs. Each simulation was continued until 1000 block errors or false alarms were observed. The capacity bounds provided in Figures 28 to 30 are provided by the $O(n^{-2})$ metaconverse PPV upper bound [88]. This characterizes the theoretical limit on the achievable BLER for QPSK modulation over an AWGN channel as a function of SNR E_s/N_0 and the block lengths A and G.

As is typical of near-capacity channel codes, Figure 28 to 30 reveal that the BLER performance of the NR polar codes improves upon increasing the information block lengths A, while keeping the coding rate A/G constant. Likewise, the BLER performance improves upon increasing the encoded block length G, while keeping A constant, although diminishing returns are observed when relying on repetition. It is notable that the BLER performance of the NR polar codes more closely approaches the capacity bound, when A and G are small. Similarly, Table VII shows that the FAR performance improves upon increasing the CRC length, as it may be expected.

Figure 28 characterizes the impact of SCL decoding list size L on BLER performance for the PBCH polar code, which uses fixed block lengths of $A = 32$ and $E = 864$. As may be expected, increasing the list size results in improved BLER performance that more closely approaches the capacity bound. However, a diminishing return can be observed, particularly between $L = 8$ and $L = 32$. Furthermore, Figure 28 compares the PBCH polar code with the NR LDPC code. In order to enable a fair comparison with this LDPC code, it is supported by a 24-bit CRC and adopts base graph-2 (BG2), which gives superior BLER performance to base graph-1 (BG1) for short block lengths and low coding rates, as described in Sections 6.2.2 and 7.2.2 of [37].

Figure 28 shows that the NR polar codes achieve superior BLER performance to the NR LDPC code at short block lengths. In particular, even the SCL polar decoder relying on a low list size of $L = 2$ is capable of outperforming the

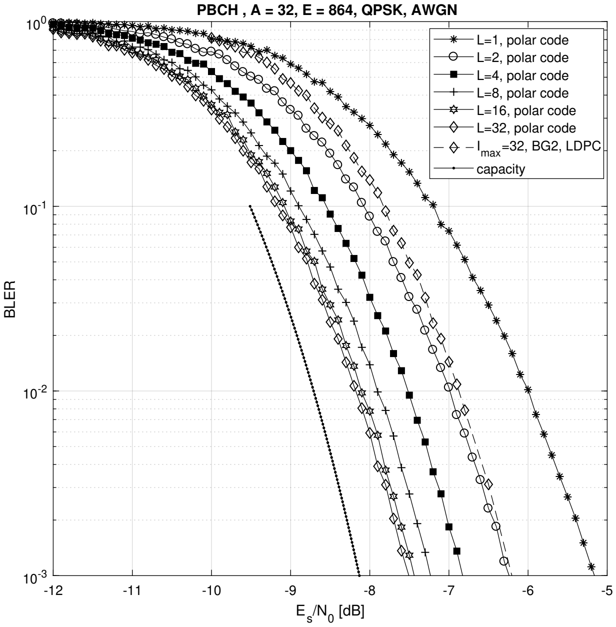


Fig. 28: Plot of BLER versus channel SNR E_s/N_0 for the PBCH polar code and the LDPC code of 3GPP NR, when using QPSK for communication over an AWGN channel.

LDPC decoder, when performing a high maximum number of decoding iterations of $I_{max} = 32$. Furthermore, Table VII compares the computational complexity of the NR polar and LDPC codes used in Figure 28, revealing that the complexity of the LDPC is much higher than that of the polar code, even when employing $L = 32$ SCL decoding. Here, the computational complexity is quantified in terms of the number of Add Compare Select (ACS) operations[3] [44] that are performed by the decoding algorithms.

As shown in Figures 29 and 30, the PUCCH and PDCCH polar decoders can satisfy the NR error correction requirements of BLER of 10^{-3} at lower SNRs E_s/N_0, when the coding rate is lower, as may be expected. It may be observed that the BLER performance closely approaches the capacity bound at short block lengths, further illustrating the advantage of polar codes these block lengths. The larger gap from the capacity bound at long encoded block lengths G may be explained by the dependence on the repetition, imposed at these block lengths owing to the limit of 1024 bits imposed upon the maximum polar core block length N.

Table VIII compares the FAR observed for the PUCCH and PDCCH polar codes during simulation with a theoretical value, for various combinations of information block length

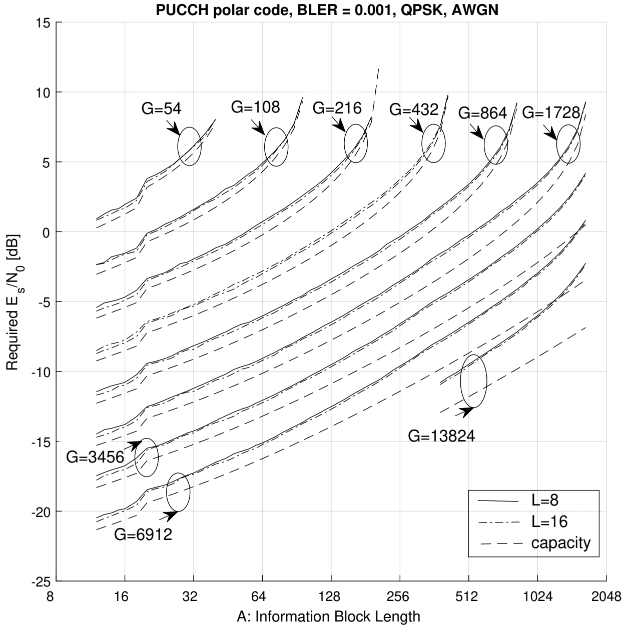


Fig. 29: Plot of SNR E_s/N_0 required to achieve a BLER of 10^{-3} versus information block length A for the PUCCH polar code of 3GPP NR, when using QPSK for communication over an AWGN channel.

TABLE VII: Computational complexity for the PBCH polar and the LDPC codes of 3GPP NR for $A = 32$ and $G = 864$.

		L **or I**$_{max}$	**Computational complexity with binary min and sum computations**
PBCH	**NR Polar (CRC-24) SCL decoding**	$L = 1$	5176
		$L = 2$	7831
		$L = 4$	13071
		$L = 8$	23531
		$L = 16$	44427
		$L = 32$	65211
	NR LDPC (CRC-24) flooding	I$_{max}$ = 32	209472

A, encoded block length G and SCL decoder with list size L. Here, the theoretical FAR performance is determined as $2^{-(P-3)}$ [86] where $P-3$ of the P are used for error detection, with the remaining 3 CRC bits used to improve error correction, as discussed in Section IV-E. It may be observed that there is a close match between the theoretical and measured FAR results of Table VIII.

[3] Computational complexity is quantified by the number of f, g and ϕ operations, as mentioned in [44].

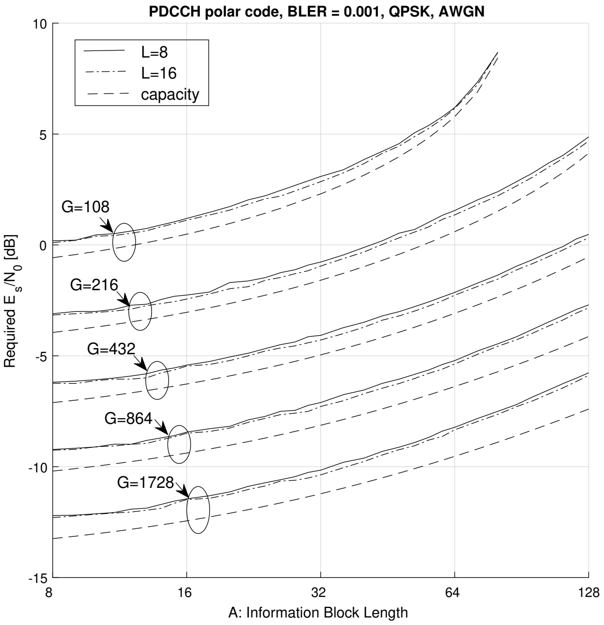


Fig. 30: Plot of SNR E_s/N_0 required to achieve a BLER of 10^{-3} versus information block length A for the PDCCH polar code of 3GPP NR, when using QPSK for communication over an AWGN channel.

TABLE VIII: Comparison of simulated FAR with theoretical FAR for the PUCCH and PDCCH polar codes, when decoding Gaussian distributed random LLRs.

					FAR	
		A	G	L	**Simulation**	**Theoretical**
PDCCH	**PC-polar (CRC-6)**	16	108	8	0.1227	$\mathbf{2^{-5} = 0.1250}$
		16	108	16	0.1439	
		16	216	8	0.1156	
		16	216	16	0.1220	
	CA-polar (CRC-11)	32	108	8	0.0035	$\mathbf{2^{-8} = 0.0039}$
		32	108	16	0.0043	
		32	216	8	0.0036	
		32	216	16	0.0038	
PDCCH	**DCA-polar (CRC-24)**	16	108	8	$4.75x10^{-7}$	$\mathbf{2^{-21} = 4.77x10^{-7}}$
		16	108	16	$4.63x10^{-7}$	
		16	216	8	$4.73x10^{-7}$	
		16	216	16	$4.84x10^{-7}$	
		32	108	8	$4.61x10^{-7}$	
		32	108	16	$4.71x10^{-7}$	
		32	216	8	$4.84x10^{-7}$	
		32	216	16	$4.73x10^{-7}$	

VII. Lessons Learned and Opportunities for Future Improvements

This section provides a summary of lessons learned and possible improvements for future iterations of the NR polar code design, as well as for future applications of polar codes beyond 5G.

A. Implementation Concerns and Potential Solutions for LDPC, Turbo and Polar Codes

During the RAN1-86bis meeting of 3GPP, the associated implementation concerns and potential solutions for channel coding schemes proposed for NR were captured. Some of the concerns about LDPC codes were related to their escalating complexity upon increasing their flexibility and that, depending on the parity check matrix design, some of the implementation parallelism may not be exploited for all code block lengths, additionally, their parallel implementation may reduce their energy and area efficiency. As a counter to these concerns, other companies suggested that, limited flexibility can provide the most attractive area and energy efficiency and that LDPC codes remain advantageous compared to other codes, even when supporting full flexibility. Secondly, some variants of min-sum based iterative decoders are implementable, and allow a trade-off between complexity and performance. Finally, it was observed that the list-32 and ordered stochastic decoding algorithms may be implementable for codeword sizes up to 1000 bits, where they can achieve near-maximum-likelihood (ML) performance. By contrast, some of the concerns about turbo codes were related to their reduced area and energy efficiency at lower block lengths, as well as the challenge of achieving high area and energy efficiency when targeting the higher-throughput and lower-latency requirements of NR. Although advanced turbo decoders were conceived for meeting the flexibility requirement of NR and fertilizing trade-off between complexity and performance, turbo codes were not adopted for 5G NR owing to the concerns discussed above. Finally, for polar codes, a concern was raised that the implementation complexity of SCL decoding increases with the list size L, especially for larger code block sizes N. The only unanimous agreement that could be reached at this time was that a list size of 1 is implementable, although some companies felt that larger list sizes were also practical, such as $L = 8$ and $L = 4$ for code block sizes N up to 4096 and 8192, respectively. Future work may be able to address the concerns identified during this 3GPP meeting.

B. Observations for the NR Polar Codes

As a complement to the observations of RAN1-86bis, we would like to offer the following observations about the NR polar codes. As described in Section IV-A, code block segmentation is used in PUCCH for decomposing long information blocks into a pair of shorter code blocks. This allows the maximum supported polar code core length to be reduced from $N = 2048$ bits to $N = 1024$ bits, while still facilitating support for the largest required information length of $A = 1706$ bits, even without relying on excessive repetition, which degrades the error correction performance [56]–[63]. Since the complexity of polar encoding and decoding scales with $N\log(N)$, the processing of two code blocks of length $N = 1024$ is about 9% lower than processing a single code block of length $N = 2048$. Furthermore, code block segmentation enables the code blocks to be processed in parallel, granting a reduced processing latency. Alternatively, the two code blocks can be processed in series, using a single processor having a 50% reduced memory requirement, which can lead to a similar improvement in ASIC area. However, these advantages are gained at the cost of losing the improved BLER that is associated with the longer core blocks length of $N = 2048$. Future applications of polar codes may wish to reconsider the use of code block segmentation, if BLER is prioritised over complexity.

Secondly, CRC bits are appended to each information block segment to perform error detection, as well as to aid its error correction. More specifically, 3 CRC bits are intended for assisting the error correction, which can be exploited by $L = 2^3 = 8$ SCL polar decoding. During the 3GPP standardisation process, the number of CRC bits intended to aid error correction was limited to 3, because exploiting any more would require higher list sizes and there was a concern that this would lead to prohibitive complexity. However, as the computational devices and decoder implementations become more sophisticated, future applications of polar codes may wish to use longer CRCs, with the intention of using more CRC bits to aid error correction with higher list sizes.

It can be argued that there are four interleaving operations used in the NR polar code design, namely the CRC interleaver of Section IV-D (which is only used in PBCH and PDCCH), the PC and frozen bit insertion operation of Section IV-E (which can be considered to be a form of interleaving), the sub-block interleaver of Section IV-G and the channel interleaver of Section IV-I (which is only used in PUCCH). However, these interleaving operations have complex implementations and required the introduction of buffers, i.e. increased hardware resources and impose extended latency. Future applications of polar codes may with to merge or remove some of these interleavers.

Finally, the application of PC-polar coding is somewhat marginalised in 3GPP NR, since it is only used for PUCCH blocks comprising $A = 12$ to $A = 19$ information bits. While PC-polar codes may have reduced polar decoding complexity [89], [90] or enable early termination [89], [90], while additionally offering superior error correction performance over conventional polar codes, they are more complex to implement and optimise. Hence, future applications of polar codes may wish to eliminate the use of PC-polar codes or adopt them more widely.

VIII. Conclusions

This paper has provided a tutorial and survey of the operation and performance of the polar codes used in the PUCCH, PBCH and PDCCH channels of 3GPP NR.

We commenced with an overview of the history of mobile communication and the motivation behind the introduction of NR. We discussed why polar codes have been selected for the NR control channels, with a summary of the 3GPP meeting outcomes that led to their specification. After that, the PUCCH, PBCH and PDCCH channels of 3GPP NR were briefly reviewed, complemented by a discussion of the encoding and decoding block diagrams of polar coding in these control channels. Then, the operation and motivation of each component in these block diagrams was detailed, with the help of schematics, flow charts and examples. An end-to-end example of PUCCH polar encoding was provided to illustrate the step by step operation of these components. Finally, the error correction and error detection performance of the NR polar codes was comprehensively characterized using our BLER and FAR analysis.

As we mentioned before, since the standardization of polar codes in 3GPP NR, the interest in polar codes has increased significantly, as shown in Figure 2. However, the current limitation of polar codes is that they do not readily lend themselves to soft-decisions, which limits their benefits in the content of both powerful turbo equalizers and multiuser-detectors (MUDs). These problems are likely to inspire coding experts and the broader 6G-research community. Another fertile area of future research is related to the conceptions of polar codes for enhancing quantum computers and other quantum systems [91].

Based on the lessons learned, as detailed in Section VII, we would like to inspire the community to circumvent the above-mentioned limitations by conceiving powerful soft-decision-aided polar coded turbo-transceivers, as well as wireless multimedia systems. Numerous open questions have to be solved, such as the design of flawless polar coded audio and video systems, Automatic Repeat Request (ARQ) solutions and quantum systems, just to name a few. Given the complex design trade-off's and conflicting metrics such as the code-rate, code-length, coding-delay, complexity etc., the radical idea of determining the optimal Pareto-front of polar codes arises. To elaborate a little further it is a challenging but extremely promising future research idea is to catalogue the set of 'best' polar codes, depending on the specific application considered. This set of codes would contain all the candidates, which exhibit for example a performance closest to capacity at a given code-rate, or codeword-length or complexity. Viewing this radical new design approach from a difficult perspective, it would not be possible to approach capacity more closely without increasing the complexity or the codeword length, or without reducing the code-rate, for example. What an inspirational challenge for coding enthusiast!

Nomenclature

AI	Agenda Item
AWGN	Additive White Gaussian Noise
BLER	Block Error Ratio
CA-polar code	CRC aided polar code
CC-HARQ	Chase Combining Hybrid Automatic Repeat reQuest
CDMA	Code Division Multiple Access
CRC	Cyclic Redundancy Check
CSI	Channel State Information
DCA-polar	Distributed CRC aided polar code
DCI	Downlink Control Information
eMBB	enhanced Mobile Broad Band
FAR	False Alarm Rate
gNodeB	New Radio Basestation
HARQ- ACK	Hybrid Automatic Repeat reQuest Acknowledgement
IR-HARQ	Incremental Redundancy Hybrid Automatic Repeat reQuest
IoT	Internet of Things
LDPC	Low Density Parity Check
LLR	Logarithmic-Likelihood Ratio
LTE	Long Term Evolution
mMTC	massive Machine Type Communication
NR	New Radio
PBCH	Physical Broadcast CHannel
PC	Parity Check
PC-polar code	Parity Check assisted polar code
PDCCH	Physical Downlink Control CHannel
PUCCH	Physical Uplink Control CHannel
PUSCH	Physical Uplink Shared CHannel
QPSK	Quadrature Phase Shift Keying
RM	Reed-Muller
RNTI	Radio Network Temporary Identifier
RRC	Radio Resource Control
SC	Successive Cancellation
SCL	Successive Cancellation List
SNR	Signal-to-Noise Ratio
TCP/IP	Transmission Control Protocol/Internet Protocol
UCI	Uplink Control Information
UE	User Equipment
UMTS	Universal Mobile Telecommunication System
URLLC	Ultra-Reliable Low-Latency Communication
V2V	Vehicle to Vehicle
XOR	eXclusive-OR
3GPP	Third Generation Partnership Project
1G	1st Generation
2G	2nd Generation
3G	3rd Generation
4G	4th Generation
5G	5th Generation
A	Number of bits in an information block
C	Number of code block segments ($C = 1$ or $C = 2$)
A'	Number of bits in an information block segment ($A' = \lceil A/C \rceil$)
I_{max}	Maximum number of iterations performed by decoder
P	Number of CRC bits appended to an information block segment
K	Number of bits in an information block segment with concatenated CRC ($K = A' + P$)
L	SCL decoder list size
G	Number of bits in an encoded block
E	Number of bits in an encoded block segment $E = \lfloor G/C \rfloor$
N	Number of bits encoded by a polar encoder core (must be a power of 2)
T	The side length of the triangle-based channel interleaver
a_k	k^{th} bit in an information block, where $k \in \{0, 1, ..., A - 1\}$
a'_k	k^{th} bit in an information block segment, where $k \in \{0, 1, ..., A' - 1\}$
p_k	k^{th} CRC bit appended to an information block segment, where $k \in \{0, 1, ..., P - 1\}$
b_k	k^{th} bit in an information block segment with appended CRC, where $k \in \{0, 1, ..., K - 1\}$
c_k	k^{th} bit in an information block segment after CRC interleaving, where $k \in \{0, 1, ..., K - 1\}$
$c'{}_k$	k^{th} bit in an information block segment after PDCCH CRC scrambling where, where $k \in \{0, 1, ..., K\}$
u_k	k^{th} bit input into a polar encoder core, where $k \in \{0, 1, ..., N - 1\}$
d_k	k^{th} bit output from a polar encoder core, where $k \in \{0, 1, ..., N - 1\}$
y_k	k^{th} bit in an encoded block segment after sub-block interleaving, where $k \in \{0, 1, ..., N - 1\}$
e_k	k^{th} bit in an encoded block segment before channel interleaving, where $k \in \{0, 1, ..., E - 1\}$
f_k	k^{th} bit in an encoded block segment after rate matching, where $k \in \{0, 1, ..., E - 1\}$
g_k	k^{th} encoded bit in an encoded block after code block concatenation, where $k \in \{0, 1, ..., G - 1\}$

References

[1] R. Steele and L. Hanzo, *Mobile Radio Communications: Second and Third Generation Cellular and WATM Systems*, 2nd ed. West Sussex, England: Wiley-IEEE Press, 2002, ch. 8, pp. 661–771.

[2] L. Hanzo, J. Blogh, and S. Ni, *3G, HSPA and FDD versus TDD Networking: Smart Antennas and Adaptive Modulation.* Wiley, 2008.

[3] L. Hanzo, H. Haas, S. Imre, D. O'Brien, M. Rupp, and L. Gyongyosi, "Wireless Myths, Realities and Futures: From 3G/4G to Optical and Quantum Wireless," *Proceedings of the IEEE*, vol. 100, no. Special Centennial Issue, pp. 1853–1888, 5 2012.

[4] MATLAB, “5G Development with MATLAB,” ebook, 2017. [Online]. Available: https://uk.mathworks.com/

[5] “3GPP TR 38.913, 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; study on Scenarios and Requirements for Next Generation Access Technologies (Release 15) V15.0.0,” Valbonne - FRANCE,” Technical Report (TR), May 2018.

[6] M. Series, “IMT Vision–Framework and overall objectives of the future development of IMT for 2020 and beyond,” 2015.

[7] “3GPP TSG RAN Meeting 80, RP-181378, WI Summary of New Radio Access Technology,” NTT DOCOMO, INC, La Jolla, USA, Technical Specification (TS), June 2018.

[8] “3GPP TSG RAN WG1 Meeting 87, R1-1611326, Channel Coding Candidates for Control Channel,” Ericsson, Reno, USA, Technical Specification (TS), November 14 - 18 2016.

[9] C. Berrou, A. Glavieux, and P. Thitimajshima, “Near Shannon Limit Error-Correcting Coding and Decoding: Turbo-codes. 1,” in *Proceedings of ICC’93-IEEE International Conference on Communications*, vol. 2. IEEE, 1993, pp. 1064–1070.

[10] D. J. MacKay and R. M. Neal, “Near Shannon Limit Performance of Low Density Parity Check Codes,” *Electronics letters*, vol. 32, no. 18, pp. 1645–1646, 1996.

[11] E. Arikan, “Channel Polarization: A Method for Constructing Capacity-Achieving Codes,” in *2008 IEEE International Symposium on Information Theory*. IEEE, 2008, pp. 1173–1177.

[12] “3GPP TSG RAN WG1 Meeting 87, R1-1613061, Comparison of Coding Candidates for DL Control Channels and Extended Applications,” MediaTek Inc., Reno, USA, Technical Specification (TS), 14th–18th November 2016.

[13] “3GPP TSG-RAN WG1 84bis, R1-162896, Channel Coding Requirements for Next Generation Radio Access Technologies,” Nokia, Alcatel-Lucent Shanghai Bell, Busan, Korea, Technical Specification (TS), 11th–115th April 2016.

[14] “3GPP TSG-RAN WG1 84bis, R1-162897, Performance and Complexity of Turbo, LDPC and Polar Codes ,” Nokia, Alcatel-Lucent Shanghai Bell, Busan, Korea, Technical Specification (TS), 11th–115th April 2016.

[15] “3GPP TSG RAN WG1 Meeting 87, R1-1611081, Final Report of 3GPP TSG RAN WG1 86bis v1.0.0,” Lisbon, Portugal,” Technical Specification (TS), 10th—14th October 2016.

[16] “3GPP RAN WG1 Meeting 86bis, R1-1610690, Way Forward on Observations for eMBB Data Channel Coding,” Samsung, Qualcomm Incorporated, Nokia, ASB, KT Corporation, Intel Corporation, Lisbon, Portugal, Technical Specification (TS), 10th–15th October 2016.

[17] “3GPP TSG RAN WG1 87, R1-1611109, Evaluation on Channel coding Candidates for eMBB Control Channel,” ZTE Microelectronics„ Reno, USA, Technical Specification (TS), 14th–18th November 2016.

[18] “3GPP TSG RAN WG1 87, R1-1613078, Evaluation on Channel Coding Candidates for eMBB Control Channel,” ZTE Microelectronics„ Reno, USA, Technical Specification (TS), 14th–18th November 2016.

[19] “Final Report of 3GPP TSG RAN WG1 84bis v1.0.0,” Busan, Korea,” Technical Specification (TS), 11th–15th April 2016.

[20] “3GPP TSG RAN WG1 84bis meeting, R1-163662, Way Forward on Channel Coding Scheme for 5G New Radio,” Samsung, Nokia, Qualcomm, ZTE, Intel, Huawei, Busan, Korea, Technical Specification (TS), 11th–15th April 2016.

[21] “3GPP TSG RAN WG1 85 meeting, R1-165637, Way Forward on Channel Coding Scheme for 5G New Radio,” Samsung, Nokia, Qualcomm, ZTE, Intel, Huawei, Nanjing, China, Technical Specification (TS), 23rd–27th May 2016.

[22] “Final Report of 3GPP TSG RAN WG1 85 v1.0.0,” Nanjing, China,” Technical Specification (TS), 23rd–27th May 2016.

[23] “3GPP TSG RAN WG1 85 meeting, R1-165643, Way Forward on Channel Coding Scheme for 5G New Radio,” ZTE Microelectronics, Nokia, Qualcomm, InterDigital, MediaTek, Intel, Samsung, Nanjing, China, Technical Specification (TS), 23rd–27th May 2016.

[24] “Final Report of 3GPP TSG RAN WG1 86 v1.0.0,” Gothenburg, Sweden,” Technical Specification (TS), 22nd–26th August 2016.

[25] “Final Report of 3GPP TSG RAN WG1 86bis v1.0.0,” Lisbon, Portugal,” Technical Specification (TS), 10th–14th October 2016.

[26] “Final Report of 3GPP TSG RAN WG1 88 v1.0.0 ,” Athens, Greece,” Technical Specification (TS), 13th–17th February 2017.

[27] “3GPP TSG RAN WG1 Meeting 88bis, R1-1706408, WF on Polar Code Design,” Spokane, USA,” Technical Specification (TS), 3rd–7th April 2017.

[28] “Final Report of 3GPP TSG RAN WG1 88bis v1.0.0,” Spokane, USA,” Technical Specification (TS), 3rd–7th April 2017.

[29] “Final Report of 3GPP TSG RAN WG1 89 v1.0.0,” Hangzhou, China,” Technical Specification (TS), 15th–19th May 2017.

[30] “Final Report of 3GPP TSG RAN WG1 90 v1.0.0,” Prague, Czech Republic,” Technical Specification (TS), 21st–25th August 2017.

[31] “3GPP TSG RAN WG1 Meeting 90, R1-1712174, Summary of email discussion [NRAH2-11] Polar code sequence,” Huawei, Prague, Czech Republic, Technical Specification (TS), 21st–25th August 2017.

[32] “3GPP TSG RAN WG1 Meeting 90, R1- 1714983, Proposed Evaluation Assumptions for Polar Channel Interleaver for DL,” Intel Corporation, Fujitsu, Prague, Czech Republic, Technical Specification (TS), 21st–25th August 2017.

[33] “3GPP TSG-RAN RAN1 90, R1-1713474, Design and Evaluation of Interleaver for Polar Codes,” Qualcomm Incorporated, Prague, Czech Republic, Technical Specification (TS), 21st–25th August 2017.

[34] “Final Report of 3GPP TSG RAN WG1 90bis v1.0.0,” Prague, Czech Republic,” Technical Specification (TS), 9th–13th October 2017.

[35] “Final Report of 3GPP TSG RAN WG1 91 v1.0.0,” Reno, USA,” Technical Specification (TS), 27th November–1st December 2017.

[36] “Final Report of 3GPP TSG RAN WG1 92bis v1.0.0,” Sanya, China,” Technical Specification (TS), 16th–20th April 2018.

[37] “3GPP TS 38.212, 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; NR; Multiplexing and Channel Coding (Release 15) V15.6.0 Technical Specification (TS),” June 2019.

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

[39] F. Hamidi-Sepehr, A. Nimbalker, and G. Ermolaev, “Analysis of 5G LDPC Codes Rate-Matching Design,” in *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*. IEEE, 2018, pp. 1–5.

[40] 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, 2018.

[41] O. İşcan, R. Böhnke, and W. Xu, “Probabilistic Shaping Using 5G New Radio Polar Codes,” *IEEE Access*, vol. 7, pp. 22 579–22 587, 2019.

[42] P. Chen, B. Bai, Z. Ren, J. Wang, and S. Sun, “Hash-polar Codes with Application to 5G,” *IEEE Access*, vol. 7, pp. 12 441–12 455, 2019.

[43] 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 and Systems II: Express Briefs*, vol. 66, no. 6, pp. 943–947, 2019.

[44] L. Xiang, Z. B. K. 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. 28 559–28 573, 2019.

[45] MATLAB. (2018) Physical channels. [Online]. Available: https://uk.mathworks.com/help/lte/physicalchannels.html

[46] “3G TR 21.905 V0.1.0 ,3rd Generation Partnership Project, Technical Specification Group Services and System Aspects Vocabulary for 3GPP Specifications, V0.1.0Technical Report (TR),” October 1999.

[47] L. Hanzo, M. El-Hajjar, and O. Alamri, “Near-capacity Wireless Transceivers and Cooperative Communications in the MIMO era:

Evolution of Standards, Waveform Design, and Future Perspectives," *Proceedings of the IEEE*, vol. 99, no. 8, pp. 1343–1385, 2011.

[48] "3GPP TS 38.213, Evolved Universal Terrestrial Radio Access (EUTRA); Physical Layer Procedures for Control (Release 15)Technical Specification (TS)," December 2017.

[49] "3GPP TS 36.212, LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Multiplexing and Channel Coding (version 9.2.0 (Release 9)Technical Specification (TS)," June 2010.

[50] E. Arikan, "A Performance Comparison of Polar Codes and Reed-Muller Codes," *IEEE Communications Letters*, vol. 12, no. 6, pp. 447–449, 2008.

[51] "3GPP TSG-RAN WG1 Meeting 91, R1-1711655, Chairman's Notes of AI 7.4 Channel Coding," Ad-Hoc Chair (Ericsson), Reno, USA, Technical Specification (TS), 27th November–1stDecember 2017.

[52] "3GPP TSG RAN WG1 Meeting 90bis, R1- 1718959, WF on Polar Code Construction for NR PBCH ," Ericsson, Qualcomm, Prague, Czech Republic, Technical Specification (TS), 9th-–13th, October 2017.

[53] "3GPP TSG-RAN WG1 88bis Meeting, R1-1705861, Design Details of Distributed CRC," Nokia, Alcatel-Lucent Shanghai Bell, Spokane, USA, Technical Specification (TS), 3rd–7th April 2017.

[54] "3GPP TSG RAN WG1 Meeting 89, R1-1708832, Early Termination Benefits of CRC Distribution," Nokia, Alcatel-Lucent Shanghai Bell, Hangzhou, P.R. China, Technical Specification (TS), 15th–19th May 2017.

[55] "3GPP TSG RAN WG1 Meeting 86bis, R1-1608862, Polar Code Construction for NR," Huawei, HiSilicon, Lisbon, Portugal, Technical Specification (TS), 10th–14th October 2016.

[56] "3GPP TSG RAN WG1 Meeting 91, R1-1720702, UCI Segmentation," Reno, USA," Technical Specification (TS), 27th November–1st December 2017.

[57] "3GPP TSG-RAN WG1 NR Ad-Hoc2, R1-1711221, Design of Polar Code for Large UCI with Segmentation," Qingdao, P. R. China," Technical Specification (TS), 27th-–30th, June 2017.

[58] "3GPP TSG RAN WG1 Meeting 88bis, R1- 1704384, Further Consideration on Polar Codes with Maximum Mother Code Size," spokane, USA," Technical Specification (TS), 3rd–7th April 2017.

[59] "3GPP TSG RAN WG1 NR Ad-Hoc3, R1-1715667, Segmentation of Polar Code for Large UCI," ZTE, Sanechips, Nagoya, Japan, Technical Specification (TS), 18th-–21st September 2017.

[60] "3GPP TSG RAN WG1 Meeting 90bis, R1-1718372, Segmentation for Polar Codes," Huawei, HiSilicon, Prague, Czech Republic, Technical Specification (TS), 9th–13th October 2017.

[61] "3GPP TSG RAN WG1 NR Ad-Hoc2, R1-1711749, Segmentation of Polar Codes," ZTE, Qingdao, China, Technical Specification (TS), 27th—30th June 2017.

[62] "3GPP TSG RAN WG1 Meeting 90, R1-1713237, Segmentation of Polar Codes for Large UCI," ZTE, Prague, Czech Republic, Technical Specification (TS), 21st–25th August 2017.

[63] "3GPP TSG RAN WG1 Meeting 90, R1-1718914, Segmentation of Polar codes for Large UCI," ZTE, Sanechips, Prague, Czech Republic, Technical Specification (TS), 9th–13th October 2017.

[64] "3GPP TSG RAN WG1 Meeting 91, R1-1720827, Uplink CRC and nFAR for Polar Codes ," NTT, DOCOMO, Reno, USA, Technical Specification (TS), 27th November–1st December 2017.

[65] C. Condo, S. A. Hashemi, and W. J. Gross, "Blind detection with polar codes," *IEEE Communications Letters*, vol. 21, no. 12, pp. 2550–2553, 2017.

[66] "33GPP TSG RAN WG1 Meeting AH 1801, R1-1800103, Text Proposals for Downlink Control Polar Coding," Huawei, HiSilicon, Vancouver, Canada, Technical Specification (TS), 22nd–26th January 2018.

[67] "3GPP TSG RAN WG1 Meeting 90bis, R1-1717752, Spreadtrum Communications, RNTI Scrambling for PDCCH Polar Code ," Prague, Czech Republic," Technical Specification (TS), 9th–13th October 2017.

[68] "3GPP TSG RAN WG1 Meeting 89,R1-1708833, Design Details of Distributed CRC ," Nokia, Alcatel-Lucent Shanghai Bell, Hangzhou, P.R. China, Technical Specification (TS), 15th–19th May 2017.

[69] "3GPP TSG RAN WG1 Meeting 90, R1-1713303, On the Hardware Implementation of CRC Interleaving ," AccelerComm, Prague, Czech Republic, Technical Specification (TS), 21st–25th August 2017.

[70] "3GPP TSG RAN WG1 Meeting 89, R1-1708836, Coding Scheme for PBCH," Nokia, Alcatel-Lucent Shanghai Bell, Hangzhou, P.R. China, Technical Specification (TS), 15th–19th May 2017.

[71] "3GPP TSG RAN WG1 90bis, R1-1718407, On Setting the Values of Frozen Bit Locations for Polar Codes," Prague, Czech Republic," Technical Specification (TS), 21st–25th August 2017.

[72] "3GPP TSG RAN WG1 Meeting 88bis, R1-1706193, Polar Coding Design for Control Channel," Huawei, HiSilicon, Spokane, USA, Technical Specification (TS), 3rd - 7th April 2017.

[73] I. Tal and A. Vardy, "List decoding of polar codes," *IEEE Transactions on Information Theory*, vol. 61, no. 5, pp. 2213–2226, 2015.

[74] "3GPP TSG RAN WG1 NR Ad-Hoc Meeting 2, R1-1710842, A Flexible, Low-latency, Hardware-efficient SCL Polar Decoder," AccelerComm, Qingdao, China, Technical Specification (TS), 27th-–30th June 2017.

[75] R. Maunder, "Polar Codes," 2017.

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

[77] K. Niu and K. Chen, "CRC-aided Decoding of Polar Codes," *IEEE Communications Letters*, vol. 16, no. 10, pp. 1668–1671, 2012.

[78] T. Murata and H. Ochiai, "On Design of CRC Codes for Polar Codes with Successive Cancellation List Decoding," in *2017 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2017, pp. 1868–1872.

[79] X. Liu, S. Wu, X. Xu, J. Jiao, and Q. Zhang, "Improved Polar SCL Decoding by Exploiting the Error Correction Capability of CRC," *IEEE Access*, vol. 7, pp. 7032–7040, 2018.

[80] "3GPP TSG RAN WG1 NR Ad-Hoc2, R1-1710750, Design of Unified Rate-Matching for Polar Codes," Samsung, Qingdao, China, Technical Specification (TS), 27th-–30th June 2017.

[81] "3GPP TSG RAN WG1 Meeting NR Ad-Hoc2, R1- 1711729, WF on Circular Buffer of Polar Code," Ericsson, Qualcomm, MediaTek, LGE, Qingdao, China, Technical Specification (TS), 27th-–30th June 2017.

[82] "3GPP TSG RAN1 Meeting 91, R1-1719612, Further Discussion on Channel Interleaver for Polar Codes of UCI," Ericsson," Technical Specification (TS), 2017.

[83] "3GPP TSG RAN WG1 Meeting 90, R1-1712649, Channel Interleaver for Polar Codes," Ericsson, Prague, P.R.Czech Republic, Technical Specification (TS), 21th–25th August 2017.

[84] "3GPP TSG-RAN WG1 NR Ad-Hoc2, R1-1711222, Design and Evaluation of Interleaver for Polar Codes," Qualcomm Incorporated, Qingdao, P. R. China, Technical Specification (TS), 27th–30th June 2017.

[85] "3GPP TSG-RAN WG1 Meeting 89, R1-1708649, Interleaver Design for Polar Codes," Qualcomm Incorporated, Hangzhou, P. R. China, Technical Specification (TS), 15th–19th May 2017.

[86] "3GPP TSG RAN WG1 88bis, R1-1706102, Improvement of FAR Performance for CA Polar Codes ," Samsung, Spokane, USA, Technical Specification (TS), 3rd–7th April 2017.

[87] "3GPP TS 38.211, 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; NR; Physical Channels and Modulation (version 2.0.0 (Release 15)Technical Specification (TS)," December 2017.

[88] T. Erseghe, "Coding in the finite-blocklength regime: Bounds based on laplace integrals and their asymptotic approximations," *IEEE Transactions on Information Theory*, vol. 62, no. 12, pp. 6854–6883, 2016.

[89] "3GPP TSG-RAN WG1 NR Ad-Hoc-2, R1-1711582, Analysis of the Benefits of PC Bits for Polar Code," Huawei, HiSilicon, Qingdao, China, Technical Specification (TS), 27th-30th– June 2017.

[90] "3GPP TSG-RAN WG1 Meeting 91, R1-1721462, Remaining Issues of Polar Code Construction for UCI," Ericsson, Reno, USA, Technical Specification (TS), 1st–December 2017.

[91] Z. Babar, Z. B. K. Egilmez, L. Xiang, D. Chandra, R. G. Maunder, S. X. Ng, and L. Hanzo, "Polar codes and their quantum-domain counterparts," *IEEE Communications Surveys & Tutorials*, 2019.

ZEYNEP B. KAYKAC EGILMEZ received her dual B.Eng. (Hons.) degrees in Electrical- Electronic Engineering and in Industrial Engineering from the Kırıkkale University, Turkey, in 2014. In November 2017, the University of Southampton has awarded her M.Sc. degree in next-generation wireless communications with distinction, Southampton, UK. Currently, she is working toward a Ph.D. degree in the next-generation wireless communication with the University of Southampton. Her research interests include channel coding, especially polar coding in wireless communication.

LUPING XIANG received the B.Eng. degree (Hons.) from Xiamen University, China, in 2015. He is currently pursuing the Ph.D. degree with the Southampton Wireless Group, University of Southampton. His research interests include channel coding and ultralow latency scheme.

ROBERT G. MAUNDER (SM'12) received the B.Eng. degree (Hons.) in electronic engineering and the Ph.D. degree in telecommunications from the School of Electronics and Computer Science, University of Southampton, U.K., in 2003 and 2007, respectively. He was a Lecturer with the NGW Group, School of Electronics and Computer Science, University of Southampton, U.K., in 2007, and was promoted to an Associate Professor in 2013 and to a Professor in 2017. He is the Founder and CTO of AccelerComm Ltd., which is commercializing his research as soft-IP. His research interests include joint source/channel coding and the holistic design of algorithms and hardware implementations for wireless communications. He has published over 100 IEEE papers in these areas. He was a Chartered Engineer of the IET in 2013 and a Fellow of the IET in 2017.

LAJOS HANZO (F'04) received the D.Sc. degree in electronics, in 1976 and the Ph.D. degree, in 1983. He received an honorary doctorate from the Technical University of Budapest, in 2009, and also from The University of Edinburgh, in 2015. In 2016, he was admitted to the Hungarian Academy of Science. During his 40-year career in telecommunications, he has held various research and academic posts in Hungary, Germany, and U.K. Since 1986, he has been with the School of Electronics and Computer Science, University of Southampton, U.K., where he holds the Chair in telecommunications. He has successfully supervised 119 Ph.D. students, co-authored 18 John Wiley/IEEE Press books on mobile radio communications totaling in excess of 10 000 pages, published 1900+ research contributions at IEEE Xplore. He is a Fellow of the Royal Academy of Engineering, IET, and EURASIP. He acted both as a TPC and a General Chair of IEEE conferences, presented keynote lectures and has been awarded a number of distinctions. He is currently directing an academic research team, working on a range of research projects in wireless multimedia communications sponsored by industry, the Engineering and Physical Sciences Research Council, U.K., the European Research Council's Advanced Fellow Grant, and the Royal Society's Wolfson Research Merit Award. He is also a Governor of the IEEE VTS. From 2008 to 2012, he was the Editor-in-Chief of the IEEE Press and a Chaired Professor at Tsinghua University, Beijing. He is an enthusiastic supporter of industrial and academic liaison. He offers a range of industrial courses.