symmetry

MDPI

Article

Polar-BBCRS: A Variant of the McEliece Cryptosystem Using Polar Codes

Jingang Liu [1], Zhiqiang Lin [2,*] and Qinrui Lv [2]

[1] School of Mathematics and Systems Science, Guangdong Polytechnic Normal University, Guangzhou 510665, China; liujingang@gpnu.edu.cn
[2] School of Information Network Security, Xinjiang Institute of Political Science and Law, Tumxuk 843900, China; 2026078@xjzfu.edu.cn
* Correspondence: linzhiqiang@gzhu.edu.cn

Abstract

Polar codes provide efficient encoding and decoding procedures for code-based cryptography, but their use in public-key constructions requires careful treatment of the exposed code structure and intentional errors. This paper presents Polar-BBCRS, a binary polar-code instantiation of the BBCRS framework. The construction combines a permutation with a rank-three perturbation and samples fixed-weight errors subject to a public affine constraint. During decryption, the perturbation reduces to a known private vector and is cancelled exactly, leaving the polar decoder with an error of unchanged Hamming weight. We prove this cancellation property, characterize the induced decoding problem, and show that the dual of the exposed subcode contains the short words of the permuted polar dual. We report classical information-set decoding estimates and an idealized Grover-accelerated Prange search baseline. The smallest proposed parameter set uses a 30.87 KiB public key and a 128-byte ciphertext, with a minimum evaluated classical decoding cost of approximately 128.31 bits on a logarithmic operation scale. The construction combines compact public matrices with a low-rank decryption transform and supports the study of security–reliability trade-offs in polar-code-based encryption.

Keywords: post-quantum cryptography; code-based cryptography; polar codes; BBCRS; information-set decoding; public-key encryption

1. Introduction

Cryptography is essential for information security. Large-scale quantum computers threaten widely used public-key algorithms: Shor's algorithm solves the factoring and discrete-logarithm problems underlying RSA, Diffie–Hellman, and elliptic-curve cryptography [1], while Grover's algorithm accelerates unstructured search [2]. These developments motivate the transition to post-quantum cryptography. In 2016, the National Institute of Standards and Technology initiated its post-quantum public-key cryptography standardization process [3]. Code-based proposals have played an important role in this effort, with historical submissions including Classic McEliece, BIKE, and LEDAcrypt [4–6].

Code-based cryptography predates this transition. McEliece proposed his public-key cryptosystem in 1978 [7], using binary Goppa codes and efficient private decoding. Its security relies on the difficulty of decoding without the private structure and of recovering an equivalent decoding trapdoor. The large public-key size motivates alternative representations and code families. Niederreiter introduced a dual construction based on parity-check

Check for updates

Academic Editors: Jie Yang, Zhijiang Li, Shengli Deng and Rui Zhang

Received: 24 October 2026
Revised: 14 November 2026
Accepted: 25 November 2026
Published: 28 November 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

matrices [8]. Other approaches have investigated algebraic-geometric codes [9], generalized Reed–Solomon (GRS) codes [10,11], low-density parity-check codes [12,13], Reed–Muller codes [14], and low-rank parity-check codes [15]. Successful attacks against several structured variants illustrate the importance of analyzing the structure exposed by the public key [16–20].

Polar codes are a class of error-correcting codes that provably achieve the capacity of binary-input symmetric discrete memoryless channels, with efficient encoding and decoding algorithms [21]. Their recursive structure is attractive for implementations, but also relevant to cryptanalysis. In 2014, Shrestha and Kim introduced a McEliece construction based on polar codes [22]. Bardet et al. subsequently developed a key-recovery attack against the permuted polar-code setting [23]. These results motivate transformations that go beyond a secret coordinate permutation.

The BBCRS framework replaces the McEliece permutation by an invertible matrix consisting of a sparse component and a low-rank component [24]. Its general construction allows different perturbation ranks. The sparse component determines how intentional errors propagate during decryption, while the low-rank component changes the public representation of the secret code. A useful instantiation must therefore coordinate the masking transformation, error distribution, and private decoder.

1.1. Related Work

Several approaches have applied polar codes to code-based cryptography. Hooshmand et al. investigated key-length reduction through polar-code constructions [25]. PolarRLCE combines polar codes with random-column insertion and local mixing [26]. Imine et al. studied a McEliece construction based on a non-permutation transformation of polar codes [27]. Redhu et al. implemented a systematic-polar encryption scheme and reported key storage and software timings [28]. Zhang et al. proposed a polar-code key encapsulation mechanism (KEM) that applies a Fujisaki–Okamoto conversion to a non-permutation-based encryption design [29].

The security of a BBCRS instantiation also depends on the underlying code family. Couvreur et al. gave a polynomial-time key-recovery attack against rank-one GRS-based BBCRS instances in a specified sparse-matrix regime [30]. The attack makes the choice of the private code and the structure of its public subcodes central to the analysis. Increasing the perturbation rank is a construction choice, not a substitute for examining those objects.

1.2. Contribution and Organization

The question addressed in this paper is how to apply the BBCRS framework to polar codes while keeping the error seen by the legitimate decoder a permutation of the original error. The main contributions are:

1. A concrete BBCRS transformation $\mathbf{Q} = \mathbf{P} + \mathbf{V}^T\mathbf{D}$, combining a permutation with a verified rank-three perturbation.
2. A constrained error-selection rule $\mathbf{e}\mathbf{V}^T = (1, 0, 0)$ that makes the perturbation a fixed private vector. Cancelling this vector leaves the polar decoder with only the permuted intentional error.
3. An explicit characterization of the public constraint, including the effective decoding dimension and a success-preserving equivalence between complete message recovery and recovery of the actual error.
4. A structural analysis of the exposed polar subcode and a parameter study covering SCL reliability, classical and quantum ISD benchmarks, and public-key storage.

The main implementation benefit is a permutation-based residual error together with a low-rank matrix operation; no additional syndrome-guessing loop is needed during

decryption. Section 2 introduces linear codes, polar codes, and the McEliece cryptosystem. Section 3 specifies Polar-BBCRS, proves its cancellation property, and experimental result. Section 4 analyzes known attacks and presents parameters and comparisons. The final section concludes the paper.

2. Preliminaries

In this section, we introduce the background of polar codes and the McEliece cryptosystem. Vectors are denoted by lower-case bold letters and matrices by upper-case bold letters. All vectors are row vectors; the proposed construction operates over $\mathbb{F}_2$.

2.1. Linear Codes

An $[n, k]$ linear code $\mathcal{C}$ is a k-dimensional subspace of $\mathbb{F}_q^n$, with code rate k/n.

Definition 1 (Generator and parity-check matrices)**.** *A generator matrix* $\mathbf{G} \in \mathbb{F}_q^{k\times n}$ *has full row rank and maps a message* $\mathbf{m}$ *to the codeword* $\mathbf{c} = \mathbf{mG}$*. A parity-check matrix* $\mathbf{H} \in \mathbb{F}_q^{(n-k)\times n}$ *has rows forming a basis of the orthogonal complement of* $\mathcal{C}$*, so that* $\mathbf{GH}^T = 0$*.*

After choosing information coordinates and applying row operations, a generator can be written in systematic form $\mathbf{G} = (\mathbf{I}_k \mid \mathbf{U})$, with parity-check matrix $\mathbf{H} = (-\mathbf{U}^T \mid \mathbf{I}_{n-k})$. Sampling a generator uniformly among full-row-rank matrices induces a uniform random linear code. We write $\mathrm{wt}(\mathbf{e})$ for Hamming weight and $\mathrm{row}(\mathbf{G})$ for the row space. A permutation preserves both Hamming weight and coordinatewise products.

2.2. Polar Codes and List Decoding

Polar codes are based on channel polarization [21]. Let $W : \{0, 1\} \to \mathcal{Y}$ be a binary-input discrete memoryless channel, with transition probabilities $W(y \mid x)$. Two useful parameters are the symmetric mutual information and the Bhattacharyya parameter:

$$I(W) = \sum_{y\in\mathcal{Y}} \sum_{x\in\{0,1\}} \frac{1}{2} W(y \mid x) \log_2 \frac{2W(y \mid x)}{W(y \mid 0) + W(y \mid 1)}, \tag{1}$$

$$Z(W) = \sum_{y\in\mathcal{Y}} \sqrt{W(y \mid 0)W(y \mid 1)}. \tag{2}$$

Channel combining and splitting produce synthetic bit-channels whose capacities approach zero or one as the block length increases. Information bits occupy the more reliable bit-channels, while the other input positions are frozen. Smaller Bhattacharyya parameters indicate greater reliability.

For $n = 2^\ell$, the polar transform is

$$\mathbf{G}_n = \mathbf{B}_n \mathbf{F}^{\otimes \ell}, \qquad \mathbf{F} = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}, \tag{3}$$

where $\mathbf{B}_n$ is the bit-reversal permutation and $\mathbf{F}^{\otimes \ell} = \mathbf{F} \otimes \mathbf{F}^{\otimes(\ell-1)}$. A polar code can be specified by $(n, k, \mathcal{A}, \mathcal{A}^c)$. Selecting the k rows indexed by the information set $\mathcal{A}$ defines $\mathbf{G}$; the complementary input positions are frozen to zero. The information set is chosen using the reliability ordering for the design channel, and encoding is $\mathbf{c} = \mathbf{mG}$ [21].

Polar codes can be decoded by successive cancellation (SC). Finite-length performance can be improved by successive cancellation list (SCL), CRC-aided SCL, and stack or hybrid decoding [31–33]. SCL retains up to L candidate paths, with standard time complexity $O(Ln \log n)$ and space complexity $O(Ln)$. The code length, rate, design channel, information set, and decoder determine the finite-length performance. We use SCL for the

reported experiments. The notation t denotes a decoding design target, not an algebraic unique-decoding radius.

2.3. The McEliece Public-Key Cryptosystem

McEliece's original construction uses binary Goppa codes with an efficient decoder correcting up to t errors [7]. Its procedures are as follows:

1. *Private-key generation.* Choose a generator $\mathbf{G} \in \mathbb{F}_2^{k\times n}$, an invertible scrambler $\mathbf{S} \in \mathbb{F}_2^{k\times k}$, and a permutation $\mathbf{P} \in \mathbb{F}_2^{n\times n}$; hold the private decoding data.
2. *Public-key generation.* Publish $\mathbf{G}_{\text{pub}} = \mathbf{SGP}$ and the error-weight parameter.
3. *Encryption.* For $\mathbf{m} \in \mathbb{F}_2^k$, choose a random error $\mathbf{e}$ of the prescribed weight at most t, and compute $\mathbf{x} = \mathbf{mG}_{\text{pub}} + \mathbf{e}$.
4. *Decryption.* Compute $\mathbf{xP}^{-1} = \mathbf{mSG} + \mathbf{eP}^{-1}$. The private decoder recovers $\mathbf{mS}$, and multiplication by $\mathbf{S}^{-1}$ gives $\mathbf{m}$.

Niederreiter's dual construction uses a parity-check matrix and encodes a fixed-weight vector through its syndrome [8]. The corresponding decoding problems underlying the two representations are closely related [34].

2.4. The BBCRS Transformation

In BBCRS, the right transformation has the form

$$\mathbf{Q} = \mathbf{T} + \mathbf{R}, \tag{4}$$

where $\mathbf{T}$ is sparse and $\mathbf{R}$ has low rank [24]. With public generator $\mathbf{G}_{\text{pub}} = \mathbf{SGQ}^{-1}$, multiplying a ciphertext $\mathbf{x} = \mathbf{mG}_{\text{pub}} + \mathbf{e}$ by $\mathbf{Q}$ produces $\mathbf{mSG} + \mathbf{eT} + \mathbf{eR}$. The error constraint controls the last term. If each row of $\mathbf{T}$ has weight at most a, then $\text{wt}(\mathbf{eT}) \leq a\,\text{wt}(\mathbf{e})$. Polar-BBCRS takes $\mathbf{T}$ to be a permutation and arranges for $\mathbf{eR}$ to be a fixed private vector.

3. The Polar-BBCRS Construction

Let $r = 3$ and $\mathbf{s} = (1, 0, 0) \in \mathbb{F}_2^3$. The public parameters specify the code length n, dimension k, and fixed intentional error weight τ.

3.1. Construction of the Transformation

The transformation is constructed in the following steps.

1. Choose distinct pairs $\{i_1, i_2\}$ and $\{j_1, j_2\}$ from $\{1, 2, 3\}$ and let i_3 and j_3 be the respective remaining indices.
2. Sample row vectors $\boldsymbol{\gamma}, \mathbf{a}_{i_1}, \mathbf{b}_{j_1}, \mathbf{a}_{i_3}, \mathbf{b}_{j_3} \in \mathbb{F}_2^n$, and set

$$\mathbf{a}_{i_2} = \mathbf{a}_{i_1} + \boldsymbol{\gamma}, \qquad \mathbf{b}_{j_2} = \mathbf{b}_{j_1} + \boldsymbol{\gamma}. \tag{5}$$

3. Let $\mathbf{A}, \mathbf{B}_0 \in \mathbb{F}_2^{3\times n}$ have rows $\mathbf{a}_i$ and $\mathbf{b}_i$, respectively. Define $\mathbf{X}, \mathbf{Y} \in \mathbb{F}_2^{3\times n}$ by setting rows i_1, i_2 of $\mathbf{X}$ and rows j_1, j_2 of $\mathbf{Y}$ to the all-one vector, and their remaining rows to zero.
4. Sample a full-row-rank matrix $\mathbf{D}$ with rows $\mathbf{d}_1, \mathbf{d}_2, \mathbf{d}_3$, and form

$$\mathbf{R} = \begin{pmatrix} \mathbf{A}^T & \mathbf{B}_0^T \end{pmatrix} \begin{pmatrix} \mathbf{D} + \mathbf{X} \\ \mathbf{D} + \mathbf{Y} \end{pmatrix}. \tag{6}$$

5. Set $\mathbf{V} = \mathbf{A} + \mathbf{B}_0$ and sample a permutation matrix $\mathbf{P}$.
6. Keep the matrices when $\text{rank}(\mathbf{V}) = 3$ and

$$\mathbf{Q} = \mathbf{P} + \mathbf{R} \tag{7}$$

is invertible; otherwise resample.

Proposition 1 (Rank-three factorization)**.** *The accepted construction satisfies* $\mathbf{R} = \mathbf{V}^T\mathbf{D}$ *and* $\mathrm{rank}(\mathbf{R}) = 3$*. Moreover,* $\mathbf{Q}$ *is invertible if and only if* $\mathbf{J} = \mathbf{I}_3 + \mathbf{D}\mathbf{P}^{-1}\mathbf{V}^T$ *is invertible. In that case,*

$$\mathbf{Q}^{-1} = \mathbf{P}^{-1} + \mathbf{P}^{-1}\mathbf{V}^T\mathbf{J}^{-1}\mathbf{D}\mathbf{P}^{-1}. \tag{8}$$

Proof. The two offset terms in (6) equal $\boldsymbol{\gamma}^T\mathbf{1}$ and cancel over $\mathbb{F}_2$, giving $\mathbf{R} = (\mathbf{A} + \mathbf{B}_0)^T\mathbf{D}$. The map $\mathbf{V}^T : \mathbb{F}_2^3 \to \mathbb{F}_2^n$ is injective and $\mathbf{D} : \mathbb{F}_2^n \to \mathbb{F}_2^3$ is surjective, so their product has rank three. The determinant identity gives $\det(\mathbf{Q}) = \det(\mathbf{P})\det(\mathbf{J})$. Multiplying the expression in (8) by $\mathbf{Q}$ verifies the inverse. □

Thus key generation can test a 3×3 matrix instead of inverting a dense $n \times n$ matrix merely to test acceptance.

3.2. Key Generation, Encryption, and Decryption

Key generation.

Choose a polar generator $\mathbf{G} \in \mathbb{F}_2^{k\times n}$ and an invertible scrambler $\mathbf{S} \in \mathbb{F}_2^{k\times k}$. Construct $\mathbf{Q}$ as above and compute

$$\mathbf{G}_{\mathrm{pub}} = \mathbf{S}\mathbf{G}\mathbf{Q}^{-1}. \tag{9}$$

The public key is $(\mathbf{G}_{\mathrm{pub}}, \mathbf{V}, t, \tau)$, and the private key contains $\mathbf{G}$, its decoding configuration, $\mathbf{S}$, $\mathbf{Q}$, and $\mathbf{d}_1 = \mathbf{s}\mathbf{D}$. Equivalently, $\mathbf{Q}$ can be stored through its factors $\mathbf{P}$, $\mathbf{V}$, and $\mathbf{D}$.

Encryption.

To encrypt $\mathbf{m} \in \mathbb{F}_2^k$, choose a support uniformly from the τ-element subsets of $\{1, \ldots, n\}$, and let $\mathbf{e}$ be its binary indicator vector. Resample the support until $\mathbf{e}\mathbf{V}^T = \mathbf{s}$. Equivalently, sample uniformly from

$$\mathcal{E}_{\tau,\mathbf{s}} = \{\mathbf{e} \in \mathbb{F}_2^n : \mathrm{wt}(\mathbf{e}) = \tau,\ \mathbf{e}\mathbf{V}^T = \mathbf{s}\}. \tag{10}$$

The ciphertext is

$$\mathbf{x} = \mathbf{m}\mathbf{G}_{\mathrm{pub}} + \mathbf{e}. \tag{11}$$

The error-set nonemptiness is required for a usable key. It can be checked by the public counting formula in Section 4.3.

Decryption.

Compute $\mathbf{y} = \mathbf{x}\mathbf{Q} + \mathbf{d}_1$, apply the private SCL decoder to recover an estimate $\widehat{\mathbf{u}}$ of $\mathbf{m}\mathbf{S}$, and return $\widehat{\mathbf{m}} = \widehat{\mathbf{u}}\mathbf{S}^{-1}$. A decoding failure includes either failure to return a candidate or return of an incorrect message.

Theorem 1 (Weight-preserving decryption)**.** *For every accepted key, every message* $\mathbf{m}$*, and every* $\mathbf{e} \in \mathcal{E}_{\tau,\mathbf{s}}$*,*

$$\mathbf{x}\mathbf{Q} + \mathbf{d}_1 = \mathbf{m}\mathbf{S}\mathbf{G} + \mathbf{e}\mathbf{P}, \qquad \mathrm{wt}(\mathbf{e}\mathbf{P}) = \tau. \tag{12}$$

Whenever the polar decoder recovers the transmitted information vector from the right-hand side, decryption returns $\mathbf{m}$*.*

Proof. The error constraint and Proposition 1 give

$$\mathbf{e}\mathbf{R} = (\mathbf{e}\mathbf{V}^T)\mathbf{D} = \mathbf{s}\mathbf{D} = \mathbf{d}_1.$$

Using $\mathbf{G}_{\text{pub}}\mathbf{Q} = \mathbf{SG}$ in (11) yields

$$\mathbf{xQ} + \mathbf{d}_1 = \mathbf{mSG} + \mathbf{eP} + \mathbf{d}_1 + \mathbf{d}_1 = \mathbf{mSG} + \mathbf{eP}.$$

A permutation preserves Hamming weight. Finally, multiplying the recovered information vector by $\mathbf{S}^{-1}$ recovers the original message. □

3.3. Decoding Reliability

We define the frame error rate (FER) as F/N, where N is the total number of encryption–decryption trials and F counts trials in which the decoder returns no message or returns the wrong message. This experiment evaluates CRC-aided polar decoding with channel soft information under the *AFF3CT* baseline [35]. Table 1 provides fields for the SCL list size, trial counts, and FER at each candidate point.

Table 1. SCL decoding reliability at the candidate parameter points.

Set	n	k	τ	L	Trials N	Failures F	FER
P1	1024	401	155	128	50,000	18	3.6×10^{-4}
P2	2048	802	252	16	50,000	5	1.0×10^{-4}
P3	4096	989	605	8	50,000	3	6.0×10^{-5}

3.4. Efficient Private Transformation

The receiver need not store or multiply a dense $\mathbf{Q}$. Instead,

$$\mathbf{y} = \mathbf{xP} + (\mathbf{xV}^T)\mathbf{D} + \mathbf{d}_1. \tag{13}$$

This uses one permutation, three binary inner products, and at most three additions of length-n rows, followed by the fixed-vector addition. Its cost is $O(rn)$, with $r = 3$, in addition to polar decoding and the final multiplication by $\mathbf{S}^{-1}$. An unstructured implementation of the latter takes $O(k^2)$ binary operations. The resulting decryption cost is

$$O(rn + Ln \log n + k^2). \tag{14}$$

The decoder is invoked once; the fixed public syndrome eliminates a search over possible low-rank contributions.

4. Security Analysis and Parameters

We consider passive recovery of the complete message and recovery of a private decoding trapdoor. The message-recovery analysis uses uniformly distributed message blocks and the fixed-weight sampler in (10). We first characterize exactly the linear information available to an adversary and then evaluate decoding and structural attacks on the resulting public objects.

4.1. Exhaustive Key Search

An exhaustive search for the scrambler ranges over the invertible binary $k \times k$ matrices, whose number is

$$|\mathrm{GL}_k(\mathbb{F}_2)| = \prod_{i=0}^{k-1}(2^k - 2^i) = 2^{k^2}\prod_{j=1}^{k}(1 - 2^{-j}). \tag{15}$$

The permutation component has $n!$ possibilities. These large spaces make direct enumeration unattractive at the proposed parameters. Since $\mathbf{Q}$ has a structured factorization, however, it is not sampled from all invertible $n \times n$ matrices; key-space counts alone do

not bound the cost of recovering an equivalent decoding trapdoor. We therefore consider structural and decoding attacks separately.

4.2. The Public Constraint and Effective Decoding Problem

Define

$$\mathbf{M} = \mathbf{G}_{\text{pub}}\mathbf{V}^T, \qquad \rho = \operatorname{rank}(\mathbf{M}) \le 3, \qquad \mathcal{C}_0 = \{\mathbf{u}\mathbf{G}_{\text{pub}} : \mathbf{u}\mathbf{M} = 0\}. \tag{16}$$

Proposition 2 (Public linear reduction)**.** *The code $\mathcal{C}_0$ is publicly computable and has dimension $k - \rho$. A ciphertext gives the relation*

$$\mathbf{m}\mathbf{M} = \mathbf{x}\mathbf{V}^T + \mathbf{s}. \tag{17}$$

If $\mathbf{m}_*$ is any solution to this relation, then

$$\mathbf{x} + \mathbf{m}_*\mathbf{G}_{\text{pub}} \in \mathcal{C}_0 + \mathbf{e}. \tag{18}$$

For any full-rank parity-check matrix $\mathbf{H}$ of $\operatorname{row}(\mathbf{G}_{\text{pub}})$, the same error satisfies

$$\begin{pmatrix}\mathbf{H}\\ \mathbf{V}\end{pmatrix}\mathbf{e}^T = \begin{pmatrix}\mathbf{H}\mathbf{x}^T\\ \mathbf{s}^T\end{pmatrix}, \qquad \operatorname{rank}\begin{pmatrix}\mathbf{H}\\ \mathbf{V}\end{pmatrix} = n - k + \rho. \tag{19}$$

Proof. Multiplying (11) by $\mathbf{V}^T$ gives (17). Since $\mathbf{G}_{\text{pub}}$ has full row rank, $\mathbf{u} \mapsto \mathbf{u}\mathbf{G}_{\text{pub}}$ is injective; rank–nullity therefore gives $\dim \mathcal{C}_0 = k - \rho$. Subtracting any solution $\mathbf{m}_*$ from $\mathbf{m}$ puts the remaining codeword in $\mathcal{C}_0$, proving (18). The kernel of the stacked matrix in (19) is precisely $\operatorname{row}(\mathbf{G}_{\text{pub}}) \cap \ker \mathbf{V} = \mathcal{C}_0$, proving its stated rank. □

Relation (17) specifies ρ independent message combinations. For a uniform message, this relation alone leaves $2^{k-\rho}$ candidates. We account for the constraint through the effective code dimension $k - \rho$, rather than through an arbitrary subtraction from an attack exponent. The security objective here is complete message recovery; the raw encryption map does not conceal every message predicate.

Proposition 3 (Equivalence of complete recovery tasks)**.** *On ciphertexts generated by* (11)*, recovery of the complete message and recovery of the actual sampled error are equivalent up to polynomial-time linear algebra, with the same success probability.*

Proof. Given the actual message, compute $\mathbf{e} = \mathbf{x} + \mathbf{m}\mathbf{G}_{\text{pub}}$. Conversely, given the actual error, solve $\mathbf{m}\mathbf{G}_{\text{pub}} = \mathbf{x} + \mathbf{e}$. Full row rank makes the solution unique. Both transformations preserve the event of recovering the actual sampled pair. □

This equivalence applies to the construction's induced decoding distribution. It supplies the precise decoding task for the attack estimates below without requiring the public code to be identified with a uniformly random code.

4.3. Fixed-Weight Sampling

Rejection sampling in (10) is uniform because every weight-τ vector has the same proposal probability and acceptance is a deterministic predicate. The number of admissible errors is also computable directly from the public matrix. Write

$$A_\tau(\mathbf{s}) = \frac{1}{8}\sum_{\mathbf{z}\in\mathbb{F}_2^3} (-1)^{\mathbf{z}\mathbf{s}^T} K_\tau\big(\operatorname{wt}(\mathbf{z}\mathbf{V})\big), \tag{20}$$

where

$$K_\tau(d) = \sum_{j=0}^{\tau} (-1)^j \binom{d}{j} \binom{n-d}{\tau-j}.$$

Equation (20) follows by expanding the indicator of $\mathbf{e}\mathbf{V}^T = \mathbf{s}$ as a sum of binary characters and grouping supports by their intersection with $\mathbf{z}\mathbf{V}$. The expected number of sampling attempts is $\binom{n}{\tau}/A_\tau(\mathbf{s})$. This formula permits an exact acceptance calculation for each public key. Approximately balanced syndromes give about eight attempts, all performed before the single public matrix multiplication for encryption.

For a fixed public key, let $p = A_\tau(\mathbf{s})/\binom{n}{\tau}$. Under independent uniform proposals, the probability of accepting a particular admissible error at trial j is $(1-p)^{j-1}/\binom{n}{\tau}$. Thus the retry count is independent of the accepted error and the message in this model. Candidate generation, branching, and memory access can still reveal information through implementation timing. A fixed-work implementation can generate and test B candidates and select the first valid candidate using masks. The error is uniform conditional on success, and the probability of sampling failure is $(1-p)^B$. This option requires fixed-work candidate generation and incurs the cost of B trials.

4.4. Structural Attacks

Public subcode.

Let $\mathcal{C}' = \mathcal{C}\mathbf{P}^{-1}$. If $\mathbf{c}\mathbf{V}^T = 0$, then $\mathbf{c}\mathbf{Q} = \mathbf{c}\mathbf{P}$ and $\mathbf{c}$ belongs to $\mathrm{row}(\mathbf{G}_{\mathrm{pub}})$ exactly when $\mathbf{c}\mathbf{P} \in \mathcal{C}$. Consequently,

$$\mathcal{C}_0 = \mathcal{C}' \cap \ker \mathbf{V}, \qquad \mathcal{C}_0^{\perp} = (\mathcal{C}')^{\perp} + \mathrm{row}(\mathbf{V}), \tag{21}$$

where $\ker \mathbf{V} = \{\mathbf{c} : \mathbf{c}\mathbf{V}^T = 0\}$. The publicly computable dual $\mathcal{C}_0^{\perp}$ therefore contains the entire permuted polar dual, including its low-weight words.

The rows of $\mathbf{G}_n^{-T}$ indexed by frozen coordinates are independent and span $\mathcal{C}^{\perp}$. For a frozen index i, its row weight is $2^{\ell - \mathrm{wt}_2(i)}$, where $\mathrm{wt}_2(i)$ counts ones in the binary index. This follows from the column weights of $\mathbf{F}^{\otimes \ell}$, since $\mathbf{F}^{-1} = \mathbf{F}$ and bit reversal preserves $\mathrm{wt}_2(i)$; see also [36] for short polar dual words in a different masked construction. There are $\sum_{j=\ell-5}^{\ell} \binom{\ell}{j}$ rows of weight at most 32, of which at most k are information rows. Hence $\mathcal{C}_0^{\perp}$ contains at least 237, 222, and 597 independent words of weight at most 32 for P1, P2, and P3, respectively. These counts certify the existence of low-weight supports without identifying their locations.

Polar-code and GRS-based attacks.

Bardet et al. [23] recovered a permuted polar code using minimum-weight supports and signatures from puncturing and shortening the full code. Here, the short dual words give an entry point, and a similar attack would need to identify informative supports and reconstruct the relevant signatures from the exposed code. The rank-one GRS attack of Couvreur et al. [30] uses the structure of GRS codes. If an attacker obtains an equivalent decoder for $\mathcal{C}'$, the public shift in (18) supplies a word in $\mathcal{C}' + \mathbf{e}$; successful decoding recovers $\mathbf{e}$ and then $\mathbf{m}$. The cost of structural recovery in this setting remains open.

Square-code calculations.

For vectors $\mathbf{a}, \mathbf{b}$, define $\mathbf{a} * \mathbf{b} = (a_1 b_1, \ldots, a_n b_n)$. The square code is the linear span

$$\mathcal{D}^{(2)} = \langle \mathbf{a} * \mathbf{b} : \mathbf{a}, \mathbf{b} \in \mathcal{D} \rangle.$$

If $\dim \mathcal{D} = d$, products of generator rows show that

$$\dim \mathcal{D}^{(2)} \leq \min\{n, d(d+1)/2\}. \tag{22}$$

More explicitly, the Schur matrix of a generator with rows $\mathbf{g}_i$ consists of $\mathbf{g}_i * \mathbf{g}_j$ for $1 \leq i \leq j \leq d$; its row space is the square code. Random-code square dimensions provide a useful reference for this test [37]. The representative public-code calculations in Table 2 attain the ambient dimension. These calculations use the earlier parameter points indicated in the table. For comparison, the corresponding GRS square dimension is $\min\{n, 2k-1\}$.

Table 2. Reported square-code dimensions for earlier representative Polar-BBCRS public instances.

n	k	Bound in (22)	Computed Dimension	GRS Reference
1024	401	1024	1024	801
2048	802	2048	2048	1603
4096	989	4096	4096	1977

Square dimension is only one structural statistic. For example, if the chosen private polar code is self-orthogonal, the low-rank transform implies $\operatorname{rank}(\mathbf{G}_{\text{pub}}\mathbf{G}_{\text{pub}}{}^T) \leq 2r$, so its public hull has dimension at least $k - 2r$. To see this, let $\mathbf{A}_0 = \mathbf{SGP}^{-1}$ and $\mathbf{W} = \mathbf{DP}^{-1}$. Equation (9) gives $\mathbf{G}_{\text{pub}} = \mathbf{A}_0 + \mathbf{MW}$; with $\mathbf{A}_0\mathbf{A}_0^T = 0$, the Gram matrix has column space contained in the span of $\mathbf{A}_0\mathbf{W}^T$ and $\mathbf{M}$, which has total dimension at most $2r$. Consequently, the structural argument is based on the stated trapdoor-recovery assumption, not on random-code indistinguishability inferred from Table 2.

4.5. Information-Set Decoding

Message-decoding attacks aim to recover the plaintext message from a ciphertext. Information-set decoding (ISD) provides a general attack approach that does not use the private code structure. Prange's basic method searches for an information set that excludes the error positions; improved variants also allow selected error patterns within that set. The resulting work factors provide a benchmark for code-based cryptosystems and depend on the field size, code length, code dimension, and error weight.

Information-set decoding attacks recover errors without the private decoder. Starting with Prange's method [38], collision and nearest-neighbor techniques yield improved algorithms, including Stern and May–Ozerov [39,40]. We evaluate Prange, Stern, Dumer, depth-two BJMM, and depth-two May–Ozerov using the modern CryptographicEstimators [41]. The estimator inputs are $(n, k_{\text{eff}}, w) = (n, k-3, \tau)$, accounting for the maximum rank of the public constraint in our scheme. The reported model uses binary-operation costs, logarithmic solution count zero (one target solution), the package's default concrete-cost parameter search, and no memory bound or extra memory-access penalty.

For a quantum search baseline, set $d = k - 3$. The probability that a uniform d-subset of coordinates avoids a specified weight-τ error support is $p_0 = \binom{n-d}{\tau}/\binom{n}{\tau}$. Grover-accelerated Prange searches for an information set by amplitude amplification [42]. With one planted error and ignoring the full-rank check, the iteration scale is $p_0^{-1/2}$. Its base-2 logarithm is 61.21, 97.90, and 131.72 for P1, P2, and P3. Circuit-level costs also depend on reversible rank checking and linear-system solving [43].

4.6. Parameter Sets and Attack Costs

Using this estimation tool, we evaluate parameter sets against classical generic-decoding targets $\kappa = 128, 192, 256$, respectively. Table 3 gives their fixed error weights, key sizes, and ciphertext sizes. Table 4 reports each ISD algorithm separately. The minimum listed costs are $2^{128.31}$, $2^{192.23}$, and $2^{256.35}$; structural attacks are considered separately above.

Table 3. Polar-BBCRS parameter sets.

Set	n	k	τ	Public Key (KiB)	Ciphertext (Bytes)
P1	1024	401	155	30.87	128
P2	2048	802	252	122.73	256
P3	4096	989	605	376.60	512

Table 4. Classical ISD estimates, expressed as $\log_2$ of the estimated binary-operation count. BJMM and May–Ozerov use depth two.

Set	Prange	Stern	Dumer	BJMM	May–Ozerov
P1	150.35	130.60	132.35	128.31	128.33
P2	226.38	204.60	204.24	193.61	192.23
P3	297.06	272.09	271.65	258.06	256.35

4.7. Public-Key Representation

The public generator can be stored as $\mathbf{G}_{\mathrm{pub}_{\mathrm{sys}}} = (\mathbf{I}_k \mid \mathbf{U})$ after choosing information coordinates. More explicitly, choose a permutation $\mathbf{\Pi}$ for which the first k columns of $\mathbf{G}_{\mathrm{pub}}\mathbf{\Pi}$ form an invertible matrix $\mathbf{K}$, and set

$$\mathbf{G}_{\mathrm{pub}_{\mathrm{sys}}} = \mathbf{K}^{-1}\mathbf{G}_{\mathrm{pub}}\mathbf{\Pi}, \qquad \mathbf{V}_{\mathrm{sys}} = \mathbf{V}\mathbf{\Pi}. \tag{23}$$

This is an equivalent representation of the same construction: absorb $\mathbf{K}^{-1}$ into the private scrambler and use $\mathbf{Q}_{\mathrm{sys}} = \mathbf{\Pi}^{-1}\mathbf{Q}$. Its permutation component is $\mathbf{\Pi}^{-1}\mathbf{P}$ and its low-rank component is $\mathbf{V}_{\mathrm{sys}}^{T}\mathbf{D}$. Neither $\mathbf{K}$ nor $\mathbf{\Pi}$ needs to be transmitted as a separate public field. The systematic matrix size is therefore

$$|\mathrm{pk}| = k(n-k) + 3n \quad \text{bits}, \qquad |\mathrm{ct}| = n \quad \text{bits}. \tag{24}$$

4.8. Comparison with Representative Constructions

Table 5 compares the main design choices. Polar-BBCRS keeps the public code length equal to the private code length and combines low-rank mixing with exact cancellation at the receiver.

Table 5. Design features of representative polar-code constructions.

Construction	Public-Code Transformation	Decoder-Side Feature
Direct polar McEliece [22]	Scrambling and permutation	Permuted intentional error
PolarRLCE [26]	Random-column insertion and mixing	Decoding adapted to the expanded construction
Polar BBCRS-type [27]	Non-permutation transformation	Error propagation depends on the chosen transform
Polar-BBCRS	Permutation plus rank-three perturbation	Exact private cancellation; residual weight τ

The principal decoding advantage over a weight-expanding sparse transform is algebraic: $\mathrm{wt}(\mathbf{eP}) = \mathrm{wt}(\mathbf{e})$ for every accepted error. For comparison, a transform with maximum row weight a has the bound $\mathrm{wt}(\mathbf{eT}) \le a\tau$. A bounded-distance budget t then imposes the sufficient restriction $\tau \le \lfloor t/a \rfloor$, whereas the permutation component does not incur this factor. For storage context, Table 6 presents the three parameter-point comparisons. The historical comparison sizes follow [26], with Classic McEliece parameters also documented in [4].

Table 6. Public-key size comparison at selected parameter points (KiB). The BBCRS entry is a historical GRS-based storage reference.

Set	Polar-BBCRS	McEliece	PolarRLCE	Classic McEliece	BBCRS
P1	30.87	187.69	97.53	255.00	65.96
P2	122.73	489.40	256.08	511.88	–
P3	376.60	936.02	379.22	1326.00	–

At the selected comparison points in Table 6, P1 uses approximately 53.2% less public-key storage than the historical GRS-BBCRS entry. Against PolarRLCE, the reductions are 68.3%, 52.1%, and 0.7%, respectively, with the largest advantages for P1 and P2. Additionally, P1 uses 87.9% less public-key storage than Classic McEliece. These percentages compare the listed storage sizes; the historical points do not constitute a benchmark under a uniform current security model.

Further key-size reductions through quasi-cyclic or seed-generated public matrices would require a separate security analysis of the additional structure or generation distribution. Square-code and puncturing/shortening tests provide specific structural checks rather than a general indistinguishability guarantee. The present public-key sizes use systematic matrix storage together with the low-rank public component $\mathbf{V}$.

The systematic-polar implementation of Redhu et al. [28] reports software timings for a different encoding and decoding procedure. The polar-code KEM of Zhang et al. [29] studies a different transform and an encapsulation protocol. In Polar-BBCRS, exact cancellation keeps the original block length and error weight and needs one SCL call, at the cost of affine error sampling and a publicly exposed subcode. Recent analyses of short duals in polarRLCE [36] and Schur-product invariants in code equivalence [44] illustrate why storage comparisons and full square-code dimensions alone do not capture the structural trade-off.

4.9. Limitations and Assumptions

The recovery analysis concerns complete messages and equivalent private decoders. For $\rho > 0$, the ciphertext reveals ρ independent linear combinations of the message by (17), so the raw encryption map does not provide semantic security for general messages. The generic ISD estimates benchmark the induced decoding problem. They do not quantify recovery from the exposed polar structure.

The historical FER applies to its own parameter point. Decryption reliability at each candidate point depends on its error distribution, polar information set, and SCL configuration. In a future KEM, observable decoding failures could expose information through decapsulation responses or timing. Such a protocol would require a separate reliability and security analysis.

5. Conclusions

Polar-BBCRS adapts the BBCRS framework to polar codes through a rank-three perturbation and fixed-weight errors subject to affine constraints. Its private transformation cancels the perturbation exactly, preserving the intended error weight and requiring only a single SCL decoding invocation, while the public constraint is explicitly embedded in the decoding model with a dimension reduction of at most three.

The candidate sets yield public-key sizes of 30.87, 122.73, and 376.60 KiB. The smallest evaluated classical ISD cost for P1 is approximately $2^{128.31}$ binary operations. We also quantify an idealized quantum search baseline and show that short polar dual words remain in the exposed subcode's dual. These results motivate further structural cryptanalysis, decoder-parameter optimization, and the study of a separate KEM construction.

https://doi.org/10.3390/sym18122000

Author Contributions: Conceptualization, Z.L.; methodology, Z.L. and J.L.; formal analysis, J.L.; validation, Q.L.; writing—original draft preparation, Z.L. and J.L.; writing—review and editing, Z.L., J.L., and Q.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Talent Special Project of Research Project of Guangdong Polytechnic Normal University (No. 2021SDKYA034); the National Natural Science Foundation of China (grant no. U25A20543).

Data Availability Statement: The analytical derivations, parameter sets, and summarized numerical results supporting this study are included in the article.

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

References

1. Shor, P.W. Algorithms for quantum computation: Discrete logarithms and factoring. In *Proceedings of the 35th Annual Symposium on Foundations of Computer Science*; IEEE Computer Society: Washington, DC, USA, 1994; pp. 124–134.
2. Grover, L.K. A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing (STOC)*; ACM Press: New York, NY, USA, 1996. https://doi.org/10.1145/237814.237866.
3. NIST. *Post-Quantum Cryptography Standardization Project*; NIST: Gaithersburg, MD, USA, 2016. https://doi.org/10.6028/NIST.IR.8105.
4. Bernstein, D.J.; Chou, T.; Lange, T.; von Maurich, I.; Misoczki, R.; Niederhagen, R.; Persichetti, E.; Peters, C.; Sendrier, N.; Szefer, J.; et al. Classic McEliece: Conservative Code-Based Cryptography. *NIST Submissions*. February 2019. Available online: https://classic.mceliece.org/nist/mceliece-20190331.pdf (accessed on 15 October 2026).
5. Aragon, N.; Barreto, P.; Bettaieb, S.; Bidoux, L.; Blazy, O.; Deneuville, J.C.; Gaborit, P.; h Ghosh, S.; Gueron, S.; Güneysu, T.; et al. BIKE: Bit Flipping Key Encapsulation. *NIST Submissions*. February 2019. Available online: https://bikesuite.org/files/v4.1/BIKE_Spec.2020.10.20.1.pdf (accessed on 15 October 2026).
6. Baldi, M.; Barenghi, A.; Chiaraluce, F.; Pelosi, G.; Santini, P. LEDAcrypt. *NIST Submissions*. February 2019. Available online: https://www.ledacrypt.org/documents/LEDAcrypt_spec_2_5.pdf (accessed on 15 October 2026).
7. McEliece, R.J. A public-key cryptosystem based on algebraic coding theory. *Jet Propuls. Lab. DSN Prog. Rep.* **1978**, *42–44*, 114–116.
8. Niederreiter, H. Knapsack-type cryptosystems and algebraic coding theory. *Prob. Contr. Inform. Theory* **1986**, *15*, 157–166.
9. Janwa, H.; Moreno, O. McEliece public key cryptosystems using algebraic-geometric codes. *Des. Codes Cryptogr.* **1996**, *8*, 293–307. https://doi.org/10.1007/bf00173300.
10. Berger, T.; Loidreau, P. How to Mask the Structure of Codes for a Cryptographic Use. *Des. Codes Cryptogr.* **2005**, *35*, 63–79. https://doi.org/10.1007/s10623-003-6151-2.
11. Wieschebrink, C. Two NP-complete Problems in Coding Theory with an Application in Code Based Cryptography. In *Proceedings of the 2006 IEEE International Symposium on Information Theory*; IEEE: Washington, DC, USA, 2006. https://doi.org/10.1109/isit.2006.261651.
12. Monico, C.; Rosenthal, J.; Shokrollahi, A. Using low density parity check codes in the McEliece cryptosystem. In *Proceedings of the 2000 IEEE International Symposium on Information Theory (Cat. No.00CH37060)*; IEEE: Washington, DC, USA, 2000. https://doi.org/10.1109/isit.2000.866513.
13. Gaborit, P. Shorter keys for code based cryptography. In Proceedings of the 2005 International Workshop on Coding and Cryptography (WCC 2005), Bergen, Norway, 14–18 March 2005; pp. 81–91.
14. Sidelnikov, V. A public-key cryptosystem based on binary Reed-Muller codes. *Discret. Math. Appl.* **1994**, *4*, 191–207. https://doi.org/10.1515/dma.1994.4.3.191.
15. Gaborit, P.; Murat, G.; Ruatta, O.; Zémor, G. Low rank parity check codes and their application to cryptography. In Proceedings of the Workshop on Coding and Cryptography (WCC), Bergen, Norway, 15–19 April 2013; Volume 2013.
16. Baldi, M.; Chiaraluce, G. Cryptanalysis of a new instance of McEliece cryptosystem based on QC-LDPC Codes. In *Proceedings of the 2007 IEEE International Symposium on Information Theory*; IEEE: Washington, DC, USA, 2007. https://doi.org/10.1109/isit.2007.4557609.
17. Couvreur, A.; Gaborit, P.; Gauthier-Umaña, V.; Otmani, A.; Tillich, J.P. Distinguisher-based attacks on public-key cryptosystems using Reed–Solomon codes. *Des. Codes Cryptogr.* **2014**, *73*, 641–666. https://doi.org/10.1007/s10623-014-9967-z.
18. Couvreur, A.; Márquez-Corbella, I.; Pellikaan, R. A polynomial time attack against algebraic geometry code based public key cryptosystems. In *Proceedings of the 2014 IEEE International Symposium on Information Theory*; IEEE: Washington, DC, USA, 2014; pp. 1446–1450. https://doi.org/10.1109/isit.2014.6875072.
19. Faugere, J.; Otmani, A.; Perret, L.; Tillich, J. Algebraic Cryptanalysis of McEliece Variants with Compact Keys. In *Advances in Cryptology—EUROCRYPT 2010*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 279–298. https://doi.org/10.1007/978-3-642-13190-5_14.
20. Minder, L.; Shokrollahi, A. Cryptanalysis of the Sidelnikov Cryptosystem. In *Advances in Cryptology—EUROCRYPT 2007*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 347–360. https://doi.org/10.1007/978-3-540-72540-4_20.

21. 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. https://doi.org/10.1109/tit.2009.2021379.
22. Shrestha, S.R.; Kim, Y.S. New McEliece Cryptosystem Based on Polar Codes as a Candidate for Post-Quantum Cryptography. In *Proceedings of the 2014 14th International Symposium on Communications and Information Technologies (ISCIT)*; IEEE: Washington, DC, USA, 2014; pp. 368–372.
23. Bardet, M.; Chaulet, J.; Dragoi, V.; Otmani, A.; Tillich, J.P. Cryptanalysis of the McEliece Public Key Cryptosystem Based on Polar Codes. In *Proceedings of the Post-Quantum Cryptography: 7th International Workshop, PQCrypto 2016, Fukuoka, Japan, 24–26 February 2016*; Proceedings 7; Springer: Berlin/Heidelberg, Germany, 2016; pp. 118–143.
24. Baldi, M.; Bianchi, M.; Chiaraluce, F.; Rosenthal, J.; Schipani, D. Enhanced Public Key Security for the McEliece Cryptosystem. *J. Cryptol.* **2016**, *29*, 1–27.
25. Hooshmand, R.; Shooshtari, M.; Eghlidos, T.; Aref, M. Reducing the Key Length of McEliece Cryptosystem Using Polar Codes. In *Proceedings of the 2014 11th International ISC Conference on Information Security and Cryptology*; IEEE: Washington, DC, USA, 2014. https://doi.org/10.1109/iscisc.2014.6994031.
26. Liu, J.; Wang, Y.; Yi, Z.; Lin, Z. PolarRLCE: A New Code-Based Cryptosystem Using Polar Codes. *Secur. Commun. Netw.* **2019**, *2019*, 3086975. https://doi.org/10.1155/2019/3086975.
27. Imine, B.; Hadj-Said, N.; Ali-Pacha, A. New McEliece Cryptosystem Based on Non-Permutation Equivalent Polar Codes. *J. Discret. Math. Sci. Cryptogr.* **2023**, *26*, 103–114. https://doi.org/10.1080/09720529.2021.1933706.
28. Redhu, R.; Narwal, E.; Gupta, S.; Hooda, R.; Ahlawat, S.; Khurana, R. Software implementation of systematic polar encoding based PKC-SPE cryptosystem for quantum cybersecurity. *Sci. Rep.* **2024**, *14*, 9994. https://doi.org/10.1038/s41598-024-60767-3.
29. Zhang, H.; Lin, Z.; Liu, J.; Zhou, H. A Novel Polar Code-Based Key Encapsulation Mechanism with Non-permutation Equivalent Public Key. In *Proceedings of the Security and Privacy in New Computing Environments, SPNCE 2023*; Springer: Berlin/Heidelberg, Germany, 2025. https://doi.org/10.1007/978-3-031-73699-5_13.
30. Couvreur, A.; Otmani, A.; Tillich, J.P.; Gauthier-Umana, V. A Polynomial-Time Attack on the BBCRS Scheme. In *Proceedings of the IACR International Workshop on Public Key Cryptography*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 175–193.
31. Chen, K.; Niu, K.; Lin, J. Improved successive cancellation decoding of polar codes. *IEEE Trans. Commun.* **2013**, *61*, 3100–3107.
32. Tal, I.; Vardy, A. List Decoding of Polar Codes. *IEEE Trans. Inf. Theory* **2015**, *61*, 2213–2226. https://doi.org/10.1109/TIT.2015.2410251.
33. Niu, K.; Chen, K. CRC-Aided Decoding of Polar Codes. *IEEE Commun. Lett.* **2012**, *16*, 1668–1671. https://doi.org/10.1109/LCOMM.2012.090312.121501.
34. Li, Y.X.; Deng, R.H.; Wang, X.M. On the equivalence of McEliece's and Niederreiter's public-key cryptosystems. *IEEE Trans. Inf. Theory* **1994**, *40*, 271–273.
35. AFF3CT Project. *AFF3CT: A Fast Forward Error Correction Toolbox*, version 4.7.3. Available online: https://github.com/aff3ct/aff3ct (accessed on 11 November 2026).
36. Wang, Y. Hidden Curves and Short Duals: Cubic Jets, Dual Leaks, and the Limits of Masked-Code Encryption. Cryptology ePrint Archive, Paper 2026/2710. 2026. Available online: https://eprint.iacr.org/2026/2710 (accessed on 13 November 2026).
37. Cascudo, I.; Cramer, R.; Mirandola, D.; Zémor, G. Squares of random linear codes. *IEEE Trans. Inf. Theory* **2015**, *61*, 1159–1173.
38. Prange, E. The use of information sets in decoding cyclic codes. *IEEE Trans. Inf. Theory* **1962**, *8*, 5–9. https://doi.org/10.1109/tit.1962.1057777.
39. Stern, J. A Method for Finding Codewords of Small Weight. In *Proceedings of the Coding Theory and Applications: 3rd International Colloquium Toulon, France, November 2–4*; 1988 Proceedings 3; Springer: Berlin/Heidelberg, Germany, 1989; pp. 106–113.
40. May, A.; Ozerov, I. On Computing Nearest Neighbors with Applications to Decoding of Binary Linear Codes. In *Advances in Cryptology—EUROCRYPT 2015*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 203–228. https://doi.org/10.1007/978-3-662-46800-5_9.
41. Esser, A.; Verbel, J.; Zweydinger, F.; Bellini, E. SoK: CryptographicEstimators—A Software Library for Cryptographic Hardness Estimation. In *Proceedings of the 19th ACM Asia Conference on Computer and Communications Security, New York, NY, USA, 2024*; ASIA CCS '24; ACM: New York, NY, USA, 2024; pp. 560–574. https://doi.org/10.1145/3634737.3645007.
42. Bernstein, D.J. Grover vs. McEliece. In *Proceedings of the Post-Quantum Cryptography*; Lecture Notes in Computer Science; Sendrier, N., Ed.; Springer: Berlin/Heidelberg, Germany, 2010; Volume 6061, pp. 73–80. https://doi.org/10.1007/978-3-642-12929-2_6.
43. Chevignard, C.; Fouque, P.A.; Schrottenloher, A. Reducing the Number of Qubits in Quantum Information Set Decoding. In *Proceedings of the Advances in Cryptology—ASIACRYPT 2024*; Lecture Notes in Computer Science; Chung, K.M., Sasaki, Y., Eds.; Springer: Berlin/Heidelberg, Germany, 2025; Volume 15491, pp. 299–329. https://doi.org/10.1007/978-981-96-0944-4_10.
44. Battagliola, M.; Mora, R.; Santini, P. Using the Schur Product to Solve the Code Equivalence Problem. *IEEE Trans. Inf. Theory* **2026**, *72*, 4675–4694. https://doi.org/10.1109/TIT.2026.3694630.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.