entropy

MDPI

Article

How to Construct Polar Codes for Ring-LWE-Based Public Key Encryption

Jiabo Wang [1,*] and Cong Ling [2]

[1] Beijing National Research Center for Information Science and Technology, Tsinghua University, Beijing 100084, China
[2] Department of Electrical and Electronic Engineering, Imperial College London, London SW7 2AZ, UK; c.ling@imperial.ac.uk
* Correspondence: wangjiabo@mail.tsinghua.edu.cn

Abstract: There exists a natural trade-off in public key encryption (PKE) schemes based on ring learning with errors (RLWE), namely: we would like a wider error distribution to increase the security, but it comes at the cost of an increased decryption failure rate (DFR). A straightforward solution to this problem is the error-correcting code, which is commonly used in communication systems and already appears in some RLWE-based proposals. However, applying error-correcting codes to those cryptographic schemes is far from simply installing an add-on. Firstly, the residue error term derived by decryption has correlated coefficients, whereas most prevalent error-correcting codes with remarkable error tolerance assume the channel noise to be independent and memoryless. This explains why only simple error-correcting methods are used in existing RLWE-based PKE schemes. Secondly, the residue error term has correlated coefficients leaving accurate DFR estimation challenging even for uncoded plaintext. It can be found in the literature that a tighter DFR estimation can effectively create a DFR margin. Thirdly, most error-correcting codes are not well designed for safety considerations, e.g., syndrome decoding has a nonconstant time nature. A code good at error correcting might be weak under a variety of attacks. In this work, we propose a polar coding scheme for RLWE-based PKE. A relaxed "independence" assumption is used to derive an uncorrelated residue noise term, and a wireless communication strategy, outage, is used to construct polar codes. Furthermore, some knowledge about the residue noise is exploited to improve the decoding performance. With the parameterization of NewHope Round 2, the proposed scheme creates a considerable DRF margin, which gives a competitive security improvement compared to state-of-the-art benchmarks. Specifically, the security is improved by 28.8%, while a DFR of 2^{-149} is achieved a for code rate pf 0.25, $n = 1024, q = 12{,}289$, and binomial parameter $k = 55$. Moreover, polar encoding and decoding have a quasilinear complexity $O(N \log_2 N)$ and intrinsically support constant-time implementations.

Keywords: ring LWE; polar codes; public key encryption; error dependency; decryption failure rate

Citation: Wang, J.; Ling, C. How to Construct Polar Codes for Ring-LWE-Based Public Key Encryption. *Entropy* **2021**, *23*, 938. https://doi.org/10.3390/e23080938

Academic Editors: Amin Sakzad and Khoa Nguyen

Received: 15 June 2021
Accepted: 17 July 2021
Published: 23 July 2021

1. Introduction

1.1. Error-Correcting for Ring-LWE-Based Public Key Encryption

The ring LWE (RLWE) problem was firstly introduced in 2010 [1], expanding on the classical version of the problem (i.e., LWE) introduced by Regev in [2]. Key establishment mechanisms based on RLWE, for example NewHope [3], are among the most attractive postquantum proposals. Their quantum security relies on the worst-case approximate shortest independent vector problem (SIVP), and they give better efficiency compared to plain LWE because of the ring structure. One topic of pressing importance is to refine such schemes for better efficiency and security. In this work, we focus on the issue of error correcting for RLWE-based public key encryption.

The key establishment based on RLWE is differentiated into two approaches regarding how to share the secret information. One is the "reconciliation-based approach" proposed

by Ding et al. in [4] where Alice and Bob extract common information from the noisy secret with a robust extractor. However, Ding et al.'s reconciliation approach was observed to produce a biased secret, which cannot be used as a secret key. Peikert proposed another reconciliation method in [5], which directly produces a uniform secret key. In the initial version of NewHope [3], four-dimensional lattice codes and lattice quantization are used to design the reconciliation mechanism. To ease the reconciliation-based NewHope, Alkim et al. replaced the reconciliation with trivial repetition codes, which encode one bit of message into four coefficients of a polynomial in R_q. This gives rise to the second approach, i.e., the "encryption-based" approach. The encryption-based NewHope enjoys the same security properties and almost the same bandwidth requirement as the reconciliation-based one.

For most of the RLWE-based PKE schemes (e.g., NewHope, LIMA), one can enable a conversion from an indistinguishability under chosen plaintext attack (IND-CPA)-secure PKE to an indistinguishability under chosen ciphertext attack (IND-CCA)-secure scheme by applying the Fujisaki–Okamoto (FO) transform or its variants in a postquantum setting [6–8]. An obvious drawback of the FO transform is that it relies on the perfect correctness of the CPA-PKE scheme, which is not true due to the small residue noise after decryption. Although robust transforms against correctness errors were designed by Hofheinz et al. in [9], the advances of CCA attacks based on the decryption failure put current PQC candidates under threat. As a high-level description, the decryption failures in an RLWE-based PKE are somehow related to the secret information. If an adversary manages to observe enough failures and identify a correlation between failures and the secret, the security will be compromised. Therefore, the decryption failure rate (DFR) is a significant factor affecting the security level, and it should be precisely calculated. Most NIST submissions choose 2^{-128} as a target DFR.

D'Anvers et al. designed attacks exploiting the decryption failures, namely the "failure boosting" and "directional failure boosting", in [10,11]. Using these techniques, an adversary can deliberately find "weak" ciphertexts that are more likely to trigger decryption failures. These failures are used to analyze the secret statistically. These attacks are verified on some basic versions of ring-/module-LWE-/LWR-based KEMs with comparable parameterization to NIST candidates, e.g., NTRUEncrypt, KYBER, and SABER, showing that the security of these KEM schemes is impacted under the proposed attacks assuming an unlimited number of decryption queries is allowed. In addition, Guo et al. proposed a CCA attack [12] targeting another RLWE-based NIST proposal, LAC. Furthermore, this novel attack exploits the high decryption failure rate of some ciphertexts caused by a certain weight property of the secret key. Though LAC was modified to resist those attacks, it was not selected as NIST's finalist due to a variety of investigated and "hidden" attacks.

Some error-correcting codes, i.e., BCH and XEf, are used to improve the DFR in LAC and Round5, respectively [13,14]. This "unusual design" distinguishes them from other lattice-based schemes, e.g., NewHope uses repetition codes, while CRYSTALS-KYBER leaves the message uncoded. On the one hand, error-correcting codes can considerably increase the noise tolerance, improve the security, and save bandwidth. We noticed that the BCH, XEf, and repetition codes in above schemes are decoded according to hard decision metrics, e.g., the Hamming distance. We expect to see an impressive improvement if more powerful error-correcting codes (e.g., polar codes, low-density parity-check (LDPC) codes) are adopted. On the other hand, what comes with the usage of error-correcting codes is the risk of side-channel attacks based on timing/caching. Again, we take LAC as an example. Given polynomial $v' = v - u \cdot s$ after decryption, the decoding of BCH codes proceeds in three steps: (1) recovering the codewords according to a hard decision metric, (2) calculating the syndrome, and (3) locating the errors if detected and correcting them. Obviously, the *if* $\cdots$ *else* statement in the last step is not constant-time.

Attempts to adapt modern error-correcting codes (e.g., LDPC) can be found in [15]. Experimental results were given to show how much LDPC codes can improve the DFR of NewHope Simple for a reasonably large enough binomial parameter k. Furthermore, the theoretical estimation of the upper bound on DFR using error-correcting codes was

given based on an "independence" assumption claiming that the correlation between the coefficients of the residue noise $e \cdot t - s \cdot e' + e''$ is negligible. However, these were actually not, and the soft decision decoding of LDPC assumes i.i.d. channels. The dependency among the noise coefficients is obvious in the vector representation of $e \cdot t - s \cdot e' + e''$, i.e.,

$$\begin{pmatrix} e_0 & -e_{n-1} & \cdots & -e_1 \\ e_1 & e_0 & \cdots & -e_2 \\ \vdots & \vdots & \ddots & \vdots \\ e_{n-1} & e_{n-2} & \cdots & e_0 \end{pmatrix} \mathbf{t} - \begin{pmatrix} s_0 & -s_{n-1} & \cdots & -s_1 \\ s_1 & s_0 & \cdots & -s_2 \\ \vdots & \vdots & \ddots & \vdots \\ s_{n-1} & s_{n-2} & \cdots & s_0 \end{pmatrix} \mathbf{e}' + \mathbf{e}''. \quad (1)$$

We have to be careful about the "independence" assumption: the assumption will overestimate (underestimate) DFR for schemes without (with) error-correction, and therefore underestimate (overestimate) the security. This "independence" assumption was relaxed by D'Anvers et al. in [16]. Specifically, the i-th coefficient of the noise term $e \cdot t - s \cdot e' + e''$ is refined in the form of $\mathbf{c}^T\mathbf{s} + g$ where vector $\mathbf{c}$ is essentially determined by polynomials e, e', vector $\mathbf{s}$ by s, t, and scalar g by the ith coefficient of e''. They assumed $\mathbf{c}^T\mathbf{s} + g$ to be i.i.d. conditional on the l_2-norm of $\mathbf{c}$ and $\mathbf{s}$. The DFR of LAC is interpreted as a weighted DFR averaged over all possible values of $\|\mathbf{s}\|$, $\|\mathbf{c}\|$. The ternary error terms in LAC make the calculation tractable. However, for a more general ring- (module)-LWE-based encryption with error terms drawn over $\mathbb{Z}$, calculating the marginal distribution $\Pr\{\|\mathbf{s}\|\}$ and $\Pr\{\|\mathbf{c}\|\}$ is no longer trivial. In their prior work [17], they gave another assumption, namely the "Gaussian" assumption, to ease the calculation.

Song et al. interpreted NewHope as a digital communication system in [18]. At the transmitter's end, binary message $m \in \{0,1\}^{256}$ is encoded as a codeword $enc(m)$ by repeating m $n/256$ times. Then, $enc(m)$ is modulated as a vector in $\{0, \lfloor q/2 \rfloor\}^n$. At the receiver's end, upon receiving $v' = e \cdot t - s \cdot e' + e'' + \lfloor q/2 \rfloor \cdot enc(m)$, the additive threshold decoder calculates $v_i'' = \sum_{l=0}^{n/256-1} v'_{i+256l}$ for $i = 0, 1, \cdots 255$ and recovers m by hard decision decoding. To analyze the DFR, one needs to take into account two types of dependencies in the noise term: (a) the dependency between the coefficients of v' conveying the same message bit of m, i.e., v'_{i+256l} for $l = 0, 1, \cdots, n/256 - 1$; (b) the dependency between the $n/4$ coefficients of v''. In [18], v_i'' was elegantly written in the form of $v_i'' = \sum_{j=0}^{511} W_{i,j} + \sum_{l=0}^{n/256-1} n_{i+256l}$ as was the sum of 512 i.i.d. random variables $W_{i,j}$ and $n/256$ i.i.d. random variables n_{i+256l} for any fixed i. Therefore, the first-type dependency was addressed. As for the second type, Song et al. proved the error term v_i'' to be identically distributed for any $i = 0, 1, \cdots, n/4$, and therefore gave a union bound on the DFR. Consequently, a tighter upper bound on the DFR is derived, which is less than 2^{-418} for $n = 1024$ and 2^{-399} for $n = 512$ (The NewHope submission claims to have an upper bound on DFR to be 2^{-216} for $n = 1024$ and 2^{-213} for $n = 512$). The improved DFR margin enhances the security level without any changes to the original protocol.

The motivation of this work was to investigate how to handle the dependency of RLWE-based PKE and how to adapt modern error-correcting codes to it. We sought a security improvement using the derived DFR margin. A concurrent work can be found in [19], where canonical embedding was employed to derive i.i.d. fading channels with channel state information (CSI) available to the recipient and polar codes were constructed. However, in reality, we do not expect to engagein canonical embedding because we can: (a) spare ourselves the trouble of switching between the canonical and polynomial representation; (b) avoid the error tolerance loss due to the tailored constellation diagram as [19] illustrated; (c) make the overall scheme comply with the most popular and practical RLWE-based PKE framework where we only deal with integers on the interval $[0, q)$.

1.2. Contribution

The contribution of this paper is as follows.

1. We formulated the RLWE-based PKE as an i.i.d. mod $2\mathbb{Z}$ additive Gaussian noise channel with channel state information (CSI) available to the receiver under a relaxed "independence" assumption;
   (a) Given the residue noise term $e \cdot t - s \cdot e' + e''$, we formulated the RLWE-based PKE as a mod $2\mathbb{Z}$ additive Gaussian noise channel within exactly one code block. We assumed the mod $2\mathbb{Z}$ additive Gaussian channel to be independent under a relaxed assumption compared to the one in [15];
   (b) Alice, the decoder, can considerably improve the DFR by exploiting the advantage that the polynomials e and s are generated on her side and she can figure out the precise distribution of the Gaussian noise;
2. We employed a telecommunication-engineering strategy, namely outage, to construct polar codes for RLWE-based PKE. The encoding and decoding routines allow quasi-linear (i.e., $(N \log_2 N)$) and constant-time implementations. Experimental results and theoretical estimation of DFR are also given. Specifically, we derived a new DFR of 2^{-149} by SC decoding for NewHope parameters $q = 12{,}289$, $n = 1024$ and code rate = 0.25 and a larger central binomial parameter $k = 55$. The DFR margin enabled us to improve the security by 28.8% while keeping the target DFR of 2^{-140} (as is the benchmark in the work of [15,18]) achievable.

1.3. Roadmap

This paper is organized as follows. A review of the ring-LWE-based public key encryption and some basics of channel models and polar codes can be found in Section 2. The problem formulation and methodology are introduced in Section 3. In Section 3.1, we explain how to formulate a typical RLWE-based PKE scheme as a mod $2\mathbb{Z}$ channel with additive Gaussian noise. A relaxed "independence" assumption is used to derive i.i.d. channels. We explain the soundness of the proposed scheme in Section 3.2 and demonstrate how to construct and decode polar codes explicitly in Section 3.3. In Section 4, we analyze the DFR theoretically and experimentally when polar decoding (SC decoding) is applied. We, in Section 5, discuss the security improvement, the constant-time implementation, and communication overhead increase by polar encoding and decoding. We conclude this paper in Section 6.

2. Preliminaries

2.1. Ring-LWE Public Key Encryption Scheme

The public key encryption scheme based on ring-LWE was first described in [20] and formally defined in a subsequent work [21]. We use the "informal" definition of ring-LWE given in [20], as it then became the most prevalent version in implementations, e.g., NewHope [22] and Peikert's KEM [5]. The scheme is parameterized by an integer modulus q, dimension n, a power of two, and a ring of integers $R := \frac{\mathbb{Z}[X]}{x^n + 1}$ and its quotient ring $R_q := R/qR$. We define an error distribution χ over R. We take the example of NewHope and define sampling from χ to be sampling each coefficient of a polynomial in R from a discrete Gaussian over $\mathbb{Z}$. The scheme proceeds as follows:

- Alice firstly samples $a \in R_q$ uniformly at random, then she samples a secret key s together with an error e according to χ. She publishes as the public key a ring-LWE sample $(a, b) = (a, a \cdot s + e \mod q) \in R_q \times R_q$;
- Bob encrypts a message $m \in \{0,1\}^n$ as $(c_1, c_2) = (a \cdot t + e' \mod q, b \cdot t + e'' + \lfloor \frac{q}{2} \rfloor \cdot m \mod q)$, where e', e'', t are sampled independently from χ;
- Alice decrypts using s by computing $d := c_2 - c_1 \cdot s = \lfloor \frac{q}{2} \rfloor \cdot m + e \cdot t - s \cdot e' + e''$.

Alice then recovers the message m by decoding: if the ith coordinate of d is closer to zero than $\lfloor q/2 \rfloor$, Alice assumes the ith coordinate of m was zero, otherwise she assumes it was one. We observe a few key facts about this scheme that we need for our work. Firstly, although its formal security proof may be found in [21], the main idea is that b, c_1, and c_2 leak no information about the secret s and the plaintext m because they

are ring-LWE samples, which are assumed to be pseudorandom by the hardness of the ring-LWE decision problem. Therefore, one could alternate the encoding term $\lfloor \frac{q}{2} \rfloor \cdot m$ without affecting security, as long as the encoding is independent of the actualization of the variables s, e, e', e'', t. We use this fact implicitly while constructing polar codes in the sequel. Secondly, we observe that Alice knows the actualization of s and e, and so may use these for decoding.

2.2. Channel Models

In wireless communications, the additive white Gaussian noise (AWGN) channel is the most primary and frequently used model to characterize how noises interfere with the channel input. A typical discrete-time AWGN channel is defined as:

$$y_i = x_i + z_i, \ \ i = 1, \cdots, N,$$

where $x_i \in \mathbb{R}$ is the channel input, $y_i \in \mathbb{R}$ is the channel output, and z_i is an additive white Gaussian noise, and there are N time slots in total. Ideally, these variables are independent in different time slots indicated by subscript i. A fading channel arises due to a time-varying attenuation of signal quality caused by either the propagation environment or by the movement of the transmitter/receiver. We consider a fading channel model W as:

$$y_i = h_i x_i + z_i, \quad i = 1, \cdots, N,$$

where h_i is the channel gain and z_i is additive white Gaussian noise. Denote by T_c the coherence interval of a fading channel W. In the context of a fading channel with memory, the channel gain h_i is believed to be a constant within one coherence interval and varies independently as the next coherence interval approaches. The realization of h_i is called channel state information (CSI), and the distribution of h_i is called channel distribution information (CDI). In the special case of $T_c = 1$, channel W is referred to as an identically independently distributed (i.i.d.) fading channel. The design and performance of error-correcting codes for i.i.d. fading channels with/without CSI is well studied [23–27].

How to design x_i to reliably transmit information at the highest rate via a specific channel has been widely and comprehensively studied over the past decades. A branch of this study is to construct capacity-achieving lattice codes for an AWGN channel and its fading variants [28–31]. At the transmitter's end, lattice coding maps binary codes to a constellation diagram in Euclidean space, called lattice modulation. At the recipient's end, the decoder recovers the binary codes by the bounded distance decoding or preferably maximum likelihood decoding for better performance. This leads to the definition of mod Λ channel and Λ/Λ' channel where Λ is a lattice and Λ' is a sublattice of Λ. We omit the formal definition here, but give an example of a mod $\mathbb{Z}$ channel and a $\mathbb{Z}/2\mathbb{Z}$ channel, which will be used in Section 3.1.

Example 1. *A mod $\mathbb{Z}$ channel is an additive white Gaussian noise (AWGN) channel with input restricted to $a \in \mathcal{V}(\mathbb{Z})$ where $\mathcal{V}(\mathbb{Z})$ is the fundamental region (A fundamental region of a lattice Λ is a region that includes one and only one point from each coset of Λ in $\mathbb{R}^n$. Algebraically, $\mathcal{V}(\Lambda)$ is a set of coset representatives for all the cosets of Λ in $\mathbb{R}^n$, e.g., we can define $\mathcal{V}(\mathbb{Z})$ to be $[0,1)$, but not necessarily to be the fundamental Voronoi cell $[-0.5, 0.5)$.) of $\mathbb{Z}$. At the receiver's end, there is a mod $\mathcal{V}(\mathbb{Z})$ operation giving the equivalent channel output as:*

$$y = a + n \ mod \ \mathbb{Z} = (a + n') \ mod \ \mathbb{Z},$$

where n is the AWGN noise and $n' = n \ mod \ \mathbb{Z}$.

Example 2. *A $\mathbb{Z}/2\mathbb{Z}$ channel is an AWGN channel with input restricted to $r \in (\mathbb{Z} + a) \cap \mathcal{V}(2\mathbb{Z})$ for some offset $a \in \mathbb{R}$. At the receiver's end, the equivalent channel output is:*

$$y = r + n \ mod \ 2\mathbb{Z} = r + n' \ mod \ 2\mathbb{Z}.$$

It can be viewed as a mod $2\mathbb{Z}$ *channel with input restricted to a set of elements of* $\mathbb{Z}+a$ *that fall in* $\mathcal{V}(2\mathbb{Z})$.

2.3. Polar Codes for BDMS Channels

Polar codes, introduced by Arıkan in [32], are linear block codes of length $N=2^n$ for a positive integer n that achieves the capacity of any binary input discrete memoryless symmetric (BDMS) channels asymptotically (In fact, the generalizations of polar codes are extended to arbitrary code length and a large class of channels.). We firstly recall some basics of polar coding for a BDMS channel. Given a BDMS channel W, there are two commonly used metrics in information theory to measure the quality of W: the mutual information (The maximum mutual information over all possible channel input distributions is the channel capacity.) and the reliability.

Definition 1 (Mutual information of BDMS channels)**.** *The mutual information* $I(W)$ *of a channel* W *is the maximum rate at which information can be successfully transmitted from the transmitter to the receiver. For a BDMS channel* $W:\mathcal{X}\to\mathcal{Y}$, $I(W)\in[0,1]$ *is defined as:*

$$I(W)\triangleq\sum_{y\in\mathcal{Y}}\sum_{x\in\mathcal{X}}\frac{1}{2}W(y|x)\log_2\frac{W(y|x)}{\frac{1}{2}W(y|0)+\frac{1}{2}W(y|1)}.$$

Here, we use the definition of symmetric mutual information assuming a uniform channel input, which is also the capacity of the BDMS channel. We use the notations $I(W)$ and $I(Y;X)$ interchangeably to denote the mutual information of W.

Definition 2 (Bhattacharyya parameter of BDMS channels)**.** *The Bhattacharyya parameter* $Z(W)$ *is a measure of channel reliability. For a BDMS channel* W, $Z(W)\in[0,1]$ *is defined as:*

$$Z(W)\triangleq\sum_{y\in\mathcal{Y}}\sqrt{W(y|0)W(y|1)}.$$

A small $Z(W)$ *indicates a more reliable channel, while a large* $Z(W)$ *implies a channel with more inferences.*

The capacity-achieving nature of polar codes arises from the so-called channel polarization phenomenon as a result of recursive applications of Arıkan's transform to identical Ws and their synthesized derivatives. The overall recursive transform can be performed in a channel-combining phase and a channel-splitting phase. In the channel-combining phase, a linear transformation defined as $X^{1:N}=U^{1:N}G_N$ is performed on a vector $U^{1:N}\in\mathcal{X}^{1:N}$ over $GF(2)$, where $G_N=B_N\begin{bmatrix}1&0\\1&1\end{bmatrix}^{\otimes n}$. B_N is a permutation matrix: if $U'^{1:N}=U^{1:N}B_N$ and $n=\log_2 N$, the $i'=((b_n,\cdots,b_2,b_1)_2+1)$-th coordinate of $U'^{1:N}$ is the $i=((b_1,b_2,\cdots,b_n)_2+1)$-th coordinate of $U^{1:N}$. By taking $X^{1:N}$ as the raw input of W, one derives a combined channel $W_N:\mathcal{X}^{1:N}\to\mathcal{Y}^{1:N}$ with a transition probability of:

$$W_N(y^{1:N}|u^{1:N})=\prod_{i\in\{1,\cdots,N\}}W(y^{(i)}|x^{(i)}=(u^{1:N}G_N)_i),\tag{2}$$

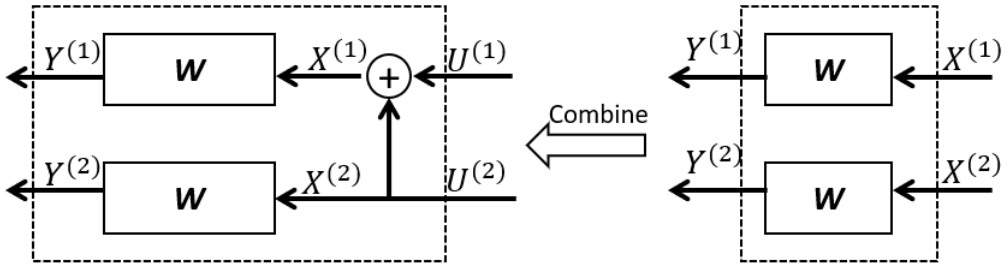
where $(\cdot)_i$ denotes i-th coordinate. Since G_N induces a one-to-one mapping between $U^{1:N}$ and $X^{1:N}$, the mutual information of W_N is:

$$I(W_N)=I(Y^{1:N};U^{1:N})=NI(W).\tag{3}$$

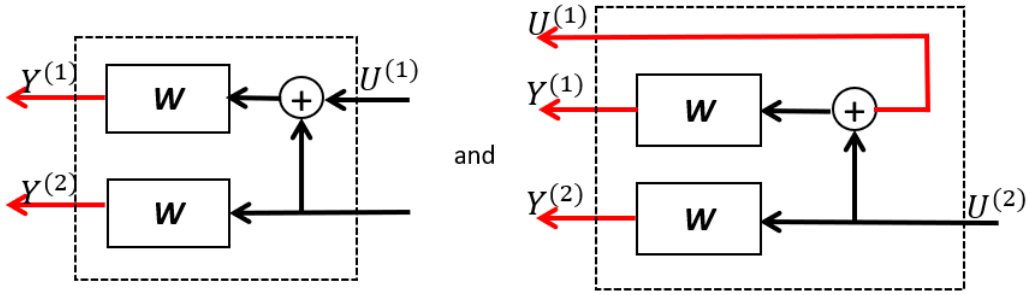
In the channel-splitting phase, W_N is further split back into N synthesized channels $W_N^{(i)}$: $\mathcal{X} \to \mathcal{Y}^N \times \mathcal{X}^{i-1}$ whose transition probability is defined by:

$$W_N^{(i)}(y^{1:N}, u^{1:i-1}|u^{(i)}) = \sum_{U^{i+1:N} \in \mathcal{X}^{N-i}} \frac{1}{2^{N-1}} W_N(Y^{1:N}|U^{1:N}). \tag{4}$$

We now demonstrate how to perform Arıkan's transform. We begin with the transform on two i.i.d. BDMS channels $W : \{0,1\} \to \mathcal{Y}$ as shown in Figure 1. Let $X^{1:2} = (X^{(1)}, X^{(2)}) \in \{0,1\}^2$ be the raw input vector of two W and $X^{1:2} = (Y^{(1)}, Y^{(2)}) \in \mathcal{Y}^2$ be the raw channel output vector. Denote by $U^{1:2} = (X^{(1)}, X^{(2)}) \in \{0,1\}^2$ the message vector. The symbol $\oplus$ indicates a mod-2 operation.



(**a**) Channel combining



(**b**) Channel splitting

Figure 1. An example of channel combining and splitting for $N = 2$.

At the channel-combining stage, the message vector $U^{1:2}$ is transformed into $X^{1:2} = U^{1:2}G_2 \mod 2$ where $G_2 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$. The two parallel Ws are seen as a combination channel $W_2 : \{0,1\}^2 \to \mathcal{Y}^2$. Since there exists a bijection between $U^{1:2}$ and $X^{1:2}$, the transition probability of W_2 is:

$$W_2(y^{1:2}|u^{1:2}) = W(y^{(1)}|u^{(1)} \oplus u^{(2)})W(y^{(2)}|u^{(2)}).$$

The channel capacity of W_2 and W satisfies:

$$I(W_2) = 2I(W). \tag{5}$$

At the channel-splitting stage, the combination channel W_2 is split into two synthesized channels $W_2^{(1)} : \{0,1\} \to \mathcal{Y}^2$ and $W_2^{(2)} : \{0,1\} \to \mathcal{Y}^2 \times \{0,1\}$. To be specific, channel $W_2^{(1)}$ takes $U^{(1)}$ as the only input and gives $Y^{(1)}, Y^{(2)}$ as the output. As for channel $W_2^{(2)}$, it

takes $U^{(2)}$ as the only channel input and gives $Y^{(1)}, Y^{(2)}$ and $U^{(1)}$ as the channel output. The channel transition probabilities of $W_2^{(1)}$ and $W_2^{(2)}$ are:

$$\begin{aligned} W_2^{(1)}(y^{1:2}|u^{(1)}) &= \sum_{u^{(2)}\in\{0,1\}} \frac{W_2(y^{1:2}|u^{1:2})\cdot P(u^{(1:2)})}{P(u^{(1)})} \\ &\overset{(a)}{=} \frac{1}{2}\sum_{u^{(2)}\in\{0,1\}} W(y^{(1)}|u^{(1)}\oplus u^{(2)})W(y^{(2)}|u^{(2)}) \end{aligned} \tag{6}$$

and:

$$\begin{aligned} W_2^{(2)}(y^{1:2},u^{(1)}|u^{(2)}) &= \frac{W_2(y^{1:2}|u^{1:2})\cdot P(u^{(1:2)})}{P(u^{(2)})} \\ &\overset{(b)}{=} \frac{1}{2}W(y^{(1)}|u^{(1)}\oplus u^{(2)})W(y^{(2)}|u^{(2)}). \end{aligned} \tag{7}$$

Note that the equalities (a) (b) are derived because $U^{(1)}$, $U^{(2)}$ are i.i.d. and they are uniformly distributed over $\{0,1\}$. More generally, a proposition follows to show the relation between $(W_N^{(i)}, W_N^{(i)})$ and $(W_{2N}^{(2i-1)}, W_{2N}^{(2i)})$.

Proposition 1 ([32]). *For $i = 1, \cdots, N$,*

$$\begin{aligned} &W_{2N}^{(2i-1)}(y^{1:2N}, u^{1:2i-2}|u^{(2i-1)}) \\ &= \frac{1}{2}\sum_{u^{(2i)}} W_N^{(i)}(y^{1:N}, u_o^{1:2i-2}\oplus u_e^{1:2i-2}|u^{(2i-1)}\oplus u^{(2i)})\cdot W_N^{(i)}(y^{N+1:2N}, u_e^{1:2i-2}|u^{(2i)}) \end{aligned} \tag{8}$$

$$\begin{aligned} &W_{2N}^{(2i)}(y^{1:2N}, u^{1:2i-1}|u^{(2i)}) \\ &= \frac{1}{2}W_N^{(i)}(y^{1:N}, u_o^{1:2i-2}\oplus u_e^{1:2i-2}|u^{(2i-1)}\oplus u^{(2i)})\cdot W_N^{(i)}(y^{N+1:2N}, u_e^{1:2i-2}|u^{(2i)}), \end{aligned} \tag{9}$$

where $u_o^{1:2i-2}$ and $u_e^{1:2i-2}$ indicate a subvector of $u^{1:2i-2}$ of odd and even indices, respectively.

It was proven in [32] that Arıkan's transform preserves the mutual information in the sense that:

$$I(W_N) = NI(W) = \sum_{i\in\{1,\cdots,N\}} I(W_N^{(i)}).$$

More importantly, the quality of the synthesized channels polarizes asymptotically as the recursion proceeds.

Theorem 1 (Channel polarization of mutual information [32]). *For any BDMS channel W, the synthesized channels $W_N^{(i)}$ polarize in the sense that, for any fixed $\delta \in (0,1)$, as N goes to infinity through powers of two, the fraction of indices $i \in \{1,\cdots,N\}$ for which $I(W_N^{(i)}) \in (1-\delta, 1]$ goes to $I(W)$ and the fraction for which $I(W_N^{(i)}) \in [0,\delta)$ goes to $1 - I(W)$.*

The channel polarization theorem from above can also be stated in the metric of the Bhattacharyya parameter by replacing $I(W_N^{(i)})$ by $Z(W_N^{(i)})$.

For any desired transmission rate $R < I(W)$, we can partition $\{1,\cdots,N\}$ into a subset $\mathcal{A}$ and its complement $\mathcal{A}^{\mathcal{C}}$ such that (i) $|\mathcal{A}| = \lfloor NR \rfloor$ and (ii) for any $i \in \mathcal{A}$ and $j \in \mathcal{A}^{\mathcal{C}}$, $Z(W_N^{(i)}) \leq Z(W_N^{(j)})$. Denote by $G_N(\mathcal{A})$ (resp. $G_N(\mathcal{A}^{\mathcal{C}})$) the rows of G_N indexed by $\mathcal{A}$ (resp.

$\mathcal{A}^C$). Given the most reliable $\lfloor NR \rfloor$ channels indexed by $\mathcal{A}$, one can construct polar codes following the encoding rule:

$$X^{1:N} = U_{\mathcal{A}} G_N(\mathcal{A}) \oplus U_{\mathcal{A}^C} G_N(\mathcal{A}^C), \tag{10}$$

where $U_{\mathcal{A}}$ is the useful information vector of length $\lfloor NR \rfloor$ and $U_{\mathcal{A}^C}$ is a predetermined vector, named frozen bits, known to both the encoder and decoder, e.g., $U_{\mathcal{A}^C} = \mathbf{0}$. In this manner, the useful information is transmitted via the most reliable synthesized channels. A question may arise about how to efficiently calculate $Z(W_N^{(i)})$. A brief review can be found in Sections 2.4 and 3.3. As a high-level description, calculating $Z(W_N^{(i)})$ according to Definition 2 for a BDMS channel with a large or even continuous output alphabet is not easy because the output alphabet of the synthesized channel $W_N^{(i)}$ increases exponentially with a factor of $\log_2 N$. One solution to handle this problem is to firstly construct an approximate channel W' of W using a degrading/upgrading technique such that W' has a countable output alphabet of a size no greater than μ and only minor and traceable capacity loss [33]. Then, one applies Arıkan's transform recursively to W', deriving synthesized channels as Proposition 1 indicates. At each recursion, one applies a merging technique to approximate the synthesized channels such that the approximation is stochastically degraded with the original one and has an output alphabet no greater than a predetermined value (e.g., ν) [34]. In this way, one can finally derive an approximation of $W_N^{(i)}$ with an output alphabet no larger than ν and negligible capacity difference. Now, one is able carry out the encoding as in Formula (10).

The successive cancellation (SC) decoder is the initial decoding algorithm for polar codes. It gives an estimation of $u^{(i)}$, the i-th coordinate of $U^{1:N}$, in the natural order of i. Given a polar code parameterized by code length N, information set $\mathcal{A}$, and frozen bits $U_{\mathcal{A}^C}$, one can derive the recovered message $\bar{u}^{(i)}$ of $u^{(i)}$ in sequential order of index i according to the decoding rule specified as:

$$\bar{u}^{(i)} = \begin{cases} u^{(i)} & i \in \mathcal{A}^C, \\ 0 & L_N^{(i)}(y^{1:N}, \bar{u}^{1:i-1}) \geq 1 \text{ and } i \in \mathcal{A}, \\ 1 & \text{otherwise}, \end{cases} \tag{11}$$

where $\bar{u}^{1:i-1}$ is the estimation of $u^{1:i-1}$ recovered before $\bar{u}^{(i)}$ and $L_N^{(i)}(y^{1:N}, \bar{u}^{1:i-1})$ is the likelihood ratio function defined as:

$$L_N^{(i)}(y^{1:N}, \bar{u}^{1:i-1}) = \frac{W_N^{(i)}(y^{1:N}, \bar{u}^{1:i-1} | u^{(i)} = 0)}{W_N^{(i)}(y^{1:N}, \bar{u}^{1:i-1} | u^{(i)} = 1)}.$$

The computational complexity of SC decoding, as is dominated by the recursive calculation of $L_N^{(i)}$ (see Appendix A), is $O(N \log_2 N)$.

Denote by P_e the average probability of block decoding error. As a result of polar encoding and SC decoding, it was proven in [32] that P_e is upper bounded as follows.

Theorem 2 (Decoding performance [32])**.** *For any BDMS channel W and any choices of parameter $(N, R, \mathcal{A})$,*

$$P_e \leq \sum_{i \in \mathcal{A}} Z(W_N^{(i)}).$$

2.4. Channel Degradation and Upgradation

The construction of polar codes can be addressed if all the Bhattacharyya parameters of synthesized channels can be efficiently calculated. In [32], an efficient solution to compute

$Z(W_N^{(i)})$ for binary erasure channels (BEC) was given, while it was suggested to use the Monte Carlo method to deal with more general BDMS channels. R. Mori and T. Tanaka made an attempt to solve this problem for arbitrary binary input memoryless symmetric (BMS) channels using the density evolution [35–37] of belief propagation (BP) decoding. However, they also mentioned that it was unclear how to handle the computational efficiency when the code length N was large and the requirement for precision was high. In [33], a quantization method was proposed to construct a degraded and upgraded approximation of a general BMS channel. If the degraded or upgraded relation exists, one can approximate $Z(W_N^{(i)})$ efficiently.

Definition 3 (Degraded and upgraded channel [33]). *A channel $\mathcal{Q}: \mathcal{X} \to \mathcal{Z}$ is (stochastically) degraded with respect to a channel $\mathcal{W}: \mathcal{X} \to \mathcal{Y}$ if there exists a channel $\mathcal{P}: \mathcal{Y} \to \mathcal{Z}$ such that:*

$$\mathcal{Q}(z|x) = \sum_{y \in \mathcal{Y}} \mathcal{W}(y|x)\mathcal{P}(z|y)$$

for all $z \in \mathcal{Z}$ and $x \in \mathcal{X}$. We denote by $\mathcal{Q} \preceq \mathcal{W}$ the relation that $\mathcal{Q}$ is degraded with respect to $\mathcal{W}$. Conversely, we denote by $\mathcal{Q}' \succeq \mathcal{W}$ the relation that $\mathcal{Q}'$ is upgraded with respect to $\mathcal{W}$ if there exists a channel $\mathcal{Q}': \mathcal{X} \to \mathcal{Z}'$ and a channel $\mathcal{P}: \mathcal{Z}' \to \mathcal{Y}$ such that:

$$\mathcal{W}(y|x) = \sum_{z' \in \mathcal{Z}'} \mathcal{Q}'(z'|x)\mathcal{P}(y|z')$$

for $y \in \mathcal{Y}$ and $x \in \mathcal{X}$.

Moreover, the synthesized channels of $\mathcal{Q}, \mathcal{W}, \mathcal{Q}'$ under Arıkan's transform also fulfill the channel degradation and upgradation relation.

Lemma 1 (Restatement of Lemma 4.7 in [38]). *Given BMS channels $\mathcal{W}, \mathcal{Q}$, and $\mathcal{Q}'$, we denote by $\mathcal{W}_N^{(i)}$, $\mathcal{Q}_N^{(i)}$, and $\mathcal{Q}'^{(i)}_N$ for $i \in [1, N]$ the synthesized channels obtained by Arıkan's transformation. If $\mathcal{Q}' \succeq \mathcal{W} \succeq \mathcal{Q}$ for all i, then $\mathcal{Q}'^{(i)}_N \succeq \mathcal{W}_N^{(i)} \succeq \mathcal{Q}_N^{(i)}$.*

If the channel degradation or upgradation relation is set up, their channel capacity, reliability, and error probability will be related as follows.

Lemma 2 ([33]). *Let $\mathcal{W}$ be a BMS channel, and suppose there exists another channel $\mathcal{Q}$ such that $\mathcal{Q} \preceq \mathcal{W}$. Then:*

$$\begin{aligned} C(\mathrm{Q}) &\leq C(\mathcal{W}), \\ Z(\mathrm{Q}) &\geq Z(\mathcal{W}), \\ P_e(\mathrm{Q}) &\geq P_e(\mathcal{W}). \end{aligned}$$

The inequality will reverse if we replace "degraded" by "upgraded".

3. Materials and Methods

3.1. RLWE-Based PKE Channel Model with Outage

In the field of telecommunication, a signal outage occurs if the signal power at the receiver's end falls below a threshold, which is related to the minimum signal-to-noise ratio (SNR) acceptable to the communication performance. The outage probability is defined as the probability with which signal outage occurs. The analysis of outage probability is of great importance to estimate fading capacities in a fading environment. A typical example is the outage estimation for fading multiple-input and multiple-output (MIMO) channels [39,40].

We already gave an RLWE-based PKE instance in Section 2. We now consider the problem of decoding the message m given the polynomial:

$$y = \lfloor \frac{q}{2} \rfloor \cdot m + e \cdot t - s \cdot e' + e'' \mod R_q, \tag{12}$$

where $e \cdot t$ and $s \cdot e'$ are polynomial multiplications in $\mathbb{Z}[x]/(1+x^n)$. It can be written in vector form as:

$$\lfloor \frac{q}{2} \rfloor \mathbf{m} + \begin{pmatrix} e_0 & -e_{n-1} & \cdots & -e_1 \\ e_1 & e_0 & \cdots & -e_2 \\ \vdots & \vdots & \ddots & \vdots \\ e_{n-1} & e_{n-2} & \cdots & e_0 \end{pmatrix} \mathbf{t} - \begin{pmatrix} s_0 & -s_{n-1} & \cdots & -s_1 \\ s_1 & s_0 & \cdots & -s_2 \\ \vdots & \vdots & \ddots & \vdots \\ s_{n-1} & s_{n-2} & \cdots & s_0 \end{pmatrix} \mathbf{e}' + \mathbf{e}'' \mod q. \tag{13}$$

Since the receiver knows matrices $\mathbf{E}, \mathbf{S}$ and we observe that the norm of each row of $\mathbf{E}, \mathbf{S}$ stays the same within one code block, the channel model of RLWE-based PKE can be described in a fading channel form as:

$$Y_i = \lfloor \frac{q}{2} \rfloor m_i + H * Z_i, \mod q, \ i = 1, \cdots n, \tag{14}$$

where $m_i \in \{0,1\}$, $Z_i \leftarrow \mathcal{N}(0, r^2)$ and the channel gain H is $H = \sqrt{1 + \sum_1^n e_i^2 + \sum_1^n s_i^2}$ where e_i and s_i are coefficients of polynomials e and s for $i \in [n]$, respectively. Note that we assume the error distribution χ to be a normal distribution $\mathcal{N}(0, r^2)$ for the convenience of analysis. A similar setting can be found in [41] where χ is defined on $\mathbb{R}/[0, q)$.

Independence assumption: Taking a close look at the channel model in Formula (14), we derive a group of n identically distributed channels rather than i.i.d. channels because every Z_i is related to every coordinate of $\mathbf{t}$ and $\mathbf{e}'$. To apply polar codes to the encoding and decoding step, we assume that the correlation between the Z_is are negligible and will not affect the decoding performance, as is a common assumption when applying modern error-correcting codes to RLWE-based PKE [15].

Now, we denote by $\epsilon \in (0,1)$ the outage probability and denote by H_ϵ the threshold such that $\Pr\{H > H_\epsilon\} = \epsilon$. Unlike in a telecommunication system where the uncertainty of channel gain would introduce difficulties in estimating the outage probability, in our RLWE channel, how the fading behaves is clearly known to the receiver. In the RLWE-based PKE instance in Section 2, both participants of the PKE process know the distribution of H. Moreover, Alice, who plays the role of the receiver in telecommunication, precisely knows the value of H, i.e., the channel state information. Examples of how H_ϵ is defined can be seen in Figure 2 where $\epsilon = 0.01$ and r is the parameter of normal distribution $\mathcal{N}(0, r^2)$.

The revised public key encryption proceeds as follows:

- The key generation step is the same as the RLWE-based PKE instance in Section 2;
- At the encryption step, Bob takes the RLWE channel as a mod $2\mathbb{Z}$ additive Gaussian channel (To be precise, it is a $\lfloor \frac{q}{2} \rfloor \mathbb{Z}/q\mathbb{Z}$ channel with additive Gaussian noise $\mathcal{N}(0, r^2H^2)$ or, equivalently, a $\mathbb{Z}/2\mathbb{Z}$ channel. To ease the notation, we instead use the mod $2\mathbb{Z}$ channel with input restricted to {0,1}. The two channels are statistically equivalent.) with the Gaussian distribution to be $\mathcal{N}(0, r^2H_\epsilon^2)$. Then, he constructs polar codes of code length $N = n$ for this channel as described in Section 2.3 and carries out encryption as normal;
- At the decryption step, Alice firstly calculates $H = \sqrt{1 + \sum_1^n e_i^2 + \sum_1^n s_i^2}$. If $H > H_\epsilon$, Alice goes back to the key generation step, and the whole process is restarted; otherwise, she decrypts and carries out SC decoding for the mod $2\mathbb{Z}$ channel with additive Gaussian noise $\mathcal{N}(0, r^2H^2)$. (An explicit illustration of polar encoding and decoding is given in Section 3.3.)

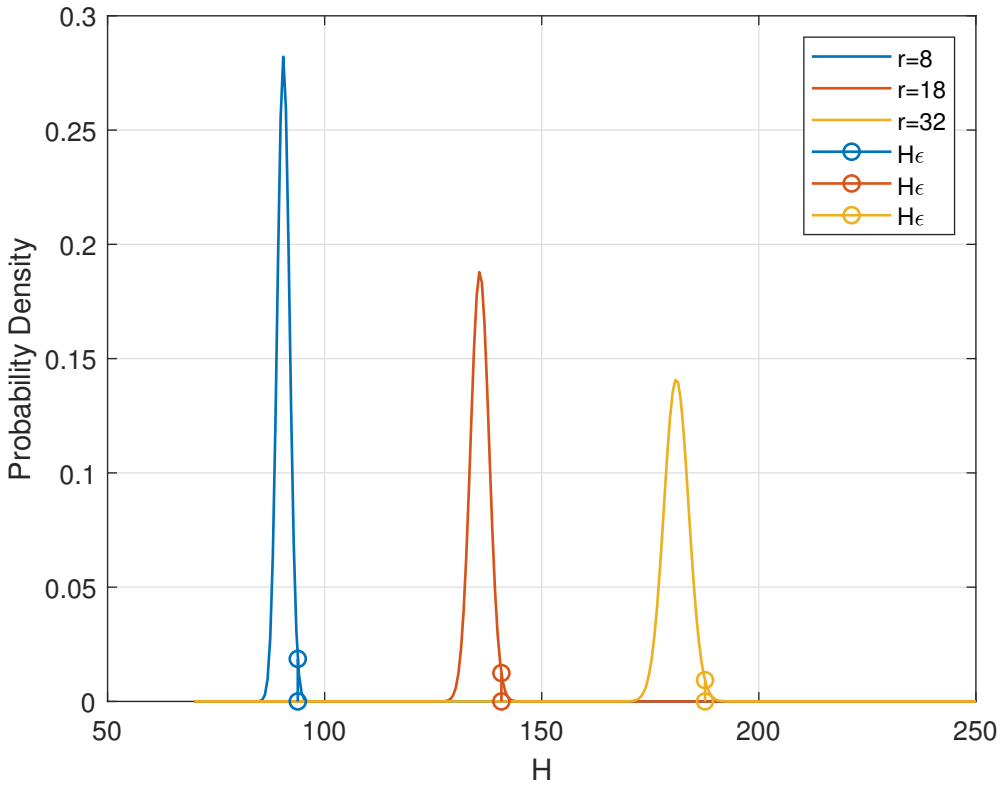


Figure 2. Outage probability and threshold $H_\epsilon, \epsilon = 0.01, n = 1024$.

3.2. The Soundness and Security of the Proposed Scheme

In the above revised RLWE-based PKE scheme, we construct polar codes for a mod $2\mathbb{Z}$ channel with additive noise $\mathcal{N}(0, r^2H_\epsilon^2)$, then apply the codes to a mod $2\mathbb{Z}$ channel with additive noise $\mathcal{N}(0, r^2H^2)$ where $H \leq H_\epsilon$. The soundness is guaranteed by the channel degradation relationship between the two channels.

Lemma 3. *If* $\sigma_1 < \sigma_2$, *the* $\mathcal{N}(0, \sigma_2^2) \mod 2\mathbb{Z}$ *channel is degraded with respect to the* $\mathcal{N}(0, \sigma_1^2) \mod 2\mathbb{Z}$ *channel.*

Proof. Suppose the channel input is X, and let $N_1 \leftarrow \mathcal{N}(0, \sigma_1^2)$ and $N_2 \leftarrow \mathcal{N}(0, \sigma_2^2)$ be additive noises. We also define an auxiliary additive noise denoted by N_{aux}, which is drawn from $\mathcal{N}(0, \sigma_2^2 - \sigma_1^2)$. At the recipient's end, the channel output after the mod $2\mathbb{Z}$ operation is $Y = (X + N_2) \mod 2\mathbb{Z} = ((X + N_1) \mod 2\mathbb{Z} + N_{aux}) \mod 2\mathbb{Z}$. As a result, $N_2 \mod 2\mathbb{Z}$ can be interpreted as a concatenation of $N_1 \mod 2\mathbb{Z}$ and $N_{aux} \mod 2\mathbb{Z}$. The proof is complete according to the definition of channel degradation as in Section 2.4. □

We now have the degradation relation between the channel models Bob and Alice have access to, i.e., $\mathcal{N}(0, H_\epsilon^2 r^2) \mod 2\mathbb{Z} \preceq \mathcal{N}(0, H^2 r^2) \mod 2\mathbb{Z}$. Recall that Lemma 2 quantitatively shows from what aspect one channel is degraded to the other and Lemma 1 shows that Arıkan's transform preserves the channel degradation relation. Meanwhile, constructing polar codes is performed by selecting the most reliable synthesized channels to convey the message. As a result, the polar code customized for $\mathcal{N}(0, H_\epsilon^2 r^2) \mod 2\mathbb{Z}$ is a subcode of the polar codes customized for the channel $\mathcal{N}(0, H^2 r^2) \mod 2\mathbb{Z}$. A similar technique by which one can construct a polar code for a degraded channel and apply it to the channel in reality can be found in [30]. The explicit polar encoding and SC decoding processes are given in Section 3.3.

Definition 4 (CPA indistinguishability experiment [42]). *Consider a public key encryption scheme* $\Pi = (Gen, Enc, Dec)$ *and an adversary* $\mathcal{A}$*; the chosen plaintext attack (CPA) indistinguishability experiment* $PubK^{cpa}_{\mathcal{A},\Pi}(n)$ *is defined as follows:*

1 $Gen(1^n)$ *is run to obtain keys* (pk, sk)*;*
2 *Adversary* $\mathcal{A}$ *is given* pk*, as well as oracle access to* $Enc_{pk}(\cdot)$*. The adversary outputs a pair of messages* m_0, m_1 *of the same length (these messages must be in the plaintext space associated with* pk*);*
3 *A random bit* $b \leftarrow \{0, 1\}$ *is chosen, and then, a ciphertext* $c \leftarrow Enc_{pk}(m_b)$ *is computed and given to* $\mathcal{A}$*. We call* c *the challenge ciphertext;*
4 $\mathcal{A}$ *continues to have access to* $Enc_{pk}(\cdot)$ *and outputs a bit* b'*;*
5 *The output of the experiment is defined to be 1 if* $b' = b$*, and 0 otherwise.*

Definition 5 (CPA secure [42]). *A public-key encryption scheme* $\Pi = (Gen, Enc, Dec)$ *has indistinguishable encryptions under a chosen plaintext attack (or is CPA secure) if for all probabilistic polynomial-time adversaries* $\mathcal{A}$ *there exists a negligible function* $negl$ *such that:*

$$Pr[PubK^{cpa}_{\mathcal{A},\Pi}(n) = 1] \leq \frac{1}{2} + negl(n).$$

For properly chosen parameters n, q and error distribution χ (e.g., in NewHope setting $n = 512, 1024$, $q = 12{,}289$; χ is the central binomial of parameter $k = 8$), RLWE-based PKE is CPA secure assuming the hardness of ring-LWE decision problem, and a concrete CPA-secure protocol was described in [43].

Proposition 2. *The revised RLWE-based PKE in Section 3.1 preserves the CPA security assuming that the standard RLWE-based PKE with properly chosen parameters* n, q *and* χ *is CPA secure.*

Proof. A standard RLWE-based PKE scheme Π is CPA secure assuming the hardness of the ring-LWE decision problem, i.e., $\Pr[\mathrm{PubK}^{\mathrm{cpa}}_{\mathcal{A},\Pi}(n) = 1] \leq \frac{1}{2} + negl(n)$. There are two modifications we made to the standard RLWE-based PKE. Firstly, at the encryption stage, Bob uses polar codes instead of uncoded plaintext. This operation has no influence on the distribution of the ciphertext and therefore preserves the security. Secondly, at the decryption step, Alice first calculates $H = \sqrt{1 + \sum_1^n e_i^2 + \sum_1^n s_i^2}$; then, she decides to repeat the key generation step if and only if $H > H_\epsilon$. Since the adversary is passive and has no idea if $H > H_\epsilon$ or not, he/she cannot determine if the ciphertext given to him/her is a valid one or not. Therefore, a polynomial-time adversary in the experiment $\mathrm{PubK}^{\mathrm{cpa}}_{\mathcal{A},\Pi'}(n)$ behaves no better than in the experiment $\mathrm{PubK}^{\mathrm{cpa}}_{\mathcal{A},\Pi}(n)$, i.e.,

$$\mathrm{PubK}^{\mathrm{cpa}}_{\mathcal{A},\Pi'}(n) \leq \mathrm{PubK}^{\mathrm{cpa}}_{\mathcal{A},\Pi}(n) \leq \frac{1}{2} + negl(n).$$

□

3.3. Polar Encoding and SC Decoding for RLWE Channel Using Outage

In this section, we show how Bob constructs polar codes using outage at the encryption step and how Alice performs decoding at the decryption step. Denote by $W : X \to Y$ the $\mathcal{N}(0, H^2r^2) \bmod 2\mathbb{Z}$ channel and by $W' : X \to Y$ its degradation $\mathcal{N}(0, H_\epsilon^2 r^2) \bmod 2\mathbb{Z}$ channel. Given the channel degradation relationship, one is able to construct polar codes for W' and apply it to W in reality. Recall in Section 2.3 that the first step to construct polar codes is to calculate the Bhattacharyya parameters for every synthesized channel $W_N^{\prime(i)}$ for $i = 1, \cdots, N$. However, as mentioned in Section 2.3, a practical solution to calculate $Z(W_N^{\prime(i)})$ is to firstly quantize the continuous output alphabet of W', then construct an approximate channel of the synthesized channel at each recursion of Arıkan's transform [33,34]. This solution proceeds as follows.

We define the likelihood ratio of W' as:

$$\lambda(y) := \frac{W'_{Y|X}(y|0)}{W'_{Y|X}(y|\lfloor \frac{q}{2} \rfloor)},\ y \in [0, q). \tag{15}$$

Since $\mathcal{N}(0, h_\epsilon^2 r^2) \bmod 2\mathbb{Z}$ is stochastically equivalent to $2\mathbb{Z}$-periodic additive Gaussian noise with variance $h_\epsilon^2 r^2$, the transition probability $W'_{Y|X}$ is defined as:

$$\begin{aligned} W'_{Y|X}(y|0) &= \sum_{\lambda \in q\mathbb{Z}} g_{0, h_\epsilon^2 r^2}(y + \lambda) \\ W'_{Y|X}(y|\lfloor \frac{q}{2} \rfloor) &= \sum_{\lambda \in q\mathbb{Z}} g_{\lfloor \frac{q}{2} \rfloor, h_\epsilon^2 r^2}(y + \lambda), \end{aligned}$$

where $g_{a,b^2}(x)$ is the density function of the Gaussian noise with mean a and variance b^2.

The channel W' is symmetric because there exists a permutation $\pi(y) = (\lfloor \frac{q}{2} \rfloor - y) \bmod q$ such that $W'(y|0) = W'(\pi(y)|\lfloor \frac{q}{2} \rfloor)$. Intuitively, a symmetric channel with binary input and continuous output can be seen as a combination of infinite binary symmetric channels (BSCs). If we focus on the likelihood ratio $\lambda(y) \geq 1$, the crossover probability of any one of these BSCs is $\frac{1}{\lambda(y)}$. The capacity of this BSC is:

$$C[\lambda(y)] = 1 - \frac{\lambda(y)}{\lambda(y)+1} \log_2 \frac{\lambda(y)+1}{\lambda(y)} - \frac{1}{\lambda(y)+1} \log_2(\lambda(y)+1),\ \lambda(y) \geq 1.$$

If we ignore the minor geometrical error introduced by rounding operation $\lfloor \cdot \rfloor$, we observe that the intervals satisfying $\lambda(y) \geq 1$ is:

$$A := [0, \lfloor \frac{q}{2} \rfloor] \cup [q - \lfloor \frac{q}{2} \rfloor, q].$$

Because $C[\lambda(y)]$ is a strict monotonic function of $\lambda(y)$, we divide A into ν segments such that for $j \in \{1, \cdots \nu\}$:

$$\begin{aligned} A_j &= \left\{ y \in A : \frac{j-1}{\nu} \leq C[\lambda(y)] \leq \frac{j}{\nu} \right\} \\ &= \left\{ y \in A : \frac{1}{\mathfrak{h}_2^{-1}\left(\frac{\nu - i + 1}{\nu}\right)} - 1 \leq \lambda(y) < \frac{1}{\mathfrak{h}_2^{-1}(\frac{\nu - i}{\nu})} - 1 \right\}, \end{aligned} \tag{16}$$

where $\mathfrak{h}_2(\cdot)$ is the entropy function of a Bernoulli random variable. Each A_j corresponds to a BSC channel with crossover probability:

$$p_j = \frac{\int_{A_j} W'_{Y|X}(y|\lfloor \frac{q}{2} \rfloor) dy}{\int_{A_j} W'_{Y|X}(y|\lfloor \frac{q}{2} \rfloor) dy + \int_{A_j} W'_{Y|X}(y|0) dy}, \tag{17}$$

where:

$$\begin{aligned} \int_{A_j} W'_{Y|X}(y|0) dy &= \int_{A_j} \sum_{\lambda \in q\mathbb{Z}} g_{0, (h_\epsilon r)^2}(y + \lambda) dy \\ \int_{A_j} W'_{Y|X}(y|\lfloor \frac{q}{2} \rfloor) dy &= \int_{A_j} \sum_{\lambda \in \lambda \in q\mathbb{Z}} g_{\lfloor \frac{q}{2} \rfloor, (h_\epsilon r)^2}(y + \lambda) dy. \end{aligned}$$

If we define z_j and its conjugate $\bar{z}_j$ to be the channel output of the BSC associated with A_j, we obtain the quantized output alphabet of W' as:

$$\mathcal{Z} := \{z_1, \bar{z}_1, z_2, \bar{z}_2, \cdots, z_\nu, \bar{z}_\nu\}.$$

If we denote by W'_Q the quantized version of the channel W', the output alphabet of W'_Q is $\mathcal{Z} := \{z_1, \bar{z}_1, \cdots, z_v, \bar{z}_v\}$. The following lemma claims that W'_Q is degraded with respect to W'.

Lemma 4. *The channel $W'_Q : X \to Z$ is degraded with respect to W'.*

Proof. We supply an intermediate channel $W_P : Y \to Z$ such that:

$$W_P(z|y) = \begin{cases} 1, & \text{if } z = z_j,\ y \in A_j, \\ 1, & \text{if } z = \bar{z}_j, \pi(y) \in A_j, \\ 0, & \text{otherwise.} \end{cases}$$

We can find that there exits a channel degradation relationship in the sense that:

$$W'_Q(z|x) = \int W'_{Y|X}(y|x) W_P(z|y) dy.$$

□

Now, we have a degraded version of W' with a finite output alphabet. Next, we apply Arıkan's transform recursively to W'_Q and calculate the $Z(W'^{(i)}_{QN})$. As the channel-combining and -splitting processes continue, the alphabet size of the synthesized channels $W'^{(i)}_{QN}$ will increase exponentially as the recursion proceeds. To handle this problem, we employed a merging technique proposed in [34], which can reduce the alphabet size of a BDMS channel with negligible and traceable loss of performance. Specifically, a BDMS channel W'_Q gives rise to BDMS synthesized channels under Arıkan's transform [32]. Any BDMS channel can be seen as a combination of BSCs. The merging technique gives an approximation of a BDMS channel by combing some of the BSCs of which it is comprised. In other words, merging approximates a BDMS channel with less BSCs, therefore a smaller output alphabet. Applying merging to the synthesized channels derived after every recursion of Arıkan's transform can effectively restrict the output alphabet. In this manner, we can approximate the synthesized channels $W'^{(i)}_{QN}$ with an output alphabet no larger than a predetermined value. This makes calculating $Z(W'^{(i)}_{QN})$ feasible.

After we finish computing the Bhattacharyya parameters of all the $W'^{(i)}_{QN}$, we can define the information set $\mathcal{A}$ and frozen set $\mathcal{A}^c$. We construct the polar codewords as:

$$x^{1:N} = u_{\mathcal{A}} G_N(\mathcal{A}) \oplus u_{\mathcal{A}^c} G_N(\mathcal{A}^c). \tag{18}$$

Upon observing the channel output $y^{1:N}$, the recipient, Alice, invokes her knowledge of the CSI h and decides to apply the decoding or to restart the protocol. The successive cancellation (SC) decoder calculates the likelihood ratio of every synthesized channel and gives an estimation of $u_{\mathcal{A}}$ according to the decision function:

$$\overline{u}^{(i)} = \begin{cases} 0, & \text{if } L_N^{(i)}(y^{1:N}, \overline{u}^{1:i-1}) \geq 1 \\ 1, & \text{otherwise} \end{cases}, \tag{19}$$

where the likelihood ratio $L_N^{(i)}(y^{1:N}, \overline{u}^{1:i-1}) \triangleq \dfrac{W_N^{(i)}(y^{1:N}, \overline{u}^{1:i-1}|0)}{W_N^{(i)}(y^{1:N}, \overline{u}^{1:i-1}|1)}$ can be calculated recursively by the SC decoding algorithm in [32]. The input of SC decoder $\lambda(y)$ is given as:

$$\lambda(y) := \frac{W_{Y|X}(y|0)}{W_{Y|X}(y|\lfloor \frac{q}{2} \rfloor)},\ y \in [0, q),$$

where the transition probability $W_{Y|X}$ is defined as:

$$W_{Y|X}(y|0) = \sum_{\lambda \in q\mathbb{Z}} g_{0,(hr)^2}(y+\lambda),$$
$$W_{Y|X}(y|\lfloor \frac{q}{2} \rfloor) = \sum_{\lambda \in q\mathbb{Z}} g_{\lfloor \frac{q}{2} \rfloor,(hr)^2}(y+\lambda).$$

A block-decoding error occurs if $\overline{u}^{1:N} \neq u^{1:N}$; we may interchangeably use the block error probability and DFR in this work. The complexity of both polar encoding and SC decoding is $O(N \log_2 N)$. Additionally, both algorithms require constant steps of operations for fixed choices of $K, N, \mathcal{A}$, making constant-time implementations plausible. According to Theorem 2, the block error probability $P_e(N, K, \mathcal{A})$ of SC decoding is upper bounded by the sum of $Z(W_N^{(i)})$. Since we have $W'_Q \preceq W' \preceq W$ and $W'^{(i)}_{QN} \preceq W'^{(i)}_N \preceq W_N^{(i)}$ according to Lemmas 1 and 2, we have:

$$P_e(N, K, \mathcal{A}) \leq \sum_{i \in \mathcal{A}} Z(W'^{(i)}_{NQ}). \tag{20}$$

4. Results: Decoding Performance Analysis

Theorem 2 gives the upper bound on the decoding error probability (DFR of PKE equivalently) of polar codes constructed for the $\mathcal{N}(0, r^2 H_\epsilon^2)$ mod $2\mathbb{Z}$ channel and applied to the $\mathcal{N}(0, H^2 r^2)$ mod $2\mathbb{Z}$ channel in reality. Figure 3 depicts the upper bound on the DFR if polar codes constructed as above are used in our revised RLWE-based PKE. In the standardization process of PQC initialized by NIST, the target DFR at code rate $1/4$ is 2^{-128}. We targeted a more conservative benchmark DFR = 2^{-140} as was used in [15,18]. Similar to NewHope, which employs a central binomial distribution with parameter k to approximate the discrete Gaussian distribution (The variance of central binomial distribution is $k/2$, and the variance of a discrete Gaussian distribution is r^2. When calculating the upper bound on the DFR, we used a continuous Gaussian distribution instead of its discrete version to ease the analysis. However, we used the central binomial of the same variance in the experiments in Figure 4), we used the parameter $k = 2r^2$ to denote different distributions χ from which e, t, s, e', e'' were drawn. We observed that by using our polar coding scheme, we could achieve the target DFR of 2^{-140} for k as large as 55, which is significantly larger than the current choice $k = 8$ in NewHope. A larger k benefits the security level of the overall scheme. Please note that schemes as NewHope compress the ciphertext before sending it out, which leads to additional compression noise. However, in this work, we only focused on the additive noise in the channel model.

The advantages of the RLWE channel model with outage are concluded as follows. Firstly, we employed an "independence" assumption so that we derived a group of i.i.d. channels. This is actually a relaxed assumption compared to the one in [15]. For example, the polynomial product $e \cdot t$ has correlated coefficients because of the polynomial convolution. However, we resolved the correlation produced by e by seeing it as a constant fading coefficient H over exactly one code block. The correlation left in our channel model only comes from t.

Secondly, the decoder is able to exploit the CSI, while the encoder makes use of the knowledge of CDI. This benefits the decoding performance significantly if compared to coding schemes that take the residue additive error term as a whole. Thirdly, the channel degradation relation makes the polar codes constructed for the degraded channel precisely fit in with the real channel. We verify our polar coding scheme in RLWE-based PKE by simulation in Figure 4. The dotted lines are the experimental results of the DFR, and the solid lines are the DFR upper bounds. At least for reasonably large code rates, the simulation results verified our estimation of the upper bounds, whereas the performance at the target code rate $1/4$ was unable to be experimentally checked.

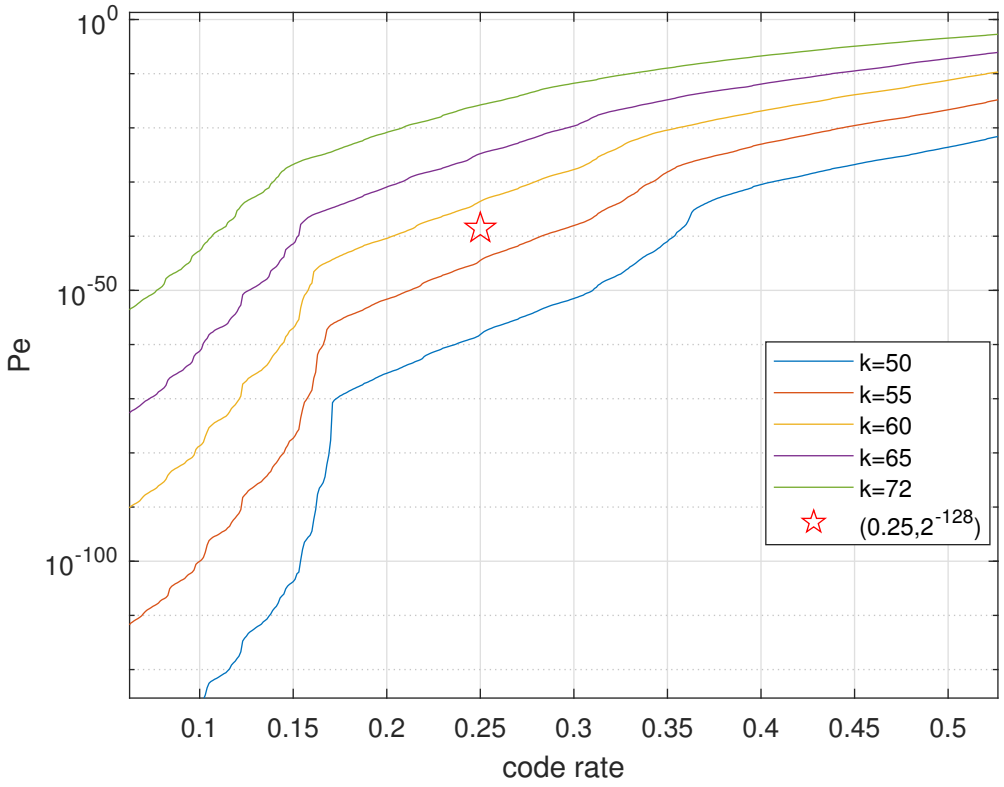


Figure 3. Upper bound on the frame error probability of SC decoding, $\epsilon = 0.01$, $N = 1024$.

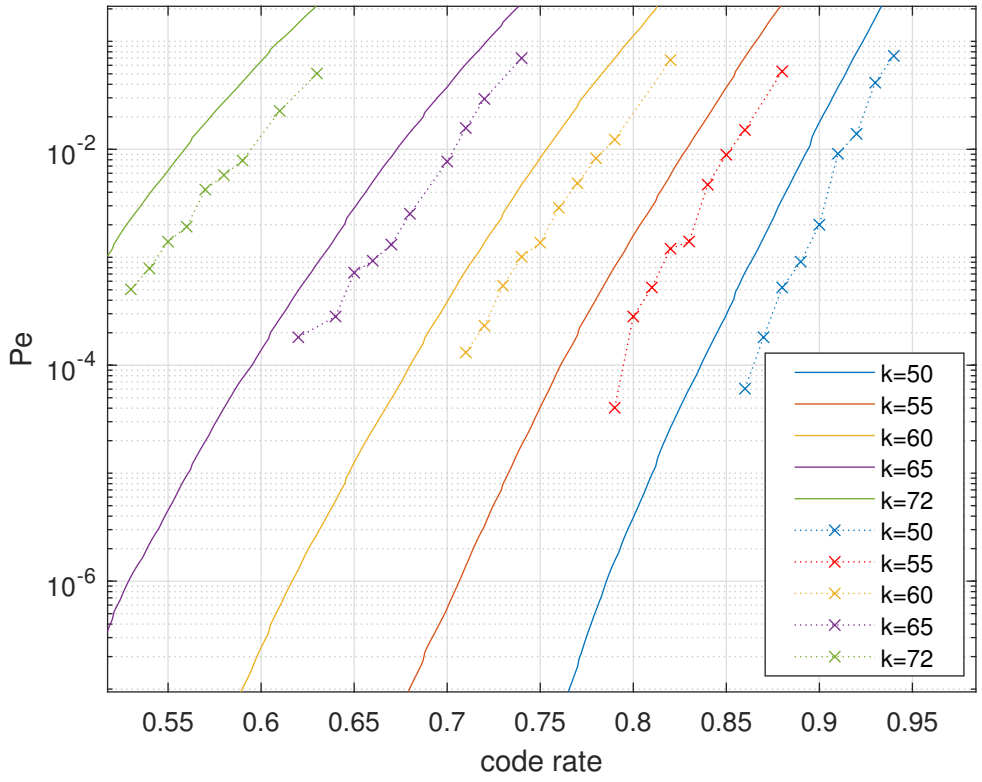


Figure 4. Decoding performance: analytical upper bound vs. simulation results

5. Discussion

5.1. Security Improvement

The new DFR margin can be exploited to increase the Gaussian noise parameter r (or the central binomial parameter $k = 2r^2$) such that the security level is increased and the DFR requirement is properly satisfied. In Table 1, we illustrate to what extent the security of RLWE-based PKE was improved for $n = 1024$, $q = 12{,}289$ compared to NewHope Round 2 if different error-correcting codes and schemes are employed. As in [15,18], a conservative target DFR was selected to be 2^{-140}. The concrete security analysis of RLWE-based PKE, so far, has been based on the hardness of LWE [22]. The security level was estimated at the cost of primal attack and dual attack (The security estimator is available at https://github.com/tpoeppelmann/newhope (accessed on 3 March 2021)).

Table 1. Improved security level of RLWE-based PKE for $n = 1024$, q = 12,289 using different error-correcting codes.

ECC Schemes	k	DFR	Classical/Quantum (bits)		Improvement
			Primal	Dual	
NewHope Round 2	8	2^{-216}	259/235	257/233	–
Polar codes in this work	55	2^{-149}	332/301	330/300	28.8%
Polar codes [19]	16	2^{-156}	282/256	281/255	9.4%
Song et al. [18]	14	2^{-156}	278/252	276/250	7.2%
Fritzmann et al. [15]	66	2^{-140}	341/309	338/307	31.76%

It was observed that the polar coding scheme described in this work gives significant security improvement compared to the one in the concurrent work using polar codes [19]. We acquired this security gain because we used the original constellation diagram $\{0, \lfloor \frac{q}{2} \rfloor\}$ rather than the closer and tailored one in [19]. Furthermore, our polar coding scheme gives a security improvement as attractive as the state-of-the-art record of 31.76% in [15], which employed nonconstant-time BCH and LDPC codes.

5.2. Constant-Time Implementation

When applying modern error-correcting codes to RLWE-based PKE, we should always be careful if the encoding and decoding enables constant-time implementations. BCH code has a good error correction capability, but its decoding proceeds in two steps: (a) locate the errors by calculating the syndrome, and (b) correct the errors if there are $t/2$ or fewer errors where t is the code distance. This is obviously not a constant-time design. LDPC code also has nonconstant-time decoding because the decoding procedure is iterative and it comes to an end when either a correct codeword is found or the maximum number of iterations is reached. Unlike the error-correcting codes (e.g., BCH, LDPC) adopted by RLWE-based PKE in the literature [15], the encoding and decoding of polar codes intrinsically enable constant-time implementations.

As for the encoding, one calculates the Bhattacharyya parameters $Z(W_N^{(i)})$ first and then carries out the encoding function as in Formula (18) (see Section 3.3). The most time-consuming step is to calculate $Z(W_N^{(i)})$; however, this can be performed offline once for all as far as the channel model in Formula (14) is known (i.e., the RLWE PKE parameters n, q, the error distribution χ, and the code rate are known). The encoding step is carried out online, and it consists of exactly $N/2 \log_2 N$ many XOR gates. An example of polar encoding for code length $N = 8$ is given in Figure 5. It can be concluded that the mod-2 additions of polar encoding are only related to the code length N, and therefore, a constant-time implementation is feasible for any fixed N.

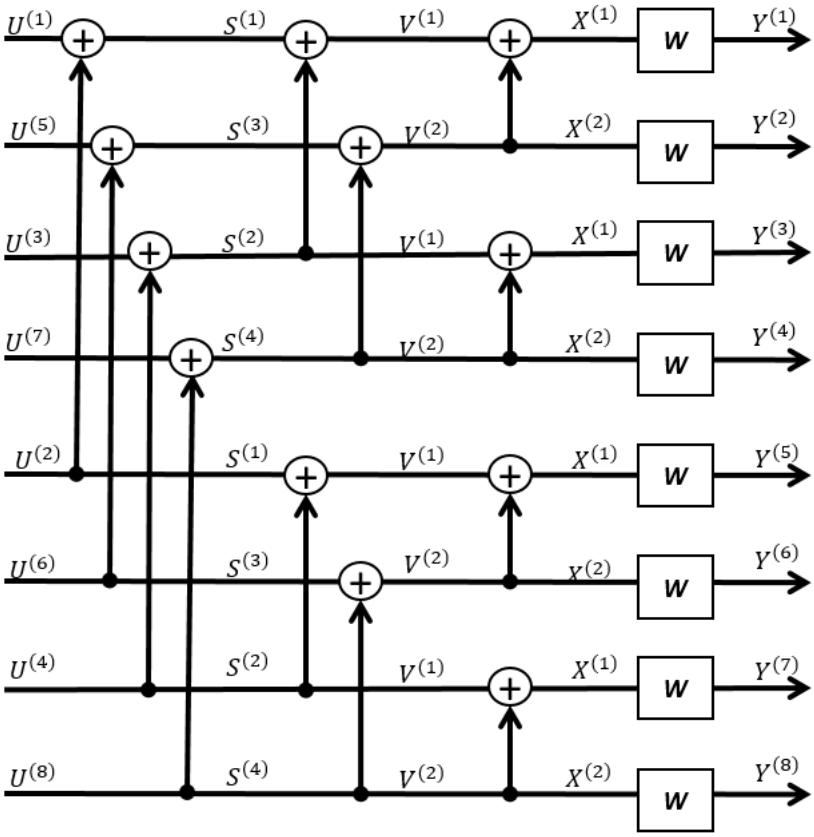


Figure 5. An example of polar encoding for code length $N = 8$.

As for polar decoding, the running time does not vary with different actualization of the message m or error term drawn from χ, as is not the case for BCH and LDPC. Given the RLWE channel output $y^{1:N}$ derived from decryption, the SC decoder recursively calculates $L_N^{(i)}(y^{1:N}, \overline{u}^{1:i-1})$ and recovers the message $u^{1:N}$ according to Formula (19). The LR calculations dominate the overall complexity of decoding, which is described in Appendix A, as well as an example for code length $N = 8$. It can be concluded that for any fixed code length N ($N =$ parameter n of RLWE-based PKE), the SC decoding require exactly $N * \log_2 N$ steps of LR calculations as in Formulas (A1) and (A2) no matter what other parameters q, χ and the code rate are. In addition, the decision-making step in Formula (19) is also constant-time because the information set $\mathcal{A}$ is uniquely determined by the channel model in Formula (14) and the parameters n, q, χ and code rate.

5.3. Complexity and Communication Overhead

Compared with the repetition codes in NewHope Round 2 [43], the proposed polar encoding and decoding scheme will for sure significantly increase the complexity. We, in this paper, mainly focused on the DFR performance and security improvement while benchmarks of the proposed scheme are not provided. Nonetheless, seeing that LDPC codes have much higher complexity than polar codes at a relatively low code rate as is explained in Appendix B (also see [44]), polar encoding and SC decoding will incur a much smaller complexity increase compared to that of 650% for LDPC, as given in [15].

Since Alice, the recipient, calculates $H = \sqrt{1 + \sum_1^n e_i^2 + \sum_1^n s_i^2}$ and goes back to the public key generation step if $H > H_\epsilon$, the averaged communication overhead is supposed to increase by a percentage of approximately ϵ for a relatively small ϵ. In this work, we set the outage probability ϵ to be a small value of 0.01, incurring a communication overhead increase by approximately 1% on average. Therefore it almost preserves the communication overhead. In addition, the proposed polar coding scheme was designed to address the additive residue noise after decryption rather than the compression noise, and we did not

improve the bandwidth efficiency compared to an improvement of 5.9% and 12.8% in [18] and [15], respectively.

6. Conclusions

In this work, we demonstrated how to construct polar codes for RLWE-based PKE. Theoretical and numerical results were given to verify the proposed coding scheme. The motivation for doing so was to give constructive guidance on how to at least relax the "dependency" and on how to design practical and efficient error-correcting codes to lower the DFR and increase the security of RLWE-based PKE.

The pros and cons of the polar coding scheme using outage are given as follows:

- The polar coding scheme using outage considerably improves the error tolerance. It significantly improves the security level (measured by bits of security) of RLWE-based PKE in the NewHope setting by 28.8%, which is as attractive as the highest record in [15];
- The proposed polar coding scheme has lower encoding and decoding complexity at a low code rate compared to other error-correcting schemes in the literature [15]. Furthermore, it intrinsically supports constant-time implementations;
- Compared with the polar coding scheme in [19], this scheme is carried out in polynomial representation and uses the original modulation constellation diagram rather than the shrunk one. This avoids the trouble of switching between the polynomial and canonical representation, and the modulation space is not compromised;
- Since the standard process of RLWE-based PKE is amended, how it will behave under a variety of attacks is left for future work, and we proved it to be at least CPA secure nonetheless.

In conclusion, using the proposed polar coding scheme in this work, one can derive a new DFR margin and therefore improve the security of a typical RLWE-based PKE scheme (e.g., NewHope). The polar coding scheme will not increase the communication overhead. For a relatively low code rate (e.g., 0.25), polar encoding and decoding are efficient compared to other modern error-correcting codes such as LDPC. Moreover, polar codes support constant-time implementations, whereas other error-correcting codes such as LDPC and BCH do not. Future work will include a solid implementation of the proposed scheme, as well as a specific benchmarking. Besides, the hidden vulnerabilities of the proposed scheme under a variety of attacks will be investigated.

Author Contributions: Conceptualization, J.W. and C.L.; methodology, J.W. and C.L.; investigation, J.W.; software, J.W.; validation, J.W. and C.L.; writing—original draft preparation, J.W.; writing—review and editing, C.L.; funding acquisition, C.L.; project administration, C.L.; supervision, C.L. All authors read and agreed to the published version of the manuscript.

Funding: This research was funded by the U.K. Engineering and Physical Sciences Research Council, Grant Number EP/S021043/1.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data is contained within the article.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Computational Complexity of SC Decoding

Consider the SC decoding for an arbitrary polar code of length N. To recover $u^{1:N}$ according to the rules in Formula (11), one needs to calculate the full set of LRs. Let $W : X \to Y$ denote a BDMS channel with input X and output Y with transition probability $W(Y|X) = P(Y|X)$. As shown in Proposition 1, the channel polarization transform com-

bines N i.i.d. copies of W in a recursive manner such that for any $0 \leq m \leq n$, $M = 2^m$, $N = 2^n$, $1 \leq \kappa \leq M/2$, the decoder calculates the LRs at the m-th layer of recursion as:

$$L_M^{(2\kappa-1)}(y_1^M, \hat{u}_1^{2\kappa-2}) = \frac{L_{M/2}^{(\kappa)}\left(y_1^{M/2}, \hat{u}_{1,o}^{2\kappa-2} \oplus \hat{u}_{1,e}^{2\kappa-2}\right) L_{M/2}^{(\kappa)}\left(y_{M/2+1}^{M}, \hat{u}_{1,e}^{2\kappa-2}\right) + 1}{L_{M/2}^{(\kappa)}\left(y_1^{M/2}, \hat{u}_{1,o}^{2\kappa-2} \oplus \hat{u}_{1,e}^{2\kappa-2}\right) + L_{M/2}^{(\kappa)}\left(y_{M/2+1}^{M}, \hat{u}_{1,e}^{2\kappa-2}\right)}, \quad \text{(A1)}$$

and:

$$L_M^{(2\kappa)}(y_1^M, \hat{u}_1^{2\kappa-1}) = \left[L_{M/2}^{(\kappa)}\left(y_1^{M/2}, \hat{u}_{1,o}^{2\kappa-2} \oplus \hat{u}_{1,e}^{2\kappa-2}\right)\right]^{1-2\hat{u}_{2\kappa-1}} \cdot L_{M/2}^{(\kappa)}\left(y_{M/2+1}^{M}, \hat{u}_{1,e}^{2\kappa-2}\right), \quad \text{(A2)}$$

where the notation $\hat{u}_{1,o}^{2\kappa-2}$ (resp. $\hat{u}_{1,e}^{2\kappa-2}$) represents a subvector of $\{\hat{u}^{(1)}, \cdots, \hat{u}^{(2\kappa-2)}\}$ with odd (resp. even) indexes (Section VIII, [32]). The stopping condition of the recursion is $L_1^{(1)}(y) = \frac{W(y|0)}{W(y|1)}$.

Observe that to calculate any LR pair $\left(L_M^{(2\kappa-1)}(y_1^M, \hat{u}_1^{2\kappa-2}), L_M^{(2\kappa)}(y_1^M, \hat{u}_1^{2\kappa-1})\right)$ at the m-th layer of the recursion, the decoder needs to know another LR pair $\left(L_{M/2}^{(\kappa)}(y_1^{M/2}, \hat{u}_{1,o}^{2\kappa-2} \oplus \hat{u}_{1,e}^{2\kappa-2}), L_{M/2}^{(\kappa)}(y_{M/2+1}^{M}, \hat{u}_{1,e}^{2\kappa-2})\right)$ at the $(m-1)$-th layer. The calculation of N LRs at layer m requires exactly N LR assembling at layer $m-1$. One can reversely compute the LRs layer-by-layer until reaching the zeroth layer, which is exactly the LR of raw channel W. Suppose that assembling an LR pair of the $(m-1)$-th layer into one LR of the m-th layer takes one complexity unit, then computing all the N LRs of the n-th layer requires $N(1+\log_2 N)$ units in total.

An example of SC decoding for code length $N = 8$ is given in Figure A1. The SC decoder recursively calculates $L_M^{(2\kappa-1)}(y_1^M, \hat{u}_1^{2\kappa-2})$ and $L_M^{(2\kappa)}(y_1^M, \hat{u}_1^{2\kappa-2})$ according to Equations (A1) and (A2) from Layer 0 to Layer 3. There are in total $8 * \log_2 8$ many LR calculations.

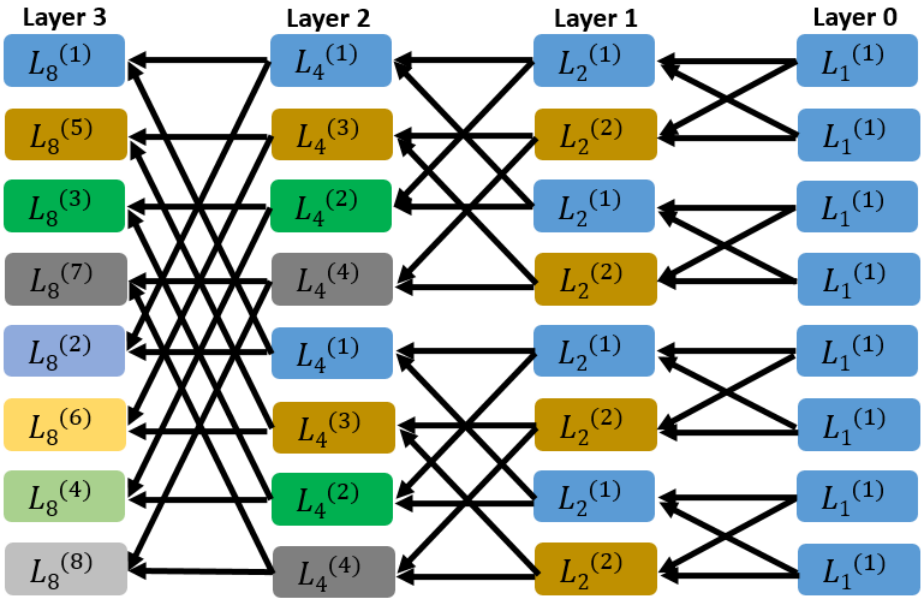


Figure A1. An example of SC decoding for code length $N = 8$.

Appendix B. Complexity: LDPC vs. Polar Codes

As discussed previously, polar encoding requires $O(N * \log_2 N)$ XOR operations, while LDPC encoding requires $O(N^2)$ matrix element multiplications [45].

To give a relatively fair comparison of decoding, the complexity can be evaluated by observing the number of addition/subtraction, multiplication, division, comparison, max/min process, and table look-up operations. In general, most of these operations

correspond to one equivalent addition, e.g., the product of two LRs can be transferred to the sum of two logarithms as is commonly used in the decoding of both LDPC and polar codes. A comparison operation in most cases corresponds to two equivalent additions, and a look-up operation takes six equivalent additions [44–46].

Normally, LDPC has larger decoding complexity than polar codes for small code rates. For both LDPC and polar codes, the basic operation at the core of decoding is the likelihood ratio (LR) calculation or equivalently the LR calculation in the log domain (LLR). Therefore, their complexity units, LR/LLR, are real numbers, and normally, we use their floating-point representations in software implementations and fixed-point on hardware. In Table A1, the decoding complexity of LDPC and polar codes is given where N, R, M denote the code length, code rate, and number of parity bits, respectively. Let L be the list size of polar SCL decoding. Denote by I_{max} the maximum number of iterations of LDPC decoding (sum-product/min-sum algorithm), by d_v the average variable degree of LDPC, and by d_c the average check degree of LDPC. When analyzing the decoding complexity, we include the number of multiplications within additions by considering log domain processing. Generally speaking, for a small code rate, a regular LDPC has a relatively large parity check matrix with relatively more nonzero elements because the code rate $R = 1 - d_v/d_c$. This will increase the message-passing complexity because there are more edges between check nodes and variable nodes.

Table A1. Complexity of LDPC and polar decoding (complexity unit: fixed/floating-point numbers) [44].

Coding Scheme	Additions	max(min)/Comparison	Look-Up Table Operations
LDPC (min-sum)	$I_{max} \cdot (2Nd_v + 2M)$	$I_{max} \cdot (2d_c - 1) \cdot M$	—
LDPC (sum-product)	$I_{max} \cdot (2Nd_v + M \cdot (2d_c - 1))$	—	$I_{max} \cdot M \cdot d_c$
Polar (SC) [47]	$N/2 \log_2 N$	$N/2 \log_2 N$	—
Polar (SCL) [47,48]	$L \cdot N/2 \log_2 N$	$L \cdot N/2 \log_2 N$	—

In Table A2, a specific complexity evaluation is given. Practically, the maximum number of iterations of LDPC decoding ranges from 20 to 50. The values of I_{max} and the list size L are selected such that the min-sum, sum-product, and SCL have comparable complexity. However, in reality, an L as large as 20 suffices in most scenarios. It can be concluded that for at least a small code rate, polar codes have lower decoding complexity than LDPC.

Table A2. Decoding complexity for an information bit length of 200 and a code rate of 1/3 (complexity unit: fixed/floating-point numbers) [44].

Coding Scheme	d_v	d_c	I_{max}	List Size	Complexity	Percentage
LDPC (min-sum)	2.576	3.864	47	—	309,400.40	100.0%
LDPC (sum-product)	2.576	3.864	20	—	301,149.40	97.3%
Polar SC (200,512)	—	—	—	—	4808.00	1.6%
Polar SCL (200,512)	—	—	—	52	309,300.57	100.0%

References

1. Lyubashevsky, V.; Peikert, C.; Regev, O. On ideal lattices and learning with errors over rings. In Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques, Monaco, France, 30 May–3 June 2010; Springer: Berlin/Heidelberg, Germany, 2010; pp. 1–23.
2. Regev, O. On Lattices, Learning with Errors, Random Linear Codes, and Cryptography. In Proceedings of the Thirty-Seventh Annual ACM Symposium on Theory of Computing, STOC '05, Baltimore, MD, USA, 22–24 May 2005; ACM: New York, NY, USA, 2005; pp. 84–93.
3. Alkim, E.; Ducas, L.; Pöppelmann, T.; Schwabe, P. Post-quantum key exchange—A new hope. In Proceedings of the 25th {USENIX} Security Symposium ({USENIX} Security 16), Austin, TX, USA, 10–12 August 2016; pp. 327–343.
4. Ding, J.; Xie, X.; Lin, X. A Simple Provably Secure Key Exchange Scheme Based on the Learning with Errors Problem. *IACR Cryptol. EPrint Arch.* **2012**, *2012*, 688.
5. Peikert, C. Lattice Cryptography for the Internet. In *Post-Quantum Cryptography*; Springer International Publishing: Cham, Switzerland, 2014; pp. 197–219.
6. Fujisaki, E.; Okamoto, T. Secure integration of asymmetric and symmetric encryption schemes. In Proceedings of the Annual International Cryptology Conference, Santa Barbara, CA, USA, 15–19 August 1999; Springer: Berlin/Heidelberg, Germany, 1999; pp. 537–554.
7. Targhi, E.E.; Unruh, D. Post-quantum security of the Fujisaki-Okamoto and OAEP transforms. In Proceedings of the Theory of Cryptography Conference, Beijing, China, 1–3 November 2016; Springer: Berlin/Heidelberg, Germany, 2016; pp. 192–216.
8. Saito, T.; Xagawa, K.; Yamakawa, T. Tightly-secure key-encapsulation mechanism in the quantum random oracle model. In Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tel Aviv, Israel, 29 April–3 May 2018; Springer: Cham, Switzerland, 2018; pp. 520–551.
9. Hofheinz, D.; Hövelmanns, K.; Kiltz, E. A modular analysis of the Fujisaki-Okamoto transformation. In Proceedings of the Theory of Cryptography Conference, Baltimore, MD, USA, 12–15 November 2017; Springer: Cham, Switzerland, 2017; pp. 341–371.
10. D'Anvers, J.P.; Guo, Q.; Johansson, T.; Nilsson, A.; Vercauteren, F.; Verbauwhede, I. Decryption failure attacks on IND-CCA secure lattice-based schemes. In Proceedings of the IACR International Workshop on Public Key Cryptography, Beijing, China, 14–17 April 2019; Springer: Cham, Switzerland, 2019; pp. 565–598.
11. D'Anvers, J.P.; Rossi, M.; Virdia, F. (One) Failure Is Not an Option: Bootstrapping the Search for Failures in Lattice-Based Encryption Schemes. In Proceedings of the Advances in Cryptology—EUROCRYPT 2020, Zagreb, Croatia, 10–14 May 2020; Springer: Cham, Switzerland, 2020; pp. 3–33.
12. Guo, Q.; Johansson, T.; Yang, J. A novel CCA attack using decryption errors against LAC. In Proceedings of the International Conference on the Theory and Application of Cryptology and Information Security, Kobe, Japan, 8–12 December 2019; Springer: Cham, Switzerland, 2019; pp. 82–111.
13. Lu, X.; Liu, Y.; Zhang, Z.; Jia, D.; Xue, H.; He, J.; Li, B.; Wang, K.; Liu, Z.; Yang, H. LAC: Practical Ring-LWE Based Public-Key Encryption with Byte-Level Modulus. *IACR Cryptol. EPrint Arch.* **2018**, *2018*, 1009.
14. Baan, H.; Bhattacharya, S.; Fluhrer, S.; Garcia-Morchon, O.; Laarhoven, T.; Rietman, R.; Saarinen, M.J.O.; Tolhuizen, L.; Zhang, Z. Round5: Compact and Fast Post-quantum Public-Key Encryption. In *Post-Quantum Cryptography*; Ding, J., Steinwandt, R., Eds.; Springer International Publishing: Cham, Switzerland, 2019; pp. 83–102.
15. Fritzmann, T.; Pöppelmann, T.; Sepúlveda, M.J. Analysis of Error-Correcting Codes for Lattice-Based Key Exchange. In Proceedings of the Selected Areas in Cryptography—SAC 2018—25th International Conference, Calgary, AB, Canada, 15–17 August 2018; Springer: Cham, Switzerland, 2018; Volume 11349, pp. 369–390.
16. D'Anvers, J.P.; Vercauteren, F.; Verbauwhede, I. The impact of error dependencies on Ring/Mod-LWE/LWR based schemes. In Proceedings of the International Conference on Post-Quantum Cryptography, Chongqing, China, 10–12 May 2019; Springer: Cham, Switzerland, 2019; pp. 103–115.
17. D'Anvers, J.P.; Vercauteren, F.; Verbauwhede, I. On the impact of decryption failures on the security of LWE/LWR based schemes. *IACR Cryptol. EPrint Arch.* **2018**, *2018*, 1089.
18. Song, M.; Lee, S.; Shin, D.; Lee, E.; Kim, Y.; No, J. Analysis of Error Dependencies on Newhope. *IEEE Access* **2020**, *8*, 45443–45456. [CrossRef]
19. Wang, J.; Ling, C. Polar Coding for Ring-LWE-Based Public Key Encryption. Cryptology ePrint Archive, Report 2021/619, 2021. Available online: https://eprint.iacr.org/2021/619 (accessed on 12 May 2021).
20. Lyubashevsky, V.; Peikert, C.; Regev, O. On ideal lattices and learning with errors over rings. *J. ACM* **2013**, *60*, 1–35. [CrossRef]
21. Lyubashevsky, V.; Peikert, C.; Regev, O. A toolkit for ring-LWE cryptography. In Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques, Athens, Greece, 26–30 May 2013; Springer: Berlin/Heidelberg, Germany, 2013; pp. 35–54.
22. Alkim, E.; Ducas, L.; Pöppelmann, T.; Schwabe, P. NewHope without reconciliation. *IACR Cryptol. EPrint Arch.* **2016**, *2016*, 1157.
23. Hall, E.; Wilson, S. Design and analysis of turbo codes on Rayleigh fading channels. *IEEE J. Sel. Areas Commun.* **1998**, *16*, 160–174. [CrossRef]
24. Trifonov, P. Design of polar codes for Rayleigh fading channel. In Proceedings of the 2015 International Symposium on Wireless Communication Systems (ISWCS), Brussels, Belgium, 25–28 August 2015; pp. 331–335.
25. Bravo-Santos, A. Polar codes for the Rayleigh fading channel. *IEEE Commun. Lett.* **2013**, *17*, 2352–2355. [CrossRef]

26. Liu, S.; Hong, Y.; Viterbo, E. Polar Codes for Block Fading Channels. In Proceedings of the 2017 IEEE Wireless Communications and Networking Conference Workshops (WCNCW), San Francisco, CA, USA, 19–22 March 2017; pp. 1–6.
27. Zheng, M.; Chen, W.; Ling, C. Polar Coding for Noncoherent Block Fading Channels. In Proceedings of the 2018 10th International Conference on Wireless Communications and Signal Processing (WCSP), Hangzhou, China, 18–20 October 2018; pp. 1–5.
28. Forney, G.D. Coset codes. I. Introduction and geometrical classification. *IEEE Trans. Inf. Theory* **1988**, *34*, 1123–1151. [CrossRef]
29. Ling, C.; Belfiore, J.C. Achieving AWGN channel capacity with lattice Gaussian coding. *IEEE Trans. Inf. Theory* **2014**, *60*, 5918–5929. [CrossRef]
30. Liu, L.; Yan, Y.; Ling, C.; Wu, X. Construction of Capacity-Achieving Lattice Codes: Polar Lattices. *IEEE Trans. Commun.* **2019**, *67*, 915–928. [CrossRef]
31. Liu, L.; Ling, C. Polar Codes and Polar Lattices for Independent Fading Channels. *IEEE Trans. Commun.* **2016**, *64*, 4923–4935. [CrossRef]
32. Arikan, E. Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels. *IEEE Trans. Inf. Theory* **2009**, *55*, 3051–3073. [CrossRef]
33. Tal, I.; Vardy, A. How to construct polar codes. *IEEE Trans. Inf. Theory* **2013**, *59*, 6562–6582. [CrossRef]
34. Pedarsani, R.; Hassani, S.H.; Tal, I.; Telatar, E. On the construction of polar codes. In Proceedings of the 2011 IEEE International Symposium on Information Theory Proceedings, St. Petersburg, Russia, 31 July–5 August 2011; pp. 11–15.
35. Mori, R.; Tanaka, T. Performance and construction of polar codes on symmetric binary-input memoryless channels. In Proceedings of the 2009 IEEE International Symposium on Information Theory, Seoul, Korea, 28 June–3 July 2009; pp. 1496–1500.
36. Mori, R.; Tanaka, T. Performance of polar codes with the construction using density evolution. *IEEE Commun. Lett.* **2009**, *13*, 519–521. [CrossRef]
37. Mori, R. Properties and Construction of Polar Codes. Master's Thesis, Kyoto University, Kyoto, Japan, 2010.
38. Korada, S.B. *Polar Codes for Channel and Source Coding*; Technical Report; EPFL: Lausanne, Switzerland, 2009.
39. Srinivasan, R.; Tiba, G. Fast estimation of outage probabilities in MIMO channels. *IEEE Trans. Commun.* **2004**, *52*, 711–715. [CrossRef]
40. Ioannou, I.; Charalambous, C.D.; Loyka, S. Outage Probability Under Channel Distribution Uncertainty. *IEEE Trans. Inf. Theory* **2012**, *58*, 6825–6838. [CrossRef]
41. Stehlé, D.; Steinfeld, R.; Tanaka, K.; Xagawa, K. Efficient public key encryption based on ideal lattices. In Proceedings of the International Conference on the Theory and Application of Cryptology and Information Security, Tokyo, Japan, 6–10 December 2009; Springer: Berlin/Heidelberg, Germany, 2009; pp. 617–635.
42. Jonathan Katz, Y.L. *Introduction to Mordern Cryptography*; CRC Press: Boca Raton, FL, USA, 2014.
43. Alkim, E.; Avanzi, R.M.; Bos, J.W.; Ducas, L.; de la Piedra, A.; Pöppelmann, T.; Schwabe, P.; Stebila, D.; Albrecht, M.R.; Orsini, E.; et al. NewHope Algorithm Specifications and Supporting Documentation. Technical Report, 2019. Available online: https://newhopecrypto.org/resources.shtml (accessed on 20 May 2021).
44. Sybis, M.; Wesolowski, K.; Jayasinghe, K.; Venkatasubramanian, V.; Vukadinovic, V. Channel Coding for Ultra-Reliable Low-Latency Communication in 5G Systems. In Proceedings of the 2016 IEEE 84th Vehicular Technology Conference (VTC-Fall), Montreal, QC, Canada, 18–21 September 2016; pp. 1–5.
45. Niu, K.; Chen, K.; Lin, J.; Zhang, Q.T. Polar codes: Primary concepts and practical decoding algorithms. *IEEE Commun. Mag.* **2014**, *52*, 192–203. [CrossRef]
46. Ryan, W.; Lin, S. *Channel Codes: Classical and Modern*; Cambridge University Press: Cambridge, UK, 2009.
47. Balatsoukas-Stimming, A.; Parizi, M.B.; Burg, A. LLR-Based Successive Cancellation List Decoding of Polar Codes. *IEEE Trans. Signal Process.* **2015**, *63*, 5165–5179. [CrossRef]
48. Tal, I.; Vardy, A. List Decoding of Polar Codes. *IEEE Trans. Inf. Theory* **2015**, *61*, 2213–2226. [CrossRef]