ANR, Appel à Projets Générique (AAPG 2019)

QCSP Project (ANR-19-CE25-0013-02)

Deliverable D1.3

Decoding algorithms, performance and complexity considerations

Editor:	Franklin Cochachin (ETIS)
Deliverable nature:	Internal (scope: Consortium and ANR)
Due date:	December 31, 2021
Delivery date:	December 31, 2021
Version:	1.0
Total number of pages:	23 pages
Keywords:	NB-Polar codes, NB-LDPC codes, NB-Turbo codes, CCSK modulation, complexity of Non-Polar codes

Abstract
This deliverable presents a simplified version of the Successive Cancellation (SC) decoder for Non-Binary Polar (NB-Polar) codes. The proposed decoder, named Successive Cancellation Min-Sum (SC-MS), is exclusively formulated in the Log-Likelihood Ratio (LLR) domain to reduce the decoding complexity of the SC decoder. The NB-Polar codes are associated with Cyclic Code-Shift Keying (CCSK) modulation to obtain a new coded modulation scheme for ultra-low signal-to-noise ratios (SNRs). A version that reduces the complexity of the SC-MS decoder is also presented. The quantized version of the SC-MS decoder is investigated using quantized LLRs on optimized size of bits. This deliverable also reports on the comparison of NB-Polar, NB-LDPC, and NB-Turbo decoders in terms of frame error rate (FER) performance.

List of Authors

Partner	Author
ETIS	Franklin Cochachin (franklin-rafael.cochachin-henostroza@ensea.fr)
ETIS	Fakhreddine Ghaffari (fakhreddine.ghaffari@ensea.fr)
ETIS	Laura Luzzi (laura.luzzi@ensea.fr)

Contents

Executive Summary

Work Package 1 (WP1: NB codes) of the QCSP project targets the design and the optimization of low coding rate CCSK-NB-codes. Three families of NB codes will be investigated: NB-Turbo code (by Lab-STICC/IMTA), NB-Polar code (by CEA and ETIS) and NB-LDPC code (by building upon the previous work of Lab-STICC/UBS, CEA).

The goal of **Task 1.3 (ETIS): Decoding algorithms, performance and complexity considerations [M9-M21]** is to investigate, strengthen, and compare different solutions to decode low-rate NB codes. On the one hand, non–simplified decoding algorithms for NB-Turbo and NB-LDPC codes have been intensively investigated in the literature. However, for very low coding rates, only little is known about their performance/complexity trade-off. Thus, extra optimizations may be required – especially for low-complexity algorithms – in order to fully exploit the correction capability of the code. On the other hand, decoding algorithms for NB-Polar codes have practically not been investigated in the literature (except for the standard SC decoding approach). We aim at investigating different solutions for decoding NB-Polar codes, including SC decoding with min-sum or min-max approximations [1], or the generalization to the NB case of more powerful SC-List and D-SC-Flip decoding algorithms [2]. Decoding algorithms should be compared in terms of both error correction performance and computational complexity. Hardware efficiency/architectural considerations will also be taken into account, to enable the transfer of the proposed solutions to WP3. Deliverable 1.3: Report (Public) M21

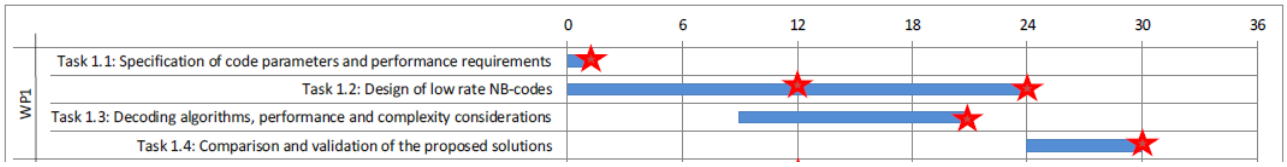


Figure 1: Gantt diagram of WP1

This deliverable is organized as follows:

Section 1 presents an introduction to Non-Binary Polar (NB-Polar) codes.

Section 2 presents NB-Polar codes and the CCSK modulation.

Section 3 describes the Log-Likelihood Ratio (LLR) calculation of the association between the Cyclic Code-Shift Keying (CCSK) modulation and the NB-Polar code, this section also proposes the Successive Cancellation Min-Sum (SC-MS) decoder and a version that reduces the complexity of the SC-MS decoder.

Section 4 presents the finite precision SC-MS decoder.

Section 5 presents the comparison of NB-Polar, NB-LDPC, and NB-Turbo decoders in terms of frame error rate (FER) performance.

1 Introduction

Binary Polar codes [3], introduced by Arikan in 2008, are the first provable capacity-achieving error correction codes for binary-input discrete memoryless channels with low encoding and decoding complexity. Binary polar codes have been adopted for the Enhanced Mobile Broadband (eMBB) control channel of 5G New Radio (NR).

In the current decade, more than 50 billion devices will be connected thanks to the Internet of Things (IoT) [4]. The new generation of mobile networks, notably the 5G system, will need to support short packet traffic [5,6]. At the link level, we can take advantage of powerful error control codes such as Non-Binary Polar (NB-Polar) codes to transmit short packets.

NB-Polar codes [7–11] of length $N = 2^n$, also known as q-ary polar codes, are powerful Forward Error Correction (FEC) codes defined over the Galois Field (GF) $\mathbb{F}_q$, $q > 2$. A NB-Polar code processes multiple bits in parallel and helps reduce the probability of frame error compared to binary polar codes.

The Cyclic Code-Shift Keying (CCSK) modulation [12] is a q-ary direct-sequence spread-spectrum (DSSS) technique that improves the spectral efficiency of spread-spectrum systems. The CCSK modulation can provide self-synchronization and identification capabilities. All CCSK sequences are obtained from a unique pseudo-random noise (PN) sequence that is circularly shifted. In [13], the authors propose the association of NB-LDPC codes and CCSK to prevent the information loss when calculating channel probabilities at the symbol level.

In this report, we associate the CCSK modulation to NB-Polar codes to obtain a new coded modulation scheme that can be used in low power networks requiring long range connectivity and high sensitivity, *e.g.* Low-Power Wide-Area (LPWA) networks for IoT applications. We examine the Successive Cancellation (SC) decoder for NB-Polar codes operating in the probability domain. The association of CCSK and NB-Polar codes enables the SC decoder to have good decoding performance at ultra-low signal-to-noise ratios (SNRs).

In order to reduce the complexity of the SC decoder, we propose a decoder named Successive Cancellation Min-Sum (SC-MS) decoder that is exclusively formulated in the Log-Likelihood Ratio (LLR) domain. To further reduce the complexity of the SC-MS decoder, we simplify the kernel transformation for construction of the NB-Polar codes. The simplified kernel also helps reduce the complexity of the NB-Polar encoder.

We investigate SC-MS decoders defined over small alphabets constructed from $Q_m \in \{2,3,4,5\}$ bits of precision for the *internal LLRs*. We also quantize the *channel LLRs* on small alphabets constructed from $Q_{ch} \in \{2,3,4,5\}$ bits of precision.

Our numerical results show that the SC-MS decoder presents a negligible performance degradation with respect to the SC decoder for code length $N \in \{64,128,256,512,1024\}$. In the case of code length $N = 2048$, the SC-MS decoder suffers a very small performance loss compared to the SC decoder, the degradation at FER $= 10^{-2}$ is around 0.1 dB for $\mathbb{F}_{2^8}$. Comparing quantized SC-MS decoders, we observe that the $(Q_{ch} = 3, Q_m = 4)$-bit SC-MS can achieve almost the same performance as the $(Q_{ch} = 5, Q_m = 5)$-bit SC-MS. We also observe that the $(Q_{ch} = 2, Q_m = 3)$-bit SC-MS decoder offers a good trade-off between performance and complexity.

When comparing the performance of the SC-MS decoder with the NB-LDPC and NB-Turbo decoders, we observe that the NB-Turbo decoder has the best performance for the different simulated codes. Comparing the SC-MS and NB-LDPC decoders, both decoders have almost the same FER performance, in some cases the SC-MS decoder has better performance and in others the NB-LDPC decoder has better performance.

2 NB-Polar Codes and CCSK Modulation

2.1 Non-Binary Polar Codes

In this report, a NB-Polar code of length $N = 2^n$ is defined over the Galois Field $\mathbb{F}_q$, with $q = 2^p$ and where $p > 1$. Let $\boldsymbol{x} = (x_0, ..., x_{N-1})$, $x_i \in \mathbb{F}_q$ for $i = 0, ..., N-1$, be the codeword that is obtained after encoding the message $\boldsymbol{u} = (u_0, ..., u_{N-1})$, $u_i \in \mathbb{F}_q$ for $i = 0, ..., N-1$. Also, let $\mathbb{F}_q^*$ denote the set of all non-zero elements of $\mathbb{F}_q$. It is worth mentioning that a GF element can be represented by a binary vector of size p, *e.g.* $x_1 = (x_1^0, ..., x_1^{p-1})$. For the construction of NB-Polar codes, we consider the kernel transformation $\mathcal{T} : (x_0, x_1) = (u_0 \oplus u_1, h \odot u_1)$, with '$\oplus$' and '$\odot$' respectively denoting the addition and multiplication rules over $\mathbb{F}_q$, and where $h \in \mathbb{F}_q^*$ is a randomly chosen element. It has been shown in [7] that the transformation $\mathcal{T}$ guarantees polarization of the NB-Polar codes if q is a prime power. A NB-Polar code of length $N = 2^n$ is obtained by applying $\mathcal{T}$ recursively (n times), and in each step of the recursion different values of h can be used. In the decoding process, $\mathcal{T}$ is composed of a check node (CN) and a variable node (VN), hence a NB-Polar code of $N = 2^n$ has $n(N/2)$ VNs/CNs. The update rule for a VN/CN is presented in section 3. Fig. 2.1 shows a graph of a NB-Polar code of $N = 2^3$ and composed of 12 VNs (represented with $\boxed{=}$) and 12 CNs (represented with $\bigoplus$).

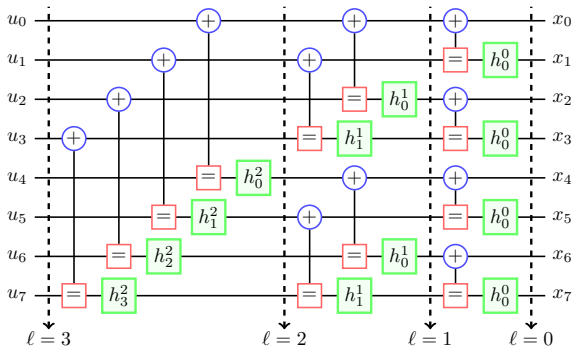


Figure 2.1: Graph of a NB-Polar code of length $N = 8$.

A (N, K) NB-Polar code has K information symbols, a total length of $N = 2^n$, $N - K$ frozen symbols, and code rate $R_c = K/N$. Thus a codeword is composed of $K \times p$ bits of information. Let $\mathcal{I} = \{i_0, ..., i_{K-1}\}$ denote the set of indices $0 \leq i_K \leq N-1$ that serves to indicate the positions of the information symbols, and let $\mathcal{I}^c$ denote the set of all elements in the set $\{0, ..., N-1\}$ that are not in $\mathcal{I}$. We use a genie-aided successive cancellation decoder [3] to compute the sets $\mathcal{I}$ and $\mathcal{I}^c$ at each SNR value. In this report, the frozen symbols $\boldsymbol{u}_{\mathcal{I}^c} = \{u_i, i \in \mathcal{I}^c\}$ are set to zero and their positions are known to both encoder and decoder. Note that $\boldsymbol{u} = (\boldsymbol{u}_{\mathcal{I}}, \boldsymbol{u}_{\mathcal{I}^c})$, where $\boldsymbol{u}_{\mathcal{I}} = \{u_i, i \in \mathcal{I}\}$ are the information symbols.

2.2 Cyclic Code-Shift Keying Modulation

The CCSK modulation is a technique that associates pseudo-random noise (PN) sequences of length $q = 2^p$ to p-bit symbols. Each sequence is derived from a unique PN sequence $\eta_0 = (\eta_0(0), ..., \eta_0(q-1))$ of length q, where $\eta_0(k) \in \{0, 1\}$ for $k = 0, ..., q-1$. We perform the circular shift to the left of η_0 in $s \in \{0, ..., q-1\}$ positions to obtain the PN sequence $\eta_s = (\eta_s(0), ..., \eta_s(q-1))$, where $\eta_s(k) = \eta_0((k+s) \mod q)$ for $k = 0, ..., q-1$. The rate of the modulation is $R_m = p/q$.

PN sequences can be constructed using a Linear Feedback Shift Register (LFSR). The PN sequence generated by an LFSR has good autocorrelation properties. Algorithms like the genetic algorithm can also be used to generate and optimize a PN sequence.

3 Successive Cancellation Based Decoders

In this section, we present the association of CCSK modulation to NB-Polar codes. We also propose a simplified version of the SC decoder. Taking advantage of the recursive structure, the decoder for the NB-Polar code proceeds in $n+1$ stages $\ell \in \{0, ...n\}$, see Fig 2.1. From this section onwards, we distinguish between two types of LLRs, the *channel LLRs* calculated at stage $\ell = 0$ and the *internal LLRs* obtained at stage $\ell \in \{1, ..., n\}$.

3.1 Definition of the Log-Likelihood Ratio

Let $\eta_0 = (\eta_0(0), ..., \eta_0(q-1))$ denote a fundamental PN sequence, and let $\boldsymbol{\eta} = (\eta_{x_0}, ..., \eta_{x_{N-1}})$ denote the codeword after applying the CCSK technique to $\boldsymbol{x} = (x_0, ..., x_{N-1})$. Each symbol $x_i \in \mathbb{F}_q$, $i = 0, ..., N-1$, of $\boldsymbol{x}$ is mapped to the PN sequence $\eta_{x_i} = (\eta_{x_i}(0), ..., \eta_{x_i}(q-1))$, where $\eta_{x_i}(k) = \eta_0((k+x_i) \mod q)$ for $k = 0, ..., q-1$, with $\eta_{x_i}(k) \in \{0, 1\}$. The code rate after the CCSK modulation is $R_e = Rc \times R_m = (K \times p)/(N \times q)$. We consider that $\boldsymbol{\eta}$ is modulated by the Binary Phase-Shift Keying (BPSK) modulation and transmitted over the Binary Input Additive White Gaussian Noise (BI-AWGN) channel with noise variance σ^2. The channel output $\boldsymbol{y} = (y_0, ..., y_{N-1})$, with $y_i = (y_i(0), ..., y_i(q-1))$ for $i = 0, ..., N-1$, is modeled by $y_i(k) = (1-2\eta_{x_i}(k)) + z_i(k)$, $k = 0, \ldots, q-1$, where $z_i(k)$ is a sequence of independent and identically distributed (i.i.d.) Gaussian random variables with zero mean and variance σ^2.

We can define the vector of channel LLRs $L_i = (L_i(0), ..., L_i(q-1))$ of a GF symbol x_i as

$$L_i(x_i) = \log\left(\frac{\Pr(y_i \mid \hat{\eta}_{x_i})}{\Pr(y_i \mid \eta_{x_i})}\right) \quad \forall x_i \in \mathbb{F}_q, \tag{3.1}$$

where $\hat{\eta}_{x_i}$ is the hard decision over y_i, *i.e.* $\hat{\eta}_{x_i}(k) = 1$ if $y_i(k) < 0$, $\hat{\eta}_{x_i}(k) = 0$ otherwise, $k = 0, \ldots, q-1$. Equation (3.1) can be expressed as:

$$L_i(x_i) = \sum_{k=0}^{q-1} \log\left(\Pr(y_i(k) \mid \hat{\eta}_{x_i}(k))\right) - \log\left(\Pr(y_i(k) \mid \eta_{x_i}(k))\right) \ \forall x_i \in \mathbb{F}_q. \tag{3.2}$$

Replacing the conditional distribution for the BI-AWGN channel $\Pr(y \mid x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(y-x)^2/2\sigma^2}$, we obtain

$$L_i(x_i) = \sum_{k=0}^{q-1} \frac{2y_i(k)}{\sigma^2} \left(\eta_{x_i}(k) - \hat{\eta}_{x_i}(k)\right) \quad \forall x_i \in \mathbb{F}_q. \tag{3.3}$$

The vector of channel LLRs computed with (3.3) allows us to obtain only positive LLR values. To set at least one element of L_i equal to zero, we use

$$L_i'(x_i) = L_i(x_i) - \min(L_i) \quad \forall x_i \in \mathbb{F}_q. \tag{3.4}$$

From the LLR values, the probability distribution $P_i(x_i) = \Pr(y_i \mid x_i)$ can be computed by:

$$P_i(x_i) = \frac{e^{-L_i(x_i)}}{\sum\limits_{x_i' \in \mathbb{F}_q} e^{-L_i(x_i')}} \quad \forall x_i \in \mathbb{F}_q. \tag{3.5}$$

3.2 Successive Cancellation Decoder

The SC decoder, defined in the probability domain [14], estimates the information symbols one by one. Note that each estimated symbol is used to decode the next symbol.

Let $\boldsymbol{P}^{\ell} = (P_0^{\ell}, ..., P_{N-1}^{\ell})$ denote the probability distribution computed at stage $\ell \in \{0, ..., n\}$ during the decoding process, where $P_i^{\ell} = (P_i^{\ell}(0), ..., P_i^{\ell}(q-1))$, $i = 0, ..., N-1$, denotes the probability distribution of the intermediate values u_i^{ℓ}, and let $\hat{u}_i^{\ell}$ denote the estimated symbol of u_i^{ℓ}. We note that $u_i^0 = x_i$. At the initialization stage of the SC decoder ($\ell = 0$), P_i^0 is initialized with (3.5), *i.e.* $P_i^0 = P_i(x_i)$ for $i = 0, ..., N-1$.

Let us consider the transformation

$$\mathcal{T} : \left(u_{\theta_t^{\ell-1}}^{\ell-1}, u_{\phi_t^{\ell-1}}^{\ell-1}\right) = \left(u_{\theta_t^{\ell-1}}^{\ell} \oplus u_{\phi_t^{\ell-1}}^{\ell}, h_{\phi_t^{\ell-1}}^{\ell-1} \odot u_{\phi_t^{\ell-1}}^{\ell}\right), \tag{3.6}$$

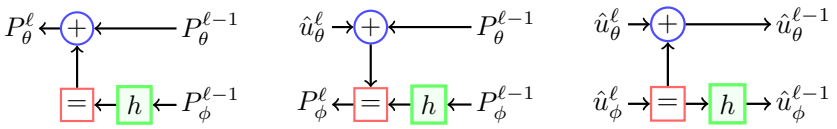
where $\theta_t^{\ell-1}$ and $\phi_t^{\ell-1}$ are given by

$$\begin{aligned} \theta_t^{\ell-1} &= 2t - (t \mod 2^{\ell-1}), \\ \phi_t^{\ell-1} &= 2^{\ell-1} + 2t - (t \mod 2^{\ell-1}), \end{aligned}$$

for $t = 0, 1, ..., N/2 - 1$.

To simplify the notations, we use θ (respectively ϕ) to denote $\theta_t^{\ell-1}$ (respectively $\phi_t^{\ell-1}$). Similarly, h is used to denote $h_{\phi_t^{\ell-1}}^{\ell-1}$ for simplicity.

The update rules considering $\mathcal{T}$ are shown in Fig. 3.1.



(a) Check node. (b) Variable node. (c) Propagation of $\hat{u}$.

Figure 3.1: Update rules for the SC decoder.

The update rule at a CN is given by

$$P_\theta^{\ell}(u_\theta^{\ell}) = \beta \sum_{u_\phi^{\ell} \in \mathbb{F}_q} P_\theta^{\ell-1}(u_\theta^{\ell} \oplus u_\phi^{\ell}) P_\phi^{\ell-1}(h \odot u_\phi^{\ell}) \quad \forall u_\theta^{\ell} \in \mathbb{F}_q, \tag{3.7}$$

where $\beta^{-1} = \sum_{u_\theta^{\ell} \in \mathbb{F}_q} P_\theta^{\ell}\left(u_\theta^{\ell}\right)$, *i.e.* β is a normalization factor.

The update rule at a VN is defined as

$$P_\phi^{\ell}\left(u_\phi^{\ell}\right) = \delta P_\theta^{\ell-1}\left(\hat{u}_\theta^{\ell} \oplus u_\phi^{\ell}\right) P_\phi^{\ell-1}\left(h \odot u_\phi^{\ell}\right), \tag{3.8}$$

where $\delta^{-1} = \sum_{u_\phi^{\ell} \in \mathbb{F}_q} P_\phi^{\ell}\left(u_\phi^{\ell}\right)$.

To propagate the estimated symbols, we use

$$\left(\hat{u}_\theta^{\ell-1}, \hat{u}_\phi^{\ell-1}\right) = \left(\hat{u}_\theta^{\ell} \oplus \hat{u}_\phi^{\ell}, h \odot \hat{u}_\phi^{\ell}\right). \tag{3.9}$$

We can estimate a message $\hat{\boldsymbol{u}} = (\hat{u}_0^n, ..., \hat{u}_{N-1}^n)$ of length $N = 2^n$ by applying (3.7), (3.8), and (3.9) at each decoding stage ℓ. At stage $\ell = n$, the hard decision of u_i^n is obtained as

$$\hat{u}_i^n = \begin{cases} 0, & \text{if} \quad i \in \mathcal{I}^c \\ \arg\max\limits_{u_i^n} P_i^n(u_i^n), & \text{if} \quad i \in \mathcal{I}. \end{cases} \tag{3.10}$$

3.3 Successive Cancellation Min-Sum Decoder

We propose a simplified version of the SC decoder named Successive-Cancellation Min-Sum (SC-MS) decoder that is exclusively formulated in the LLR domain.

3.3.1 Update Rules of SC-MS Decoders

Let $\boldsymbol{L}^{\ell} = (L_0^{\ell}, ..., L_{N-1}^{\ell})$ denote the LLRs computed at stage $\ell \in \{0, ..., n\}$ during the decoding process, and let $L_i^{\ell} = (L_i^{\ell}(0), ..., L_i^{\ell}(q-1))$, $i = 0, ..., N-1$, denote the LLR of u_i^{ℓ}. The SC-MS decoder is initialized using (3.4), *i.e.* we have $L_i^0 = L_i'(x_i)$ for $i = 0, ..., N-1$. With these notations and considering $\mathcal{T}$ of (3.6), the update rule at a CN is given by

$$L_{\theta}^{\ell}(u_{\theta}^{\ell}) = \min_{u_{\phi}^{\ell} \in \mathbb{F}_q} \left(L_{\theta}^{\ell-1}(u_{\theta}^{\ell} \oplus u_{\phi}^{\ell}) + L_{\phi}^{\ell-1}(h \odot u_{\phi}^{\ell}) \right) \quad \forall u_{\theta}^{\ell} \in \mathbb{F}_q. \tag{3.11}$$

The update rule at a VN is defined as

$$L_{\phi}'\left(u_{\phi}^{\ell}\right) = L_{\theta}^{\ell-1}\left(\hat{u}_{\theta}^{\ell} \oplus u_{\phi}^{\ell}\right) + L_{\phi}^{\ell-1}\left(h \odot u_{\phi}^{\ell}\right). \tag{3.12}$$

$$L_{\phi}^{\ell}\left(u_{\phi}^{\ell}\right) = L_{\phi}'\left(u_{\phi}^{\ell}\right) - \min\left(L_{\phi}'\right). \tag{3.13}$$

Equation (3.13) is necessary for numerical reasons to ensure the nondivergence of the SC-MS decoder. In the decoding process without the use of (3.13), the internal LLRs converge to very large numerical values that are computationally intractable.

We can estimate $\hat{\boldsymbol{u}} = (\hat{u}_0^n, ..., \hat{u}_{N-1}^n)$ by applying (3.11), (3.12), (3.13), and (3.9), where the hard decision of u_i^n is obtained as

$$\hat{u}_i^n = \begin{cases} 0, & \text{if} \quad i \in \mathcal{I}^c \\ \arg\min\limits_{u_i^n} L_i^n(u_i^n), & \text{if} \quad i \in \mathcal{I}. \end{cases} \tag{3.14}$$

3.3.2 Performance of SC-MS Decoders

In this section, we present the frame error rate (FER) performance of SC-based decoders over the BI-AWGN channel for various NB-Polar codes and two high-order GFs.

A genie-aided SC decoder is used to compute the set $\mathcal{I}$. Additionally, the set of symbol positions $\mathcal{I}_{MS}$ is obtained using a genie-aided SC-MS decoder. The sets $\mathcal{I}$ and $\mathcal{I}_{MS}$ are calculated at each value of SNR; the results show that the two sets are not equal although they may have many elements in common.

We consider four cases for the Monte Carlo simulations: *(i)* $(K = 20, \mathbb{F}_{2^6})$, *(ii)* $(K = 160, \mathbb{F}_{2^6})$, *(iii)* $(K = 30, \mathbb{F}_{2^8})$, and *(iv)* $(K = 240, \mathbb{F}_{2^8})$. Also, we consider $N \in \{64, 128, 256\}$ for $K \in \{20, 30\}$, and $N \in \{512, 1024, 2048\}$ for $K \in \{160, 240\}$. For all NB-Polar codes, the SC decoder performance is shown as a benchmark.

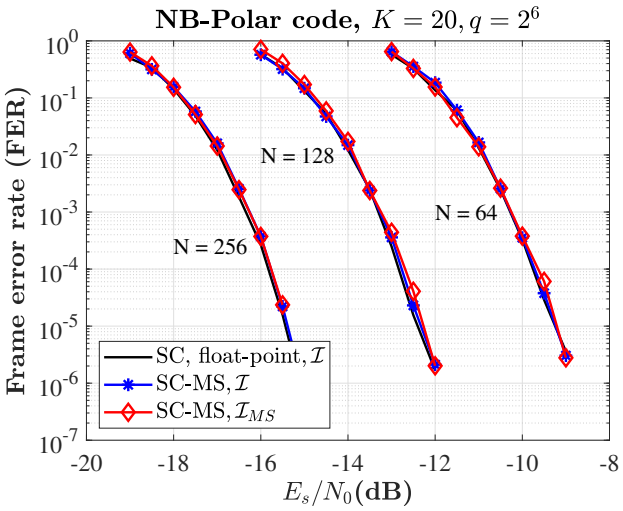
Fig. 3.2 shows the FER performance comparison between the SC and SC-MS using $\mathcal{I}$ and $\mathcal{I}_{MS}$, for six code lengths $N \in \{64, 128, 256, 512, 1024, 2048\}$ and for the GF $\mathbb{F}_{2^6}$. The results obtained show that the SC-MS decoders present a negligible performance degradation with respect to the SC decoders when the code length is $N \in \{64, 128, 256\}$.

Simulation results for the code length $N \in \{64, 128, 256, 512, 1024, 2048\}$ and the field $\mathbb{F}_{2^8}$ are provided in Fig. 3.3, we can see a negligible performance loss for SC-MS decoders when the code length is $N \in \{64, 128, 256\}$. We can observe at FER $= 10^{-2}$ a very small performance loss of about 0.1 dB for $(N = 2048, K = 240)$, 0.08 dB for $(N = 1024, K = 240)$, and 0.05 dB for $(N = 512, K = 240)$. Similar behavior is observed for $K = 160$, $\mathbb{F}_{2^6}$, and $N \in \{512, 1024, 2048\}$.

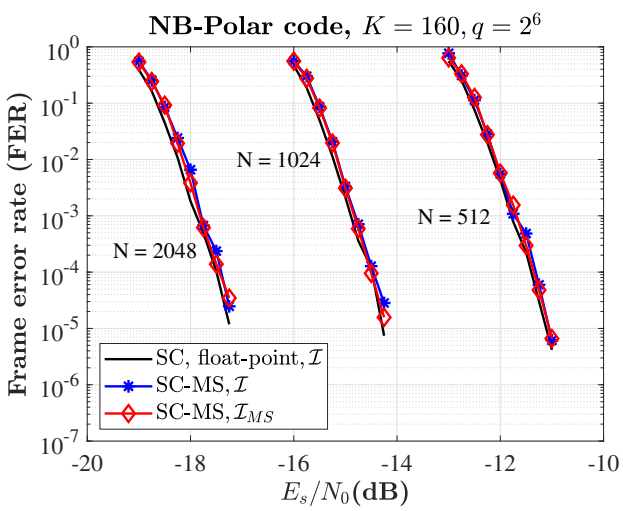
When comparing the FER performance of the SC-MS decoders using the sets $\mathcal{I}$ and $\mathcal{I}_{MS}$, the SC-MS decoders implemented with $\mathcal{I}$ have (almost) the same decoding performance as SC-MS decoders implemented with $\mathcal{I}_{MS}$.

3.4 Low-complexity decoding for NB-Polar codes

In this section, we optimize the SC-MS decoder to reduce its complexity.

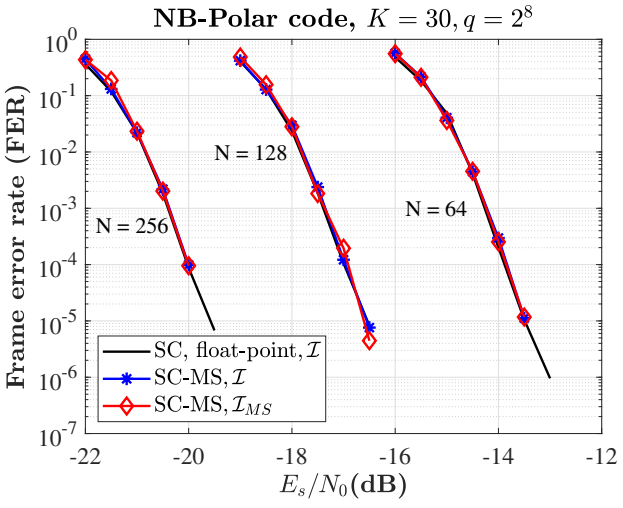


(a) $K = 20$ and $\mathbb{F}_{2^6}$.

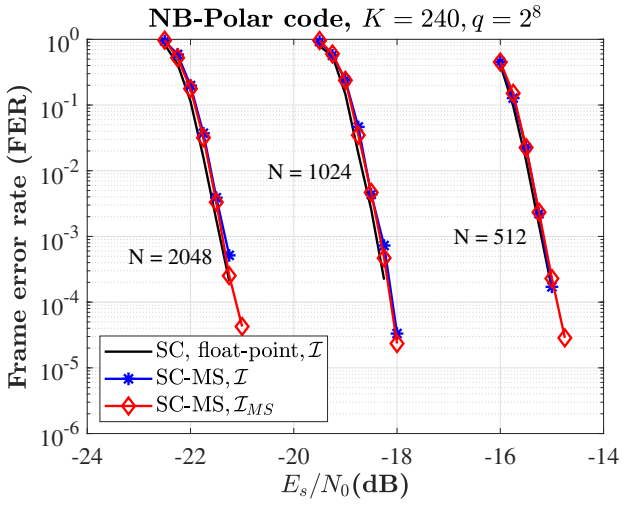


(b) $K = 160$ and $\mathbb{F}_{2^6}$.

Figure 3.2: FER performance of SC-based decoders for $N \in \{64, 128, 256, 512, 1024, 2048\}$ over $\mathbb{F}_{2^6}$.



(a) $K = 30$ and $\mathbb{F}_{2^8}$.



(b) $K = 240$ and $\mathbb{F}_{2^8}$.

Figure 3.3: FER performance of SC-based decoders for $N \in \{64, 128, 256, 512, 1024, 2048\}$ over $\mathbb{F}_{2^8}$.

3.4.1 Optimized Encoding of NB-Polar Codes

In section 2, we considered the kernel transformation $\mathcal{T} : (x_0, x_1) = (u_0 \oplus u_1, h \odot u_1)$ for the construction of NB-Polar codes, where $h \in \mathbb{F}_q^*$ is a randomly chosen element. A NB-Polar code of length $N = 2^n$ is obtained by applying $\mathcal{T}$ recursively (n times). Once the construction of a NB-Polar of length $N = 2^n$ code is finished, we obtain n layers of coefficients h. For example, Fig. 2.1 shows a graph of a NB-Polar code of $N = 2^3$, we can see three layers of coefficients h.

Let d_ℓ denote the configuration where the first ℓ layers of h are equal to 1. Then, we obtain that d_0 corresponds to all the coefficients h different from 1, and also that d_n corresponds to all the coefficients h equal to 1. Let us take the example of $N = 2^3$. Figure 3.4 shows all the possible configurations for d_ℓ, we can observe that for configuration d_3 all coefficients are equal to 1.

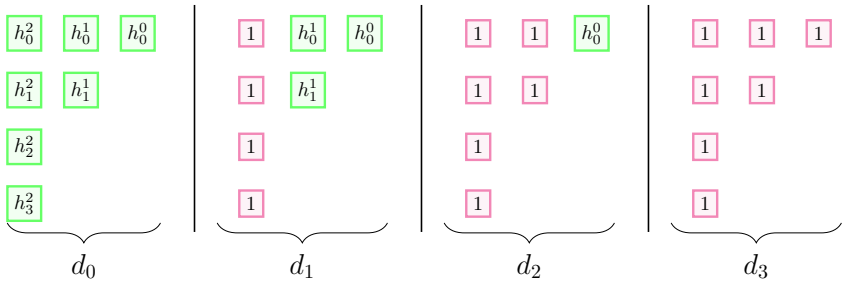


Figure 3.4: All the possible configurations of d_ℓ.

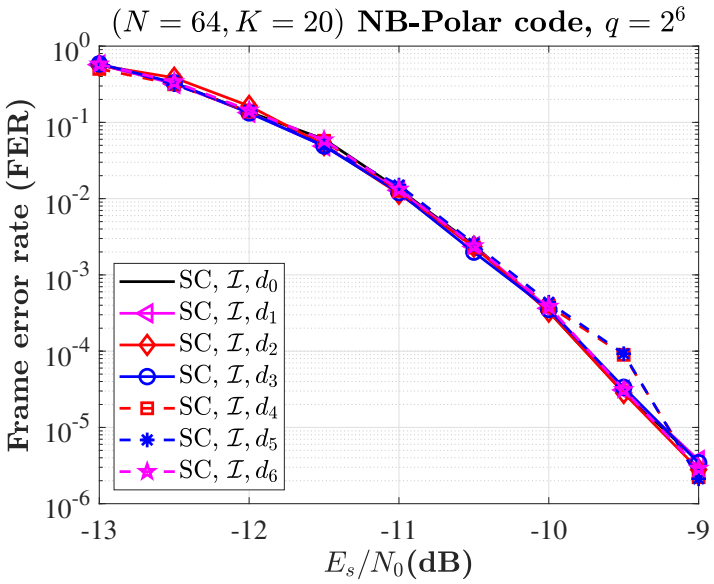
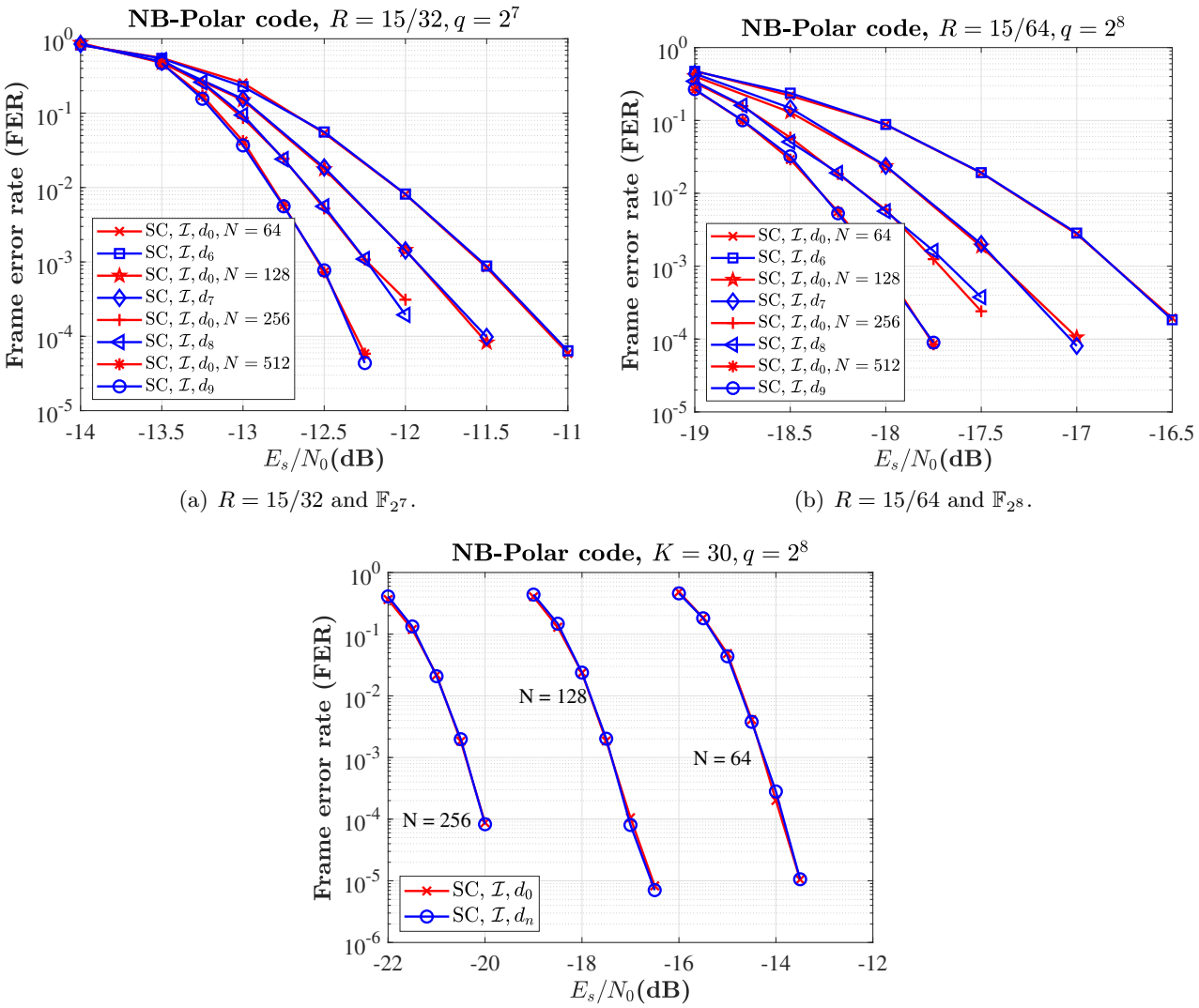


Figure 3.5: FER performance of SC decoders for $(N = 64, K = 20)$ and for $d_\ell \in \{d_0, ..., d_6\}$ over $\mathbb{F}_{2^6}$.

For the optimization process of d_ℓ, we use the SC decoder, we also take into account the association of CCSK modulation and NB-Polar codes. For the good choice of d_ℓ, we compare the FER performance results of the SC decoder by varying d_ℓ, and we choose the configuration d_ℓ that does not degrade the FER performance of the SC decoder.

Figure 3.5 represents the FER performance of the SC decoder for $(N = 64, K = 20)$ and for $d_\ell \in \{d_0, d_1, d_2, d_3, d_4, d_5, d_6\}$ over $\mathbb{F}_{2^6}$. We can clearly see that the FER performance of the SC decoder is the same for any configuration d_ℓ, hence the best choice for d_ℓ is $d_\ell = d_6 = d_n$, that is, we can put the value of 1 to all the coefficients.

Figure 3.6 and Figure 3.7 present the FER performance results for $d_\ell = d_0$ and $d_\ell = d_n$ considering different codes lengths $N = 2^n$, different code rates $R = K/N$, and different fields $\mathbb{F}_q$. The simulation results again show that $d_\ell = d_n$ is the best choice.



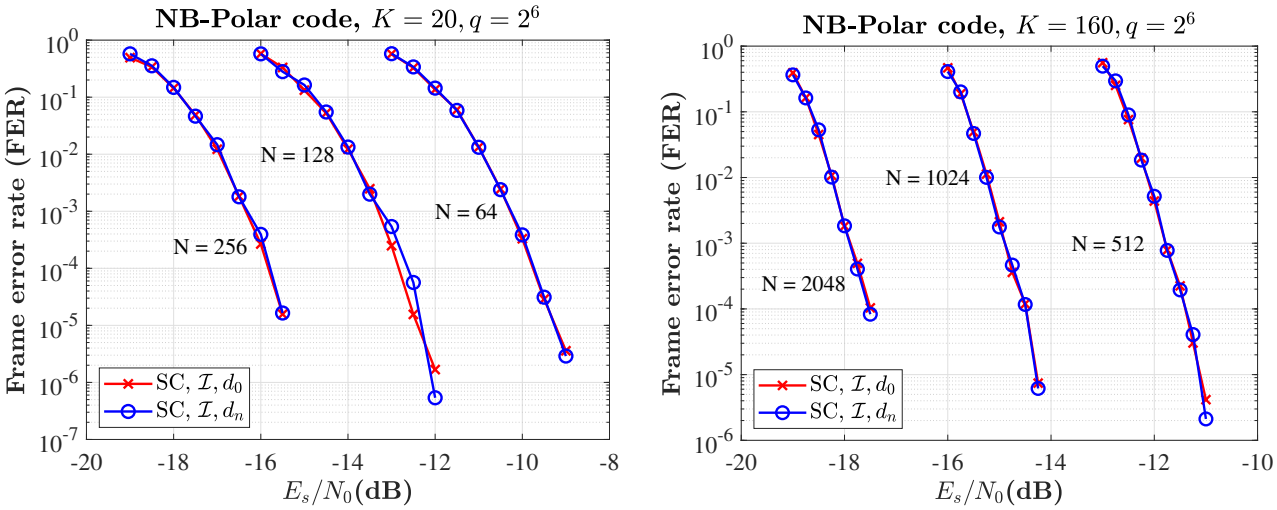
(a) $R = 15/32$ and $\mathbb{F}_{2^7}$.

(b) $R = 15/64$ and $\mathbb{F}_{2^8}$.

(c) $K = 30$ and $\mathbb{F}_{2^8}$.

Figure 3.6: FER performance of SC decoders for $d_\ell \in \{d_0, d_6\}$ over $\mathbb{F}_{2^7}$ and $\mathbb{F}_{2^8}$.

Additional simulation results are presented in Figure 3.7 for $(N = 64, K = 20)$ and for $d_\ell \in \{d_0, d_n\}$ over $\mathbb{F}_q \in \{\mathbb{F}_{2^6}, \mathbb{F}_{2^7}, \mathbb{F}_{2^8}, \mathbb{F}_{2^9}\}$, once again we can see that $d_\ell = d_n$ is the best choice.



(a) $K = 20$. (b) $K = 160$.

Figure 3.7: FER performance of SC decoders for $K \in \{20, 60\}$ and for $d_\ell \in \{d_0, d_6\}$ over $\mathbb{F}_{2^6}$.

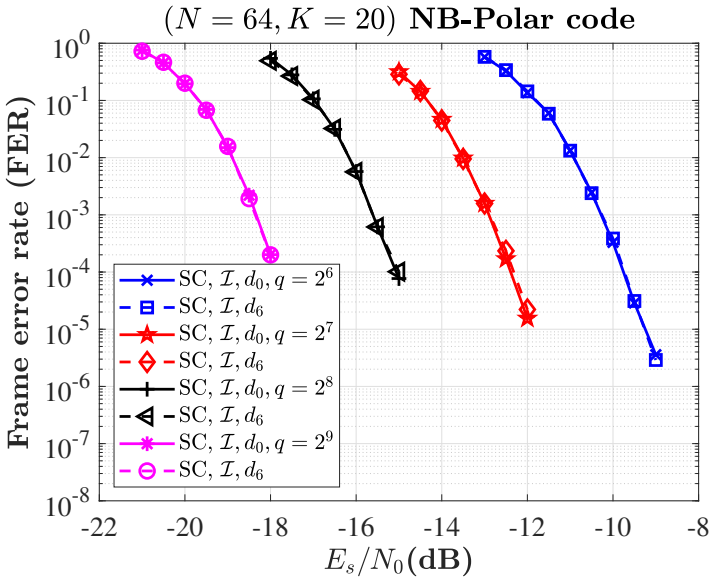


Figure 3.8: FER performance of SC decoders for $(N = 64, K = 20)$ and for $d_\ell \in \{d_0, d_n\}$ over $\mathbb{F}_q \in \{\mathbb{F}_{2^6}, \mathbb{F}_{2^7}, \mathbb{F}_{2^8}, \mathbb{F}_{2^9}\}$.

From all the results obtained, we can conclude that all the coefficients h can be equal to 1 and therefore it is not necessary to use random coefficients. Setting all the coefficients equal to 1 implies a 100% reduction in the use of a memory to store the coefficients. Also, in the process of construction of NB-Polar codes, the multiplication '$\odot$' is no longer necessary, hence the kernel transformation $\mathcal{T} : (x_0, x_1) = (u_0 \oplus u_1, h \odot u_1)$ becomes $\mathcal{T}_O : (x_0, x_1) = (u_0 \oplus u_1, u_1)$. Using the kernel transformation $\mathcal{T}_O$ implies only using XOR logic gates for encoding the NB-Polar codes.

Note that the transformation given in equation 3.6 becomes:

$$\mathcal{T}_O : \left(u^{\ell-1}_{\theta^{\ell-1}_t}, u^{\ell-1}_{\phi^{\ell-1}_t}\right) = \left(u^{\ell}_{\theta^{\ell-1}_t} \oplus u^{\ell}_{\phi^{\ell-1}_t}, u^{\ell}_{\phi^{\ell-1}_t}\right). \tag{3.15}$$

The update rules considering $\mathcal{T}_O$ are described in Section 3.4.2.

3.4.2 Optimized Decoding of NB-Polar Codes

In this section, we present the update rules used to reduce decoding complexity of SC-MS decoders. Let n_o be a natural number such that $0 < n_o < q$, and let $\mathbb{F}_{n_o}^{\ell} = \{\lambda_0^{\ell}, \lambda_1^{\ell}, ..., \lambda_{n_o-1}^{\ell}\}$ be a subset of $\mathbb{F}_q$, where $\lambda_i^{\ell} \in \mathbb{F}_q$ for $i = 0, ..., n_o - 1$. Also let $L_{\phi_O}^{\ell} = (L_{\phi_O}^{\ell}(\lambda_0^{\ell}), ..., L_{\phi_O}^{\ell}(\lambda_{n_o-1}^{\ell}))$ denote the n_o smallest values of the vector $L_{\phi}^{\ell} = (L_{\phi}^{\ell}(0), ..., L_{\phi}^{\ell}(q-1))$. With these notations and considering $\mathcal{T}_O$ of (3.15), the update rule at a CN is given by

$$L_{\theta}^{\ell}(u_{\theta}^{\ell}) = \min_{\lambda_i^{\ell} \in \mathbb{F}_{n_o}^{\ell}} \left(L_{\theta}^{\ell-1}(u_{\theta}^{\ell} \oplus \lambda_i^{\ell}) + L_{\phi_O}^{\ell-1}(\lambda_i^{\ell}) \right) \quad \forall u_{\theta}^{\ell} \in \mathbb{F}_q. \tag{3.16}$$

One can see that equation 3.16 is less complex than equation 3.11. For a fixed value of u_{θ}^{ℓ}, equation 3.16 performs $n_o < q$ operations, while equation 3.11 performs q operations. Hence the importance of choosing a small value of n_o.

For the case of the VN, the update rule is given by

$$L'_{\phi}\left(u_{\phi}^{\ell}\right) = L_{\theta}^{\ell-1}\left(\hat{u}_{\theta}^{\ell} \oplus u_{\phi}^{\ell}\right) + L_{\phi}^{\ell-1}\left(u_{\phi}^{\ell}\right). \tag{3.17}$$

$$L_{\phi}^{\ell}\left(u_{\phi}^{\ell}\right) = L'_{\phi}\left(u_{\phi}^{\ell}\right) - \min\left(L'_{\phi}\right). \tag{3.18}$$

We can estimate $\hat{\boldsymbol{u}} = (\hat{u}_0^n, ..., \hat{u}_{N-1}^n)$ by applying (3.16), (3.17), (3.18), and (3.9), where the hard decision of u_i^n is obtained as

$$\hat{u}_i^n = \begin{cases} 0, & \text{if} \quad i \in \mathcal{I}^c \\ \arg\min\limits_{u_i^n} L_i^n(u_i^n), & \text{if} \quad i \in \mathcal{I}. \end{cases} \tag{3.19}$$

The value of n_o is optimized using Monte Carlo simulations. To compare the FER performance of the optimized decoders, the SC decoder performance is shown as a benchmark using $n_o = q$ and $d_{\ell} = d_0$. Figure 3.9 shows the FER performance of SC-MS decoders for $(N = 64, K = 20)$ and $d_{\ell} = d_n$ over $\mathbb{F}_q \in \{\mathbb{F}_{2^6}, \mathbb{F}_{2^7}\}$. For the case $\mathbb{F}_{2^6}$, we can see that n_o between 16 and 20 is a good choice due to negligible performance loss. In the case of $\mathbb{F}_{2^7}$, n_o between 30 and 40 is a good choice. Of course, we always choose the smallest possible value of n_o.

Simulation results for $(N = 64, K = 20)$ and $\mathbb{F}_q \in \{\mathbb{F}_{2^6}, \mathbb{F}_{2^7}, \mathbb{F}_{2^8}, \mathbb{F}_{2^9}\}$ are provided in Figure 3.10. We can see that the performance losses of the SC-MS decoders are negligible for small values of n_o compared to q. From Figure 3.9 and Figure 3.10, we can see that for a fixed code rate, the ratio n_o/q decreases as q increases.

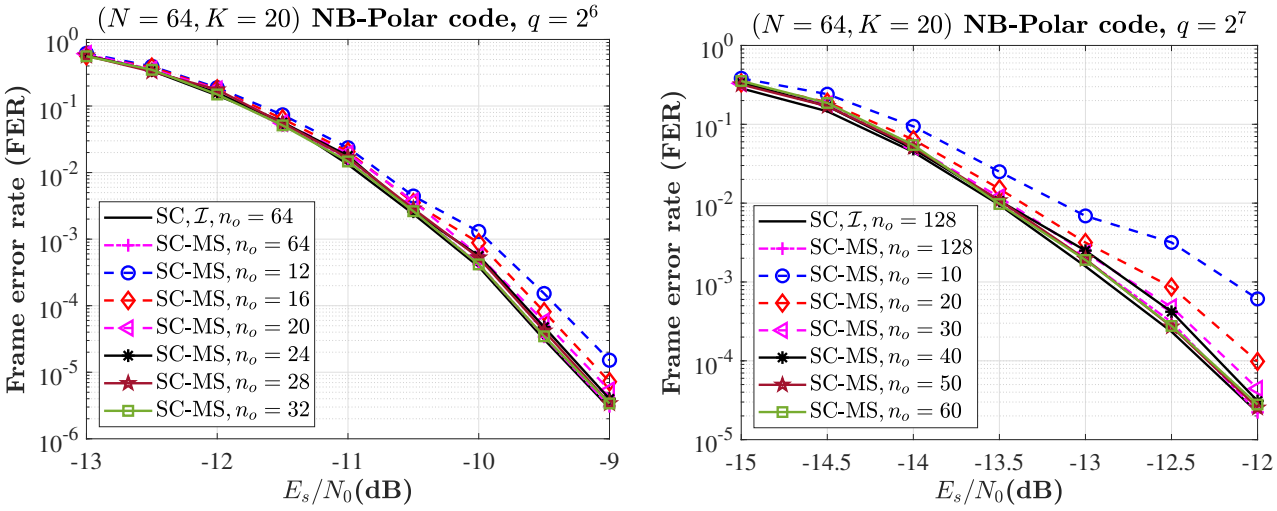
Figure 3.11 depicts the FER performance of the SC-MS decoders for $N \in \{64, 128, 256\}$ and $R = 5/16$ over $\mathbb{F}_{2^6}$. Comparing the decoders, the SC-MS decoders with small values of n_o can reach the FER performance of the SC decoders. From these results, we can see that for a fixed code rate and fixed q, n_o increases as the length N increases.

3.5 Complexity of SC-based Decoders

We measure decoding complexity by counting the adders, multipliers, and comparators used by the decoder over the field $\mathbb{F}_q$. In an SC decoder, a CN requires q^2 multiplications and $q(q-1)$ additions; and a VN requires q multiplications. The normalization process requires $q-1$ additions and q divisions. In the case of an SC-MS decoder, a CN requires q^2 additions and $q(q-1)$ comparisons; and a VN requires q additions, $q-1$ comparisons, and q subtractions.

We report in Table 3.1 the complexity of a CN and a VN. Note that the subtractions are considered as additions and the divisions as multiplications.

When the optimized SC-MS decoder is considered, the complexity of the CN is reduced. In Table 3.1, we can observe that the number of additions is reduced from q^2 to qn_o and the number of comparisons is reduced from $q(q-1)$ to $q(n_o-1)$.



(a) $n_o \in \{12, 16, 20, 24, 28, 32, 64\}$ and $\mathbb{F}_{2^6}$.

(b) $n_o \in \{10, 20, 30, 40, 50, 60, 128\}$ and $\mathbb{F}_{2^7}$.

Figure 3.9: FER performance of SC-MS decoders for $(N = 64, K = 20)$ over $\mathbb{F}_q \in \{\mathbb{F}_{2^6}, \mathbb{F}_{2^7}\}$.

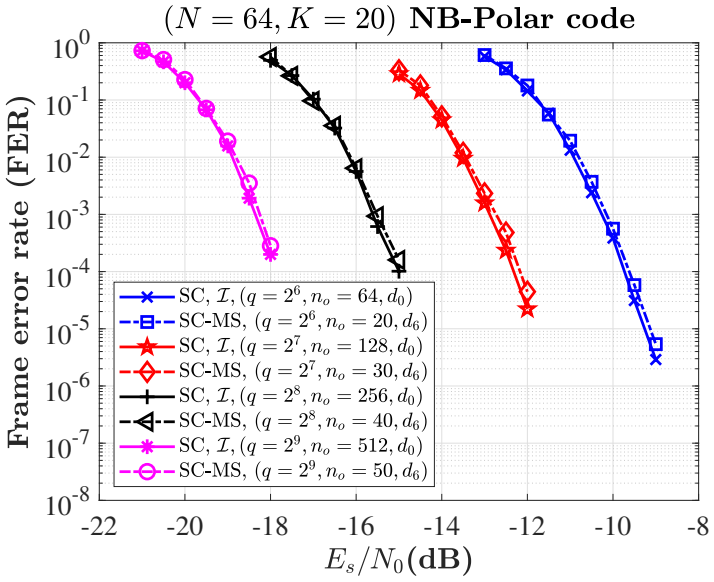


Figure 3.10: FER performance of SC-MS decoders for $(N = 64, K = 20)$ over $\mathbb{F}_q \in \{\mathbb{F}_{2^6}, \mathbb{F}_{2^7}, \mathbb{F}_{2^8}, \mathbb{F}_{2^9}\}$.

Table 3.1: Complexity of a single CN and a single VN for $\mathbb{F}_q$.

Decoder	Node	# of multipliers	# of adders	# of comparators
SC	CN	$q^2 + q$	$q^2 - 1$	-
	VN	$2q$	$q - 1$	-
SC-MS	CN	-	q^2	$q(q-1)$
	VN	-	$2q$	$q - 1$
Optimized SC-MS	CN	-	qn_o	$q(n_o - 1)$
	VN	-	$2q$	$q - 1$

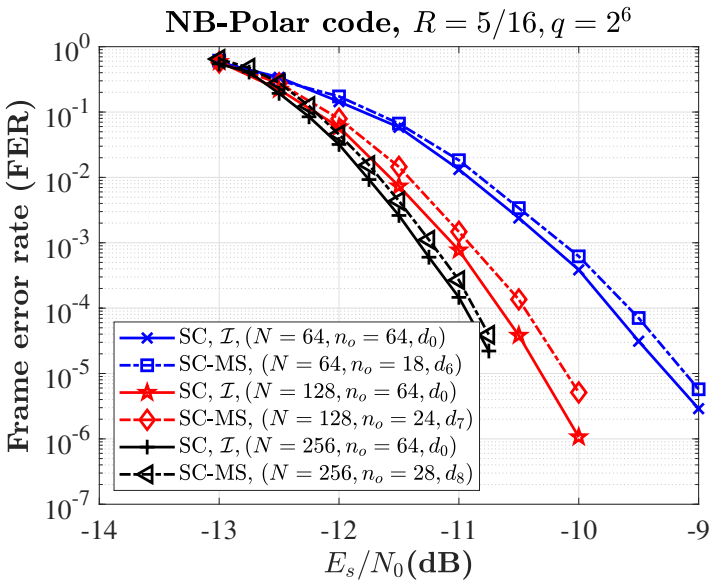


Figure 3.11: FER performance of SC-MS decoders for $N \in \{64, 128, 256\}$ and $R = 5/16$ over $\mathbb{F}_{2^6}$.

4 Finite Precision SC-MS Decoders

In this section, we present a finite precision version of the SC-MS decoder for low complexity hardware implementation.

4.1 Quantization used for SC-MS Decoders

For quantized SC-MS decoders, the LLRs have to be quantized and saturated. Let Q_{ch} denote the number of precision bits of quantized channel LLRs, and let $\mathcal{A}_{ch}$ denote the alphabet of channel LLRs defined as $\mathcal{A}_{ch} = \{0, +1, ..., +N_{ch}\}$, composed of $N_{ch} + 1$ states, where $N_{ch} = 2^{Q_{ch}} - 1$.

Let us also denote the quantizer by $\mathcal{Q} : \mathbb{R} \to \mathcal{A}_{ch}$ for the quantized SC-MS decoders, defined as

$$\mathcal{Q}(a) = \min\left(\lfloor \alpha \times a \rfloor, N_{ch}\right), \tag{4.1}$$

where $\lfloor . \rfloor$ depicts the floor function. The parameter α is called *channel gain factor* and is used to increase or decrease the amplitude of channel LLRs at the decoder input.

After computing L'_i with (3.3) and (3.4), the quantized version $I_i = (I_i(0), ..., I_i(q-1))$ of L'_i can be obtained as

$$I_i(x_i) = \mathcal{Q}\left(L'_i(x_i)\right) \quad \forall x_i \in \mathbb{F}_q. \tag{4.2}$$

The quantized channel LLR I_i is used to initialize the decoder.

Let us denote by $\mathcal{A}_L$ the alphabet of the internal LLRs defined as $\mathcal{A}_L = \{0, +1, ..., +N_m\}$, composed of $N_m + 1$ states, with $N_m = 2^{Q_m} - 1$ and where Q_m is the number of precision bits of quantized internal LLRs.

4.2 Update Rules

4.2.1 SC-MS Decoders

Let $\boldsymbol{I}^\ell = (I_0^\ell, ..., I_{N-1}^\ell)$ denote the quantized LLRs computed at stage $\ell \in \{0, ..., n\}$ during the decoding process, and let $I_i^\ell = (I_i^\ell(0), ..., I_i^\ell(q-1))$, $i = 0, ..., N-1$, denote the quantized LLR of u_i^ℓ. At stage $\ell = 0$, the input of the quantized decoder is obtained with (4.2), *i.e.* $I_i^0 = I_i(x_i)$ for $i = 0, ..., N-1$.

Considering $\mathcal{T}$ of (3.6), the update rule at a CN is given by

$$I_\theta^\ell\left(u_\theta^\ell\right) = \min_{u_\phi^\ell \in \mathbb{F}_q} \left(I_\theta^{\ell-1}\left(u_\theta^\ell \oplus u_\phi^\ell\right) + I_\phi^{\ell-1}\left(h \odot u_\phi^\ell\right)\right) \ \forall u_\theta^\ell \in \mathbb{F}_q. \tag{4.3}$$

For the case of a VN, the update rule is determined by

$$I'_\phi\left(u_\phi^\ell\right) = I_\theta^{\ell-1}\left(\hat{u}_\theta^\ell \oplus u_\phi^\ell\right) + I_\phi^{\ell-1}\left(h \odot u_\phi^\ell\right). \tag{4.4}$$

$$I_\phi^\ell\left(u_\phi^\ell\right) = \min\left(I'_\phi\left(u_\phi^\ell\right) - \min\left(I'_\phi\right), N_m\right). \tag{4.5}$$

The message $\hat{\boldsymbol{u}} = (\hat{u}_0^n, ..., \hat{u}_{N-1}^n)$ can be estimated using (4.3), (4.4), (4.5), and (3.9). The hard decision u_i^n is obtained using (3.14), where L_i^n is replaced by I_i^n for $i = 0, ..., N-1$.

The operators used in the quantized SC-MS decoder require $Q_m + 1$ bits of precision. Since it is easier to implement fixed point operations, the (Q_{ch}, Q_m)-bit SC-MS decoder reduces computational complexity. Comparing the $(64, 64)$-bit SC-MS (floating-point) with the $(2, 3)$-bit SC-MS, we can roughly obtain a 93.7% reduction in complexity. Note that we compare the number of bits of the operators, *i.e.* $100(1 - (Q_m + 1)/64)$.

4.2.2 Optimized SC-MS Decoders

Let $I^{\ell}_{\phi_O} = (I^{\ell}_{\phi_O}(\lambda^{\ell}_0), ..., I^{\ell}_{\phi_O}(\lambda^{\ell}_{n_o-1}))$ denote the quantized LLR of $L^{\ell}_{\phi_O} = (L^{\ell}_{\phi_O}(\lambda^{\ell}_0), ..., L^{\ell}_{\phi_O}(\lambda^{\ell}_{n_o-1}))$. Considering $\mathcal{T}_O$ of (3.15), the update rule at a CN can be written as

$$I^{\ell}_{\theta}\left(u^{\ell}_{\theta}\right) = \min_{\lambda^{\ell}_i \in \mathbb{F}^{\ell}_{n_o}} \left(I^{\ell-1}_{\theta}\left(u^{\ell}_{\theta} \oplus \lambda^{\ell}_i\right) + I^{\ell-1}_{\phi_O}\left(\lambda^{\ell}_i\right)\right) \ \forall u^{\ell}_{\theta} \in \mathbb{F}_q. \tag{4.6}$$

For the case of a VN, the update rule is given by

$$I'_{\phi}\left(u^{\ell}_{\phi}\right) = I^{\ell-1}_{\theta}\left(\hat{u}^{\ell}_{\theta} \oplus u^{\ell}_{\phi}\right) + I^{\ell-1}_{\phi}\left(u^{\ell}_{\phi}\right). \tag{4.7}$$

$$I^{\ell}_{\phi}\left(u^{\ell}_{\phi}\right) = \min\left(I'_{\phi}\left(u^{\ell}_{\phi}\right) - \min\left(I'_{\phi}\right), N_m\right). \tag{4.8}$$

Then, $\hat{\boldsymbol{u}} = (\hat{u}^n_0, ..., \hat{u}^n_{N-1})$ can be computed using (4.6), (4.7), (4.8), and (3.9).

4.3 Performance of Fixed-Point SC-MS Decoders

In this section, we present the simulation results of finite precision SC decoders over the field $\mathbb{F}_{2^6}$. We focus on the $(N = 64, K = 20)$ NB-Polar code and different bits of precision for channel LLRs and internal LLRs. All quantized decoders use the set $\mathcal{I}$ for the symbol positions.

The channel gain factor α used to quantize the channel LLRs are optimized using Monte Carlo simulations. In Table 4.1, we indicate the optimal values of α for (Q_{ch}, Q_m)-bit SC-MS decoders. We also report in Table 4.1 the SNR losses of the (Q_{ch}, Q_m)-bit SC-MS decoder with respect to the SC decoder.

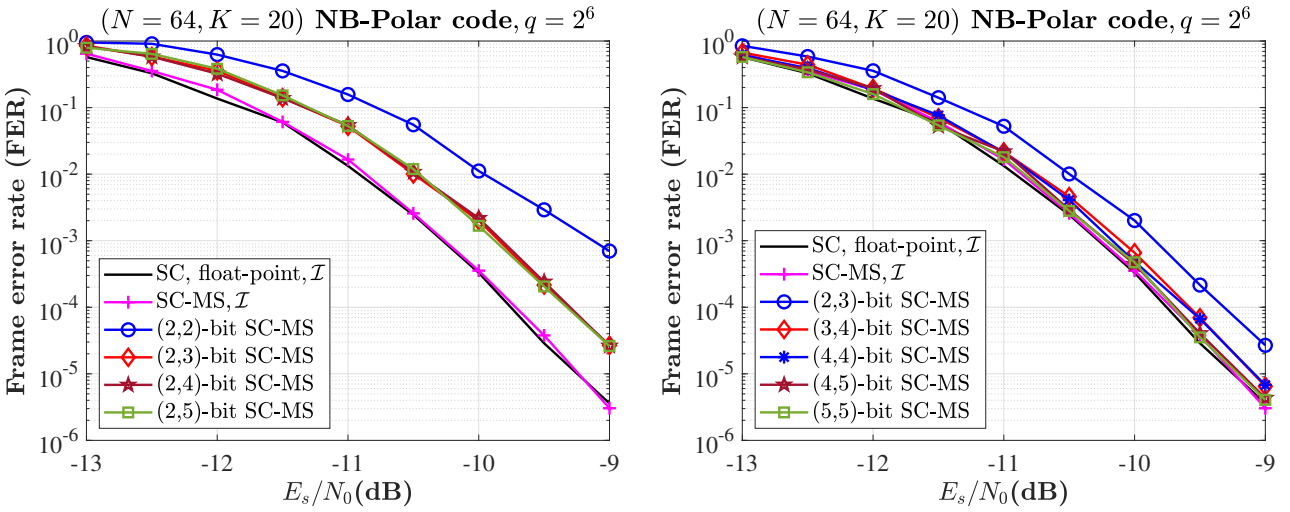
Table 4.1: SNR losses of finite precision SC-MS decoders for the $(N = 64, K = 20)$ NB-Polar code and for the field $\mathbb{F}_{2^6}$.

(Q_{ch}, Q_m)	α^*	SNR loss (dB) @ FER $= 10^{-2}$	(Q_{ch}, Q_m)	α^*	SNR loss (dB) @ FER $= 10^{-2}$
$(2, 2)$	0.40	1.00	$(3, 4)$	0.90	0.17
$(2, 3)$	0.55	0.45	$(3, 5)$	0.90	0.15
$(2, 4)$	0.55	0.45	$(4, 4)$	1.10	0.14
$(2, 5)$	0.55	0.45	$(4, 5)$	1.40	0.11
$(3, 3)$	0.60	0.38	$(5, 5)$	1.90	0.07

Fig. 4.1 depicts the FER performance of the best quantized SC-MS decoders for low precision (Q_{ch}, Q_m). We can see that the $(5, 5)$-bit SC-MS decoder has the best FER performance, and the $(3, 4)$-bit SC-MS decoder can achieve almost the same performance as the $(5, 5)$-bit SC-MS decoder.

The FER performance curves plotted in Fig. 4.2 shows the performance of the optimized (Q_{ch}, Q_m)-bit SC-MS decoders. The SNR losses of the optimized (Q_{ch}, Q_m)-bit SC-MS decoder with respect to the SC decoder are listed in Table 4.2.

From all results obtained, the $(2, 3)$-bit SC-MS decoder is a good option for applications that require low hardware complexity. In the case where the decoding performance is privileged by the applications, the $(3, 4)$-bit SC-MS decoder is a good choice for a hardware implementation.

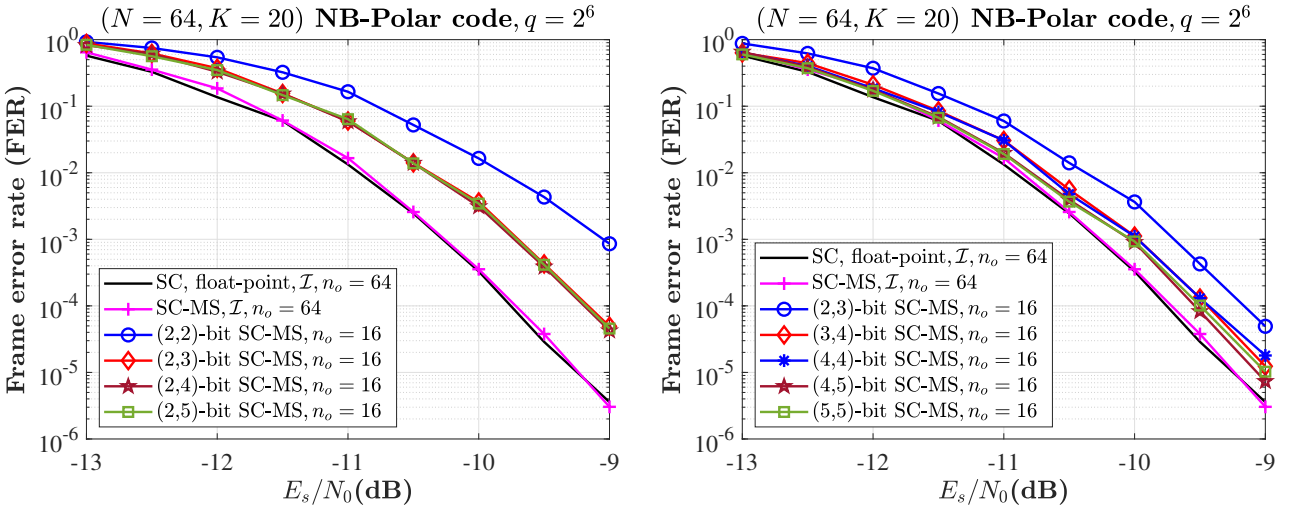


(a) FER performance. (b) FER performance.

Figure 4.1: FER performance of the (Q_{ch}, Q_m)-bit SC-MS decoders over $\mathbb{F}_{2^6}$, $N = 64$, and $K = 20$.

Table 4.2: SNR losses of optimized SC-MS decoders for the $(N = 64, K = 20)$ NB-Polar code, for the field $\mathbb{F}_{2^6}$, and for $n_o = 16$.

(Q_{ch}, Q_m)	α^*	SNR loss (dB) @ FER $= 10^{-2}$	(Q_{ch}, Q_m)	α^*	SNR loss (dB) @ FER $= 10^{-2}$
$(2, 2)$	0.45	1.00	$(3, 4)$	1.05	0.21
$(2, 3)$	0.55	0.48	$(3, 5)$	1.05	0.20
$(2, 4)$	0.60	0.48	$(4, 4)$	1.10	0.20
$(2, 5)$	0.60	0.48	$(4, 5)$	1.80	0.12
$(3, 3)$	0.63	0.40	$(5, 5)$	2.00	0.10



(a) FER performance. (b) FER performance.

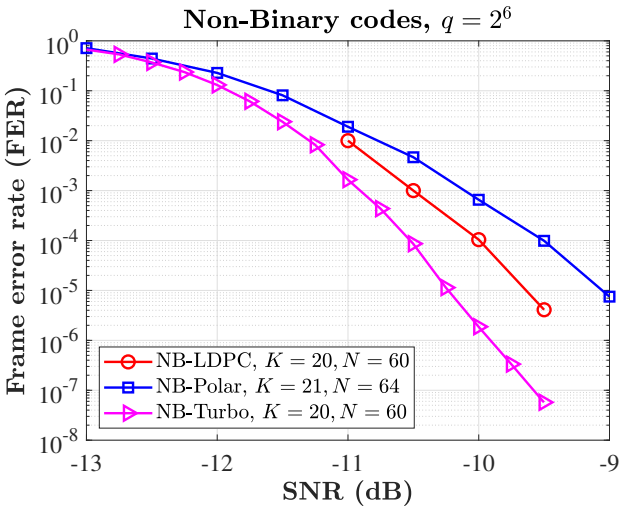
Figure 4.2: FER performance of the (Q_{ch}, Q_m)-bit SC-MS decoders over $\mathbb{F}_{2^6}$, $N = 64$, and $K = 20$.

5 Performance of Non-Binary Decoders

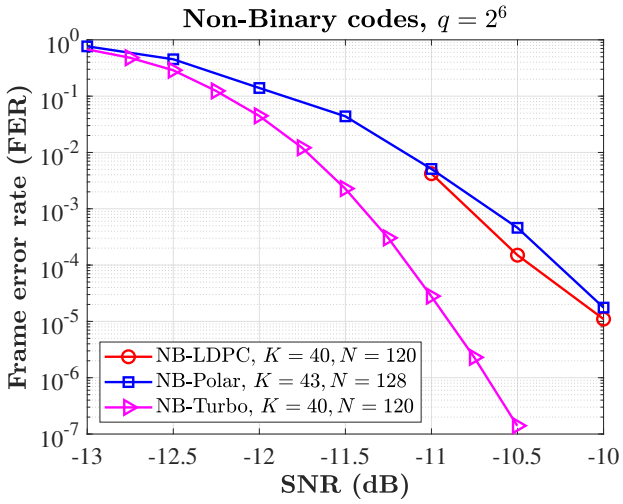
In this section, we compare the FER performance of the SC-MS decoder (NB-Polar decoder) with the NB-LDPC and NB-Turbo decoders. We consider the BI-AWGN channel, the GF $\mathbb{F}_{2^6}$, and the code rate $R \in \{1/3, 2/3\}$. We also consider $N \in \{64, 128, 256\}$ for the NB-Polar codes and $N \in \{60, 120, 240\}$ for the NB-LDPC and NB-Turbo codes.

The Monte Carlo simulations for NB-Polar, NB-LDPC, and NB-Turbo decoders with CCSK modulation are presented in Figure 5.1 and Figure 5.2. We can observe that the NB-Turbo decoder has the best performance.

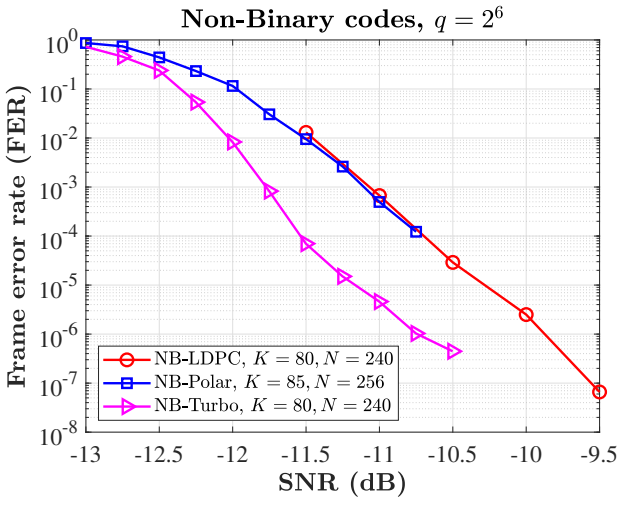
Comparing the NB-Polar and NB-LDPC decoders, both decoders have almost the same FER performance. In some cases the NB-Polar decoder has better FER performance as shown in Figure 5.2(c), and in others the NB-LDPC decoder has better performance as seen in Figure 5.1(a).



(a) $K \in \{20, 21\}$.

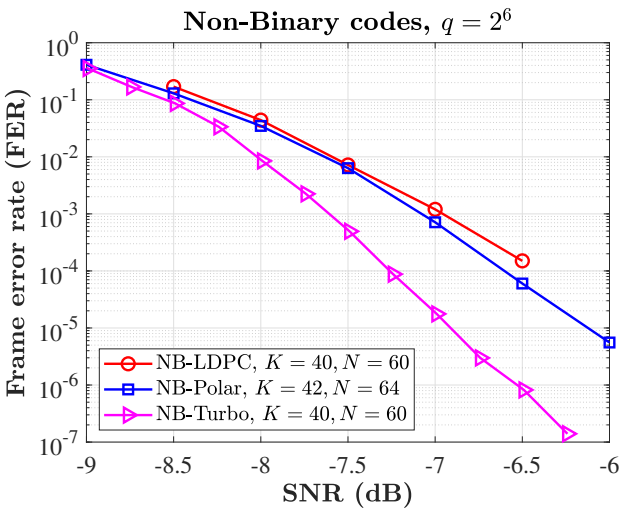


(b) $K \in \{40, 43\}$.

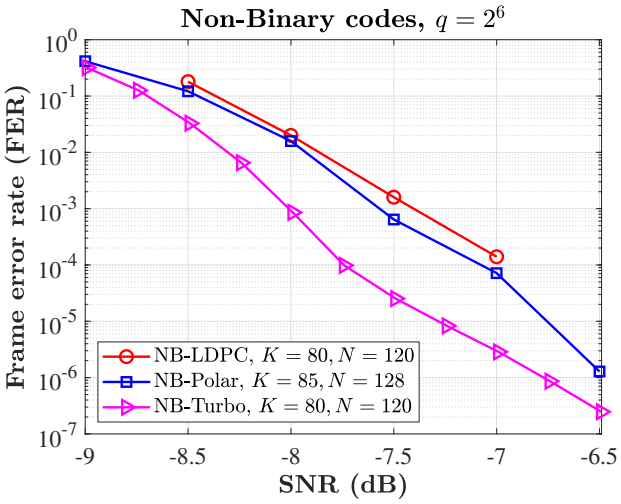


(c) $K \in \{80, 85\}$.

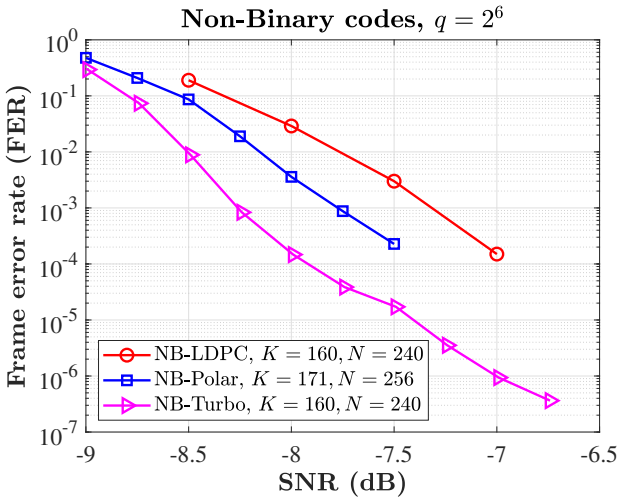
Figure 5.1: Performance comparison of non-binary codes using code rate $R = 1/3$ and field $\mathbb{F}_{2^6}$.



(a) $K \in \{40, 42\}$.



(b) $K \in \{80, 85\}$.



(c) $K \in \{160, 171\}$.

Figure 5.2: Performance comparison of non-binary codes using code rate $R = 2/3$ and field $\mathbb{F}_{2^6}$.

6 General Conclusion

In this document, we first have shown that NB-Polar codes associated with CCSK modulation can achieve very low FER at ultra-low SNR. Moreover, we have proposed a simplified version of the SC decoder. The SC-MS decoder has been defined in the LLR domain and it reduces the decoding complexity. To further reduce complexity of the SC-MS decoder, we have simplified the kernel transformation for construction of NB-Polar codes. We have also proposed a quantized version of the SC-MS decoder.

The Monte Carlo simulations have shown that the SC-MS decoders present a negligible performance degradation with respect to the SC decoders for code length $N \in \{64, 128, 256, 512, 1024\}$. In this study, we have demonstrated that the $(2, 3)$-bit SC-MS and $(3, 4)$-bit SC-MS offer a good trade-off between decoding performance and complexity.

On the other hand, the FER performance of the SC-MS decoder has been compared with the FER performance of the NB-LDPC and NB-Turbo decoders. We have shown that the NB-Turbo decoder has the best FER performance, and the SC-MS decoder and the NB-LDPC decoder have almost the same FER performance.

Bibliography

[1] V. Savin, "Min-max decoding for non binary ldpc codes," in *2008 IEEE International Symposium on Information Theory*, 2008, pp. 960–964.

[2] L. Chandesris, V. Savin, and D. Declercq, "Dynamic-scflip decoding of polar codes," *IEEE Transactions on Communications*, vol. 66, no. 6, pp. 2333–2345, 2018.

[3] E. Arikan, "Channel Polarization: A Method for Constructing Capacity-Achieving Codes for Symmetric Binary-Input Memoryless Channels," *IEEE Transactions on Information Theory*, vol. 55, no. 7, pp. 3051–3073, 2009.

[4] K. Mekki, E. Bajic, F. Chaxel, and F. Meyer, "A Comparative Study of LPWAN Technologies for Large-Scale IoT Deployment," *ICT Express*, vol. 5, no. 1, pp. 1–7, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405959517302953

[5] G. Durisi, T. Koch, and P. Popovski, "Toward Massive, Ultrareliable, and Low-Latency Wireless Communication With Short Packets," *Proceedings of the IEEE*, vol. 104, no. 9, pp. 1711–1726, 2016.

[6] A. Bana, K. F. Trillingsgaard, P. Popovski, and E. de Carvalho, "Short Packet Structure for Ultra-Reliable Machine-Type Communication: Tradeoff Between Detection and Decoding," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 6608–6612.

[7] E. Şaşoğlu, E. Telatar, and E. Arikan, "Polarization for Arbitrary Discrete Memoryless Channels," in *2009 IEEE Information Theory Workshop*, 2009, pp. 144–148.

[8] R. Mori and T. Tanaka, "Channel Polarization on q-ary Discrete Memoryless Channels by Arbitrary Kernels," in *2010 IEEE International Symposium on Information Theory*, 2010, pp. 894–898.

[9] E. Şaşoğlu, "Polar Codes for Discrete Alphabets," in *2012 IEEE International Symposium on Information Theory Proceedings*, 2012, pp. 2137–2141.

[10] W. Park and A. Barg, "Polar Codes for Q-Ary Channels, $q = 2^r$," *IEEE Transactions on Information Theory*, vol. 59, no. 2, pp. 955–969, 2013.

[11] M. Chiu, "Non-Binary Polar Codes with Channel Symbol Permutations," in *2014 International Symposium on Information Theory and its Applications*, 2014, pp. 433–437.

[12] A. Y.-C. Wong and V. C. M. Leung, "Code-Phase-Shift Keying: A Power and Bandwidth Efficient Spread Spectrum Signaling Technique for Wireless Local Area Network Applications," in *CCECE '97. Canadian Conference on Electrical and Computer Engineering. Engineering Innovation: Voyage of Discovery. Conference Proceedings*, vol. 2, 1997, pp. 478–481 vol.2.

[13] O. Abassi, L. Conde-Canencia, M. Mansour, and E. Boutillon, "Non-Binary Low-Density Parity-Check Coded Cyclic Code-Shift Keying," in *2013 IEEE Wireless Communications and Networking Conference (WCNC)*, 2013, pp. 3890–3894.

[14] L. Chandesris, "Contribution à la Construction et au Décodage des Codes Polaires," Ph.D. dissertation, Université de Cergy-Pontoise, Cergy-Pontoise, France, 2019.