(Interleaved) Extended Gabidulin Codes Decoding up to Gilbert-Varshamov Bound and Their Applications to RQC

Yongcheng Song[1], Rongmao Chen[2], Fangguo Zhang[3], Xinyi Huang[1⋆], Jian Weng[1], and Huaxiong Wang[4]

[1] College of Cyber Security, Jinan University, Guangzhou, China
yongchengsong@outlook.com, {xyhuang81,cryptjweng}@gmail.com
[2] School of Computer, National University of Defense Technology, Changsha, China
chromao@nudt.edu.cn
[3] School of Computer Science and Engineering, Sun Yat-Sen University, Guangzhou, China
zhangfg@mail.sysu.edu.cn
[4] School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore
hxwang@ntu.edu.sg

Abstract. In this paper, we investigate the Extended Gabidulin (EG) codes and the Interleaved EG (IEG) codes, and enhance the Rank Quasi-Cyclic (RQC) encryption scheme. Our primary contribution is the development of a general decoding algorithm for (I)EG codes, for which we precisely provide the DFR, bound the decoding capacity, and estimate the decoding complexity. As the core tool, we demonstrate that the Linear Reconstruction (LR) problem derived from the decoding (I)EG codes problem can be probabilistically solved, enabling (I)EG codes to achieve arbitrarily small DFRs, decode up to the rank Gilbert-Varshamov bound (even close to the minimal distance), and decode by the Welch-Berlekamp like algorithm. An interesting and important byproduct is that we demonstrate that decoding interleaved Gabidulin codes can be achieved deterministically by solving the LR problem. We finally apply the EG codes to improve RQC (NIST PQC & Asiacrypt 2023). For 128-bit security, our optimized RQC reduces bandwidth by 69% and 34% compared to the original versions, respectively. The scheme also achieves at least 50% improvement in efficiency and mitigates MM algebraic attacks (as discussed in Eurocrypt 2020, Asiacrypt 2020 & 2023) as EG codes facilitate schemes operating over smaller finite fields. Overall, our scheme outperforms code-based schemes of NIST PQC Round 4 submissions, such as HQC, BIKE, and Classic McEliece, in terms of bandwidth. A conservative parameters set still remains competitive bandwidths.

Keywords: Extended Gabidulin Codes, Post-Quantum Cryptography, Code-Based Cryptography, NIST PQC, RQC

⋆ Corresponding Author

1 Introduction

Rank Metric Codes and Gabidulin Codes. The rank metric codes were introduced by Delsarte [19] in 1978, and later were also found by Gabidulin [22], along with Gabidulin codes. Since then, rank metric codes have been used for many applications: coding theory and space time coding in particular, and also for cryptography. The rank metric codes used in cryptography are compressed over an extension field $\mathbb{F}_{q^m}$ of degree m of the finite field $\mathbb{F}_q$ to save the size of cryptosystems, and are called $\mathbb{F}_{q^m}$-linear codes. An $\mathbb{F}_{q^m}$-linear code ($[n,k]_{q^m}$-linear code) of length n and dimension k is a k-dimensional subspace of $\mathbb{F}_{q^m}^n$. Any word can be associated with an $m \times n$ matrix and the rank weight is defined as the rank of this matrix. Currently, the families of well-known rank metric codes are mainly Gabidulin codes [22], Low Rank Parity Check (LRPC) codes [7,23], and Simple codes [24]. In this paper, we focus on Gabidulin codes and its variants [13,9,15,31,44]. An $[n,k]_{q^m}$-Gabidulin code is the evaluation of q-polynomials [40] bounded degree $k-1$ on a fixed generator of weight n with the condition $k \leq n \leq m$. The Gabidulin codes are Maximum Rank Distance (MRD) codes and are viewed as the rank metric analogues of classical Reed-Solomon codes in Hamming metric. They therefore have a strong algebraic structure. The Gabidulin codes benefit from an efficient decoding algorithm that corrects weight errors up to $\lfloor \frac{n-k}{2} \rfloor$ in a deterministic way. As is well known, there is a gap between the decoding capacity and the Rank Gilbert-Varshamov (RGV) bound.

Extended Gabidulin Codes. Among variants of Gabidulin codes [13,9,15] , Extended Gabidulin (EG) codes [13] proposed by Berger and Ourivski in 2009 are interesting. Unlike Gabidulin codes, the EG codes include many non-MRD codes and feature a weaker algebraic structure. The EG codes still are the evaluation of q-polynomials, but with a wide parameters region. Let q, m, n, t, k be integers and $k \leq t \leq \min\{n, m\}$. Let $\boldsymbol{g} = (g_1, g_2, \ldots, g_n) \in \mathbb{F}_{q^m}^n$ be a generator of weight t. Let $\mathcal{L}_{\leq k-1}[x]$ be the set of q-polynomials of q-degree $\leq k-1$. The EG code of dimension k and length n generated by $\boldsymbol{g}$ is defined as

$$\mathrm{EG}_k(\boldsymbol{g}) = \Big\{ f(\boldsymbol{g}) = (f(g_1), f(g_2), \ldots, f(g_n)) \ : \ f(x) \in \mathcal{L}_{\leq k-1}[x] \Big\}.$$

One can observe that if $m \geq n$ and $t = n$, namely the coordinates $g_1, g_2, \ldots, g_n$ of the generator is linearly independent, the EG codes are exactly Gabidulin codes. From the Gabidulin codes to the EG codes, the main difference is that the generator is allowed to be linearly dependent, i.e., $t \leq \min\{n, m\}$. For long time, the linear independence of generator has been considered as the key of successfully and deterministically decoding Gabidulin codes. The community views Gabidulin codes as the rank metric analogues of the RS codes, the linearly independent generator of Gabidulin codes corresponds to the mutually different locator of the RS codes, then adapt the decoding algorithms of the RS codes to Gabidulin codes. The linear independence and mutual difference are mainly used to determine the locations that the error occurs. However, once introducing the

change of linear dependence, the situation is rather different, and it intuitively does not to determine the locations that the error occurs. In the case of EG codes, the following questions naturally arise:

1. How can EG codes be decoded directly?
2. What are the decoding performance characteristics of EG codes, including decoding failure rate, decoding capacity, and decoding complexity?
3. Can we adapt existing decoding algorithms for Gabidulin codes to EG codes?
4. How should the interleaved EG codes be defined and decoded?

These problems were partially raised by Berger and Ourivski [13] in 2009. One could note that the EG codes are equivalent to the Augmented Gabidulin (AG) codes [15], where the proposed decoding technique can also be applied to the EG codes (See Subsection 1.3 for more explanations). However, developing other tailored and efficient decoding techniques for the EG codes has been remained open and interesting. In this paper, we try to develop other decoding for the EG codes, and deal with the above problems, exploring the potential of EG codes for designing code-based cryptographic algorithms.

Code-Based Cryptography. Code-based cryptography is a promising candidate for post-quantum cryptography. Three code-based systems using Hamming metric codes, i.e., BIKE, Classic McEliece, and HQC, have advanced to the fourth round of the NIST Post-Quantum Cryptography (NIST PQC) standardization process for potential future adoption [39]. As an alternative, rank-based cryptosystems, which use rank metric error-correcting codes, offer competitive bandwidth [38]. Recent studies [46,6,4,15] have further explored the potential of rank metrics, demonstrating that McEliece [6], NTRU [36,4], and Alekhnovich [46,4] approaches outperform the counterparts of Hamming metric systems in bandwidth and even rival lattice-based cryptosystems. Notably, cryptosystems using the Alekhnovich approach, namely RQC and its variants [34,37,15,46,4], achieve a bandwidth of 1.4 KB for 128-bit security. These developments highlight the promise of rank-based cryptosystems and warrant further investigation.

The RQC Scheme. The Alekhnovich approach was introduced in [1] and allows the security to rely solely on random decoding assumptions. In the early stages, this approach was highly inefficient. A few years later, a more optimized approach was proposed with HQC, which relies on Quasi-Cyclic codes. It has been generalized to rank metric with RQC [34,37]. In RQC, two types of codes are used: the random ideal codes are used to ensure security, and the public Gabidulin codes are used to decrypt the ciphertext. The main advantage compared to McEliece and NTRU is that the security is completely based on random decoding assumptions and is independent of the specific codes whereas McEliece and NTRU require an additional indistinguishability assumption on a specific code for the security of the private key; however, this advantage comes at a price of the short plaintext size (low encryption rate).

RQC was proposed to the NIST PQC in 2017 with competitive sizes. The combinatorial attacks [41,25,8] were once considered to be the most efficient

attacks against the parameters region of RQC. However, it turned out later that the dedicated algebraic attacks [10,12,11], particularly MaxMinors (MM) attack, greatly undermine the concrete security of RQC. The main reason is that RQC must work over a large field whose extension degree m is at least length n of Gabidulin codes, while the MM attack is very powerful over such a large finite field because it is highly probable to solve an overdetermined system. It was just because of the MM attack that RQC was not selected for the third round of NIST PQC. New parameter sets [34] were proposed to provide adequate security against algebraic attacks and still remain competitive sizes. In [46], the authors introduced blockwise errors into RQC to decrease the noise growth, which brings large decoding gains and saves parameters scale. The bandwidth is ultimately reduced to about 2.5 KB.

The more recent works [15,4] enriched the designs of RQC and made a breakthrough in bandwidth. The work [15] (TIT 2024) introduced the AG codes to improve the decoding capacity, and proposed two variants of RQC. The first one is Unstructured RQC with Multiple Syndromes, which we term U-RQC-MS. The U-RQC-MS scheme considers a long-term security and features a conservative design as it relies purely on random decoding problems without any ideal structure. The second one is RQC with Multiple Syndromes, which we term RQC-MS. The RQC-MS scheme is ideally suited for the AG codes and allows a tradeoff between public key size and ciphertext size. By blockwise structure [46] and AG codes, the work [4] further decreased the bandwidth of RQC-MS to 1.4 KB. This bandwidth has outperformed latticed-based Kyber finalized by NIST PQC.

Overall, RQC and its variants feature reliable security and very competitive bandwidths. However, we stress that further optimizations are still desirable for the original RQC. For the improved RQC [4,15]: (1) The original RQC uses only one syndrome while the aforementioned improvements use multiple syndromes, which results in a change of the structure; and (2) The implementation and efficiency are unknown. Moreover, Gabidulin codes used in the original RQC [34,46] impose a strong limitation on the freedom of parameters due to a necessary condition $m \geq n$, which also makes RQC more vulnerable to the MM attack. A very promising optimization routine is to introduce error-correcting codes with a desirable decoding capability (efficient decoding algorithm, negligible decoding failure rate, and high error-correcting capacity) into cryptosystems.

In this paper, our goal is to decode the EG codes by solving the linear reconstruction problem, apply the EG codes to rank-based cryptosystems, and try to optimize the original RQC in both size and efficiency.

1.1 Our Contributions

– We analyze the decoding of the EG codes by solving the Linear Reconstruction (LR) problem. We show that the EG codes possess an efficient decoding algorithm and feature the DFR which can be made arbitrarily small. More interestingly, for appropriate code parameters, the EG codes can exactly decode up to the RGV bound (even close to the minimal distance), which

outperforms Gabidulin codes. From this property, we obtain that by the EG codes, any syndrome and word can be decoded efficiently to an error of weight the RGV bound. We then show that the Welch-Berlekamp algorithm can be adapted to decode the EG codes with the complexity $\mathcal{O}(n^2)$. Our work essentially shows that the decoding conditions of codes defined by q-polynomial are not necessarily as strong as those of Gabidulin codes. For decoding the same errors, the EG codes are more efficient than Gabidulin codes if allowing a DFR. Further, we introduce the Interleaved EG (IEG) codes and analyze its decoding performance. The IEG codes allow to decode more errors with a negligible DFR.

– Building on our decoding approach, we present a method to demonstrate that decoding interleaved Gabidulin codes can be achieved deterministically by solving the Linearized Reconstruction (LR) problem. This addresses an open question left partially unresolved in [31], where the authors provided a proof only for the specific case where the number of columns in the system equals the number of rows plus one. This limitation led to a misconception in subsequent works [44,50,6], which assumed that decoding interleaved Gabidulin codes required *probabilistic* algorithms.
– We apply EG codes to the original RQC (NIST PQC & Asiacrypt 2023) and demonstrate improvements of approximately 50% in both size and efficiency. A very conservative parameters set still remains competitive bandwidths. A detailed comparison with related works is provided in Table 1. All improvements are attributed to our enhanced decoding performance of the EG codes.

1.2 Technical Overview

We first recall the decoding model of the EG codes. Given the EG codes of the dimension k with the generator $\boldsymbol{g}$ of weight t, we assume that the message q-polynomial is $f(x)$ of degree $k-1$, the error occurring at the channel is $\boldsymbol{e}$ and $\|\boldsymbol{e}\|_{\mathrm{R}} \leq r$, the received word is $\boldsymbol{y} = f(\boldsymbol{g}) + \boldsymbol{e}$. The aim of decoding EG codes is to recover $\boldsymbol{e}$ and $f(x)$ from $\boldsymbol{y}$. For any $\boldsymbol{e} = (e_1, e_2, \ldots, e_n) \in \mathbb{F}_{q^m}^n$, we denote the $(s+1) \times n$ Moore matrix of order s of $\boldsymbol{e}$ as

$$\mathbf{Moore}(\boldsymbol{e}, s) = \begin{bmatrix} e_1 & e_2 & \cdots & e_n \\ e_1^q & e_2^q & \cdots & e_n^q \\ \vdots & \vdots & \ddots & \vdots \\ e_1^{q^s} & e_2^{q^s} & \cdots & e_n^{q^s} \end{bmatrix}.$$

Decoding Errors of Exact Weight r. We reduce the decoding EG codes problem to solving the Linear Reconstruction (LR) problem. We show that the decoding EG codes problem is still equivalent to solving the well-known and hard Non-Linear Reconstruction (NLR) problems, then the obtained NLR problem can be reduced to the LR problem. We first analyze the method solving the LR problem by solving a linear system. Directly decoding the EG codes consists in solving the system $\boldsymbol{A}\boldsymbol{x}^\top = \mathbf{0}_n$, where $\boldsymbol{A} = \left[\mathbf{Moore}(\boldsymbol{y}, r)^\top \ \mathbf{Moore}(\boldsymbol{g}, k+r-1)^\top\right]$. The weight value r such that the system has one-dimensional right kernel determines

the decoding capacity of the EG codes. We conduct a comprehensive analysis on the dimension of the right kernel of the system. A key step is that we derive that the matrix $\boldsymbol{A}$ is equivalent to $\left[\mathbf{Moore}(\boldsymbol{e}, r)^\top \mathbf{Moore}(\boldsymbol{g}, k+r-1)^\top\right]$, which greatly facilitates analysis of the right kernel. A careful reader should note that the similar problem is still evaded in previous works [30,31,9] because analyzing the right kernel of the linear system is rather cumbersome and challenging. We find that for appropriate code parameters, the probability that one-dimensional kernel does not occur is negligible. Eventually, the EG codes can decode up to $r = \min\left\{t-k, \left\lfloor \frac{n-k}{2} \right\rfloor\right\}$, which exactly reaches to the RGV bound. We give a specific experiment to verify the decoding capacity and DFR (see Section 5.4).

Decoding Errors of Weight $< r$. A careful reader could note that, in the above decoding idea, we only consider the decoding the maximal-weight error. However, a crucial problem of how to decode smaller errors is omitted because the errors and their weight are unpredictable in practical channels. How to decode the smaller errors ? What the sufficient and necessary condition of successful decoding is ? Trivially, one could try to solve r linear systems by increasing weight values from 1 to r, but this is very cost (about $\mathcal{O}(n^4)$). We analyze this problem for EG codes with a deep level. We show that one actually can decode smaller errors by solving only "one" linear system with the cost of about $\mathcal{O}(n^3)$. Specifically, when the error of weight w occurs ($w < r$), the successful decoding iff the right kernel of $\boldsymbol{A}$ is of dimension $r - w + 1$. Note that the similar problem is omitted in the case of Gabidulin codes (Algorithm 1 [9]), where authors did not consider the sufficient and necessary condition of successful decoding for smaller errors, instead of crudely assuming existence of non-zero right kernel. The remaining problem now is that while the decoding is easy to implement, it costs the complexity of $\mathcal{O}(n^3)$, which is less efficient and should be improved.

Improved Decoding Algorithms. We consider two methods to improve decoding complexity. The first one is the improvement of solving the system $\boldsymbol{A}\boldsymbol{x}^\top = \mathbf{0}_n$. By observing the structure of $\boldsymbol{A}$, its part is independent of the received word $\boldsymbol{y} \in \mathbb{F}_{q^m}^n$ and depends only on the generator $\boldsymbol{g} \in \mathbb{F}_{q^m}^n$. This fact allows us to solve a smaller system with $r+1$ unknowns and $n-k-r$ equations. The decoding complexity is $\mathcal{O}(r^3)$. The Welch-Berlekamp like algorithm is an efficient technique [30,9] solving the LR problem. Our second improvement is that adapting the Welch-Berlekamp like algorithm [30,9] to solve the LR problem, further decode EG codes. The decoding complexity is at most $\mathcal{O}(n^2)$. We refer to Section 6 for the details of improvements.

Interleaved EG Codes and Decoding. We introduce the Interleaved EG (IEG) codes and analyze its decoding performance. Interleaving a code consists in considering several codewords at the same time, corrupted by errors sharing the same support. This specific structure allows to decode more errors. Let $\boldsymbol{y}_i = f_i(\boldsymbol{g}) + \boldsymbol{e}_i, \quad i \in [N]$ where all $\boldsymbol{e}_i$'s share the support of dimension r. The decoding IEG codes problem is that given N words $\boldsymbol{y}_i$'s, the goal is to recover N

q-polynomials $f_i(x)$ and errors $\boldsymbol{e}_i$. We reduce the decoding IEG codes problem to solving the LR problem. We show that if one receives such a set of N words $\boldsymbol{y}_i$'s, it can correct up to $r = \min\left\{t-k, \frac{N(n-k)}{(N+1)}\right\}$ and the DFR can be made arbitrarily small. We refer to Section 7 for details.

The DFR of Interleaved Gabidulin Codes. We analyze the DFR of the interleaved Gabidulin codes by solving the LR problem with Gaussian elimination. Provided that the Gabidulin codes are the special case of the EG codes, by our DFR of the (Interleaved) EG codes, we derive that the obtained linear system has always one-dimensional right kernel, further decoding interleaved Gabidulin codes is deterministic. This problem is left in [31], where the authors only considered the vague case that the number of columns of the system is exactly equal to number of rows plus 1. This leads to a mislead that the most previous works [44,50,6] thought that there exists only *probabilistic* decoding algorithms for interleaved Gabidulin codes. Our DFR is based on a heuristic argument (in Theorem 3). Because the interleaved Gabidulin codes cover the Gabidulin codes when $N = 1$, if our heuristic argument is invalid, then decoding Gabidulin codes by the Gaussian elimination would not be deterministic. This challenges the well-known conclusion: decoding Gabidulin codes is deterministic, which has stood for over 40 years. The details are presented in Section 8.

Applications to RQC. We apply the EG codes to rank-based cryptosystem RQC (NIST PQC & Asiacyrpt 2023) without any structural changes such as multiple syndromes. This optimization is not considered in [15] and conference version of [4]. We note that, very recently, RQC used AG codes is added into the eprint version of [4]. As concurrent and independent work, we obtain an almost same bandwidth, especially we provide a practical and efficient implementation by exploiting the explicit decoding algorithm of our EG codes. Recall that Gabidulin codes used in the previous RQC require $m \geq n$, which makes RQC vulnerable to powerful MM attacks (Eurocrypt 2020, Asiacrypt 2020 & 2023) and imposes a strong limitation on security parameters. The use of the EG codes can alleviate this case. As the EG codes can work over a smaller finite field and allow $m < n$, our RQC can efficiently mitigate the advantage of the MM attack and has more degree of freedom while choosing security parameters. This leads to a significant improvement in both size and efficiency. A detailed comparison with several classic code-based PKEs and lattice-based Kyber is summarized in Table 1. We consider two types of parameters set: **Our RQC** for the current attack and **Our Conservative RQC** for the potential attacks in the future.

For 128-bit security, our RQC has a bandwidth of 1690 bytes, which is about 69% and 34% more compact than RQC (NIST PQC) and RQC (Asiacrypt 2023), respectively. The improvement is more significant for higher security levels. For 192-bit security, we obtain about 71% and 55% improvement, respectively. We refer to Table 4 in Section 9 for the improvement in efficiency. Our RQC achieves about 60% improvement in timings over RQC (NIST PQC).

Table 1. Comparisons of RQC, HQC, BIKE, Classic McEliece, and Kyber.

Schemes	Security Level	pt (bits)	sk (bytes)	pk (bytes)	ct (bytes)	total (bytes)	DFR
Our RQC	128 (2^{163})	159	40	590	1100	1690	2^{-133}
	192 (2^{192})	236	40	837	1594	2431	2^{-202}
	256 (2^{262})	292	40	1291	2566	3857	2^{-258}
Our Conservative RQC	128 (2^{167})	171	40	796	1512	2308	2^{-138}
	192 (2^{243})	249	40	1711	3342	5053	2^{-207}
	256 (2^{281})	339	40	3190	6300	9490	2^{-274}
RQC (Asiacrypt [46])	128 (2^{127})	581	40	860	1704	2564	-
	192 (2^{214})	381	40	1834	3652	5486	-
	256 (2^{267})	417	40	2421	4826	7247	-
RQC (NIST [34])	128	381	40	1834	3652	5486	-
	192	755	40	2853	5690	8543	-
	256	543	40	4090	8164	12254	-
RQC-MS (PQC [4])	128 (2^{145})	129	40	320	1118	1438	2^{-145}
	192 (2^{206})	201	40	610	2278	2888	2^{-206}
HQC (NIST [35])	128	128	40	2249	4497	6746	2^{-128}
BIKE (NIST [2])	128	256	281	1541	1573	3114	2^{-128}
Classic McEliece (NIST [14])	128	256	6492	261120	96	261216	-
Kyber512 (NIST [43])	118	256	1632	800	768	1568	2^{-139}
Kyber768 (NIST [43])	182	256	2400	1184	1088	2272	2^{-164}
Kyber1024 (NIST [43])	256	256	3168	1568	1568	3136	2^{-174}

pt: plaintext size or encryption rate; sk: private key size; pk: public key size; ct: ciphertext size; total: bandwidth (pk + ct). In column "Security Level", the practical security strength is given in the bracket.

Compared to RQC-MS [4], our bandwidth is slightly shorter for 192-bit security. Moreover, we improve the original RQC and do not consider any structural changes because the original RQC might have more applications due to its flexible structure with only one syndrome.

Compared to the NIST Round 4 code-based submissions: HQC, BIKE, and Classic McEliece, our bandwidth is the smallest, and we obtain at least 75% improvement over HQC. We only present submissions of 128-bit security. In fact, for 192-bit and 256-bit security, the bandwidth of our RQC still remains optimal. Compared to Kyber finalized by NIST PQC, our RQC features comparable performance, particularly, shorter key sizes and smaller DFRs.

Consider that the complexity of the blockwise rank decoding problem [46,4] needs more time to mature, to avoid potential accelerated attacks in the future, we also choose a very conservative set of parameters for RQC (**Our Conservative RQC**). The bandwidths still outperform HQC, BIKE, and Classic McEliece. Compared to Kyber, our conservative RQC also features comparable key sizes and smaller DFRs.

Overall, unlike Kyber, RQC lacks of the ciphertext compress technique and the accelerated implementation. Developing efficient decoding algorithms for the EG codes is crucial for the accelerated implementation.

1.3 Comparisons with the Decoding of the AG Codes [15]

Considering the EG codes are equivalent to the Augmented Gabidulin (AG) codes [15]. In this section, we highlight some differences with the decoding technique (Proposition 2 in [15]) of the AG codes. We first stress some objective facts. For some families of codes, the DFR and that decoding complexity depends on the specific decoding techniques, instead of the equivalence property of codes. Naturally, the different decoding techniques could lead to different DFR and decoding complexity. As in [31] and [44], by different decoding techniques, authors presented different DFR and decoding complexity for the Interleaved Gabidulin codes. Again, for Gabidulin codes, Extended Euclidean Algorithm for Linearized polynomials (LEEA)[49], Gao-like algorithm [48], and Welch-Berlekamp like algorithm [30,9] present different decoding complexity. Thus it should be encouraged to develop different techniques for decoding a family of codes.

We now recall the AG codes [15]. In the definitions of the EG codes, if we set the first t coordinates of $\boldsymbol{g}$ to be linearly independent and the last $n-t$ coordinates to be zeros, then the EG codes are exactly the AG codes. In the case of the rank metric, there exists an invertible matrix on the base field $\mathbb{F}_q$ which permits to turn a code into the other and such an invertible matrix is called an isometry. Assume that one obtains an EG code with generator $\boldsymbol{g}$ of weight t, then it is possible to apply an isometry to turn n coordinates of $\boldsymbol{g}$ into t linearly independent coordinates plus $n-t$ zero coordinates. The n new coordinates define an AG code that is equivalent to the EG code. While the work [15] serves as a pioneering effort in providing a technique to decode the EG (AG) codes, developing other efficient decoding techniques is still interesting. In this paper, we adopt different decoding idea. The differences with [15] are as follows.

First, the decoding techniques are obviously different. The technique [15] decoding the AG codes can be viewed as the first specific method to decode the EG codes. The authors in [15] used the support erasures technique exposing errors' support, and reduced the decoding EG code problem to solving a linear system with exposed errors' support. However, it seems to be hard to improve this decoding complexity. They must solve a linear system with an implicit structure (Equation (3), [15]) due to adding of exposed errors' support. Currently, it is at least unknown whether there are more efficient techniques to solve such a linear system. Differently, we do not consider any support erasures technique exposing errors' support. We reduce the decoding EG code problem to solving the LR problem, try the underlying techniques solving the LR problem, further improve decoding efficiency. While directly solving the LR problem also consists in solving a linear system by Gaussian elimination, the structure of the linear system is explicit, which is beneficial for observing the essence of decoding and developing efficient decoding algorithms. Specifically, once the EG code can be

decoded by solving the LR problem, one could develop and adapt efficient algorithms solving the LR problem to decode the EG codes such as the (improved) Gaussian elimination [30] and the Welch-Berlekamp like algorithm [30,9]. As a result, by solving the LR problem with the (improved) Gaussian elimination and the Welch-Berlekamp like algorithm, we obtain a lower decoding complexity ($\mathcal{O}(r^3)$ and $\mathcal{O}(n^2)$) than the support erasures technique. The Gaussian elimination costs $\mathcal{O}(n^3)$, the improved cost is about $\mathcal{O}(r^3)$, and the Welch-Berlekamp like algorithm costs $\mathcal{O}(n^2)$, while decoding AG codes in [15] costs about $\mathcal{O}(m^3)$. Note that $n \geq m > r$. More importantly, the (improved) Gaussian elimination and the Welch-Berlekamp like algorithm can be easily implemented.

Second, we definitely show that EG codes can "exactly" decode up to the errors of weight the RGV bound, instead of "asymptotical" upper bound (means a might gap). This property is not mentioned in the case of the AG codes [15]. The codes "exactly" decoding up to the RGV bound is very interesting for code-based cryptography as such codes are one of the key techniques designing efficient code-based hash-sign signatures. For example, the low signing efficiency of code-based hash-sign signature, CFS, roots in that Goppa code only "asymptotically" decodes up to the Hamming GV bound as the upper bound, instead of "exactly". This must try to find decodable syndrome in many times to obtain a signature. Our work definitely shows that the EG codes can overcome this obstacle since any syndrome can be decoded with an overwhelming probability. The remaining challenge is technique securely hiding EG codes. We note that an interesting hiding technique for Gabidulin codes was developed in recent work [5], which seems to be a solution to securely hiding EG codes.

Third, we introduce the Interleaved EG codes and analyze its decoding performance. The IEG codes can correct up to $r = \min\left\{t-k, \frac{N(n-k)}{(N+1)}\right\}$ and the DFR can be made arbitrarily small. By our DFR of the (I)EG codes, we obtain the null DFR of the interleaved Gabidulin codes by solving the LR problem with Gaussian elimination. This addresses an open question left partially unresolved in [31], where the authors provided a proof only for the specific case where the number of columns in the system equals the number of rows plus one.

1.4 Organization

In Section 2, we present some notations and recall some preliminaries. Section 3 proves some useful results for estimating the decoding failure rate. We give the EG codes in Section 4 together with a reduction from decoding EG code problem to the LR problem. Section 5 analyzes the decoding algorithm, decoding complexity, decoding failure rate, and decoding capacity. Section 6 presents two improvements of decoding algorithms. Section 7 introduces and analyzes the interleaved EG codes. In Section 8, we show the relation between the interleaved EG codes and the interleaved Gabidulin codes. In Section 9, we apply the EG codes to improve RQC. We conclude this paper in Section 10.

2 Preliminaries

In this section, we define some notations, and recall $\mathbb{F}_{q^m}$-linear codes and q-polynomials.

2.1 Notations

We denote by $\mathbb{N}$ the set of non-negative integer numbers, q prime or prime power, and $\mathbb{F}_{q^m}$ an extension of degree m of the finite field $\mathbb{F}_q$. We denote by $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_m) \in \mathbb{F}_{q^m}^m$ a basis of $\mathbb{F}_{q^m}$ viewed as an m-dimensional vector space over $\mathbb{F}_q$. Vectors (resp. matrices) are denoted by lower-case (resp. upper-case) bold letters. We use the notation $[N] := \{1, 2, \ldots, N\}$ for the first N natural numbers. Let $\mathcal{A}$ be an algorithm. We say that $\mathcal{A}$ is a PPT algorithm if it is a probabilistic polynomial-time algorithm. The linear span of a set of vectors $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m$ is denoted by $\langle \boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m \rangle$. The row space and the rank of a matrix $\boldsymbol{X} \in \mathbb{F}_q^{m \times n}$ are denoted by $\langle \boldsymbol{X} \rangle$ and $\mathsf{rank}(\boldsymbol{X})$, respectively. By definition, $\mathsf{rank}(\boldsymbol{X}) = \mathsf{dim}\langle \boldsymbol{X} \rangle$.

2.2 $\mathbb{F}_{q^m}$-Linear Codes with Rank Metric

Definition 1 (Rank Metric). *For $\boldsymbol{x} = (x_1, x_2, \ldots, x_n) \in \mathbb{F}_{q^m}^n$, each coordinate x_i is associated to a vector of $\mathbb{F}_q^m$ w.r.t. the basis $\boldsymbol{\alpha}$. Then $\boldsymbol{x}$ is associated to an $m \times n$ matrix given by $\mathbf{Mat}(\boldsymbol{x}) = (x_{ij})_{i \in [m], j \in [n]}$: $\boldsymbol{x} = \boldsymbol{\alpha}\mathbf{Mat}(\boldsymbol{x})$.* The rank weight $\|\boldsymbol{x}\|_{\mathrm{R}}$ *of $\boldsymbol{x}$ is defined as the rank of $\mathbf{Mat}(\boldsymbol{x})$. The rank distance $d_{\mathrm{R}}(\boldsymbol{x}, \boldsymbol{y})$ between $\boldsymbol{x}$ and $\boldsymbol{y}$ in $\mathbb{F}_{q^m}^n$ is defined by $d_{\mathrm{R}}(\boldsymbol{x}, \boldsymbol{y}) := \|\boldsymbol{x} - \boldsymbol{y}\|_{\mathrm{R}}$.*

Let $\langle x_1, x_2, \ldots, x_n \rangle_{\mathbb{F}_q}$ be the $\mathbb{F}_q$-linear subspace of $\mathbb{F}_{q^m}$ generated by linear combinations over $\mathbb{F}_q$ of coordinates of $\boldsymbol{x}$. *The support* $\mathrm{Supp}(\boldsymbol{x})$ of $\boldsymbol{x}$ is defined as the $\mathbb{F}_q$-linear subspace $\langle x_1, x_2, \ldots, x_n \rangle_{\mathbb{F}_q}$, i.e., $\mathrm{Supp}(\boldsymbol{x}) = \langle x_1, x_2, \ldots, x_n \rangle_{\mathbb{F}_q}$. It follows from definition that $\|\boldsymbol{x}\|_{\mathrm{R}} = \dim(\mathrm{Supp}(\boldsymbol{x}))$. The set of such errors of weight r and length n is denoted by $\mathcal{S}_r^n$.

Support and Coefficient Matrices of the Error. For an error $\boldsymbol{e} \in \mathcal{S}_r^n$, let $\boldsymbol{\varepsilon} = (\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_r) \in \mathbb{F}_{q^m}^r$ be a basis of $\mathrm{Supp}(\boldsymbol{e})$, then there exists a matrix $\boldsymbol{C} \in \mathbb{F}_q^{r \times n}$ of rank r such that $\boldsymbol{e} = \boldsymbol{\varepsilon}\boldsymbol{C}$. Under the basis $\boldsymbol{\alpha}$, there exists a matrix $\boldsymbol{S} \in \mathbb{F}_q^{m \times r}$ of rank r such that $\boldsymbol{\varepsilon} = \boldsymbol{\alpha}\boldsymbol{S}$. Then $\boldsymbol{e} = \boldsymbol{\alpha}\boldsymbol{S}\boldsymbol{C}$. We call $\boldsymbol{S}$ and $\boldsymbol{C}$ respectively support matrix and coefficient matrix, and denote these two matrices as $\mathbf{SM}(\boldsymbol{e})$ and $\mathbf{CM}(\boldsymbol{e})$, respectively. From Definition 1, under the same basis $\boldsymbol{\alpha}$, $\boldsymbol{e} = \boldsymbol{\varepsilon}\mathbf{CM}(\boldsymbol{e}) = \boldsymbol{\alpha}\mathbf{SM}(\boldsymbol{e})\mathbf{CM}(\boldsymbol{e}) = \boldsymbol{\alpha}\mathbf{Mat}(\boldsymbol{e})$.

Definition 2 ($\mathbb{F}_{q^m}$-Linear Codes with Rank Metric). *An $\mathbb{F}_{q^m}$-linear code embedded with rank metric of length n and dimension k is a subspace of dimension k of $\mathbb{F}_{q^m}^n$. Such $\mathbb{F}_{q^m}$-linear codes are denoted by $[n, k]_{q^m}$.*

Given an $[n, k]_{q^m}$-linear code $\mathcal{C}$, a matrix $\boldsymbol{G} \in \mathbb{F}_{q^m}^{k \times n}$ is called *generator matrix* iff $\mathcal{C} = \left\{\boldsymbol{m}\boldsymbol{G} \ : \ \boldsymbol{m} \in \mathbb{F}_{q^m}^k\right\}$ and a matrix $\boldsymbol{H} \in \mathbb{F}_{q^m}^{(n-k) \times n}$ is called *parity-check matrix* iff $\mathcal{C} = \left\{\boldsymbol{c} \in \mathbb{F}_{q^m}^n \ : \ \boldsymbol{H}\boldsymbol{c}^\top = \boldsymbol{0}\right\}$. The systematic forms of $\boldsymbol{G}$ and $\boldsymbol{H}$ are respectively defined as $[\boldsymbol{I}_k \ \boldsymbol{P}]$ and $\left[-\boldsymbol{P}^\top \ \boldsymbol{I}_{n-k}\right]$ where $\boldsymbol{P} \in \mathbb{F}_{q^m}^{k \times (n-k)}$.

2.3 q-Polynomial and Annihilator Polynomial

The construction of the EG code uses q-polynomials introduced originally by Ore in the 1930s [40].

Definition 3 (q-Polynomial). *A q-polynomial of q-degree n over $\mathbb{F}_{q^m}$ is a polynomial of the form $f(x) = \sum_{i=0}^{n} f_i x^{q^i}$ for $f_i \in \mathbb{F}_{q^m}$, $f_n \neq 0$.*

We denote the set of q-polynomials by $\mathcal{L}[x]$ and denote the q-degree of a q-polynomial $f(x)$ by $\mathsf{deg}_q f(x)$. Let $\mathcal{L}_{\leq r}[x]$ be the set of q-polynomials of q-degree $\leq r$. Let $a(x) = \sum_i a_i x^{q^i}$ and $b(x) = \sum_i b_i x^{q^i} \in \mathcal{L}[x]$:

- The addition "$+$" of $a(x)$ and $b(x)$: $a(x) + b(x) = \sum_i (a_i + b_i) x^{q^i}$;
- The symbolic product "$\circ$" of $a(x)$ and $b(x)$: $a(x) \circ b(x) = a(b(x))$.

It is well-known that the set of q-polynomials together with addition and symbolic product forms a noncommutative ring. In such a ring, the symbolic product is associative and distributive w.r.t. both right and left product. The identity element is $I(x) = x$ and there are no divisors of zero, i.e., $a(x) \circ b(x) = 0$ implies $a(x) = 0$ or $b(x) = 0$. A q-polynomial $c(x)$ is said to be symbolically right divisible by $b(x)$ if $c(x) = a(x) \circ b(x)$. When $c(x) = a(x) \circ b(x)$, we say that $c(x)$ is symbolically left divisible by $a(x)$, which is denoted by $b(x) = a(x) \setminus c(x)$.

If $a(x) = \sum_{i=0}^{r} a_i x^{q^i}$, $b(x) = \sum_{i=0}^{s} b_i x^{q^i}$, and $c(x) = a(x) \circ b(x) = \sum_{i=0}^{s+r} c_i x^{q^i}$, one easily checks that

$$\begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_{s+r} \end{bmatrix} = \underbrace{\begin{bmatrix} b_0 & 0 & \cdots & 0 \\ b_1 & b_0^q & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ b_s & b_{s-1}^q & \cdots & b_0^{q^r} \\ 0 & b_s^q & \cdots & b_1^{q^r} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & b_s^{q^r} \end{bmatrix}}_{:=\mathbf{PM}(b(x),r) \in \mathbb{F}_{q^m}^{(s+r+1)\times(r+1)}} \times \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_r \end{bmatrix} = \mathbf{PM}(b(x), r) \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_r \end{bmatrix}.$$

A q-polynomial $f \in \mathcal{L}[x]$ satisfies:

- $\forall\ x_1, x_2 \in \mathbb{F}_{q^m}$, $\beta_1, \beta_2 \in \mathbb{F}_q$, $f(\beta_1 x_1 + \beta_2 x_2) = \beta_1 f(x_1) + \beta_2 f(x_2)$.
- If x_1 and x_2 are any two roots of f, then $f(x_1) = f(x_2) = f(x_1 + x_2) = 0$.

These two properties imply that all roots of a q-polynomial span an $\mathbb{F}_q$-subspace of $\mathbb{F}_{q^m}$. Proposition 1 shows that the q-degree of a q-polynomial is lower bounded by the dimension of its root space. Definition 4 shows that given an $\mathbb{F}_q$-subspace of $\mathbb{F}_{q^m}$, there exists a unique monic q-polynomial vanishing on such an $\mathbb{F}_q$-subspace.

Proposition 1. *Let $f(x) \in \mathcal{L}[x]$ be a q-polynomial. Assume that $e_1, e_2, \ldots, e_n \in \mathbb{F}_{q^m}$ span an $\mathbb{F}_q$-subspace of $\mathbb{F}_{q^m}$ of dimension r. If $f(e_i) = 0$ for all $i \in [n]$, then either $f(x) = 0$, or $\mathsf{deg}_q f(x) \geqslant r$.*

Definition 4 (Annihilator Polynomial). *For any $\mathbb{F}_q$-subspace E of $\mathbb{F}_{q^m}$ of dimension r, there exists a unique monic q-polynomial $\mathcal{A}_E(x)$ of q-degree r that vanishes on E, i.e., $\mathcal{A}_E(x) = 0$ for any $x \in E$. Such a monic q-polynomial is called the annihilator polynomial of E.*

Since the coordinates of any error of weight r uniquely span a r-dimensional $\mathbb{F}_q$-subspace of $\mathbb{F}_{q^m}$, i.e., its r-dimensional support, we have Definition 5.

Definition 5 (Annihilator Polynomial of Vector). *For any $\boldsymbol{e} = (e_1, e_2, \ldots, e_n) \in \mathcal{S}_r^n$, there exists a unique annihilator polynomial of q-degree r that vanishes on $\boldsymbol{e}$, i.e., $\mathcal{A}_{\boldsymbol{e}}(e_i) = 0$ for all $i \in [n]$. We defined the annihilator polynomial of $\boldsymbol{e}$ by $\mathcal{A}_{\boldsymbol{e}}(x)$.*

Proposition 2 shows that the dimension of $\mathbb{F}_q$-subspace spanned by some roots of a q-polynomial is upper bounded by its q-degree.

Proposition 2. *Let $f(x)$ be a q-polynomial of q-degree r. Let $\{e_i : f(e_i) = 0, i \in [n]\}$ be a set consisting of any n roots of $f(x)$. Then the dimension of the $\mathbb{F}_q$-subspace spanned by such a set is at most r, in other words, if $\boldsymbol{e} = (e_1, e_2, \ldots, e_n)$, then $\|\boldsymbol{e}\|_{\mathrm{R}} \leq r$.*

3 Our Results on Rank of Moore Matrix

For analyzing the decoding capacity and properties of the EG codes, in this section, we derive some results on the rank of the Moore matrix. We first recall the relation between q-polynomial and Frobenius automorphism.

The q-polynomial is essentially related to Frobenius automorphism from $\mathbb{F}_{q^m}$ to $\mathbb{F}_{q^m}$. We denote such a Frobenius automorphism by θ:

$$\begin{aligned} \theta : \quad \mathbb{F}_{q^m} &\to \mathbb{F}_{q^m} \\ x &\mapsto \theta(x) := x^q. \end{aligned}$$

The Frobenius automorphism θ is an $\mathbb{F}_q$-automorphism of $\mathbb{F}_{q^m}$. We use its following properties:

- $\forall\ x_1, x_2 \in \mathbb{F}_{q^m},\ \beta_1, \beta_2 \in \mathbb{F}_q$: $\theta(x_1 x_2) = \theta(x_1)\theta(x_2)$, $\theta(\beta_1 x_1 + \beta_2 x_2) = \beta_1\theta(x_1) + \beta_2\theta(x_2)$ (linearity over $\mathbb{F}_q$),
- $\forall\ i \in \mathbb{N}$, $\theta^i(x) = \theta^{i-1}(\theta(x)) = \theta^{i-1}(x^q) = \cdots = \theta\left(x^{q^{i-1}}\right) = x^{q^i}$,
- $\forall\ i \in \mathbb{N}$, θ^i is also an Frobenius automorphism. This is because the set of all Frobenius automorphisms is a cyclic group of order m.

A q-polynomial in variable x can be viewed as a polynomial of θ acting on the variable x. The addition (resp. symbolic product "$\circ$") of two q-polynomials corresponds to the addition (resp. composition "$\bullet$") of two polynomials of θ. Let $a(x) = \sum_i a_i x^{q^i}$ and $b(x) = \sum_i b_i x^{q^i} \in \mathcal{L}[x]$. Let $a(\theta) = \sum_i a_i \theta^i$ and $b(\theta) = \sum_i b_i \theta^i$.

- $a(\theta)(x) = \left(\sum_i a_i\theta^i\right)(x) = \sum_i a_i\theta^i(x) = \sum_i a_i x^{q^i} = a(x)$,
- $(a(\theta) + b(\theta))(x) = \left(\sum_i (a_i + b_i)\,\theta^i\right)(x) = \sum_i (a_i + b_i)\,x^{q^i} = a(x) + b(x)$,
- $(a(\theta) \bullet b(\theta))(x) = a(b(\theta))(x) = a(b(x)) = a(x) \circ b(x)$.

Let $\boldsymbol{x} = (x_1, x_2, \ldots, x_n) \in \mathbb{F}_{q^m}^n$. The Frobenius automorphism θ can be applied to $\boldsymbol{x}$ coordinate-wise: $\theta(\boldsymbol{x}) = (\theta(x_1), \theta(x_2), \ldots, \theta(x_n)) = (x_1^q, x_2^q, \ldots, x_n^q)$. Further, for any $i \in \mathbb{N}$, $\theta^i(\boldsymbol{x}) = \left(\theta^i(x_1), \theta^i(x_2), \ldots, \theta^i(x_n)\right) = \left(x_1^{q^i}, x_2^{q^i}, \ldots, x_n^{q^i}\right)$.

Definition 6. *Let $s \in \mathbb{N}$. For any $\boldsymbol{e} = (e_1, e_2, \ldots, e_n) \in \mathbb{F}_{q^m}^n$, we denote the $(s+1) \times n$ Moore matrix of order s of $\boldsymbol{e}$ as*

$$\mathbf{Moore}(\boldsymbol{e}, s) = \begin{bmatrix} e_1 & e_2 & \cdots & e_n \\ e_1^q & e_2^q & \cdots & e_n^q \\ \vdots & \vdots & \ddots & \vdots \\ e_1^{q^s} & e_2^{q^s} & \cdots & e_n^{q^s} \end{bmatrix} = \begin{bmatrix} \boldsymbol{e} \\ \theta(\boldsymbol{e}) \\ \vdots \\ \theta^s(\boldsymbol{e}) \end{bmatrix}.$$

Proposition 3. *Let $s \in \mathbb{N}$ and $\boldsymbol{e} \in \mathcal{S}_r^n$. Let $\boldsymbol{\varepsilon} \in \mathbb{F}_{q^m}^r$ be a basis of* $\mathrm{Supp}(\boldsymbol{e})$. *Let $\mathbf{CM}(\boldsymbol{e}) \in \mathbb{F}_q^{r \times n}$ of rank r be the coefficient matrix of $\boldsymbol{e}$ under $\boldsymbol{\varepsilon}$ such that $\boldsymbol{e} = \boldsymbol{\varepsilon}\mathbf{CM}(\boldsymbol{e})$. Then $\mathbf{Moore}(\boldsymbol{e}, s) = \mathbf{Moore}(\boldsymbol{\varepsilon}, s) \cdot \mathbf{CM}(\boldsymbol{e})$.*

Proof. Since for any $i \in \mathbb{N}$, θ^i is an $\mathbb{F}_q$-automorphism, $\theta^i(\boldsymbol{e}) = \theta^i\left(\boldsymbol{\varepsilon}\mathbf{CM}(\boldsymbol{e})\right) = \theta^i(\boldsymbol{\varepsilon})\mathbf{CM}(\boldsymbol{e})$. Thus, we have

$$\mathbf{Moore}(\boldsymbol{e}, s) = \begin{bmatrix} \boldsymbol{e} \\ \theta(\boldsymbol{e}) \\ \vdots \\ \theta^s(\boldsymbol{e}) \end{bmatrix} = \begin{bmatrix} \boldsymbol{\varepsilon}\mathbf{CM}(\boldsymbol{e}) \\ \theta(\boldsymbol{\varepsilon})\mathbf{CM}(\boldsymbol{e}) \\ \vdots \\ \theta^s(\boldsymbol{\varepsilon})\mathbf{CM}(\boldsymbol{e}) \end{bmatrix} = \mathbf{Moore}(\boldsymbol{\varepsilon}, s) \cdot \mathbf{CM}(\boldsymbol{e}).$$

□

Proposition 4. *Let $s \in \mathbb{N}$. If $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_m) \in \mathbb{F}_{q^m}^m$ is a basis of $\mathbb{F}_{q^m}$ viewed as an m-dimensional vector space over $\mathbb{F}_q$, then the rank of* $\mathbf{Moore}(\boldsymbol{\alpha}, s)$ *satisfies:*

$$\mathsf{rank}\left(\mathbf{Moore}(\boldsymbol{\alpha}, s)\right) = \begin{cases} s+1, & \text{if } s < m; \\ m, & \text{if } s \geq m. \end{cases} \tag{1}$$

Proof. By the definition of Moore matrix in Definition 6,

$$\mathbf{Moore}(\boldsymbol{\alpha}, s) = \begin{bmatrix} \alpha_1 & \alpha_2 & \cdots & \alpha_m \\ \alpha_1^q & \alpha_2^q & \cdots & \alpha_m^q \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_1^{q^s} & \alpha_2^{q^s} & \cdots & \alpha_m^{q^s} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\alpha} \\ \theta(\boldsymbol{\alpha}) \\ \vdots \\ \theta^s(\boldsymbol{\alpha}) \end{bmatrix}.$$

- **Case** $s < m$**:** If $\mathsf{rank}(\mathbf{Moore}(\boldsymbol{\alpha}, s)) \leq s$, then there exist a non-zero vector $(a_0, a_1, \ldots, a_s) \in \mathbb{F}_{q^m}^s$ such that $\sum_{i=0}^{s} a_i \theta^i(\boldsymbol{\alpha}) = 0$. This means that there exist a q-polynomial $\sum_{i=0}^{s} a_i x^{q^i}$ of q-degree $\leq s$ that vanishes on $\boldsymbol{\alpha}$. Since $\|\boldsymbol{\alpha}\|_{\mathrm{R}} = m$, by Definition 5, the q-degree of annihilator polynomial of $\boldsymbol{\alpha}$ must be m. This leads a contradiction, thus $\mathsf{rank}(\mathbf{Moore}(\boldsymbol{\alpha}, s)) = s + 1$.
- **Case** $s = m$**:** By the case of $s < m$, we have $\mathsf{rank}(\mathbf{Moore}(\boldsymbol{\alpha}, m-1)) = m$, further

$$\begin{aligned}\mathsf{rank}(\mathbf{Moore}(\boldsymbol{\alpha}, m)) &= \mathsf{rank}\left(\begin{bmatrix}\mathbf{Moore}(\boldsymbol{\alpha}, m-1) \\ \theta^m(\boldsymbol{\alpha})\end{bmatrix}\right) \\ &\geq \mathsf{rank}(\mathbf{Moore}(\boldsymbol{\alpha}, m-1)) = m.\end{aligned}$$

  Moreover, $\mathsf{rank}(\mathbf{Moore}(\boldsymbol{\alpha}, m)) \leq m$ because the number of columns of $\mathbf{Moore}(\boldsymbol{\alpha}, m)$ is m. Thus, $\mathsf{rank}(\mathbf{Moore}(\boldsymbol{\alpha}, m)) = m$.
- **Case** $s > m$**:** By the case of $s = m$, we have $\mathsf{rank}(\mathbf{Moore}(\boldsymbol{\alpha}, m)) = m$, further

$$\mathsf{rank}(\mathbf{Moore}(\boldsymbol{\alpha}, s)) = \mathsf{rank}\left(\begin{bmatrix}\mathbf{Moore}(\boldsymbol{\alpha}, m) \\ \theta^{m+1}(\boldsymbol{\alpha}) \\ \vdots \\ \theta^s(\boldsymbol{\alpha})\end{bmatrix}\right) \geq \mathsf{rank}(\mathbf{Moore}(\boldsymbol{\alpha}, m)) = m.$$

  Similarly, $\mathsf{rank}(\mathbf{Moore}(\boldsymbol{\alpha}, s)) \leq m$ because the number of columns of $\mathbf{Moore}(\boldsymbol{\alpha}, s)$ is m. Thus, $\mathsf{rank}(\mathbf{Moore}(\boldsymbol{\alpha}, s)) = m$.

Finally, we obtain a conclusion in Equation (1). □

Corollary 1. *Let $s \in \mathbb{N}$. Let $\boldsymbol{\varepsilon} = (\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_r) \in \mathbb{F}_{q^m}^r$ be a basis of any r-dimensional $\mathbb{F}_q$-subspace of $\mathbb{F}_{q^m}$. Then the rank of $\mathbf{Moore}(\boldsymbol{\varepsilon}, s)$ satisfies:*

$$\mathsf{rank}\left(\mathbf{Moore}(\boldsymbol{\varepsilon}, s)\right) = \begin{cases} s+1, & \text{if } s < r; \\ r, & \text{if } s \geq r. \end{cases} \tag{2}$$

Proposition 5. *Let $s \in \mathbb{N}$. For any $\boldsymbol{e} \in \mathcal{S}_r^n$, the rank of $\mathbf{Moore}(\boldsymbol{e}, s)$ satisfies:*

$$\mathsf{rank}\left(\mathbf{Moore}(\boldsymbol{e}, s)\right) = \begin{cases} s+1, & \text{if } s < r; \\ r, & \text{if } s \geq r. \end{cases} \tag{3}$$

Proof. Let $\boldsymbol{\varepsilon} \in \mathbb{F}_{q^m}^r$ be a basis of $\mathrm{Supp}(\boldsymbol{e})$. Let $\mathbf{CM}(\boldsymbol{e}) \in \mathbb{F}_q^{r \times n}$ of rank r be the coefficient matrix of $\boldsymbol{e}$ under $\boldsymbol{\varepsilon}$ such that $\boldsymbol{e} = \boldsymbol{\varepsilon}\mathbf{CM}(\boldsymbol{e})$. By Proposition 3, for any $s \in \mathbb{N}$, $\mathbf{Moore}(\boldsymbol{e}, s) = \mathbf{Moore}(\boldsymbol{\varepsilon}, s) \cdot \mathbf{CM}(\boldsymbol{e})$. Since $\mathbf{CM}(\boldsymbol{e}) \in \mathbb{F}_q^{r \times n}$ is a row full-rank matrix, we have

$$\mathsf{rank}(\mathbf{Moore}(\boldsymbol{e}, r)) = \mathsf{rank}(\mathbf{Moore}(\boldsymbol{\varepsilon}, s)) = \begin{cases} s+1, & \text{if } s < r; \\ r, & \text{if } s \geq r. \end{cases}$$

The last equality uses Corollary 1. □

4 Decoding EG Code to Solving LR Problem

In this section, we recall the definition of the EG codes [13], and reduce decoding EG Codes to solving LR Problem.

We slightly relax the conditions with a wider code parameters region: the weight of a generator $\leq \min\{m, n\}$, which covers Gabidulin codes.

Definition 7 (EG Code). *Let q, m, n, t, k be integers and $k \leq t \leq \min\{n, m\}$. Let $\boldsymbol{g} = (g_1, g_2, \ldots, g_n) \in \mathbb{F}_{q^m}^n$ be the generator of weight t. The EG code of dimension k and length n generated by $\boldsymbol{g}$ is defined as*

$$\mathrm{EG}_k(\boldsymbol{g}) = \Big\{ f(\boldsymbol{g}) = (f(g_1), f(g_2), \ldots, f(g_n)) : f(x) \in \mathcal{L}_{\leq k-1}[x] \Big\}.$$

The dimension and minimal distance of the EG codes is formally presented in Propositions 6 and 7. The proofs are put in Appendices A.3 and A.4.

Proposition 6. *The dimension of the $\mathrm{EG}_k(\boldsymbol{g})$ codes is k.*

Proposition 7. *The minimal distance of the $\mathrm{EG}_k(\boldsymbol{g})$ codes is $t - k + 1$.*

Next, we reduce the decoding EG codes to solving the LR problem.

4.1 From Decoding EG Codes to Solving NLR Problem

Given the EG codes in Definition 7 (the generator $\boldsymbol{g}$ of weight t), we assume that the message q-polynomial is $f(x) \in \mathcal{L}_{\leq k-1}[x]$, the error occurring at the channel is $\boldsymbol{e}$ and $\|\boldsymbol{e}\|_{\mathrm{R}} \leq r$, the received word is $\boldsymbol{y} = f(\boldsymbol{g}) + \boldsymbol{e}$. The aim of decoding EG codes is to recover $\boldsymbol{e}$ and $f(x)$ from $\boldsymbol{y}$. We define formally the decoding EG code problem as follows.

Definition 8 (Decoding EG Code Problem $\mathsf{DecEGCode}(\boldsymbol{g}, \boldsymbol{y})$).
Input : $\boldsymbol{g} = (g_1, g_2, \ldots, g_n) \in \mathbb{F}_{q^m}^n$ *and* $\|\boldsymbol{g}\|_{\mathrm{R}} = t$; $\boldsymbol{y} = (y_1, y_2, \ldots, y_n) \in \mathbb{F}_{q^m}^n$.
Output : $f(x) \in \mathcal{L}[x]$ *and* $\boldsymbol{e} \in \mathbb{F}_{q^m}^n$ *such that:*

1. $\mathsf{deg}_q f(x) \leq k - 1$; 2. $\boldsymbol{e} \neq \boldsymbol{0}_n$ *and* $\|\boldsymbol{e}\|_{\mathrm{R}} \leq r$; 3. $\boldsymbol{y} = f(\boldsymbol{g}) + \boldsymbol{e}$.

From Definition 5, we known that for any $\boldsymbol{e} \in \mathbb{F}_{q^m}^n$ of weight r, there exists a unique annihilator polynomial $\mathcal{A}_{\boldsymbol{e}}(x)$ of q-degree r that vanishes on $\boldsymbol{e}$, i.e., $\mathcal{A}_{\boldsymbol{e}}(e_i) = 0$ for all $i \in [n]$. Thus, it is natural to transform the $\mathsf{DecEGCode}(\boldsymbol{g}, \boldsymbol{y})$ problem into the Non Linear Reconstruction (NLR) problem (Definition 9).

Definition 9 (Non Linear Reconstruction Problem $\mathsf{NLR}(\boldsymbol{g}, \boldsymbol{y})$).
Input : $\boldsymbol{g} = (g_1, g_2, \ldots, g_n) \in \mathbb{F}_{q^m}^n$ *and* $\|\boldsymbol{g}\|_{\mathrm{R}} = t$; $\boldsymbol{y} = (y_1, y_2, \ldots, y_n) \in \mathbb{F}_{q^m}^n$.
Output : $f(x), v(x) \in \mathcal{L}[x]$ *such that:*

1. $\mathsf{deg}_q f(x) \leq k - 1$; 2. $v(x) \neq 0$ *and* $\mathsf{deg}_q v(x) \leq r$; 3. $v(\boldsymbol{y}) = (v \circ f)(\boldsymbol{g})$.

Theorem 1. *Solving the $\mathsf{DecEGCode}(\boldsymbol{g}, \boldsymbol{y})$ problem can be reduced to solving the $\mathsf{NLR}(\boldsymbol{g}, \boldsymbol{y})$ problem.*

Proof. Let $f(x)$ and $v(x)$ be a solution of $\mathsf{NLR}(\boldsymbol{g}, \boldsymbol{y})$. Then $\mathsf{deg}_q f(x) \leq k-1$, $v(x) \neq 0$, $\mathsf{deg}_q v(x) \leq r$, and $v(\boldsymbol{y}) = (v \circ f)(\boldsymbol{g})$. From $v(\boldsymbol{y}) = (v \circ f)(\boldsymbol{g})$, we have

$$v(\boldsymbol{y}) = (v \circ f)(\boldsymbol{g}) \implies v(\boldsymbol{y}) = v(f(\boldsymbol{g})) \implies v(\boldsymbol{y} - f(\boldsymbol{g})) = \mathbf{0}_n.$$

We set $\boldsymbol{e} = \boldsymbol{y} - f(\boldsymbol{g}) \neq \mathbf{0}_n$ and have $v(\boldsymbol{e}) = \mathbf{0}_n$. Since $\mathsf{deg}_q v(x) \leq r$, from Proposition 2, we have $\|\boldsymbol{e}\|_{\mathrm{R}} \leq r$. This means that $f(x)$ and $\boldsymbol{e}$ are a solution to $\mathsf{DecEGCode}(\boldsymbol{g}, \boldsymbol{y})$. □

Theorem 1 shows that for decoding EG code, it is sufficient to solve the NLR problem, and the maximum r solved in the NLR problem determines the decoding capacity of the EG codes.

4.2 From the NLR Problem to the LR Problem

A naive way to solve the NLR problem is that viewing the coefficients of $v(x)$ and $f(x)$ as two groups of unknowns, respectively, then solving a system obtained from the equation $v(\boldsymbol{y}) = (v \circ f)(\boldsymbol{g})$. Since the related equation involves products of two groups of unknowns, one has to solve a multivariate system over $\mathbb{F}_{q^m}$. However, solving a multivariate system is an NP-hard problem on average.

Facing this challenge, to find more solvable instances, we adapt the linearized technique in [9,30] to our NLR problem. We view $v(x) \circ f(x)$ as a new unknown q-polynomial $u(x)$ of q-degree $\leq k+r-1$, and build a linear system with unknowns in the coefficients of $v(x)$ and $u(x)$. Once $v(x)$ and $u(x)$ are solved, we can obtain $f(x)$ by left division. At this time, solving the NLR problem is reduced to solving the Linear Reconstruction (LR) problem (Definition 10).

Definition 10 (Linear Reconstruction Problem $\mathsf{LR}(\boldsymbol{g}, \boldsymbol{y})$).
Input : $\boldsymbol{g} = (g_1, g_2, \ldots, g_n) \in \mathbb{F}_{q^m}^n$ *and* $\|\boldsymbol{g}\|_{\mathrm{R}} = t$; $\boldsymbol{y} = (y_1, y_2, \ldots, y_n) \in \mathbb{F}_{q^m}^n$.
Output : $v(x), u(x) \in \mathcal{L}[x]$ *such that:*

1. $v(x) \neq 0$ *and* $\mathsf{deg}_q v(x) \leq r$; 2. $u(x) \leq k + r - 1$; 3. $v(\boldsymbol{y}) = u(\boldsymbol{g})$.

A solution $(f(x), v(x))$ of $\mathsf{NLR}(\boldsymbol{g}, \boldsymbol{y})$ clearly gives a solution $(v(x), v(x) \circ f(x))$ of $\mathsf{LR}(\boldsymbol{g}, \boldsymbol{y})$. However, the converse argument is tricky. The proof of converse argument depends on specific methods solving the LR problem. For solving this LR problem, there mainly exist two pioneering and classical methods [30,9]: the Gaussian elimination and the Welch-Berlekamp like algorithm.

In this paper, we mainly study the Gaussian elimination. Solving the $\mathsf{LR}(\boldsymbol{g}, \boldsymbol{y})$ problem by the Gaussian elimination consists in solving a liner system

$$\boldsymbol{A}\boldsymbol{x}^{\top} = \mathbf{0}_n \tag{4}$$

where

$$\boldsymbol{A} = \begin{bmatrix} y_1 & y_1^q & \cdots & y_1^{q^r} & g_1 & g_1^q & \cdots & g_1^{q^{k+r-1}} \\ y_2 & y_2^q & \cdots & y_2^{q^r} & g_2 & g_2^q & \cdots & g_2^{q^{k+r-1}} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ y_n & y_n^q & \cdots & y_n^{q^r} & g_n & g_n^q & \cdots & g_n^{q^{k+r-1}} \end{bmatrix} = \left[\mathbf{Moore}(\boldsymbol{y}, r)^{\top} \ \mathbf{Moore}(\boldsymbol{g}, k+r-1)^{\top} \right]. \tag{5}$$

In this case, to deal with the converse argument: A solution $(v(x), v(x) \circ f(x))$ of $\mathsf{LR}(\boldsymbol{g}, \boldsymbol{y})$ can give a solution $(f(x), v(x))$ of $\mathsf{NLR}(\boldsymbol{g}, \boldsymbol{y})$, we show that:

- The vector consisting of the coefficients of $v(x)$ and $-v(x) \circ f(x)$ must be included in the right kernel of $\boldsymbol{A}$ (see Proposition 8);
- If the right kernel of $\boldsymbol{A}$ is of dimension 1, then the LR problem can be solved up to the largest q-degree (see Theorem 2).

Next, we will comprehensively analyze the right kernel of the linear system $\boldsymbol{A}\boldsymbol{x}^\top = \boldsymbol{0}_n$ to prove the converse argument and develop decodable parameters.

For the proof of converse argument, a careful reader may ask why not adapt the ideas of Theorem 6 in [9] (or Proposition 2 in [30]) to the parameters of the EG codes? Our answer is that it is infeasible. Note that, to decode by solving the LR problem in the rank setting, authors in [30,9] actually presented two pioneering and classical methods: the Gaussian elimination and the Welch-Berlekamp like algorithm. The reasons are as follows:

- Theorem 6 in [9] (or Proposition 2 in [30]) is proven from the perspective of roots of q-polynomial. Specifically, they use Fundamental Theorem of Algebra for q-polynomials: If a q-polynomial of q-degree $\leq n-1$ vanishes on n $\mathbb{F}_q$-linearly independent elements, then the q-polynomial must be zero q-polynomial. The conclusions only hold for $m \geq n = t$ and $r \leq \lfloor \frac{n-k}{2} \rfloor$ (i.e., parameters of Gabidulin codes).
- Theorem 6 in [9] (or Proposition 2 in [30]) allows well prove the converse argument in the case where the Welch-Berlekamp like algorithm is applied to solve the LR problem. The conclusion seems only support that the Welch-Berlekamp like algorithm is deterministic because the proofs of Theorem 6 (or Proposition 2) and the design of the Welch-Berlekamp like algorithm are almost made from the perspective of roots of q-polynomial.

Even if the ideas of Theorem 6 in [9] (or Proposition 2 in [30]) are forcibly adapted to the parameters of the EG codes, it would also be hard to estimate DFR. In the case where the Gaussian elimination is applied to solve the LR problem, the decoding performance such as DFR must be made by strictly analyzing the right kernel of the linear system. This is also necessary because the different methods could lead to different DFR.

However, analyzing the right kernel of the linear system is rather cumbersome, which is evaded and left in [30,9,31]. As the left question in [31] (below Proposition 1): "Now we investigate the question: when is the system of dimension 1?" The authors there provided a proof only for a specific case where the number of columns in the system equals the number of rows plus one. Again, a careful reader could also note that [30,9] (Algorithm 1 [9]) omitted two crucial problems: How to decode the smaller errors? What the sufficient and necessary condition of successful decoding is? For the latter, the authors only assume the existence of non-zero right kernel. Our conclusion shows that this is not the case. And if our heuristic argument (in Theorem 3) deriving DFR is invalid, then decoding Gabidulin codes by the Gaussian elimination would not be deterministic

(Theorem 8). This challenges the well-known conclusion: decoding Gabidulin codes is deterministic, which has stood for over 40 years.

It is also because the authors in [30,9,31] did not deal with the right kernel of the linear system that they missed many decodable parameters and the existence of the (interleaved) EG codes. We will directly analyze the right kernel of the linear system and try to answer the above open problems.

Proposition 8. *Let $f(x)$ and $v(x)$ be a solution to the $\mathsf{NLR}(\boldsymbol{g},\boldsymbol{y})$ problem defined in Definition 9. Let $\boldsymbol{A}$ be an $n\times(k+2r+1)$ matrix defined in Equation (5). The solution space of the system $\boldsymbol{A}\boldsymbol{x}^{\top}=\mathbf{0}_n$ must contain a vector consisting of the coefficients of $v(x)$ and $-v(x)\circ f(x)$.*

Proof. From Definition 9, we known that if $f(x)$ and $v(x)$ are the solution to the $\mathsf{NLR}(\boldsymbol{g},\boldsymbol{y})$ problem, then $\mathsf{deg}_q f(x)\le k-1$, $v(x)\neq 0$, $\mathsf{deg}_q v(x)\le r$, and $v(\boldsymbol{y})=(v\circ f)(\boldsymbol{g})$. Let $f(x)=\sum_{i=0}^{k-1} f_i x^{q^i}$ and $v(x)=\sum_{i=0}^{r} v_i x^{q^i}$. Let $u(x)=v(x)\circ f(x)=\sum_{i=0}^{k+r-1} u_i x^{q^i}$.

From $v(\boldsymbol{y})=(v\circ f)(\boldsymbol{g}) \iff v(\boldsymbol{y})=v(f(\boldsymbol{g})) \iff v(\boldsymbol{y}-f(\boldsymbol{g}))=\mathbf{0}_n$, one can check:

$$\begin{bmatrix} y_1-f(g_1) & (y_1-f(g_1))^q & \cdots & (y_1-f(g_1))^{q^r} \\ y_2-f(g_2) & (y_2-f(g_2))^q & \cdots & (y_2-f(g_2))^{q^r} \\ \vdots & \vdots & \ddots & \vdots \\ y_n-f(g_n) & (y_n-f(g_n))^q & \cdots & (y_n-f(g_n))^{q^r} \end{bmatrix} \begin{bmatrix} v_0 \\ v_1 \\ \vdots \\ v_r \end{bmatrix} = \mathbf{0}_n \tag{6}$$

$$\Longleftrightarrow$$

$$\begin{bmatrix} y_1 & y_1^q & \cdots & y_1^{q^r} & g_1 & g_1^q & \cdots & g_1^{q^{k+r-1}} \\ y_2 & y_2^q & \cdots & y_2^{q^r} & g_2 & g_2^q & \cdots & g_2^{q^{k+r-1}} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ y_n & y_n^q & \cdots & y_n^{q^r} & g_n & g_n^q & \cdots & g_n^{q^{k+r-1}} \end{bmatrix} \begin{bmatrix} \boldsymbol{I}_{(r+1)\times(r+1)} \\ \hline -\mathbf{PM}(f(x),r) \end{bmatrix} \begin{bmatrix} v_0 \\ v_1 \\ \vdots \\ v_r \end{bmatrix} = \mathbf{0}_n \tag{7}$$

$$\Longleftrightarrow$$

$$\underbrace{\begin{bmatrix} y_1 & y_1^q & \cdots & y_1^{q^r} & g_1 & g_1^q & \cdots & g_1^{q^{k+r-1}} \\ y_2 & y_2^q & \cdots & y_2^{q^r} & g_2 & g_2^q & \cdots & g_2^{q^{k+r-1}} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ y_n & y_n^q & \cdots & y_n^{q^r} & g_n & g_n^q & \cdots & g_n^{q^{k+r-1}} \end{bmatrix}}_{=\boldsymbol{A}\in\mathbb{F}_{q^m}^{n\times(k+2r+1)}} \begin{bmatrix} v_0 \\ v_1 \\ \vdots \\ v_r \\ -u_0 \\ -u_1 \\ \vdots \\ -u_{k+r-1} \end{bmatrix} = \mathbf{0}_n. \tag{8}$$

From Equation (8), we can observe that the vector $(v_0, v_1, \ldots, v_r, -u_0, -u_1, \ldots, -u_{k+r-1})$ consisting of the coefficients of $v(x)$ and $-v(x)\circ f(x)$ must belong to the solution space of the linear system (4). □

Theorem 2. *If the dimension of the right kernel of matrix $\boldsymbol{A}$ defined in Equation (5) is 1, then the* $\mathsf{NLR}(\boldsymbol{g}, \boldsymbol{y})$ *problem in Definition 9 can be solved up to q-degree r.*

Proof. If the dimension of the right kernel of $\boldsymbol{A}$ is 1, then the solution to the linear system (4): $\boldsymbol{A}\boldsymbol{x}^\top = \boldsymbol{0}_n$ is unique up to a multiplicative factor in $\mathbb{F}_{q^m}$. Let $\boldsymbol{b} = (b_0, b_1, \ldots, b_{k+2r})$ be this unique solution. We set $v'(x) = \sum_{i=0}^{r} b_i x^{q^i}$ and $u'(x) = \sum_{i=0}^{k+r-1} b_{i+r+1} x^{q^i}$. Next, we show that $v'(x)$ and $-v'(x) \setminus u'(x)$ are the solution to the $\mathsf{NLR}(\boldsymbol{g}, \boldsymbol{y})$ problem.

From Definition 9, if $f(x)$ and $v(x)$ is a solution to the $\mathsf{NLR}(\boldsymbol{g}, \boldsymbol{y})$ problem, then $\mathsf{deg}_q f(x) \leq k-1$, $v(x) \neq 0$, $\mathsf{deg}_q v(x) \leq r$, and $v(\boldsymbol{y}) = (v \circ f)(\boldsymbol{g})$. From Proposition 8, the vector consisting of the coefficients of $v(x)$ and $-v(x) \circ f(x)$ must belong to the solution space of the linear system (4). This means that there exists a non-zero $\beta \in \mathbb{F}_{q^m}$ such that $v'(x) = \beta \cdot v(x)$ and $u'(x) = -\beta \cdot v(x) \circ f(x)$. By left division, we have $f(x) = -v'(x) \setminus u'(x)$. It is easy to check that $v'(x) \neq 0$, $\mathsf{deg}_q v'(x) = \mathsf{deg}_q v(x) \leq r$, and $v'(\boldsymbol{y}) = (v' \circ f)(\boldsymbol{g})$. Thus, $v'(x)$ and $-v'(x) \setminus u'(x)$ are exactly the solution to the $\mathsf{NLR}(\boldsymbol{g}, \boldsymbol{y})$ problem. □

Theorem 2 shows that when the right kernel (denote as $\mathsf{kernel}(\boldsymbol{A})$) of $\boldsymbol{A}$ is of dimension 1, any non-zero element can determine the solution to the $\mathsf{NLR}(\boldsymbol{g}, \boldsymbol{y})$ problem for the largest q-degree r. Further, by Theorem 1, the $\mathsf{DecEGCode}(\boldsymbol{g}, \boldsymbol{y})$ problem is solved up to errors of weight r.

At this point, we proof the converse argument and obtain decoding EG codes can perfectly reduced to solving the LR problem by the Gaussian elimination. The solved largest q-degree r in the LR problem determines decoding errors of the maximal weight r. To determine the largest q-degree r for the LR problem, we must analyze when the right kernel of $\boldsymbol{A}$ is of dimension 1.

5 Decoding EG Codes

In this section, we analyze the right kernel of $\boldsymbol{A}$, analyze decoding failure rate and decoding capacity, and give the decoding algorithm of the EG codes.

5.1 Dimension of Right Kernel of $\boldsymbol{A}$

Fist, we argue that $\mathsf{dim}(\mathsf{kernel}(\boldsymbol{A})) > 0$ for any $\boldsymbol{g}$ and $\boldsymbol{y}$, that is, there must exist a non-zero element in $\mathsf{kernel}(\boldsymbol{A})$. Following Propositions 9 and 10, we get $\mathsf{rank}(\boldsymbol{A}) \leq k + 2r$. Since $\mathsf{rank}(\boldsymbol{A}) + \mathsf{dim}(\mathsf{kernel}(\boldsymbol{A})) = k + 2r + 1$, we must have $\mathsf{dim}(\mathsf{kernel}(\boldsymbol{A})) \geq 1$.

Proposition 9. *Let $f(x)$ and $\boldsymbol{e}$ be the solution to the* $\mathsf{DecEGCode}(\boldsymbol{g}, \boldsymbol{y})$ *problem defined in Definition 8. Then the matrix $\boldsymbol{A}$ defined in Equation (5) is equivalent to an $n \times (k + 2r + 1)$ matrix $\boldsymbol{B}$ defined in Equation (9).*

$$\boldsymbol{B} = \left[\mathbf{Moore}(\boldsymbol{e}, r)^\top \quad \mathbf{Moore}(\boldsymbol{g}, k + r - 1)^\top \right]. \tag{9}$$

Proof. From Definition 8, if $f(x)$ and $\boldsymbol{e}$ are the solution to the $\mathsf{DecEGCode}(\boldsymbol{g}, \boldsymbol{y})$ problem, then $\mathsf{deg}_q f(x) \leq k-1$, $\boldsymbol{e} \neq \mathbf{0}_n$, $\|\boldsymbol{e}\|_{\mathrm{R}} \leq r$, and

$$\boldsymbol{y} = f(\boldsymbol{g}) + \boldsymbol{e} \iff \boldsymbol{y} - f(\boldsymbol{g}) = \boldsymbol{e} \iff y_i - f(g_i) = e_i \text{ for all } i \in [n].$$

Combining Equation (7), we have

$$\begin{bmatrix} y_1 & y_1^q & \cdots & y_1^{q^r} & g_1 & g_1^q & \cdots & g_1^{q^{k+r-1}} \\ y_2 & y_2^q & \cdots & y_2^{q^r} & g_2 & g_2^q & \cdots & g_2^{q^{k+r-1}} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ y_n & y_n^q & \cdots & y_n^{q^r} & g_n & g_n^q & \cdots & g_n^{q^{k+r-1}} \end{bmatrix} \left[\begin{array}{c|c} \boldsymbol{I}_{(r+1)\times(r+1)} & \mathbf{0}_{(r+1)\times(k+r)} \\ \hline -\mathbf{PM}(f(x), r) & \boldsymbol{I}_{(k+r)\times(k+r)} \end{array}\right]$$

$$= \begin{bmatrix} y_1 - f(g_1) & (y_1 - f(g_1))^q & \cdots & (y_1 - f(g_1))^{q^r} & g_1 & g_1^q & \cdots & g_1^{q^{k+r-1}} \\ y_2 - f(g_2) & (y_2 - f(g_2))^q & \cdots & (y_2 - f(g_2))^{q^r} & g_2 & g_2^q & \cdots & g_2^{q^{k+r-1}} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ y_n - f(g_n) & (y_n - f(g_n))^q & \cdots & (y_n - f(g_n))^{q^r} & g_n & g_n^q & \cdots & g_n^{q^{k+r-1}} \end{bmatrix}$$

$$= \begin{bmatrix} e_1 & e_1^q & \cdots & e_1^{q^r} & g_1 & g_1^q & \cdots & g_1^{q^{k+r-1}} \\ e_2 & e_2^q & \cdots & e_2^{q^r} & g_2 & g_2^q & \cdots & g_2^{q^{k+r-1}} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ e_n & e_n^q & \cdots & e_n^{q^r} & g_n & g_n^q & \cdots & g_n^{q^{k+r-1}} \end{bmatrix} = \boldsymbol{B}.$$

□

Proposition 10. *For any uniform error $\boldsymbol{e} \in \mathbb{F}_{q^m}^n$ and $\|\boldsymbol{e}\|_{\mathrm{R}} = r$, if $k + r \leq t \leq \min\{n, m\}$ and $k + 2r \leq n$, then $\mathsf{rank}(\boldsymbol{B}) \leq k + 2r$.*

Proof. We consider the rank of $\boldsymbol{B}^\top$. Let

$$\boldsymbol{B}_1 = \mathbf{Moore}(\boldsymbol{e}, r), \quad \boldsymbol{B}_2 = \mathbf{Moore}(\boldsymbol{g}, k + r - 1), \quad \boldsymbol{B}^\top = \begin{bmatrix} \boldsymbol{B}_1 \\ \boldsymbol{B}_2 \end{bmatrix}.$$

Since $k+r \leq t$, from Proposition 5, we have $\mathsf{rank}(\boldsymbol{B}_1) = r$ and $\mathsf{rank}(\boldsymbol{B}_2) = k+r$. Thus,

$$\mathsf{rank}(\boldsymbol{B}) = \mathsf{rank}\left(\boldsymbol{B}^\top\right) = \mathsf{rank}\left(\begin{bmatrix} \boldsymbol{B}_1 \\ \boldsymbol{B}_2 \end{bmatrix}\right) \leq \mathsf{rank}(\boldsymbol{B}_1) + \mathsf{rank}(\boldsymbol{B}_2) = k + 2r.$$

□

Propositions 9 and 10 show $\mathsf{rank}(\boldsymbol{A}) = \mathsf{rank}(\boldsymbol{B}) \leq k + 2r$. Since $\mathsf{rank}(\boldsymbol{A}) + \mathsf{dim}(\mathsf{kernel}(\boldsymbol{A})) = k + 2r + 1$, we must have $\mathsf{dim}\left(\mathsf{kernel}(\boldsymbol{A})\right) \geq 1$. Thus, there must exist a non-zero element in $\mathsf{kernel}(\boldsymbol{A})$. Next, we analyze when $\mathsf{kernel}\left(\boldsymbol{A}\right)$ is one-dimensional, and derive the DFR and the decoding capacity.

5.2 Decoding Failure Rate for the Largest r

The failure depends on the event $\mathsf{dim}(\mathsf{kernel}(\boldsymbol{A})) \neq 1$. In Subsection 5.1, we have got $\mathsf{dim}(\mathsf{kernel}(\boldsymbol{A})) \geq 1$ due to $\mathsf{rank}(\boldsymbol{A}) \leq k+2r$ and $\mathsf{rank}(\boldsymbol{A}) + \mathsf{dim}(\mathsf{kernel}(\boldsymbol{A})) = k + 2r + 1$. Thus, the DFR depends on the probability of $\mathsf{dim}(\mathsf{kernel}(\boldsymbol{A})) > 1$, which is equivalent to $\mathsf{rank}(\boldsymbol{A}) < k + 2r$. Since $\mathsf{rank}(\boldsymbol{A}) = \mathsf{rank}(\boldsymbol{B})$ (see Proposition 9), we now analyze the probability of $\mathsf{rank}(\boldsymbol{B}) < k + 2r$.

Theorem 3. *For any uniform error $\boldsymbol{e} \in \mathbb{F}_{q^m}^n$ and $\|\boldsymbol{e}\|_{\mathrm{R}} = r$, if $k + r \leq t \leq \min\{n, m\}$ and $k + 2r \leq n$, then*

$$\Pr[\mathsf{rank}(\boldsymbol{B}) < k + 2r] \leq \gamma_q \cdot q^{a(t+r-a-n)}, \tag{10}$$

where $a = t - k - r + 1$.

Proof. We consider $\boldsymbol{B}^\top$ and estimate $\Pr[\mathsf{rank}(\boldsymbol{B}^\top) < k + 2r]$. Let $\boldsymbol{\varepsilon} \in \mathbb{F}_{q^m}^r$ be a basis of $\mathrm{Supp}(\boldsymbol{e})$. Let $\mathbf{CM}(\boldsymbol{e}) \in \mathbb{F}_q^{r\times n}$ of rank r be the coefficient matrix of $\boldsymbol{e}$ under $\boldsymbol{\varepsilon}$ such that $\boldsymbol{e} = \boldsymbol{\varepsilon}\mathbf{CM}(\boldsymbol{e})$. Let $\boldsymbol{\gamma} \in \mathbb{F}_{q^m}^t$ be a basis of $\mathrm{Supp}(\boldsymbol{g})$. Let $\mathbf{CM}(\boldsymbol{g}) \in \mathbb{F}_q^{t\times n}$ of rank t be the coefficient matrix of $\boldsymbol{g}$ under $\boldsymbol{\gamma}$ such that $\boldsymbol{g} = \boldsymbol{\gamma}\mathbf{CM}(\boldsymbol{g})$. Following Definition 6 and Proposition 3,

$$\begin{aligned}
\boldsymbol{B}^\top &= \begin{bmatrix} \boldsymbol{B}_1 \\ \boldsymbol{B}_2 \end{bmatrix} = \left[\begin{array}{c} \mathbf{Moore}(\boldsymbol{e}, r) \\ \hline \mathbf{Moore}(\boldsymbol{g}, k+r-1) \end{array}\right] = \left[\begin{array}{c} \mathbf{Moore}(\boldsymbol{\varepsilon}, r)\cdot\mathbf{CM}(\boldsymbol{e}) \\ \hline \mathbf{Moore}(\boldsymbol{\gamma}, k+r-1)\cdot\mathbf{CM}(\boldsymbol{g}) \end{array}\right] \\
&= \underbrace{\left[\begin{array}{c|c} \mathbf{Moore}(\boldsymbol{\varepsilon}, r) & \mathbf{0}_{(r+1)\times t} \\ \hline \mathbf{0}_{(k+r)\times r} & \mathbf{Moore}(\boldsymbol{\gamma}, k+r-1) \end{array}\right]}_{:=\boldsymbol{L}\in\mathbb{F}_{q^m}^{(k+2r+1)\times(r+t)}} \underbrace{\left[\begin{array}{c} \mathbf{CM}(\boldsymbol{e}) \\ \hline \mathbf{CM}(\boldsymbol{g}) \end{array}\right]}_{:=\boldsymbol{M}\in\mathbb{F}_q^{(r+t)\times n}}.
\end{aligned} \tag{11}$$

By Corollary 1, we have $\mathsf{rank}\left(\mathbf{Moore}(\boldsymbol{\varepsilon}, r)\right) = r$, and as $k + r - 1 < t$, $\mathsf{rank}\left(\mathbf{Moore}(\boldsymbol{\gamma}, k + r - 1)\right) = k + r$. Thus,

$$\mathsf{rank}(\boldsymbol{L}) = \mathsf{rank}\left(\mathbf{Moore}(\boldsymbol{\varepsilon}, r)\right) + \mathsf{rank}\left(\mathbf{Moore}(\boldsymbol{\gamma}, k + r - 1)\right) = k + 2r \leq n.$$

As $\mathsf{rank}(\boldsymbol{B}^\top) \leq \min\{\mathsf{rank}(\boldsymbol{L}), \mathsf{rank}(\boldsymbol{M})\}$ and $\boldsymbol{M} \in \mathbb{F}_q^{(t+r)\times n}$ is defined in the basis field $\mathbb{F}_q$, we have an **Heuristic Argument**: if $\mathsf{rank}(\boldsymbol{L}) = k + 2r$ and $\mathsf{rank}(\boldsymbol{M}) \geq k + 2r$, then $\mathsf{rank}\left(\boldsymbol{B}^\top\right) = k + 2r$. We also ran a lot of experiments to verify this argument. Please see the formal **Heuristic Argument** after the proof. Hence, we have

$$\mathsf{rank}\left(\boldsymbol{B}^\top\right) < k + 2r \implies \mathsf{rank}(\boldsymbol{M}) < k + 2r. \tag{12}$$

Moreover,

$$\begin{aligned}
\mathsf{rank}(\boldsymbol{M}) &= \mathsf{rank}\begin{bmatrix} \mathbf{CM}(\boldsymbol{e}) \\ \mathbf{CM}(\boldsymbol{g}) \end{bmatrix} = \mathsf{dim}\Big(\langle\mathbf{CM}(\boldsymbol{e})\rangle + \langle\mathbf{CM}(\boldsymbol{g})\rangle\Big) \\
&= \mathsf{rank}\left(\mathbf{CM}(\boldsymbol{e})\right) + \mathsf{rank}\left(\mathbf{CM}(\boldsymbol{g})\right) - \mathsf{dim}\Big(\langle\mathbf{CM}(\boldsymbol{e})\rangle \cap \langle\mathbf{CM}(\boldsymbol{g})\rangle\Big) \\
&= r + t - \mathsf{dim}\Big(\langle\mathbf{CM}(\boldsymbol{e})\rangle \cap \langle\mathbf{CM}(\boldsymbol{g})\rangle\Big).
\end{aligned}$$

Let $\Delta = \mathsf{dim}\Big(\langle \mathbf{CM}(\boldsymbol{e})\rangle \cap \langle \mathbf{CM}(\boldsymbol{g})\rangle\Big)$, we have

$$\mathsf{rank}(\boldsymbol{M}) < k + 2r \iff \Delta > t - k - r. \tag{13}$$

Let $a = t - k - r + 1$. Finally, combining Equations (12) and (13), we have:

– When $a \leq \min\{t, r\}$,

$$\begin{aligned}&\Pr[\mathsf{rank}\left(\boldsymbol{B}^\top\right) < k + 2r] \leq \Pr[\mathsf{rank}(\boldsymbol{M}) < k + 2r] = \Pr\Big[\Delta > t - k - r\Big] \\ &= \sum_{i=a}^{\min\{t,r\}} \Pr\Big[\Delta = i\Big] \approx \Pr\Big[\Delta = a\Big] \leq \gamma_q \cdot q^{a(t+r-a-n)}. \end{aligned} \tag{14}$$

The first inequality "$\leq$" due to "$\Longrightarrow$" of Equation (12). The second equation "$=$" due to "$\Longleftrightarrow$" of Equation (13). The approximation "$\approx \Pr\Big[\Delta = a\Big]$" is natural. By the probability in Lemma 2 (see Appendix A), as the dimension i increases, the probability decreases significantly. Of course, one can also use upper bounds "$\leq \big(\min\{t, r\} - a + 1\big)\Pr\Big[\Delta = a\Big]$". We use the approximation because it simulates closely. The last inequality also follows Lemma 2.

– When $a > \min\{t, r\}$, it is impossible to obtain an intersection space of dimension $\geq a$. Hence,

$$\Pr[\mathsf{rank}\left(\boldsymbol{B}^\top\right) < k + 2r] \leq \Pr[\mathsf{rank}(\boldsymbol{M}) < k + 2r] = \Pr\Big[\Delta \geq a\Big] = 0. \tag{15}$$

This means that the DFR is null and the decoding algorithm is deterministic. □

Heuristic Argument. If $\mathsf{rank}(\boldsymbol{L}) = k + 2r$ and $\mathsf{rank}(\boldsymbol{M}) \geq k + 2r$, then $\mathsf{rank}\left(\boldsymbol{B}^\top\right) = \mathsf{rank}\left(\boldsymbol{LM}\right) = k + 2r$.

To validate the heuristic argument, we provide two evidences:

– We performed an experiment. With a fixed matrix $\boldsymbol{L}$, a lot of random matrices $\boldsymbol{M}$ of rank $\geq k+2r$ always lead to a matrix $\boldsymbol{B}^\top$ of rank $k+2r$. The scripts are available online at https://github.com/RQCPKE/EGCodesRQC.
– In Section 8 (Theorem 8), the heuristic argument is applied to prove the null DFR of the (Interleaved) Gabidulin codes. If our heuristic argument is invalid, then decoding Gabidulin codes by the Gaussian elimination would not be deterministic. This challenges the well-known conclusion: decoding Gabidulin codes is deterministic, which has stood for over 40 years.

These two evidences show that our heuristic argument is reliable, sound, and valid.

5.3 Decoding Capacity

Theorem 4. *The* $\mathrm{EG}_k(\boldsymbol{g})$ *codes in Definition 7 can decode errors of weight up to* $\min\left\{t-k, \left\lfloor \frac{n-k}{2} \right\rfloor\right\}$.

Proof. For any error $\boldsymbol{e} \in \mathbb{F}_{q^m}^n$ and $\|\boldsymbol{e}\|_{\mathrm{R}} = r$, from Theorem 3, we known that when $k + r \leq t$ and $k + 2r \leq n$, the $\mathrm{EG}_k(\boldsymbol{g})$ codes can decode such an error $\boldsymbol{e}$. This leads to conclusion. □

Theorem 5. *If* $t = m$ *and* $k = 2m - n$*, then the* $\mathrm{EG}_k(\boldsymbol{g})$ *codes in Definition 7 can decode errors of weight up to the RGV bound in Definition 14 (Appendix A.1).*

Proof. From Theorem 4, we know that the $\mathrm{EG}_k(\boldsymbol{g})$ codes can decode errors of weight up to $\min\left\{t-k, \left\lfloor \frac{n-k}{2} \right\rfloor\right\}$. Let $r = \min\left\{t-k, \left\lfloor \frac{n-k}{2} \right\rfloor\right\}$. By the definition of the RGV bound (Definition 14, Appendix A.1), we only need to prove that such a value r satisfies $q^{r(m+n-r)} = q^{(n-k)m}$. If $t = m$ and $k = 2m - n$, then $t - k = \left\lfloor \frac{n-k}{2} \right\rfloor = n - m$. Hence, $r = \min\left\{t-k, \left\lfloor \frac{n-k}{2} \right\rfloor\right\} = n - m$. It is easy to check that $q^{r(m+n-r)} = q^{(n-k)m} = q^{2m(n-m)}$. □

Recall that the RGV bound in Appendix A.1, $q^{r(m+n-r)}$ is the lower bound of $\mathcal{V}(q, m, n, r)$. Thus, by the EG codes, any syndrome (resp. word) in $\mathbb{F}_{q^m}^{n-k}$ (resp. $\mathbb{F}_{q^m}^n$) can be decoded to an error of weight the RGV bound. In the experiment, for some code parameters, any syndrome (resp. word) can even decode up to a larger error than the RGV bound with an overwhelming probability.

5.4 Decoding Algorithm

Theorem 3 considers the errors of weight exactly r, and the decoding will fail if $\mathsf{rank}(\boldsymbol{A}) < k + 2r$. However, when an error of weight $< r$ (say weight is w) really occurs, we must have $\mathsf{rank}(\boldsymbol{A}) < k + 2r$:

$$\mathsf{rank}(\boldsymbol{L}) = \mathsf{rank}\left(\mathbf{Moore}(\boldsymbol{\varepsilon}, r)\right) + \mathsf{rank}\left(\mathbf{Moore}(\boldsymbol{\gamma}, k + r - 1)\right) = w + k + r < k + 2r,$$

$$\mathsf{rank}(\boldsymbol{A}) = \mathsf{rank}(\boldsymbol{B}^{\top}) \leq \min\{\mathsf{rank}(\boldsymbol{L}), \mathsf{rank}(\boldsymbol{M})\} \leq \mathsf{rank}(\boldsymbol{L}) < k + 2r,$$

where $\boldsymbol{\varepsilon} \in \mathbb{F}_{q^m}^w$ is a basis of support of the error of weight w.

Then, in this case, whether it can decode successfully, how to recover $f(x)$, and how to determine DFR ? We discuss this case in Appendix B. We there show that if $\mathsf{dim}\left(\mathsf{kernel}(\boldsymbol{A})\right)$ is controlled as $r - w + 1$, any non-zero element in $\mathsf{kernel}(\boldsymbol{A})$ can still be used to recover $f(x)$, and the DFR $\leq \gamma_q \cdot q^{a(t+w-a-n)}$. The DFR quickly decreases in w and is upper bounded by that of decoding weight r.

The resulting decoding procedure is given in Algorithm 1. We present a general decoding procedure: given an EG code decoding up to weight r, when the error of weight w occurs ($w \leq r$), if $\mathsf{dim}\left(\mathsf{kernel}(\boldsymbol{A})\right)$ is controlled as $r - w + 1$, any non-zero element in $\mathsf{kernel}(\boldsymbol{A})$ can determine $f(x)$ by Steps 5 - 6. The DFR depends on $\mathsf{dim}\left(\mathsf{kernel}(\boldsymbol{A})\right)$, and Algorithm 1 returns failure if $\mathsf{dim}\left(\mathsf{kernel}(\boldsymbol{A})\right) \neq r - w + 1$.

Algorithm 1 Decoding errors of weight w for the EG codes ($w \leq r$)

Input: $q, m, n, k, t, w, r \in \mathbb{N}$, $k \leq t \leq \min\{n, m\}$,
$\boldsymbol{g} \in \mathbb{F}_{q^m}^n$ and $\|\boldsymbol{g}\|_{\mathrm{R}} = t$, $\boldsymbol{y} \in \mathbb{F}_{q^m}^n$.
Output: $f(x) \in \mathcal{L}_{\leq k-1}[x]$; $\boldsymbol{e} \in \mathbb{F}_{q^m}^n$ and $\|\boldsymbol{e}\|_{\mathrm{R}} = w$

1: Compute matrices $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$, and construct matrix $\boldsymbol{A}$:

$$\boldsymbol{Y} = \mathbf{Moore}(\boldsymbol{y}, r)^\top, \quad \boldsymbol{Z} = \mathbf{Moore}(\boldsymbol{g}, k + r - 1)^\top, \quad \boldsymbol{A} = [\boldsymbol{Y} \ \ \boldsymbol{Z}].$$

2: Compute the right kernel $\mathsf{kernel}(\boldsymbol{A})$ of the linear system $\boldsymbol{A}\boldsymbol{x}^\top = \mathbf{0}_n$.
3: Recover message q-polynomial $f(x)$ and error $\boldsymbol{e}$:
4: Let $\boldsymbol{b} = (b_0, b_1, \ldots, b_{k+2r})$ be a random non-zero element in $\mathsf{kernel}(\boldsymbol{A})$.
5: Set $v'(x) = \sum_{i=0}^{r} b_i x^{q^i}$ and $u'(x) = \sum_{i=0}^{k+r-1} b_{i+r+1} x^{q^i}$.
6: Set $f(x) = -v'(x) \backslash u'(x)$ by left division. // Holds if $\mathsf{dim}\,(\mathsf{kernel}(\boldsymbol{A})) = r - w + 1$
7: **if** $\mathsf{deg}_q f(x) \leq k - 1$ and $\|\boldsymbol{y} - f(\boldsymbol{g})\|_{\mathrm{R}} = w$:
8: **return** $f(x)$ and $\boldsymbol{e} := \boldsymbol{y} - f(\boldsymbol{g})$
9: **else**:
10: **return** $\perp$

Decoding Complexity. The complexity of Algorithm 1 is dominated by Step 2. The right kernel of the system $\boldsymbol{A}\boldsymbol{x}^\top = \mathbf{0}_n$ can be found in $\mathcal{O}(n^3)$ operations in $\mathbb{F}_{q^m}$ by standard Gaussian elimination. Thus, the complexity of decoding EG codes is bounded by $\mathcal{O}(n^3)$.

5.5 Simulated DFR for Decoding Errors of Weight r

In Table 2, we chose eight types of simulable code parameters, and set five groups of parameters for each type. We simulated DFR for each group by performing decoding algorithms (Algorithm 1) in 10^5 times. The theoretical DFR is estimated by Equations (14) and (15) with $\gamma_2 = 4$, and $\gamma_q = 2$ if $q > 2$. The theoretical DFR for the types decoding up to the RGV bound is always greater than 1, and we recorded these DFR as 1 (see No. 26 - 35). Here, d is the minimal distance, d_{RGV} is the RGV bound (Definition 14) and d_{RS} is the RS bound (Definition 15). The test scripts are available online at https://github.com/RQCPKE/EGCodesRQC.

6 Improved Decoding Algorithm

It is clear that the cost $\mathcal{O}(n^3)$ of decoding Algorithm 1 is too much for an efficiently implementation. In this section, we provide two improvements and obtain the cost $\mathcal{O}(r^3)$ and $\mathcal{O}(n^2)$. These also show that our decoding idea starting from solving the LR problem is potential for developing efficient decoding algorithms.

6.1 The Improved Gaussian Elimination

Here, we adapt the idea of decoding Gabidulin codes by Loidreau [30] (Section 5.1). Recall that decoding Algorithm 1 consists in solving the system (4): $\boldsymbol{A}\boldsymbol{x}^\top =$

Table 2. Theoretical and simulated DFR of the EG codes for errors of weight r.

Types	No.	Parameters (q, m, n, t, k, r)	Theoretical DFR	Simulated DFR	d	d_{RGV}	d_{RS}
Increase q $(t = m < n)$	1	$(2, 5, 7, 5, 2, 2)$	(2^{-2}) 0.2500	0.0579	4	3	4
	2	$(3, 5, 7, 5, 2, 2)$	$(2^{-5.3})$ 0.0247	0.0123	4	3	4
	3	$(5, 5, 7, 5, 2, 2)$	$(2^{-8.3})$ 0.0032	0.0016	4	3	4
	4	$(7, 5, 7, 5, 2, 2)$	$(2^{-10.2})$ 0.00085	0.00034	4	3	4
	5	$(11, 5, 7, 5, 2, 2)$	$(2^{-12.8})$ 0.00014	0.00004	4	3	4
Increase m $(t = m < n)$	6	$(2, 31, 41, 31, 9, 16)$	(2^{-5}) 0.0313	0.0154	23	19	25
	7	$(2, 32, 41, 32, 9, 16)$	(2^{-6}) 0.0156	0.0078	24	19	25
	8	$(2, 33, 41, 33, 9, 16)$	(2^{-7}) 0.0078	0.0037	25	20	26
	9	$(2, 34, 41, 34, 9, 16)$	(2^{-8}) 0.0039	0.0018	26	20	27
	10	$(2, 35, 41, 35, 9, 16)$	(2^{-9}) 0.0020	0.0009	27	20	28
Increase n $(t = m < n)$	11	$(2, 27, 41, 27, 9, 16)$	(2^{-1}) 0.5000	0.2312	19	17	22
	12	$(2, 27, 42, 27, 9, 16)$	(2^{-4}) 0.0625	0.0388	19	17	22
	13	$(2, 27, 43, 27, 9, 16)$	(2^{-7}) 0.0078	0.0052	19	18	22
	14	$(2, 27, 44, 27, 9, 16)$	(2^{-10}) 0.00098	0.00085	19	18	22
	15	$(2, 27, 45, 27, 9, 16)$	(2^{-13}) 0.00012	0.00008	19	18	22
Increase t $(t < m < n)$	16	$(2, 35, 41, 30, 9, 16)$	(2^{-4}) 0.0625	0.0300	22	20	28
	17	$(2, 35, 41, 31, 9, 16)$	(2^{-5}) 0.0313	0.0151	23	20	28
	18	$(2, 35, 41, 32, 9, 16)$	(2^{-6}) 0.0156	0.0083	24	20	28
	19	$(2, 35, 41, 33, 9, 16)$	(2^{-7}) 0.0078	0.0036	25	20	28
	20	$(2, 35, 41, 34, 9, 16)$	(2^{-8}) 0.0039	0.0018	26	20	28
Increase t $(t < n < m)$	21	$(2, 29, 26, 16, 5, 10)$	(2^{-2}) 0.0250	0.1326	12	16	22
	22	$(2, 29, 26, 17, 5, 10)$	(2^{-4}) 0.0625	0.0371	13	16	22
	23	$(2, 29, 26, 18, 5, 10)$	(2^{-6}) 0.0156	0.0098	14	16	22
	24	$(2, 29, 26, 19, 5, 10)$	(2^{-8}) 0.0039	0.0024	15	16	22
	25	$(2, 29, 26, 20, 5, 10)$	(2^{-10}) 0.00098	0.00058	16	16	22
Decoding RGV Bound $(k > r)$	26	$(2, 30, 37, 30, 23, 7)$	1	0.7122	8	7	12
	27	$(2, 30, 38, 30, 22, 8)$	1	0.7086	9	8	13
	28	$(2, 30, 39, 30, 21, 9)$	1	0.7102	10	9	14
	29	$(2, 30, 40, 30, 20, 10)$	1	0.7109	11	10	16
	30	$(2, 30, 41, 30, 19, 11)$	1	0.7142	12	11	17
Decoding RGV Bound $(k < r)$	31	$(2, 21, 34, 21, 8, 13)$	1	0.7125	14	13	17
	32	$(2, 22, 36, 22, 8, 14)$	1	0.7119	15	14	18
	33	$(2, 23, 38, 23, 8, 15)$	1	0.7118	16	15	19
	34	$(2, 24, 40, 24, 8, 16)$	1	0.7122	17	16	20
	35	$(2, 25, 42, 25, 8, 17)$	1	0.7123	18	17	21
Gabidulin Codes $(t = n \leq m)$	36	$(2, 27, 27, 27, 7, 10)$	0	0	21	14	21
	37	$(2, 28, 27, 27, 7, 10)$	0	0	21	14	21
	38	$(2, 29, 27, 27, 7, 10)$	0	0	21	14	21
	39	$(2, 30, 27, 27, 7, 10)$	0	0	21	14	21
	40	$(2, 31, 27, 27, 7, 10)$	0	0	21	15	21

$\mathbf{0}_n$. The first $r + 1$ coordinates of $\boldsymbol{x}$ determine $v(x)$ of q-degree r and the last $k + r$ coordinates of $\boldsymbol{x}$ determine $u(x) = -v(x) \circ f(x)$ of q-degree $\leq k + r - 1$.

By observing $\boldsymbol{A}$, its part is independent of the received word $\boldsymbol{y} \in \mathbb{F}_{q^m}^n$ and depends only on the generator $\boldsymbol{g} \in \mathbb{F}_{q^m}^n$. Let $\boldsymbol{y}_1 \in \mathbb{F}_{q^m}^{k+r}$ and $\boldsymbol{g}_1 \in \mathbb{F}_{q^m}^{k+r}$ be vectors consisting of the first $k+r$ coordinates of $\boldsymbol{y}$ and $\boldsymbol{g}$, respectively. Let $\boldsymbol{y}_2 \in \mathbb{F}_{q^m}^{n-k-r}$

and $\boldsymbol{g}_2 \in \mathbb{F}_{q^m}^{n-k-r}$ be vectors consisting of the last $n-k-r$ coordinates of $\boldsymbol{g}$ and $\boldsymbol{y}$, respectively. To adapt the decoding idea in [30], we assume that the coordinates of $\boldsymbol{g}_1$ are linearly independent over $\mathbb{F}_q$, i.e., $\|\boldsymbol{g}_1\|_{\mathrm{R}} = k+r$. Otherwise, we perform a permutation on $\boldsymbol{g}$ and $\boldsymbol{y}$ such that the coordinates of $\boldsymbol{g}_1$ are linearly independent. We write $\boldsymbol{A}$ as

$$\boldsymbol{A} = \left[\begin{array}{c|c} \boldsymbol{T}_1 & \boldsymbol{G}_1 \\ \hline \boldsymbol{T}_2 & \boldsymbol{G}_2 \end{array}\right] \in \mathbb{F}_{q^m}^{n\times(k+2r+1)},$$

where $\boldsymbol{T}_1 = \mathbf{Moore}(\boldsymbol{y}_1, r)^\top \in \mathbb{F}_{q^m}^{(k+r)\times(r+1)}$, $\boldsymbol{T}_2 = \mathbf{Moore}(\boldsymbol{y}_2, r)^\top \in \mathbb{F}_{q^m}^{(n-k-r)\times(r+1)}$, $\boldsymbol{G}_1 = \mathbf{Moore}(\boldsymbol{g}_1, k+r-1)^\top \in \mathbb{F}_{q^m}^{(k+r)\times(k+r)}$, and $\boldsymbol{G}_2 = \mathbf{Moore}(\boldsymbol{g}_2, k+r-1)^\top \in \mathbb{F}_{q^m}^{(n-k-r)\times(k+r)}$.

Let $\boldsymbol{x}_1 \in \mathbb{F}_{q^m}^{r+1}$ be a vector consisting of the first $r+1$ coordinates of $\boldsymbol{x}$. Let $\boldsymbol{x}_2 \in \mathbb{F}_{q^m}^{k+r}$ be a vector consisting of the last $k+r$ coordinates of $\boldsymbol{x}$. As $\|\boldsymbol{g}_1\|_{\mathrm{R}} = k+r$, we have $\mathsf{rank}(\boldsymbol{G}_1) = k+r$, i.e., $\boldsymbol{G}_1$ is invertible. Solving the system (4): $\boldsymbol{A}\boldsymbol{x}^\top = \mathbf{0}_n$ is equivalent to solving

$$\begin{cases} \boldsymbol{T}_1\boldsymbol{x}_1^\top + \boldsymbol{G}_1\boldsymbol{x}_2^\top = \mathbf{0}_{k+r} \\ \boldsymbol{T}_2\boldsymbol{x}_1^\top + \boldsymbol{G}_2\boldsymbol{x}_2^\top = \mathbf{0}_{n-k-r} \end{cases} \iff \begin{cases} \boldsymbol{x}_2^\top = -\boldsymbol{G}_1^{-1}\boldsymbol{T}_1\boldsymbol{x}_1^\top \\ \left(\boldsymbol{T}_2 - \boldsymbol{G}_2\boldsymbol{G}_1^{-1}\boldsymbol{T}_1\right)\boldsymbol{x}_1^\top = \mathbf{0}_{n-k-r} \end{cases} \tag{16}$$

For decoding by solving the system (16), one *precomputes* $\boldsymbol{G}_1^{-1}$ and $\boldsymbol{G}_2\boldsymbol{G}_1^{-1}$. Once receiving $\boldsymbol{y}$, one first solves $\boldsymbol{x}_1$ from the subsystem $\left(\boldsymbol{T}_2 - \boldsymbol{G}_2\boldsymbol{G}_1^{-1}\boldsymbol{T}_1\right)\boldsymbol{x}_1^\top = \mathbf{0}_{n-k-r}$ with $r+1$ unknowns and $n-k-r$ equations, then computes $\boldsymbol{x}_2$. The overall complexity is $\mathcal{O}(r^3)$ for solving $\boldsymbol{x}_1$ when $r+1 \approx n-k-r$. This is the case of cryptographic parameters.

6.2 The Welch-Berlekamp Like Algorithm for the EG Codes

The Welch-Berlekamp algorithm [27] is an efficient algorithm solving the linear reconstruction problem. This algorithm was adapted to decoding Gabidulin codes by Loidreau [30,9] (called Welch-Berlekamp like algorithm). The decoding algorithm presented in the previous sections consists in solving the so-called LR problem (Definition 10) by Gaussian elimination. In this section, we find that the Welch-Berlekamp like algorithm can be adapted to decode EG codes.

The idea is to compute two pairs (u_0, v_0) and (u_1, v_1) of q-polynomials which satisfy the interpolation conditions of the LR problem (see Definition 10):

$$v\left(y_i\right) = u\left(g_i\right), \quad i \in [n]$$

and such that at least one of the pairs satisfies the final degree conditions:

$$\begin{aligned} &\mathsf{deg}_q(u(x)) \leq \begin{cases} k + \left\lfloor \frac{n-k}{2} \right\rfloor - 1, & \text{if } n-k \text{ even} \\ k + \left\lfloor \frac{n-k}{2} \right\rfloor, & \text{if } n-k \text{ odd} \end{cases} \\ &v(x) \neq 0; \quad \mathsf{deg}_q(v(x)) \leq r. \end{aligned}$$

For a negligible DFR, n is often sufficient large. In this case, $\lfloor \frac{n-k}{2} \rfloor$ is greater than r. When applying the Welch-Berlekamp like algorithm to decode EG codes, the last $n-2r-k$ elements of $\boldsymbol{y}$ are redundant, and the first $2r+k$ elements are sufficient for successful decoding. We thus can tune the final degree conditions as:

$$\begin{aligned} &\mathsf{deg}_q(u(x)) \leq k+r, \\ &v(x) \neq 0; \quad \mathsf{deg}_q(v(x)) \leq r. \end{aligned}$$

The resulting decoding algorithm is presented in Algorithm 2. The overall decoding complexity is $\mathcal{O}\left((2r+k)^2\right) \approx \mathcal{O}\left(n^2\right)$. The simulated DFR is presented in Appendix A.5 (See Table 5). The test scripts are available online at https://github.com/RQCPKE/EGCodesRQC.

Algorithm 2 Decoding errors of weight $\leq r$ by the Welch-Berlekamp algorithm

Input: $q, m, n, k, t, r \in \mathbb{N}$, $k \leq t \leq \min\{n, m\}$, $\boldsymbol{g} \in \mathbb{F}_{q^m}^n$ and $\|\boldsymbol{g}\|_{\mathrm{R}} = t$, $\boldsymbol{y} \in \mathbb{F}_{q^m}^n$.
Output: $f(x) \in \mathcal{L}_{\leq k-1}[x]$; $\boldsymbol{e} \in \mathbb{F}_{q^m}^n$ and $\|\boldsymbol{e}\|_{\mathrm{R}} \leq r$

1: Call Algorithm 5 in [9]
2: Return (N_1, W_1)
3: Set $u(x) = N_1$ and $v(x) = W_1$
4: Compute $f(x) = v(x) \setminus u(x)$
5: **if** $\mathsf{deg}_q f(x) \leq k-1$ and $\|\boldsymbol{y} - f(\boldsymbol{g})\|_{\mathrm{R}} \leq r$:
6: **return** $f(x)$ and $\boldsymbol{e} := \boldsymbol{y} - f(\boldsymbol{g})$
7: **else**:
8: **return** $\perp$

7 Interleaved EG Codes

Interleaving a code considers several codewords at the same time, corrupted by errors sharing the same support. This specific structure allows to design algorithms being able to decode more errors. In this section, we introduce the Interleaved EG (IEG) codes and analyze its decoding capacity.

Definition 11 (IEG Code). *Let q, m, n, t, k, N be integers and $k \leq t \leq \min\{n, m\}$. Let $\boldsymbol{g} = (g_1, g_2, \ldots, g_n) \in \mathbb{F}_{q^m}^n$ be the generator of weight t. The IEG code of dimension k and length n generated by $\boldsymbol{g}$ is defined as*

$$\mathrm{IEG}_k(\boldsymbol{g}) = \left\{ \begin{bmatrix} f_1(\boldsymbol{g}) \\ f_2(\boldsymbol{g}) \\ \vdots \\ f_N(\boldsymbol{g}) \end{bmatrix} \; : \; f_i(x) \in \mathcal{L}_{\leq k-1}[x], \quad i \in [N] \right\}.$$

Let $\boldsymbol{y}_i = f_i(\boldsymbol{g}) + \boldsymbol{e}_i, \quad i \in [N]$ where all $\boldsymbol{e}_i$'s share the support of dimension r. We show that if one receives such a set of N words $\boldsymbol{y}_i$'s, it can decode more

errors than the EG codes, specifically correct up to $r = \min\left\{t-k, \frac{N(n-k)}{(N+1)}\right\}$. We give a decoding algorithm based on solving the LR problem and bound the DFR. First, like the LR problem for decoding the EG codes (see Definition 10), we define the LR problem for decoding the IEG codes.

Definition 12 (The $\mathsf{LR}\left(\boldsymbol{g}, \{\boldsymbol{y}_i\}_{i\in[N]}\right)$ Problem for Decoding IEG Codes).
Input : $\boldsymbol{g} = (g_1, g_2, \ldots, g_n) \in \mathbb{F}_{q^m}^n$; $\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_N \in \mathbb{F}_{q^m}^n$.
Output : $v(x), u_i(x) \in \mathcal{L}[x]$ *such that:*

1. $v(x) \neq 0$ *and* $\mathsf{deg}_q v(x) \leq r$; 2. $u_i(x) \leq k+r-1$; 3. $v\left(\boldsymbol{y}_i\right) = u_i\left(\boldsymbol{g}\right)$.

Let $\boldsymbol{Z} = \mathbf{Moore}(\boldsymbol{g}, k+r-1)^\top \in \mathbb{F}_{q^m}^{n\times(k+r)}$ and $\boldsymbol{Y}_i = \mathbf{Moore}(\boldsymbol{y}_i, r)^\top \in \mathbb{F}_{q^m}^{n\times(r+1)}$. Solving the $\mathsf{LR}\left(\boldsymbol{g}, \{\boldsymbol{y}_i\}_{i\in[N]}\right)$ problem is equivalent to solving a linear system $\widehat{\boldsymbol{A}}\boldsymbol{x} = \mathbf{0}_{Nn}$ where

$$\widehat{\boldsymbol{A}} = \left[\begin{array}{c|ccc} \boldsymbol{Y}_1 & \boldsymbol{Z} & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \boldsymbol{Y}_2 & \boldsymbol{0} & \boldsymbol{Z} & \cdots & \boldsymbol{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{Y}_N & \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{Z} \end{array}\right] := \left[\begin{array}{c|c} \boldsymbol{Y}_1 & \\ \boldsymbol{Y}_2 & \\ \vdots & \mathbf{Diag}(\boldsymbol{Z}, N) \\ \boldsymbol{Y}_N & \end{array}\right] \in \mathbb{F}_{q^m}^{nN\times[r+1+N(k+r)]}. \quad (17)$$

If the right kernel $\mathsf{kernel}\left(\widehat{\boldsymbol{A}}\right)$ is one-dimensional, let $\boldsymbol{b} = (b_0, b_1, \ldots, b_{r+N(k+r)})$ be a random non-zero element in $\mathsf{kernel}\left(\widehat{\boldsymbol{A}}\right)$, we set $v'(x) = \sum_{i=0}^{r} b_i x^{q^i}$ and $u_i'(x) = \sum_{j=0}^{k+r-1} b_{r+1+(i-1)(k+r)+j} x^{q^j}$. Then $f_i(x) = -v'(x) \setminus u_i'(x)$. Next, we analyze when $\mathsf{kernel}\left(\widehat{\boldsymbol{A}}\right)$ is one-dimensional, and derive the DFR and the decoding capacity.

Let $\boldsymbol{E}_i = \mathbf{Moore}(\boldsymbol{e}_i, r)^\top \in \mathbb{F}_{q^m}^{n\times(r+1)}$ and $\boldsymbol{\Gamma}_i = -\mathbf{PM}(f_i(x), r) \in \mathbb{F}_{q^m}^{(k+r)\times(r+1)}$. Following Proposition 9, we have

$$\widehat{\boldsymbol{A}} \underbrace{\left[\begin{array}{cccccc} \boldsymbol{I} & \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \boldsymbol{\Gamma}_1 & \boldsymbol{I} & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{\Gamma}_N & \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{I} \end{array}\right]}_{\text{invertible matrix}} = \left[\begin{array}{c|c} \boldsymbol{E}_1 & \\ \boldsymbol{E}_2 & \\ \vdots & \mathbf{Diag}(\boldsymbol{Z}, N) \\ \boldsymbol{E}_N & \end{array}\right] := \widehat{\boldsymbol{B}}.$$

Thus, $\widehat{\boldsymbol{A}}$ is equivalent to $\widehat{\boldsymbol{B}}$. Next, we analyze $\mathsf{dim}\left(\mathsf{kernel}\left(\widehat{\boldsymbol{A}}\right)\right) \geq 1$.

Proposition 11. *For all errors $\boldsymbol{e}_i$'s sharing the same support of dimension r, if $k+r \leq t \leq \min\{n, m\}$ and $r+N(k+r) \leq nN$, then* $\mathsf{rank}\left(\widehat{\boldsymbol{B}}\right) \leq r+N(k+r)$.

The proof of Proposition 11 is put in Appendix A.6. By Proposition 11, we have $\mathsf{rank}\left(\widehat{\boldsymbol{A}}\right) = \mathsf{rank}\left(\widehat{\boldsymbol{B}}\right) \leq r+N(k+r)$. Since $\mathsf{rank}\left(\widehat{\boldsymbol{A}}\right)+\mathsf{dim}\left(\mathsf{kernel}\left(\widehat{\boldsymbol{A}}\right)\right) = r+N(k+r)+1$, we must have $\mathsf{dim}\left(\mathsf{kernel}\left(\widehat{\boldsymbol{A}}\right)\right) \geq 1$. Thus, there must exists

a non-zero element in $\mathsf{kernel}\left(\widehat{\boldsymbol{A}}\right)$.

The DFR decoding the errors of maximal weight depends on $\mathsf{dim}\left(\mathsf{kernel}\left(\widehat{\boldsymbol{A}}\right)\right) > 1$, which is equivalent to $\mathsf{rank}\left(\widehat{\boldsymbol{A}}\right) < r + N(k+r)$. Since $\mathsf{rank}\left(\widehat{\boldsymbol{A}}\right) = \mathsf{rank}\left(\widehat{\boldsymbol{B}}\right)$, the DFR depends on the probability of $\mathsf{rank}\left(\widehat{\boldsymbol{B}}\right) < r + N(k+r)$. This leads to Theorem 6 whose proof is put in Appendix A.7.

Theorem 6. *For any N uniform errors $\boldsymbol{e}_i$'s sharing the same support of dimension r, if $k + r \leq t \leq \min\{n, m\}$ and $r + N(k+r) \leq nN$, then*

$$\Pr\left[\mathsf{rank}\left(\widehat{\boldsymbol{B}}\right) < r + N(k+r)\right] \leq \gamma_q \cdot q^{a(Nt+r-a-Nn)}, \tag{18}$$

where $a = N(t-k-r)+1$.

Theorem 7. *The IEG codes can decode errors of weight up to* $\min\left\{t-k, \left\lfloor \frac{N(n-k)}{N+1} \right\rfloor\right\}$.

Proof. For all errors $\boldsymbol{e}_i$ sharing the same support of dimension r, from Theorem 6, we known that when $k + r \leq t$ and $r + N(k+r) \leq nN$, the IEG codes can decode such errors $\boldsymbol{e}_i$'s. This leads to conclusion. □

Theorem 6 considers N errors sharing r-dimensional support, and the decoding will fail if $\mathsf{rank}\left(\widehat{\boldsymbol{A}}\right) < N(k+r) + r$. However, when N errors share the support of dimension $< r$ (say dimension is w) really occurs, we must have $\mathsf{rank}\left(\widehat{\boldsymbol{A}}\right) < N(k+r) + r$:

$$\mathsf{rank}(\boldsymbol{L}) = \mathsf{rank}\left(\mathbf{Moore}(\boldsymbol{\varepsilon}, r)\right) + \mathsf{rank}\left(\mathbf{Diag}(\boldsymbol{N}_2, N)\right) = w + N(k+r) < N(k+r) + r,$$

$$\mathsf{rank}\left(\widehat{\boldsymbol{A}}\right) = \mathsf{rank}\left(\widehat{\boldsymbol{A}}^{\top}\right) \leq \min\{\mathsf{rank}(\boldsymbol{L}), \mathsf{rank}(\boldsymbol{M})\} \leq \mathsf{rank}(\boldsymbol{L}) < N(k+r) + r,$$

where $\boldsymbol{\varepsilon} \in \mathbb{F}_{q^m}^w$ is a basis of support of the error of weight w.

Then, in this case, whether it can decode successfully, how to recover $f_i(x)$, and how to determine DFR ? We discuss this case in Appendix C. We there show that if $\mathsf{dim}\left(\mathsf{kernel}\left(\widehat{\boldsymbol{A}}\right)\right)$ is controlled as $r - w + 1$, any non-zero element in $\mathsf{kernel}\left(\widehat{\boldsymbol{A}}\right)$ can determine $f_i(x)$, and the DFR $\leq \gamma_q \cdot q^{a(Nt+w-a-Nn)}$. The resulting decoding procedure is presented in Algorithm 3.

Decoding Complexity. The complexity of Algorithm 3 is dominated by Step 2. The right kernel of the system $\boldsymbol{A}\boldsymbol{x}^{\top} = \boldsymbol{0}_n$ can be found in $\mathcal{O}\left((Nn)^3\right)$ operations in $\mathbb{F}_{q^m}$ by Gaussian elimination. Thus, the complexity is bounded by $\mathcal{O}\left((Nn)^3\right)$. We leave more improvements as independent interest for further study.

Algorithm 3 Decoding errors of weight w for the IEG codes ($w \le r$)

Input: $q, m, n, k, t, w, r, N \in \mathbb{N}$, $k \le t \le \min\{n, m\}$,
$\boldsymbol{g} \in \mathbb{F}_{q^m}^n$ and $\|\boldsymbol{g}\|_{\mathrm{R}} = t$; $\{\boldsymbol{y}_i\}_{i\in[N]} \in (\mathbb{F}_{q^m}^n)^N$.
Output: $f_i(x) \in \mathcal{L}_{\le k-1}[x]$; $\boldsymbol{e}_i \in \mathbb{F}_{q^m}^n$ and $\mathsf{dim}(\mathrm{Supp}(\boldsymbol{e}_i)) = w$

1: Compute $\boldsymbol{Y}_i = \mathbf{Moore}(\boldsymbol{y}_i, r)^\top \in \mathbb{F}_{q^m}^{n\times(r+1)}$ and $\boldsymbol{Z} = \mathbf{Moore}(\boldsymbol{g}, k+r-1)^\top \in \mathbb{F}_{q^m}^{n\times(k+r)}$, construct the matrix $\widehat{\boldsymbol{A}} \in \mathbb{F}_{q^m}^{nN\times[r+1+N(k+r)]}$ as in Equation (17).
2: Compute the right kernel $\mathsf{kernel}\left(\widehat{\boldsymbol{A}}\right)$ of the linear system $\widehat{\boldsymbol{A}}\boldsymbol{x}^\top = \boldsymbol{0}_{Nn}$.
3: Recover message q-polynomials $f_i(x)$ and errors $\boldsymbol{e}_i$ for $i \in [N]$:
4: Let $\boldsymbol{b} = (b_0, b_1, \ldots, b_{r+N(k+r)})$ be a random non-zero element in $\mathsf{kernel}\left(\widehat{\boldsymbol{A}}\right)$.
5: Set $v'(x) = \sum_{i=0}^{r} b_i x^{q^i}$ and $u_i'(x) = \sum_{j=0}^{k+r-1} b_{r+1+(i-1)(k+r)+j} x^{q^j}$.
6: Set $f_i(x) = -v'(x) \setminus u_i'(x)$ by left division.
// Holds if $\mathsf{dim}\left(\mathsf{kernel}\left(\widehat{\boldsymbol{A}}\right)\right) = r - w + 1$
7: **if** $\mathsf{deg}_q f_i(x) \le k-1$ and $\|\boldsymbol{y}_i - f_i(\boldsymbol{g})\|_{\mathrm{R}} = w$:
8: **return** $f_i(x)$ and $\boldsymbol{e}_i := \boldsymbol{y}_i - f_i(\boldsymbol{g})$
9: **else**:
10: **return** $\perp$

8 Relations with (Interleaved) Gabidulin Codes

In this section, by our DFR idea of decoding EG codes, we show that decoding interleaved Gabidulin codes is deterministic. This problem is left in [31], where the authors only presented a proof in a vague case that the number of columns of the system is exactly equal to number of rows plus 1. This leads to a mislead that the most previous works [44,50,17,6] thought that there exists only efficient *probabilistic* decoding algorithms for interleaved Gabidulin codes. The result covers the fact that decoding Gabidulin codes is deterministic. This work also supports the validity of our heuristic argument for Equation (12). If our heuristic argument is invalid, then decoding Gabidulin codes by the Gaussian elimination would not be deterministic. This challenges the well-known conclusion: decoding Gabidulin codes is deterministic, which has stood for over 40 years.

Definition 13 (Gabidulin Code). *Let q, m, n, k be integers and $k \le n \le m$. Let $\boldsymbol{g} = (g_1, \ldots, g_n) \in \mathbb{F}_{q^m}^n$ be a vector of weight n, which is called a generator. The Gabidulin code of dimension k and length n generated by $\boldsymbol{g}$ is defined as*

$$\mathrm{Gab}_k(\boldsymbol{g}) = \Big\{ f(\boldsymbol{g}) = (f(g_1), f(g_2), \ldots, f(g_n)) \ : \ f(x) \in \mathcal{L}_{\le k-1}[x] \Big\}.$$

Given N q-polynomials $f_i(x) \in \mathcal{L}_{\le k-1}[x]$ for $i \in [N]$, the decoding model of the interleaved Gabidulin codes is

$$\boldsymbol{y}_i = f_i(\boldsymbol{g}) + \boldsymbol{e}_i, \quad i \in [N]$$

where all $\boldsymbol{e}_i$'s share the support of dimension r. Note that when $N = 1$, the decoding model is exactly that of the Gabidulin codes.

By the definition of the EG codes (Definition 7), when $t = n$ and $m \geq n$, the EG codes are exactly Gabidulin codes. We adapt the results of our DFR to the case of interleaved Gabidulin codes and obtain the null DFR in Theorem 8.

Theorem 8. *For any N uniform errors $\boldsymbol{e}_i$'s sharing the same support of dimension r, if $r \leq \left\lfloor \frac{N(n-k)}{N+1} \right\rfloor$, then the decoding of interleaved Gabidulin codes is deterministic, i.e., the DFR is null.*

Theorem 8 can be viewed as a direct application of Theorem 6. If $r \leq \left\lfloor \frac{N(n-k)}{N+1} \right\rfloor$, then $Nn \geq N(k+r)+r$. In the case of Gabidulin codes, $a = N(t-k-r)+1 = N(n-k-r)+1 \geq r+1 > r$. This is an impossible event, hence $\Pr[\mathsf{rank}\left(\widehat{\boldsymbol{B}}^{\top}\right) < N(k+r)+r] = 0$. This means that the DFR is null. We present the formal proof in Appendix A.8.

9 Applications to RQC PKE

In this section, we apply the EG codes to RQC PKE [46] with the ideal blockwise rank decoding problems. The details are presented in Appendix I. We choose two types of parameters sets: **Our RQC** and **Our Conservative RQC**. Here, we only present the overall performance comparison. The definition of PKE is recalled in Appendix D. The ideal blockwise decoding problems are recalled in Appendices E-G. The best attacks on decoding problems are recalled in Appendix H. The specific security parameters for our RQC are given in Table 7 in Appendix I.2.

In Table 3, we present sizes and DFR for our RQC and their original versions. For 128-bit security, our RQC has a bandwidth of 1690 bytes, which is about 69% and 34% more compact than RQC (NIST PQC) and RQC (Asiacrypt 2023), respectively. This improvement is more significant for higher security levels.

In Table 4, we provide reference timings for our RQC and their original versions. Our RQC achieves about 60% improvement over RQC (NIST PQC). These timings are roughly 1000 times greater than when running on C language. Thus, the efficiency is practical.

10 Conclusion and Future Work

In this paper, we investigated the (interleaved) EG codes and applied the EG codes to rank-based cryptosystems. By our analysis by solving the LR problem, the EG codes have the DFR which can be made arbitrarily small and can exactly decode up to the RGV bound. We exploited these decoding gains to improve RQC in size and efficiency. The resulting bandwidth and efficiency are quite practical. Compared to HQC finalized by NIST PQC, our RQC also offers a very competitive bandwidth. We also proved the null DFR of the Interleaved Gabidulin codes. Our work provided a broad space for enriching coding theory and designing rank-based cryptosystems. In the future, we would like to conduct more study.

Table 3. Comparison of sizes and DFR for RQC.

Schemes	pt	sk	pk	ct	total	DFR
Our RQC-128	159	40	590	1100	1690	2^{-133}
Our RQC-192	236	40	837	1594	2431	2^{-202}
Our RQC-256	292	40	1291	2566	3857	2^{-258}
Our Conservative RQC-128	171	40	796	1512	2308	2^{-138}
Our Conservative RQC-192	249	40	1711	3342	5053	2^{-207}
Our Conservative RQC-256	339	40	3190	6300	9490	2^{-274}
RQC-128 (Asiacrypt [46])	581	40	860	1704	2564	-
RQC-192 (Asiacrypt [46])	381	40	1834	3652	5486	-
RQC-256 (Asiacrypt [46])	417	40	2421	4826	7247	-
RQC-128 (NIST [34])	381	40	1834	3652	5486	-
RQC-192 (NIST [34])	755	40	2853	5690	8543	-
RQC-256 (NIST [34])	543	40	4090	8164	12254	-

Plaintext size (pt); private key size (st); public key size (pk); ciphertext size (ct); bandwidth (total): pk + ct; Decryption Failure Rate (DFR).

Table 4. Comparison of timings for RQC.

Schemes	KGen (ms)	Enc (ms)	Dec (ms)	Total (ms)
Our RQC-128	90	98	201	389
Our RQC-192	129	136	464	721
Our RQC-256	195	201	825	1221
Our Conservative RQC-128	109	116	448	673
Our Conservative RQC-192	235	250	1080	1565
Our Conservative RQC-256	458	481	3480	4419
RQC-128 (Asiacrypt [46])	134	136	249	519
RQC-192 (Asiacrypt [46])	285	350	665	1300
RQC-256 (Asiacrypt [46])	383	405	1620	2408
RQC-128 (NIST PQC [34])	238	250	447	935
RQC-192 (NIST PQC [34])	359	399	1250	2008
RQC-256 (NIST PQC [34])	527	561	2430	3518

The schemes are implemented on SageMath 9.5. The benchmark is Ubuntu-22.04 + WSL + Intel(R) Core(TM) i5-8265U CPU @ 1.60GHz with SageMath 9.5. The test scripts are available online at https://github.com/RQCPKE/EGCodesRQC. Total: KGen + Enc + Dec.

- We hope the Heuristic argument deriving DFR is well proven.
- The DFR of the Welch-Berlekamp like algorithm is based on a lot of simulations, and we hope give more theoretical supports in the future.
- Try to adapt the other decoding algorithms of Gabidulin codes, such as Extended Euclidean Algorithm for Linearized polynomials (LEEA) [49] and Gao-like algorithm [48], to decode the EG codes and derive DFR.
- Study the list decoding of the EG codes for the radius τ: $\min\left\{t-k, \left\lfloor \frac{n-k}{2} \right\rfloor\right\} \leq \tau \leq m$.
- Generalize the EG codes over fields of any characteristic.

- Apply the EG codes to matrix/subspace codes in random network coding [45].
- Mask EG codes and try to improve LowMS [6] and MinRank-McEliece cryptosystems [5], even design full domain Hash-and-Sign signatures.
- Apply the interleaved EG codes to improve and design the rank-based cryptosystems.

Acknowledgement

We would like to thank the anonymous reviewers from Eurocrypt & Crypto 2025, and also welcome the suggestions from the community.

References

1. Alekhnovich, M.: More on average case vs approximation complexity. In: Proceedings of the 44th Symposium on Foundations of Computer Science (FOCS). pp. 298–307. IEEE Computer Society (2003) 3
2. Aragon, N., Barreto, P., Bettaieb, S., et al.: BIKE. Fourth round submission to the NIST post-quantum cryptography call (2022), https://bikesuite.org/ 8
3. Aragon, N., Blazy, O., Deneuville, J., Gaborit, P., Zémor, G.: Ouroboros: An efficient and provably secure KEM family. IEEE Transactions on Information Theory **68**(9), 6233–6244 (2022) 53, 54
4. Aragon, N., Briaud, P., Dyseryn, V., Gaborit, P., Vinçotte, A.: The blockwise rank syndrome learning problem and its applications to cryptography. In: Post-Quantum Cryptography (PQCrypto). vol. 14771, pp. 75–106. Springer (2024) 3, 4, 7, 8, 54
5. Aragon, N., Couvreur, A., Dyseryn, V., Gaborit, P., Vinçotte, A.: MinRank Gabidulin encryption scheme on matrix codes. In: Advances in Cryptology - ASIACRYPT. vol. 15487, pp. 68–100. Springer (2024) 10, 34
6. Aragon, N., Dyseryn, V., Gaborit, P., Loidreau, P., Renner, J., Wachter-Zeh, A.: LowMS: a new rank metric code-based KEM without ideal structure. Designs, Codes and Cryptography **92**(4), 1075–1093 3, 5, 7, 31, 34
7. Aragon, N., Gaborit, P., Hauteville, A., Ruatta, O., Zémor, G.: Low rank parity check codes: New decoding algorithms and applications to cryptography. IEEE Transactions on Information Theory **65**(12), 7697–7717 (2019) 2, 53
8. Aragon, N., Gaborit, P., Hauteville, A., Tillich, J.: A new algorithm for solving the rank syndrome decoding problem. In: International Symposium on Information Theory (ISIT). pp. 2421–2425. IEEE (2018) 3, 54, 55
9. Augot, D., Loidreau, P., Robert, G.: Generalized Gabidulin codes over fields of any characteristic. Designs, Codes and Cryptography **86**(8), 1807–1848 (2018) 2, 6, 9, 10, 17, 18, 19, 27, 28
10. Bardet, M., Briaud, P., Bros, M., Gaborit, P., Neiger, V., Ruatta, O., Tillich, J.: An algebraic attack on rank metric code-based cryptosystems. In: Advances in Cryptology - EUROCRYPT. vol. 12107, pp. 64–93. Springer (2020) 4, 56
11. Bardet, M., Briaud, P., Bros, M., Gaborit, P., Tillich, J.: Revisiting algebraic attacks on MinRank and on the rank decoding problem. Designs, Codes and Cryptography **91**(11), 3671–3707 (2023) 4

12. Bardet, M., Bros, M., Cabarcas, D., Gaborit, P., Perlner, R.A., Smith-Tone, D., Tillich, J., Verbel, J.A.: Improvements of algebraic attacks for solving the rank decoding and MinRank problems. In: Advances in Cryptology - ASIACRYPT. vol. 12491, pp. 507–536. Springer (2020) 4, 56
13. Berger, T.P., Ourivski, A.: Construction of new mds codes from Gabidulin codes. In: Proceedings of ACCT. pp. 40–47 (2009) 2, 3, 16
14. Bernstein, D.J., Chou, T., Cid, C., et al.: Classic McEliece. Fourth round submission to the NIST post-quantum cryptography call (2022), https://classic.mceliece.org/ 8
15. Bidoux, L., Briaud, P., Bros, M., Gaborit, P.: RQC revisited and more cryptanalysis for rank-based cryptography. IEEE Transactions on Information Theory **70**(3), 2271–2286 (2024) 2, 3, 4, 7, 9, 10, 54
16. Buss, J.F., Frandsen, G.S., Shallit, J.O.: The computational complexity of some problems of linear algebra. Journal of Computer and System Sciences **58**(3), 572–596 (1999) 51
17. Couvreur, A., Bombar, M.: Right-hand side decoding of Gabidulin codes and applications. In: International Workshop on Coding and Cryptography (WCC) (2022) 31
18. Debris-Alazard, T., Tillich, J.: Two attacks on rank metric code-based schemes: RankSign and an IBE Scheme. In: Advances in Cryptology - ASIACRYPT. vol. 11272, pp. 62–92. Springer (2018) 39
19. Delsarte, P.: Bilinear forms over a finite field, with applications to coding theory. Journal of Combinatorial Theory **25**(3), 226–241 (1978) 2
20. Etzion, T., Vardy, A.: Error-correcting codes in projective space. IEEE Transactions on Information Theory **57**(2), 1165–1173 (2011) 39
21. Faugère, J., Levy-dit-Vehel, F., Perret, L.: Cryptanalysis of MinRank. In: Advances in Cryptology - CRYPTO. vol. 5157, pp. 280–296. Springer (2008) 51
22. Gabidulin, E.M.: Theory of codes with maximum rank distance. Problemy peredachi informatsii **21**(1), 3–16 (1985) 2
23. Gaborit, P., Murat, G., Ruatta, O., Zémor, G.: Low rank parity check codes and their application to cryptography. In: The Workshop on Coding and Cryptography (WCC). http://www.selmer.uib.no/WCC2013/pdfs/Gaborit.pdf 2
24. Gaborit, P., Hauteville, A., Phan, D.H., Tillich, J.: Identity-based encryption from codes with rank metric. In: Advances in Cryptology - CRYPTO. vol. 10403, pp. 194–224. Springer (2017) 2, 38, 39
25. Gaborit, P., Ruatta, O., Schrek, J.: On the complexity of the rank syndrome decoding problem. IEEE Transactions on Information Theory **62**(2), 1006–1019 (2016) 3, 54, 55
26. Gaborit, P., Zémor, G.: On the hardness of the decoding and the minimum distance problems for rank codes. IEEE Transactions on Information Theory **62**(12), 7245–7252 (2016) 51
27. Gemmell, P., Sudan, M.: Highly resilient correctors for polynomials. Information Processing Letters **43**(4), 169–174 (1992) 27
28. Hauteville, A.: Décodage en métrique rang et attaques sur un système de chiffrement à base de codes LRPC, Master's thesis, University of Limoges, September, 2014 39
29. Hauteville, A., Tillich, J.: New algorithms for decoding in the rank metric and an attack on the LRPC cryptosystem. In: IEEE International Symposium on Information Theory (ISIT). pp. 2747–2751. IEEE (2015) 54, 59

30. Loidreau, P.: A Welch-Berlekamp like algorithm for decoding Gabidulin codes. In: International Workshop on Coding and Cryptography (WCC). vol. 3969, pp. 36–45. Springer (2005) 6, 9, 10, 17, 18, 19, 25, 27
31. Loidreau, P.: Decoding rank errors beyond the error-correcting capability. In: International Workshop on Algebraic and Combinatorial Coding Theory (ACCT). pp. 168–190 (2006) 2, 5, 6, 7, 9, 10, 18, 19, 30
32. Loidreau, P.: Asymptotic behaviour of codes in rank metric over finite fields. Designs, Codes and Cryptography. **71**(1), 105–118 (2014) 38, 39
33. Melchor, C.A., Aragon, N., Bardet, M., et al.: ROLLO. Second round submission to the NIST post-quantum cryptography call (2020), https://pqc-rollo.org/ 54
34. Melchor, C.A., Aragon, N., Bettaieb, S., et al.: RQC. Second round submission to the NIST post-quantum cryptography call (2020), http://pqc-rqc.org/ 3, 4, 8, 33, 53, 54, 57, 58
35. Melchor, C.A., Aragon, N., Bettaieb, S., et al.: HQC. Fourth round submission to the NIST post-quantum cryptography call (2022), http://pqc-hqc.org 8
36. Melchor, C.A., Aragon, N., Dyseryn, V., Gaborit, P., Zémor, G.: LRPC codes with multiple syndromes: Near ideal-size KEMs without ideals. In: Post-Quantum Cryptography (PQCrypto). vol. 13512, pp. 45–68. Springer (2022) 3
37. Melchor, C.A., Blazy, O., Deneuville, J., Gaborit, P., Zémor, G.: Efficient encryption from random quasi-cyclic codes. IEEE Transactions on Information Theory **64**(5), 3927–3943 (2018) 3
38. NIST: Status report on the second round of the NIST post-quantum cryptography standardization process (2020), https://nvlpubs.nist.gov/nistpubs/ir/2020/NIST.IR.8309.pdf 3
39. NIST: Status report on the third round of the NIST post-quantum cryptography standardization process (2022), https://doi.org/10.6028/NIST.IR.8413-upd1 3
40. Ore, O.: On a special class of polynomials. Transactions of the American Mathematical Society **35**(3), 559–584 (1933) 2, 12
41. Ourivski, A.V., Johansson, T.: New technique for decoding codes in the rank metric and its cryptography applications. Problems of Information Transmission **38**(3), 237–246 (2002) 3
42. Puchinger, S., Renner, J., Rosenkilde, J.: Generic decoding in the sum-rank metric. IEEE Transactions on Information Theory **68**(8), 5075–5097 (2022) 55
43. Schwabe, P., Avanzi, R., Bos, J., et al.: Kyber. Third round submission to the NIST post-quantum cryptography call (2022), https://pq-crystals.org/ 8
44. Sidorenko, V., Jiang, L., Bossert, M.: Skew-feedback shift-register synthesis and decoding interleaved Gabidulin codes. IEEE Transactions on Information Theory **57**(2), 621–632 (2011) 2, 5, 7, 9, 31
45. Silva, D., Kschischang, F.R., Koetter, R.: A rank-metric approach to error control in random network coding. IEEE Transactions on Information Theory **54**(9), 3951–3967 (2008) 34
46. Song, Y., Zhang, J., Huang, X., Wu, W.: Blockwise rank decoding problem and LRPC codes: Cryptosystems with smaller sizes. In: Advances in Cryptology - ASIACRYPT. vol. 14444, pp. 284–316. Springer (2023) 3, 4, 8, 32, 33, 51, 54, 56, 59
47. Song, Y., Zhang, J., Huang, X., Wu, W.: Blockwise rank decoding problem and LRPC codes: Cryptosystems with smaller sizes. IEEE Transactions on Information Theory (2025). https://doi.org/10.1109/TIT.2025.3555075 54
48. Wachter, A., Sidorenko, V., Bossert, M.: A fast linearized Euclidean algorithm for decoding Gabidulin codes. In: Proceedings of ACCT. pp. 298–303 (2010) 9, 33

49. Wachter-Zeh, A., Afanassiev, V.B., Sidorenko, V.: Fast decoding of gabidulin codes. Designs, Codes and Cryptography **66**(1-3), 57–73 (2013) 9, 33
50. Wachter-Zeh, A., Zeh, A.: List and unique error-erasure decoding of interleaved Gabidulin codes with interpolation techniques. Designs, Codes and Cryptography **73**(2), 547–570 (2014) 5, 7, 31

Auxiliary Supporting Material

A Omitted Definitions, Lemmata, Proofs, and Tables

A.1 Omitted Definitions

Definition 14 (Rank Gilbert-Varshamov (RGV) Bound [32,24]). *Let $S(q,m,n,r)$ be the number of errors of weight r in $\mathbb{F}_{q^m}^n$, which equals to the number of $m \times n$ matrices in $\mathbb{F}_q^{m\times n}$ of rank r, i.e.,*

$$S(q,m,n,r) = \begin{bmatrix} n \\ r \end{bmatrix}_q \prod_{j=0}^{r-1}(q^m - q^j) = \prod_{j=0}^{r-1} \frac{(q^m - q^j)(q^n - q^j)}{q^r - q^j}.$$

Let $\mathcal{V}(q,m,n,r)$ be the number of errors of weight $\leq r$ in $\mathbb{F}_{q^m}^n$:

$$\mathcal{V}(q,m,n,r) = \sum_{i=0}^{r} S(q,m,n,i) = \sum_{i=0}^{r}\prod_{j=0}^{i-1} \frac{(q^m - q^j)(q^n - q^j)}{q^i - q^j}.$$

The RGV bound $d_{\mathrm{RGV}}(m,n,k)$ is the smallest $r \in \mathbb{N}$ such that $\mathcal{V}(q,m,n,r) \geq q^{(n-k)m}$. When either m or n tends to infinity, $\mathcal{V}(q,m,n,r) \approx q^{r(m+n-r)}$. Let $q^{r(m+n-r)} = q^{(n-k)m}$, we have

$$d_{\mathrm{RGV}}(m,n,k) = \begin{cases} \frac{m+n-\sqrt{(m-n)^2+4km}}{2}, & m \neq n; \\ n\left(1 - \sqrt{\frac{k}{n}}\right), & m = n. \end{cases} \tag{19}$$

From Definition 14, the RGV bound for an $[n,k]_{q^m}$ code $\mathcal{C}$ with parity-check matrix $\boldsymbol{H} \in \mathbb{F}_{q^m}^{(n-k)\times n}$, corresponds to the smallest weight r for which, for any syndrome $\boldsymbol{s} \in \mathbb{F}_{q^m}^{n-k}$, there exists on average an error $\boldsymbol{e}$ of weight r such that $\boldsymbol{H}\boldsymbol{e}^\top = \boldsymbol{s}$. Note that the approximation $q^{r(m+n-r)}$ is very tight and is the lower bound of $\mathcal{V}(q,m,n,r)$ (see Proposition 1 of [32]):

$$q^{r(m+n-r)} \leq \mathcal{V}(q,m,n,r) \leq \gamma_q \cdot q^{r(m+n-r)},$$

where $\gamma_q = \prod_{j=1}^{\infty} \frac{1}{1-q^{-j}}$ is always greater than 1 and is monotonically decreasing in q, e.g., $\gamma_2 \approx 3.463$, $\gamma_3 \approx 1.785$, and $\gamma_4 \approx 1.452$. This means that the value r satisfying $\mathcal{V}(q,m,n,r) = q^{m(n-k)}$ must be less than or equal to one satisfying $q^{r(m+n-r)} = q^{(n-k)m}$. In other words, the accurate RGV bound must be less than or equal to the (asymptotical) RGV bound in Equation (19). Further, if the code $\mathcal{C}$ can decode up to the RGV bound in Equation (19), then its decoding capacity is actually beyond the accurate RGV bound.

Hence, we obtain that if the code $\mathcal{C}$ can efficiently decode up to the RGV bound in Equation (19), then any syndrome (resp. word) in $\mathbb{F}_{q^m}^{n-k}$ (resp. $\mathbb{F}_{q^m}^{n}$) can be decoded efficiently. The EG codes are exactly this case (see Theorem 5).

Definition 15 (Rank Singleton (RS) Bound [32,18]). *The RS bound* $d_{\mathrm{RS}}(m,n,k)$ *for an* $[n,k]_{q^m}$*-linear code is defined as* $d_{\mathrm{RS}}(m,n,k) = \left\lfloor \frac{m(n-k)}{\max(m,n)} \right\rfloor + 1$.

The minimal distance of rank metric codes is less than or equal to the RS bound. The code with the minimal distance reaching the RS bound is called a Maximum Rank Distance (MRD) code.

A.2 Omitted Lemmata

Lemma 1. *Let* $q,n,r \in \mathbb{N}$ *and* $n \geq r$. $q^{r(n-r)} \leq \begin{bmatrix} n \\ r \end{bmatrix}_q < \gamma_q \cdot q^{r(n-r)}$.

Proof. By definition of Gaussian binomials, we have

$$\begin{bmatrix} n \\ r \end{bmatrix}_q = \prod_{i=0}^{r-1} \frac{q^n - q^i}{q^r - q^i} = q^{r(n-r)} \prod_{i=0}^{r-1} \frac{1-q^{i-n}}{1-q^{i-r}}.$$

Since $n \geq r$, we get $\prod_{i=0}^{r-1} \frac{1-q^{i-n}}{1-q^{i-r}} \geq 1$ which yields the left-hand inequality. To get the other inequality, we bound the product:

$$\prod_{i=0}^{r-1} \frac{1-q^{i-n}}{1-q^{i-r}} \leq \prod_{i=0}^{r-1} \frac{1}{1-q^{i-r}} = \prod_{k=0}^{r-1} \frac{1}{1-\frac{1}{q^{k+1}}} < \prod_{j=1}^{\infty} \frac{1}{1-\frac{1}{q^j}}.$$

The right-hand inequality is obtained by taking $\gamma_q = \prod_{j=1}^{\infty} \frac{1}{1-q^{-j}}$. Note that γ_q is always greater than 1 and is monotonically decreasing in q, e.g., $\gamma_2 \approx 3.463$, $\gamma_3 \approx 1.785$, and $\gamma_4 \approx 1.452$. □

Lemma 2 is to quantify the probability that the fixed subspace U and the random subspace V have an intersection of a certain dimension. We will use it to estimate the decoding failure rate. We found a trace of this probability in [28] (Proposition 3.2) and its application in [24] (Lemma 5). Here, we provide a concise proof and precise upper bound.

Lemma 2. *Let* $t,r,a \in \mathbb{N}$ *and* $a \leq \min\{r,t\}$. *Let* U *be a fixed* t*-dimensional subspace of* $\mathbb{F}_q^n$. *Let* V *be a random* r*-dimensional subspace of* $\mathbb{F}_q^n$. *Then*

$$\Pr\left[\dim(U \cap V) = a\right] \leq \gamma_q \cdot q^{a(t+r-a-n)}.$$

Proof. By Lemma 7 in [20], given subspace U of dimension t, there are $\begin{bmatrix} t \\ a \end{bmatrix}_q$ ways to choose an a-dimensional subspace W of U. For a fixed W, the number of r-dimensional subspaces V such that $U \cap V = W$ is

$$\frac{(q^n - q^t)\left(q^n - q^{t+1}\right) \cdots \left(q^n - q^{t+r-a-1}\right)}{(q^r - q^a)\left(q^r - q^{a+1}\right) \cdots (q^r - q^{r-1})} = \begin{bmatrix} n-t \\ r-a \end{bmatrix}_q q^{(r-a)(t-a)}.$$

The total number of r-dimensional subspace V of $\mathbb{F}_q^n$ is $\begin{bmatrix} n \\ r \end{bmatrix}_q$. Thus, we have

$$\Pr\left[\dim(U \cap V) = a\right] = \frac{\begin{bmatrix} t \\ a \end{bmatrix}_q \begin{bmatrix} n-t \\ r-a \end{bmatrix}_q q^{(r-a)(t-a)}}{\begin{bmatrix} n \\ r \end{bmatrix}_q} \leq \gamma_q \cdot q^{a(t+r-a-n)}.$$

The inequality follows the conclusion in Lemma 1. □

A.3 The Proof of Proposition 6

Proof (Proposition 6). Given any $f(x) \in \mathcal{L}_{\leq k-1}[x]$, let $f(x) = \sum_{i=0}^{k-1} f_i x^{q^i}$. Then the codeword $f(\boldsymbol{g})$ of the $\mathrm{EG}(\boldsymbol{g})$ code can be expressed as

$$f(\boldsymbol{g}) = (f(g_1), f(g_2), \ldots, f(g_n)) = (f_0, f_1, \ldots, f_{k-1})\,\mathbf{Moore}(\boldsymbol{g}, k-1).$$

Let $\boldsymbol{G} = \mathbf{Moore}(\boldsymbol{g}, k-1)$. Since $k-1 < t \leq \min\{n, m\}$, from Proposition 5, we have $\mathsf{rank}(\boldsymbol{G}) = k$. Thus, the dimension of the $\mathrm{EG}(\boldsymbol{g})$ code is k, and $\boldsymbol{G}$ is its generator matrix. □

A.4 The Proof of Proposition 7

Proof (Proposition 7). Assume that the first t coordinates of $\boldsymbol{g}$ are linearly independent; otherwise, performing a permutation. Let $\boldsymbol{\tau} \in \mathbb{F}_{q^m}^t$ be a basis of $\mathrm{Supp}(\boldsymbol{g})$. Let $\mathbf{CM}(\boldsymbol{g}) \in \mathbb{F}_q^{t\times n}$ of rank t be the coefficient matrix of $\boldsymbol{g}$ under $\boldsymbol{\tau}$ such that $\boldsymbol{g} = \boldsymbol{\tau}\mathbf{CM}(\boldsymbol{g})$. Then there exists an invertible matrix $\boldsymbol{P} \in \mathbb{F}_q^{n\times n}$ s.t., the last $n-t$ columns of $\mathbf{CM}(\boldsymbol{g})$ are zeros, i.e.,

$$\mathbf{CM}(\boldsymbol{g})\boldsymbol{P} = [\boldsymbol{V} \quad \mathbf{0}_{t\times n-t}], \qquad \boldsymbol{V} \in \mathbb{F}_q^{t\times t}.$$

Further there exists a vector $\boldsymbol{g}' = (g'_1, g'_2, ..., g'_t, 0, ..., 0)$ s.t., $\boldsymbol{g}' = \boldsymbol{\tau}\mathbf{CM}(\boldsymbol{g})\boldsymbol{P} = \boldsymbol{g}\boldsymbol{P}$.

We have $\boldsymbol{G}\boldsymbol{P} = \mathbf{Moore}(\boldsymbol{g}, k-1)\boldsymbol{P} = \mathbf{Moore}(\boldsymbol{g}\boldsymbol{P}, k-1) = \mathbf{Moore}(\boldsymbol{g}', k-1)$. For rank metric codes over $\mathbb{F}_{q^m}$, the invertible matrix $\boldsymbol{P}$ over the basis field is an isometry. Then the EG codes defined by $\boldsymbol{G}$ are equivalent to the codes defined by $\boldsymbol{G}\boldsymbol{P}$. The first t columns of $\boldsymbol{G}\boldsymbol{P}$ define a Gabidulin code of dimension k and length t with the minimal distance $t-k+1$. Thus, the EG codes have the minimal distance $t-k+1$. □

A.5 Simulated DFR for the Welch-Berlekamp Like Algorithm

In Table 5, we test five types of simulable code parameters, and set five groups of parameters for each type. We simulated DFR for each group by performing decoding algorithms (Algorithm 2) in 10^5 times. The theoretical DFR is estimated by Equations (14) and (15) with $\gamma_2 = 4$, and $\gamma_q = 2$ if $q > 2$.

The theoretical DFR for the types decoding up to the RGV bound is always greater than 1, and we recorded these DFR as 1 (see No. 16 - 25). Here, d is the minimal distance, d_{RGV} is the RGV bound (Definition 14), and d_{RS} is the RS bound (Definition 15). The test scripts are available online at https://github.com/RQCPKE/EGCodesRQC.

Table 5. Theoretical and simulated DFR of the EG codes for errors of weight r using **the Welch-Berlekamp like algorithm**.

Types	No.	Parameters (q, m, n, t, k, r)	Theoretical DFR	Simulated DFR	d	d_{RGV}	d_{RS}
Increase m $(t = m < n)$	1	$(2, 31, 41, 31, 9, 16)$	(2^{-5}) 0.0313	0.0150	13	19	25
	2	$(2, 32, 41, 32, 9, 16)$	(2^{-6}) 0.0156	0.0076	14	19	25
	3	$(2, 33, 41, 33, 9, 16)$	(2^{-7}) 0.0078	0.0038	15	20	26
	4	$(2, 34, 41, 34, 9, 16)$	(2^{-8}) 0.0039	0.0017	16	20	27
	5	$(2, 35, 41, 35, 9, 16)$	(2^{-9}) 0.0020	0.0008	17	20	28
Increase t $(t < m < n)$	6	$(2, 35, 41, 30, 9, 16)$	(2^{-4}) 0.0625	0.0310	22	20	28
	7	$(2, 35, 41, 31, 9, 16)$	(2^{-5}) 0.0313	0.0150	23	20	28
	8	$(2, 35, 41, 32, 9, 16)$	(2^{-6}) 0.0156	0.0084	24	20	28
	9	$(2, 35, 41, 33, 9, 16)$	(2^{-7}) 0.0078	0.0036	25	20	28
	10	$(2, 35, 41, 34, 9, 16)$	(2^{-8}) 0.0039	0.0018	26	20	28
Increase t $(t < n < m)$	11	$(2, 29, 26, 16, 5, 10)$	(2^{-2}) 0.0250	0.1320	11	16	22
	12	$(2, 29, 26, 17, 5, 10)$	(2^{-4}) 0.0625	0.0372	13	16	22
	13	$(2, 29, 26, 18, 5, 10)$	(2^{-6}) 0.0156	0.0098	14	16	22
	14	$(2, 29, 26, 19, 5, 10)$	(2^{-8}) 0.0039	0.0025	15	16	22
	15	$(2, 29, 26, 20, 5, 10)$	(2^{-10}) 0.00098	0.00058	16	16	22
Decoding RGV Bound $(k > r)$	16	$(2, 30, 37, 30, 23, 7)$	1	0.7120	8	7	12
	17	$(2, 30, 38, 30, 22, 8)$	1	0.7087	9	8	13
	18	$(2, 30, 39, 30, 21, 9)$	1	0.7100	10	9	14
	19	$(2, 30, 40, 30, 20, 10)$	1	0.7105	11	10	16
	20	$(2, 30, 41, 30, 19, 11)$	1	0.7141	12	11	17
Decoding RGV Bound $(k < r)$	21	$(2, 21, 34, 21, 8, 13)$	1	0.7124	14	13	17
	22	$(2, 22, 36, 22, 8, 14)$	1	0.7120	15	14	18
	23	$(2, 23, 38, 23, 8, 15)$	1	0.7118	16	15	19
	24	$(2, 24, 40, 24, 8, 16)$	1	0.7120	17	16	20
	25	$(2, 25, 42, 25, 8, 17)$	1	0.7125	18	17	21

A.6 The proof of Proposition 11

Proof (Proposition 11). We consider the rank of $\widehat{\boldsymbol{B}}^{\top}$. Let

$$\widehat{\boldsymbol{B}}_1 = \left[\boldsymbol{E}_1^{\top} \ \boldsymbol{E}_2^{\top} \ \cdots \ \boldsymbol{E}_N^{\top} \right], \quad \widehat{\boldsymbol{B}}_2 = \mathbf{Diag}\left(\boldsymbol{Z}^{\top}, N\right), \quad \widehat{\boldsymbol{B}}^{\top} = \begin{bmatrix} \widehat{\boldsymbol{B}}_1 \\ \widehat{\boldsymbol{B}}_2 \end{bmatrix}.$$

Since $k+r \leq t$, from Proposition 5, we have $\mathsf{rank}\left(\boldsymbol{E}_i^{\top}\right) = r$ and $\mathsf{rank}\left(\boldsymbol{Z}\right) = k+r$. Thus,

$$\mathsf{rank}\left(\widehat{\boldsymbol{B}}\right) = \mathsf{rank}\left(\widehat{\boldsymbol{B}}^{\top}\right) \leq \mathsf{rank}\left(\widehat{\boldsymbol{B}}_1\right) + \mathsf{rank}\left(\widehat{\boldsymbol{B}}_2\right) = r + N(k+r).$$

□

A.7 The proof of Theorem 6

Proof (Theorem 6). We consider $\widehat{\boldsymbol{B}}^\top$ and estimate $\Pr\left[\mathsf{rank}\left(\widehat{\boldsymbol{B}}^\top\right) < r + N(k+r)\right]$. Let $\boldsymbol{\varepsilon} \in \mathbb{F}_{q^m}^r$ be a basis of support for N errors $\boldsymbol{e}_i$'s. Let $\mathbf{CM}(\boldsymbol{e}_i) \in \mathbb{F}_q^{r\times n}$ of rank r be the coefficient matrix of $\boldsymbol{e}_i$ under $\boldsymbol{\varepsilon}$ such that $\boldsymbol{e}_i = \boldsymbol{\varepsilon}\mathbf{CM}(\boldsymbol{e}_i)$. Let $\boldsymbol{\gamma} \in \mathbb{F}_{q^m}^t$ be a basis of $\mathrm{Supp}(\boldsymbol{g})$. Let $\mathbf{CM}(\boldsymbol{g}) \in \mathbb{F}_q^{t\times n}$ of rank t be the coefficient matrix of $\boldsymbol{g}$ under $\boldsymbol{\gamma}$ such that $\boldsymbol{g} = \boldsymbol{\gamma}\mathbf{CM}(\boldsymbol{g})$. Let $\boldsymbol{N}_1 = \mathbf{Moore}(\boldsymbol{\varepsilon}, r) \in \mathbb{F}_{q^m}^{(r+1)\times r}$ and $\boldsymbol{N}_2 = \mathbf{Moore}(\boldsymbol{\gamma}, k+r-1) \in \mathbb{F}_{q^m}^{(k+r)\times t}$. By Definition 6 and Proposition 3,

$$\begin{aligned}\widehat{\boldsymbol{B}}^\top &= \left[\begin{array}{c|c|c|c} \boldsymbol{E}_1^\top & \boldsymbol{E}_2^\top & \cdots & \boldsymbol{E}_N^\top \end{array}\right] = \left[\begin{array}{c|c|c|c} \boldsymbol{N}_1\cdot\mathbf{CM}(\boldsymbol{e}_1) & \boldsymbol{N}_1\cdot\mathbf{CM}(\boldsymbol{e}_2) & \cdots & \boldsymbol{N}_1\cdot\mathbf{CM}(\boldsymbol{e}_N) \end{array}\right] \\ &= \underbrace{\left[\begin{array}{c|c} \boldsymbol{N}_1 & \boldsymbol{0} \\ \hline \boldsymbol{0} & \mathbf{Diag}(\boldsymbol{N}_2, N) \end{array}\right]}_{:=\boldsymbol{L}\in\mathbb{F}_{q^m}^{[r+N(k+r)+1]\times(r+Nt)}} \underbrace{\left[\begin{array}{c|c|c|c} \mathbf{CM}(\boldsymbol{e}_1) & \mathbf{CM}(\boldsymbol{e}_2) & \cdots & \mathbf{CM}(\boldsymbol{e}_N) \\ \hline \multicolumn{4}{c}{\mathbf{Diag}(\mathbf{CM}(\boldsymbol{g}), N)} \end{array}\right]}_{:=\boldsymbol{M}\in\mathbb{F}_q^{(r+Nt)\times Nn}}. \end{aligned} \tag{20}$$

By Corollary 1, we have $\mathsf{rank}\,(\boldsymbol{N}_1) = r$, and as $k+r-1 < t$, $\mathsf{rank}\,(\mathbf{Diag}(\boldsymbol{N}_2, N)) = N(k+r)$. Thus, $\mathsf{rank}(\boldsymbol{L}) = \mathsf{rank}\,(\boldsymbol{N}_1)+\mathsf{rank}\,(\mathbf{Diag}(\boldsymbol{N}_2, N)) = r+N(k+r) \le Nn$.

As $\mathsf{rank}\left(\widehat{\boldsymbol{B}}^\top\right) \le \min\{\mathsf{rank}(\boldsymbol{L}), \mathsf{rank}(\boldsymbol{M})\}$ and $\boldsymbol{M} \in \mathbb{F}_q^{(r+Nt)\times Nn}$ is defined in the basis field $\mathbb{F}_q$, Following the Heuristic Argument presented in Theorem 3: if $\mathsf{rank}(\boldsymbol{M}) \ge r + N(k+r)$, then $\mathsf{rank}\left(\widehat{\boldsymbol{B}}^\top\right) = r + N(k+r)$. Hence, we have

$$\mathsf{rank}\left(\widehat{\boldsymbol{B}}^\top\right) < r + N(k+r) \implies \mathsf{rank}(\boldsymbol{M}) < r + N(k+r). \tag{21}$$

Moreover, since

$$\begin{aligned}\mathsf{rank}(\boldsymbol{M}) &= \mathsf{rank}\left(\left[\begin{array}{c|c|c|c} \mathbf{CM}(\boldsymbol{e}_1) & \mathbf{CM}(\boldsymbol{e}_2) & \cdots & \mathbf{CM}(\boldsymbol{e}_N) \\ \hline \multicolumn{4}{c}{\mathbf{Diag}(\mathbf{CM}(\boldsymbol{g}), N)} \end{array}\right]\right) \\ &= \mathsf{dim}\Big(\big\langle\big[\mathbf{CM}(\boldsymbol{e}_1)\ \mathbf{CM}(\boldsymbol{e}_2)\ \cdots\ \mathbf{CM}(\boldsymbol{e}_N)\big]\big\rangle + \langle\mathbf{CM}(\boldsymbol{g})\rangle\Big) \\ &= r + Nt - \Delta,\end{aligned}$$

where $\Delta = \mathsf{dim}\Big(\big\langle\big[\mathbf{CM}(\boldsymbol{e}_1)\ \mathbf{CM}(\boldsymbol{e}_2)\ \cdots\ \mathbf{CM}(\boldsymbol{e}_N)\big]\big\rangle \cap \langle\mathbf{CM}(\boldsymbol{g})\rangle\Big)$, we have

$$\mathsf{rank}(\boldsymbol{M}) < r + N(k+r) \iff \Delta > N(t-k-r). \tag{22}$$

Let $a = N(t-k-r)+1$. Finally, combining Equations (21) and (22), we have:

– When $a \le \min\{Nt, r\}$,

$$\begin{aligned}&\Pr[\mathsf{rank}\left(\widehat{\boldsymbol{B}}^\top\right) < r + N(k+r)] \le \Pr[\mathsf{rank}(\boldsymbol{M}) < r + N(k+r)] \\ &= \Pr\Big[\Delta > N(t-k-r)\Big] = \sum_{i=a}^{\min\{Nt,r\}} \Pr\Big[\Delta = i\Big] \approx \Pr\Big[\Delta = a\Big] \\ &\le \gamma_q \cdot q^{a(Nt+r-a-Nn)}.\end{aligned} \tag{23}$$

The first inequality "$\leq$" due to "$\Longrightarrow$" of Equation (21). The second equation "$=$" due to "$\Longleftrightarrow$" of Equation (22). The approximation "$\approx$" is natural. By the probability in Lemma 2 (see Appendix A), as the dimension i increases, the probability decreases significantly. The last inequality also follows Lemma 2.

– When $a > \min\{Nt, r\}$, it is impossible to obtain an intersection space of dimension $\geq a$. Hence,

$$\Pr[\mathsf{rank}\left(\widehat{\boldsymbol{B}}^\top\right) < r + N(k+r)] \leq \Pr[\mathsf{rank}(\boldsymbol{M}) < r + N(k+r)] = \Pr\Big[\Delta \geq a\Big] = 0. \tag{24}$$

This means that the DFR is null and the decoding algorithm is deterministic. □

A.8 The proof of Theorem 8

Proof (Theorem 8). We adapt the proof of Theorem 6, consider $\widehat{\boldsymbol{B}}^\top$, and estimate $\Pr\left[\mathsf{rank}\left(\widehat{\boldsymbol{B}}^\top\right) < r + N(k+r)\right]$. Let $\boldsymbol{\varepsilon} \in \mathbb{F}_{q^m}^r$ be a basis of support for N errors $\boldsymbol{e}_i$'s. Let $\mathbf{CM}(\boldsymbol{e}_i) \in \mathbb{F}_q^{r\times n}$ of rank r be the coefficient matrix of $\boldsymbol{e}_i$ under $\boldsymbol{\varepsilon}$ such that $\boldsymbol{e}_i = \boldsymbol{\varepsilon}\mathbf{CM}(\boldsymbol{e}_i)$. Differently, $\boldsymbol{g}$ has weight n. Let $\boldsymbol{\gamma} \in \mathbb{F}_{q^m}^n$ be a basis of $\mathrm{Supp}(\boldsymbol{g})$. Let $\mathbf{CM}(\boldsymbol{g}) \in \mathbb{F}_q^{n\times n}$ of rank n be the coefficient matrix of $\boldsymbol{g}$ under $\boldsymbol{\gamma}$ such that $\boldsymbol{g} = \boldsymbol{\gamma}\mathbf{CM}(\boldsymbol{g})$. Let $\boldsymbol{N}_1 = \mathbf{Moore}(\boldsymbol{\varepsilon}, r) \in \mathbb{F}_{q^m}^{(r+1)\times r}$ and $\boldsymbol{N}_2 = \mathbf{Moore}(\boldsymbol{\gamma}, k+r-1) \in \mathbb{F}_{q^m}^{(k+r)\times n}$. By Equation (20), let $t = n$, we have

$$\begin{aligned}
\widehat{\boldsymbol{B}}^\top &= \left[\begin{array}{c|c|c|c} \boldsymbol{E}_1^\top & \boldsymbol{E}_2^\top & \cdots & \boldsymbol{E}_N^\top \\ \hline \multicolumn{4}{c}{\mathbf{Diag}(\boldsymbol{Z}^\top, N)} \end{array}\right] = \left[\begin{array}{c|c|c|c} \boldsymbol{N}_1\cdot\mathbf{CM}(\boldsymbol{e}_1) & \boldsymbol{N}_1\cdot\mathbf{CM}(\boldsymbol{e}_2) & \cdots & \boldsymbol{N}_1\cdot\mathbf{CM}(\boldsymbol{e}_N) \\ \hline \multicolumn{4}{c}{\mathbf{Diag}(\boldsymbol{N}_2\cdot\mathbf{CM}(\boldsymbol{g}), N)} \end{array}\right] \\
&= \underbrace{\left[\begin{array}{c|c} \boldsymbol{N}_1 & \boldsymbol{0} \\ \hline \boldsymbol{0} & \mathbf{Diag}(\boldsymbol{N}_2, N) \end{array}\right]}_{:=\boldsymbol{L}\in\mathbb{F}_{q^m}^{[r+N(k+r)+1]\times(r+Nn)}} \underbrace{\left[\begin{array}{c|c|c|c} \mathbf{CM}(\boldsymbol{e}_1) & \mathbf{CM}(\boldsymbol{e}_2) & \cdots & \mathbf{CM}(\boldsymbol{e}_N) \\ \hline \multicolumn{4}{c}{\mathbf{Diag}(\mathbf{CM}(\boldsymbol{g}), N)} \end{array}\right]}_{:=\boldsymbol{M}\in\mathbb{F}_q^{(r+Nn)\times Nn}}.
\end{aligned}$$

By Corollary 1, we have $\mathsf{rank}\,(\boldsymbol{N}_1) = r$, and as $k+r-1 < n$, we have $\mathsf{rank}\,(\mathbf{Diag}(\boldsymbol{N}_2, N)) = N(k+r)$. Thus,

$$\mathsf{rank}(\boldsymbol{L}) = \mathsf{rank}\,(\boldsymbol{N}_1) + \mathsf{rank}\,(\mathbf{Diag}(\boldsymbol{N}_2, N)) = r + N(k+r) \leq Nn.$$

As $\boldsymbol{M} \in \mathbb{F}_q^{(r+Nn)\times Nn}$ and $\mathsf{rank}(\mathbf{CM}(\boldsymbol{g})) = n$, we have

$$\left\langle \left[\mathbf{CM}(\boldsymbol{e}_1)\ \mathbf{CM}(\boldsymbol{e}_2)\cdots\mathbf{CM}(\boldsymbol{e}_N)\right]\right\rangle \subset \left\langle \mathbf{Diag}(\mathbf{CM}(\boldsymbol{g}), N)\right\rangle.$$

Hence,

$$\begin{aligned}
\mathsf{rank}(\boldsymbol{M}) &= \mathsf{dim}\Big(\left\langle \left[\mathbf{CM}(\boldsymbol{e}_1)\ \mathbf{CM}(\boldsymbol{e}_2)\cdots\mathbf{CM}(\boldsymbol{e}_N)\right]\right\rangle + \left\langle \mathbf{Diag}(\mathbf{CM}(\boldsymbol{g}), N)\right\rangle\Big) \\
&= r + Nn - r = Nn.
\end{aligned}$$

If $r \leq \left\lfloor \frac{N(n-k)}{N+1} \right\rfloor$, then $Nn \geq N(k+r)+r$, further $\Pr\left[\mathsf{rank}(\boldsymbol{M}) < N(k+r)+r\right] = 0$. Following the Heuristic Argument presented in Theorem 3, we have

$$\mathsf{rank}\left(\widehat{\boldsymbol{B}}^{\top}\right) < N(k+r)+r \implies \mathsf{rank}(\boldsymbol{M}) < N(k+r)+r.$$

$$\Pr\left[\mathsf{rank}\left(\widehat{\boldsymbol{B}}^{\top}\right) < N(k+r)+r\right] \leq \Pr\left[\mathsf{rank}(\boldsymbol{M}) < N(k+r)+r\right] = 0.$$

□

B Decoding Errors of Weight w ($w < r$) for the EG Codes

Assume that the value r is the maximum q-degree solved in the LR problem. In this case, the EG codes can decode up to weight r. Let $w < r$. Now, we analyze how to decode when the error of weight w occurs, and show the relation between the solutions of the decoding problem and LR problem.

Theorem 9. *Assume that the* LR *problem can be solved up to q-degree r. Let $f(x)$ be a fixed q-polynomial of q-degree $\leq k-1$. Let $(\boldsymbol{g}, \boldsymbol{y})$ be the instance of the* $\mathsf{DecEGCode}(\boldsymbol{g}, \boldsymbol{y})$ *problem with $\boldsymbol{y} = f(\boldsymbol{g}) + \boldsymbol{e}$ and $\|\boldsymbol{e}\|_{\mathrm{R}} = w$. The solutions of the* $\mathsf{LR}(\boldsymbol{g}, \boldsymbol{y})$ *problem are included in a set $\mathcal{V}_w^r$:*

$$\mathcal{V}_w^r = \left\{ \left(v(x),\ -v(x) \circ f(x)\right)\ :\ v(x) = z(x) \circ \mathcal{A}_{\boldsymbol{e}}(x),\ z(x) \in \mathcal{L}_{\leq r-w}[x] \right\}.$$

Then the set $\mathcal{V}_w^r$ is isomorphic to a linear space over $\mathbb{F}_{q^m}$ of dimension $r-w+1$.

Proof. It is clear that any $\left(v(x),\ -v(x) \circ f(x)\right) \in \mathcal{V}_w^r$ satisfies

$$-v\left(\boldsymbol{y}\right) = -v(f\left(\boldsymbol{g}\right) + \boldsymbol{e}) = -v(f\left(\boldsymbol{g}\right)) + z \circ \mathcal{A}_{\boldsymbol{e}}(\boldsymbol{e}) = -v(f\left(\boldsymbol{g}\right)) = -\left(v \circ f\right)\left(\boldsymbol{g}\right).$$

When $z(x) \neq 0$, $v(x) = z(x) \circ \mathcal{A}_{\boldsymbol{e}}(x) \neq 0$, $\mathsf{deg}_q v(x) \leq r$, and $\mathsf{deg}_q(-v \circ f)(x) \leq k+r-1$, such $\left(v(x),\ -v(x) \circ f(x)\right)$ exactly is the solution to the $\mathsf{LR}(\boldsymbol{g}, \boldsymbol{y})$ problem. We denote a map by

$$\begin{aligned} \varphi :\quad \mathcal{L}_{\leq r}[x] &\ \to\ \mathbb{F}_{q^m}^{r+1} \\ b(x) = \sum_{i=0}^{r} b_i x^{q^i} &\ \mapsto\ (b_0, b_1, \ldots, b_r). \end{aligned}$$

By some abuse of notation, we also denote by φ its natural extension to $\mathcal{V}_w^r$:

$$\begin{aligned} \varphi :\quad \mathcal{V}_w^r &\ \to\ \mathbb{F}_{q^m}^{k+2r+1} \\ \left(v(x),\ -v(x) \circ f(x)\right) &\ \mapsto\ \left(\varphi(v(x)), \varphi\left(-v(x) \circ f(x)\right)\right) \end{aligned}$$

Let $v(x) = z(x) \circ \mathcal{A}_{\boldsymbol{e}}(x) = \sum_{i=0}^{r} v_i x^{q^i}$ and $u(x) = -v(x) \circ f(x) = \sum_{i=0}^{k+r-1} u_i x^{q^i}$. Let $f(x) = \sum_{i=0}^{k-1} f_i x^{q^i}$, and $\mathcal{A}_{\boldsymbol{e}}(x) = \sum_{i=0}^{r} a_i x^{q^i}$, $(a_r = 1)$. Let $z(x) = \sum_{i=0}^{r-w} z_i x^{q^i}$. One easily checks equations:

$$(u_0, u_1, \ldots, u_{k+r-1})^\top = -\mathbf{PM}(f(x), r)\,(v_0, v_1, \ldots, v_r)^\top \tag{25}$$

$$(v_0, v_1, \ldots, v_r)^\top = \mathbf{PM}\left(\mathcal{A}_{\boldsymbol{e}}(x), r-w\right)(z_0, z_1, \ldots, z_{r-w})^\top. \tag{26}$$

Therefore, we have

$$\begin{aligned}
&\Big(\varphi\big(v(x),\ -v(x) \circ f(x)\big)\Big)^\top = \Big(\varphi(v(x)), \varphi\left(-v(x) \circ f(x)\right)\Big)^\top \\
&= (v_0, v_1, \ldots, v_r, u_0, u_1, \ldots, u_{k+r-1})^\top \\
&= \left[\frac{\boldsymbol{I}_{(r+1)\times(r+1)}}{-\mathbf{PM}(f(x), r)}\right] \mathbf{PM}\left(\mathcal{A}_{\boldsymbol{e}}(x), r-w\right)(z_0, z_1, \ldots, z_{r-w})^\top.
\end{aligned} \tag{27}$$

In Equation (27), we denote the first matrix by $\boldsymbol{F}$ that is fixed and is column full-rank (rank $r+1$), and $\boldsymbol{W} := \mathbf{PM}\left(\mathcal{A}_{\boldsymbol{e}}(x), r-w\right)$ is fixed and is of rank $r-w+1$ due to $a_r = 1$. Thus, $\boldsymbol{FW}$ is a column full-rank $(k+2r+1) \times (r-w+1)$ matrix over $\mathbb{F}_{q^m}$. This shows that under the map φ, the set $\mathcal{V}_w^r$ is isomorphic to a linear space $\langle(\boldsymbol{FW})^\top\rangle$ over $\mathbb{F}_{q^m}$ of dimension $r-w+1$. □

By Theorem 9, we can get that in $\mathcal{V}_w^r$, any no-zero pair $(v(x), u(x))$ satisfies $-v(x) \backslash u(x) = f(x)$. Thus, once the set $\mathcal{V}_w^r$ is determined, we can obtain the solution to the $\mathsf{LR}(\boldsymbol{g}, \boldsymbol{y})$ problem.

Theorem 10. *Let $f(x)$ be a fixed q-polynomial of $\mathsf{deg}_q f(x) \leq k-1$. Let $\boldsymbol{y} = f(\boldsymbol{g}) + \boldsymbol{e}$ and $\|\boldsymbol{e}\|_{\mathrm{R}} = w$. Let $\mathsf{LR}(\boldsymbol{g}, \boldsymbol{y})$ be the instance of the LR problem. If the dimension of the right kernel $\mathsf{kernel}(\boldsymbol{A})$ of matrix $\boldsymbol{A}$ defined in Equation (5) is $r-w+1$, then $\mathsf{kernel}(\boldsymbol{A})$ is isomorphic to $\mathcal{V}_w^r$ defined in Theorem 9.*

Proof. If $f(x)$ and $(v \circ f)(x)$ are the solution to the $\mathsf{LR}(\boldsymbol{g}, \boldsymbol{y})$ problem, then $v(x) \neq 0$, $\mathsf{deg}_q v(x) \leq r$, $\mathsf{deg}_q (v \circ f)(x) \leq k+r-1$, and

$$v\left(\boldsymbol{y}\right) = (v \circ f)\left(\boldsymbol{g}\right) \iff v\left(\boldsymbol{y}\right) = v(f\left(\boldsymbol{g}\right)) \iff v\left(\boldsymbol{y} - f(\boldsymbol{g})\right) = \mathbf{0}_n \iff v\left(\boldsymbol{e}\right) = \mathbf{0}_n.$$

Then $v(x)$ must have the form of $v(x) = z(x) \circ \mathcal{A}_{\boldsymbol{e}}(x)$, where $z(x) \in \mathcal{L}[x]$ is random, $\mathsf{deg}_q z(x) \leq r-w$, and $\mathcal{A}_{\boldsymbol{e}}(x)$ is the annihilator polynomial of $\boldsymbol{e}$ (Definition 5, $\mathcal{A}_{\boldsymbol{e}}(x)$ is unique and monic, $\mathsf{deg}_q \mathcal{A}_{\boldsymbol{e}}(x) = w$, and $\mathcal{A}_{\boldsymbol{e}}(\boldsymbol{e}) = \mathbf{0}_n$).

Let $f(x) = \sum_{i=0}^{k-1} f_i x^{q^i}$ and $v(x) = z(x) \circ \mathcal{A}_{\boldsymbol{e}}(x) = \sum_{i=0}^{r} v_i x^{q^i}$. Let $\mathcal{A}_{\boldsymbol{e}}(x) = \sum_{i=0}^{r} a_i x^{q^i}$ $(a_r = 1)$ and $z(x) = \sum_{i=0}^{r-w} z_i x^{q^i}$. From $v\left(\boldsymbol{y} - f(\boldsymbol{g})\right) = \mathbf{0}_n$, combing Equation (6), we have

$$\begin{bmatrix}
y_1 - f(g_1) & (y_1 - f(g_1))^q & \cdots & (y_1 - f(g_1))^{q^r} \\
y_2 - f(g_2) & (y_2 - f(g_2))^q & \cdots & (y_2 - f(g_2))^{q^r} \\
\vdots & \vdots & \ddots & \vdots \\
y_n - f(g_n) & (y_n - f(g_n))^q & \cdots & (y_n - f(g_n))^{q^r}
\end{bmatrix}
\begin{bmatrix} v_0 \\ v_1 \\ \vdots \\ v_r \end{bmatrix} = \mathbf{0}_n$$

$$\Longleftrightarrow$$

$$\boldsymbol{A}\left[\frac{\boldsymbol{I}_{(r+1)\times(r+1)}}{-\mathbf{PM}(f(x),r)}\right](v_0,v_1,\ldots,v_r)^\top=\mathbf{0}_n$$

$$\Longleftrightarrow$$

$$\boldsymbol{A}\left[\frac{\boldsymbol{I}_{(r+1)\times(r+1)}}{-\mathbf{PM}(f(x),r)}\right]\mathbf{PM}\left(\mathcal{A}_{\boldsymbol{e}}(x),r-w\right)(z_0,z_1,\ldots,z_{r-w})^\top=\mathbf{0}_n \tag{28}$$

In Equation (28), we denote the second matrix by $\boldsymbol{F}$ that is fixed and is a column full-rank (rank $r+1$), and $\boldsymbol{W}:=\mathbf{PM}\left(\mathcal{A}_{\boldsymbol{e}}(x),r-w\right)$ is fixed and is of rank $r-w+1$ due to $a_r=1$. Thus, $\boldsymbol{FW}$ is a column full-rank $(k+2r+1)\times(r-w+1)$ matrix over $\mathbb{F}_{q^m}$. This shows that $\mathsf{dim}\left(\mathsf{kernel}(\boldsymbol{A})\right)=r-w+1$ and $\left\langle(\boldsymbol{FW})^\top\right\rangle=\mathsf{kernel}(\boldsymbol{A})$.

Consider the linear system

$$\boldsymbol{A}\boldsymbol{x}^\top=\mathbf{0}_n. \tag{29}$$

Combing Equation (27), we clearly have that if $\mathsf{dim}\left(\mathsf{kernel}(\boldsymbol{A})\right)=r-w+1$, then $\mathsf{kernel}(\boldsymbol{A})$ is isomorphic to $\mathcal{V}_w^r$. □

By Theorem 9 and Theorem 10, if $\mathsf{dim}\left(\mathsf{kernel}(\boldsymbol{A})\right)=r-w+1$, for any non-zero solution $\boldsymbol{x}=(v_0,v_1,\ldots,v_r,u_0,u_1,\ldots,u_{k+r-1})$ to the system (29), let $v(x)=\sum_{i=0}^{r}v_ix^{q^i}$ and $u(x)=\sum_{i=0}^{k+r-1}u_ix^{q^i}$, then $f(x)=-v(x)\backslash u(x)$. Such $(v(x),f(x))$ is solution to the $\mathsf{NLR}(\boldsymbol{g},\boldsymbol{y})$ problem. This means that, given the EG codes decoding weight up to r, any error of weight $w\leq r$ can decoded by solving the $\mathsf{LR}(\boldsymbol{g},\boldsymbol{y})$ problem if $\mathsf{dim}\left(\mathsf{kernel}(\boldsymbol{A})\right)=r-w+1$.

Proposition 12. *Assume that the LR problem can be solved up to q-degree r. For any error $\boldsymbol{e}\in\mathbb{F}_{q^m}^n$ and $\|\boldsymbol{e}\|_{\mathrm{R}}=w$, if $k+r\leq t\leq\min\{n,m\}$ and $k+2r\leq n$, then $\mathsf{rank}(\boldsymbol{B})\leq k+r+w$.*

Proof. Recall that when the LR problem can be solved up to q-degree r, following Proposition 9, the matrix $\boldsymbol{A}$ is equivalent to $\boldsymbol{B}$ in Equation (9).

$$\boldsymbol{B}=\left[\mathbf{Moore}(\boldsymbol{e},r)^\top\quad\mathbf{Moore}(\boldsymbol{g},k+r-1)^\top\right].$$

We consider the rank of $\boldsymbol{B}^\top$. Let

$$\boldsymbol{B}_1=\mathbf{Moore}(\boldsymbol{e},r),\quad\boldsymbol{B}_2=\mathbf{Moore}(\boldsymbol{g},k+r-1),\quad\boldsymbol{B}^\top=\begin{bmatrix}\boldsymbol{B}_1\\\boldsymbol{B}_2\end{bmatrix}.$$

Since $k+r\leq t$, from Proposition 5, we have $\mathsf{rank}(\boldsymbol{B}_1)=w$ and $\mathsf{rank}(\boldsymbol{B}_2)=k+r$. Thus,

$$\mathsf{rank}(\boldsymbol{B})=\mathsf{rank}\left(\boldsymbol{B}^\top\right)=\mathsf{rank}\left(\begin{bmatrix}\boldsymbol{B}_1\\\boldsymbol{B}_2\end{bmatrix}\right)\leq\mathsf{rank}(\boldsymbol{B}_1)+\mathsf{rank}(\boldsymbol{B}_2)\leq k+r+w.$$

□

From Theorem 10, the DFR of decoding weight $w < r$ depends on the probability that $\mathsf{dim}\,(\mathsf{kernel}(\boldsymbol{A})) \neq r - w + 1$. Proposition 12 shows $\mathsf{rank}(\boldsymbol{A}) = \mathsf{rank}(\boldsymbol{B}) \leq k + r + w$, since $\mathsf{rank}(\boldsymbol{A}) + \mathsf{dim}(\mathsf{kernel}(\boldsymbol{A})) = k + 2r + 1$, we have $\Pr[\mathsf{dim}(\mathsf{kernel}(\boldsymbol{A})] \geq r-w+1$. Thus, the DFR depends on probability of $\mathsf{dim}(\mathsf{kernel}(\boldsymbol{A})) > r-w+1$, which is equivalent to $\mathsf{rank}(\boldsymbol{A}) < k+r+w$. We next analyze the probability of $\mathsf{rank}(\boldsymbol{A}) < k + r + w$ (or $\mathsf{rank}(\boldsymbol{B}) < k + r + w$) in Theorem 11.

Theorem 11. *For any uniform error $\boldsymbol{e} \in \mathbb{F}_{q^m}^n$ and $\|\boldsymbol{e}\|_{\mathrm{R}} = w$, if $k + r \leq t \leq \min\{n, m\}$ and $k + 2r \leq n$, then*

$$\Pr[\mathsf{rank}(\boldsymbol{B}) < w + k + r] \leq \gamma_q \cdot q^{a(t+w-a-n)}, \tag{30}$$

where $a = t - k - r + 1$.

Proof. We consider $\boldsymbol{B}^\top$ and estimate $\Pr[\mathsf{rank}(\boldsymbol{B}^\top) < w + k + r]$. Let $\boldsymbol{\varepsilon} \in \mathbb{F}_{q^m}^w$ be a basis of $\mathrm{Supp}(\boldsymbol{e})$. Let $\mathbf{CM}(\boldsymbol{e}) \in \mathbb{F}_q^{w\times n}$ of rank w be the coefficient matrix of $\boldsymbol{e}$ under $\boldsymbol{\varepsilon}$ such that $\boldsymbol{e} = \boldsymbol{\varepsilon}\mathbf{CM}(\boldsymbol{e})$. Let $\boldsymbol{\gamma} \in \mathbb{F}_{q^m}^t$ be a basis of $\mathrm{Supp}(\boldsymbol{g})$. Let $\mathbf{CM}(\boldsymbol{g}) \in \mathbb{F}_q^{t\times n}$ of rank t be the coefficient matrix of $\boldsymbol{g}$ under $\boldsymbol{\gamma}$ such that $\boldsymbol{g} = \boldsymbol{\gamma}\mathbf{CM}(\boldsymbol{g})$. Following Definition 6 and Proposition 3,

$$\begin{aligned}\boldsymbol{B}^\top &= \begin{bmatrix}\boldsymbol{B}_1\\ \boldsymbol{B}_2\end{bmatrix} = \left[\frac{\mathbf{Moore}(\boldsymbol{e}, r)}{\mathbf{Moore}(\boldsymbol{g}, k + r - 1)}\right] = \left[\frac{\mathbf{Moore}(\boldsymbol{\varepsilon}, r)\cdot \mathbf{CM}(\boldsymbol{e})}{\mathbf{Moore}(\boldsymbol{\gamma}, k + r - 1)\cdot \mathbf{CM}(\boldsymbol{g})}\right]\\ &= \underbrace{\left[\begin{array}{c|c}\mathbf{Moore}(\boldsymbol{\varepsilon}, r) & \mathbf{0}_{(r+1)\times t}\\ \hline \mathbf{0}_{(k+r)\times r} & \mathbf{Moore}(\boldsymbol{\gamma}, k + r - 1)\end{array}\right]}_{:=\boldsymbol{L}\in\mathbb{F}_{q^m}^{(k+2r+1)\times(t+w)}} \underbrace{\left[\frac{\mathbf{CM}(\boldsymbol{e})}{\mathbf{CM}(\boldsymbol{g})}\right]}_{:=\boldsymbol{L}\in\mathbb{F}_q^{(t+w)\times n}}.\end{aligned} \tag{31}$$

By Corollary 1, as $w < r$, we have $\mathsf{rank}\,(\mathbf{Moore}(\boldsymbol{\varepsilon}, r)) = w$, and as $k+r-1 < t$, we have $\mathsf{rank}\,(\mathbf{Moore}(\boldsymbol{\gamma}, k + r - 1)) = k + r$. Thus,

$$\mathsf{rank}(\boldsymbol{L}) = \mathsf{rank}\,(\mathbf{Moore}(\boldsymbol{\varepsilon}, r)) + \mathsf{rank}\,(\mathbf{Moore}(\boldsymbol{\gamma}, k + r - 1)) = w + k + r.$$

As $\mathsf{rank}\left(\boldsymbol{B}^\top\right) \leq \min\{\mathsf{rank}(\boldsymbol{L}), \mathsf{rank}(\boldsymbol{M})\}$ and $\boldsymbol{M} \in \mathbb{F}_q^{(t+w)\times n}$ is defined in the basis field $\mathbb{F}_q$, we have a similar heuristic argument: if $\mathsf{rank}(\boldsymbol{M}) \geq w+k+r$, then $\mathsf{rank}\left(\boldsymbol{B}^\top\right) = w + k + r$. Hence, we have

$$\mathsf{rank}\left(\boldsymbol{B}^\top\right) < w + k + r \implies \mathsf{rank}(\boldsymbol{M}) < w + k + r. \tag{32}$$

Moreover,

$$\begin{aligned}\mathsf{rank}(\boldsymbol{M}) &= \mathsf{rank}\begin{bmatrix}\mathbf{CM}(\boldsymbol{e})\\ \mathbf{CM}(\boldsymbol{g})\end{bmatrix} = \mathsf{dim}\Big(\langle\mathbf{CM}(\boldsymbol{e})\rangle + \langle\mathbf{CM}(\boldsymbol{g})\rangle\Big)\\ &= \mathsf{dim}\Big(\langle\mathbf{CM}(\boldsymbol{e})\rangle\Big) + \mathsf{dim}\Big(\langle\mathbf{CM}(\boldsymbol{g})\rangle\Big) - \mathsf{dim}\Big(\langle\mathbf{CM}(\boldsymbol{e})\rangle \cap \langle\mathbf{CM}(\boldsymbol{g})\rangle\Big)\\ &= w + t - \mathsf{dim}\Big(\langle\mathbf{CM}(\boldsymbol{e})\rangle \cap \langle\mathbf{CM}(\boldsymbol{g})\rangle\Big).\end{aligned}$$

Let $\Delta = \mathsf{dim}\Big(\langle\mathbf{CM}(\boldsymbol{e})\rangle \cap \langle\mathbf{CM}(\boldsymbol{g})\rangle\Big)$, we have

$$\mathsf{rank}(\boldsymbol{M}) < w + k + r \iff \Delta > t - k - r. \tag{33}$$

Let $a = t - k - r + 1$. Finally, combining Equations (32) and (33), we have:

– When $a \leq \min\{t, w\}$,

$$\Pr[\mathsf{rank}\left(\boldsymbol{B}^{\top}\right) < w+k+r] \leq \Pr[\mathsf{rank}(\boldsymbol{M}) < w+k+r] = \Pr\Big[\Delta > t-k-r\Big]$$
$$= \sum_{i=a}^{\min\{t,w\}} \Pr\Big[\Delta = i\Big] \approx \Pr\Big[\Delta = a\Big] \leq \gamma_q \cdot q^{a(t+w-a-n)}. \tag{34}$$

The first inequality "$\leq$" due to "$\Longrightarrow$" of Equation (32). The second equation "$=$" due to "$\Longleftrightarrow$" of Equation (33). The approximation "$\approx$" is natural. By the probability in Lemma 2 (see Appendix A), as the dimension i increases, the probability decreases significantly. The last inequality also follows Lemma 2.

– When $a > \min\{t, w\}$, it is impossible to obtain an intersection space of dimension $\geq a$. Hence,

$$\Pr[\mathsf{rank}\left(\boldsymbol{B}^{\top}\right) < w+k+r] \leq \Pr[\mathsf{rank}(\boldsymbol{M}) < w+k+r] = \Pr\Big[\Delta \geq a\Big] = 0. \tag{35}$$

This means that DFR is null and the decoding algorithm is deterministic.

□

Theorem 11 shows that the DFR quickly decreases in w and is upper bounded by $\gamma_q \cdot q^{a(t+r-a-n)}$. The probability $\gamma_q \cdot q^{a(t+r-a-n)}$ is exactly one of decoding weight r. Thus, given EG codes decoding up to weight r, such EG codes can decode errors of weight $\leq r$ with the maximum DFR: $\gamma_q \cdot q^{a(t+r-a-n)}$. The decoding procedure is given in Algorithm 1.

Simulated DFR for Decoding Errors of Weight $w \leq r$. In Table 6, we chose four types of simulable code parameters, and set five groups of parameters for each type. We simulated DFR for each group by performing decoding algorithms (Algorithm 1) in 10^5 times. The theoretical DFR is estimated by Equations (34) and (35) with $\gamma_2 = 4$, and $\gamma_q = 2$ if $q > 2$. From Table 6, we can observe that the theoretical and simulated DFR are close, and the theoretical DFR is always greater than the simulated one. These show the correctness of our theoretical DFR.

C Decoding Errors of Weight w ($w < r$) for the IEG Codes

Assume that the value r is the maximum q-degree solved in the LR problem (Definition 12) obtained from the decoding IEG codes. In this case, the IEG codes can decode up to weight r. Let $w < r$. Now, we analyze how to decode when N errors $\boldsymbol{e}_i$'s share the same w-dimensional support, and show the relation between the solutions of the decoding problem and LR problem. Adapting theorems and propositions in Appendix B, we can easily obtain the following Theorems 12-14 and Proposition 13.

Table 6. Theoretical and simulated DFR of the EG codes for errors of weight $w \leq r$.

Types	No.	Parameters (q, m, n, t, k, r, w)	Theoretical DFR	Simulated DFR	d_{RGV}	d_{RS}
Decrease w $(t = m < n)$	1	$(2, 27, 41, 27, 9, 16, 16)$	(2^{-1}) 0.5000	0.2312	17	22
	2	$(2, 27, 41, 27, 9, 16, 15)$	(2^{-4}) 0.0625	0.0380	17	22
	3	$(2, 27, 41, 27, 9, 16, 14)$	(2^{-7}) 0.0078	0.0050	17	22
	4	$(2, 27, 41, 27, 9, 16, 13)$	(2^{-10}) 0.00098	0.00062	17	22
	5	$(2, 27, 41, 27, 9, 16, 12)$	(2^{-13}) 0.00012	0.00007	17	22
Decrease w $(t < n < m)$	6	$(2, 29, 26, 16, 5, 10, 10)$	(2^{-2}) 0.0250	0.1326	16	22
	7	$(2, 29, 26, 16, 5, 10, 9)$	(2^{-4}) 0.0625	0.0362	16	22
	8	$(2, 29, 26, 16, 5, 10, 8)$	(2^{-6}) 0.0156	0.0097	16	22
	9	$(2, 29, 26, 16, 5, 10, 7)$	(2^{-8}) 0.0039	0.0026	16	22
	10	$(2, 29, 26, 16, 5, 10, 6)$	(2^{-10}) 0.00098	0.00069	16	22
Decrease w (Decoding RGV Bound $k > r$)	11	$(2, 30, 37, 30, 23, 7, 7)$	(2^{1}) 2	0.7122	7	12
	12	$(2, 30, 37, 30, 23, 7, 6)$	(2^{0}) 1	0.4197	7	12
	13	$(2, 30, 37, 30, 23, 7, 5)$	(2^{-1}) 0.5000	0.2217	7	12
	14	$(2, 30, 37, 30, 23, 7, 4)$	(2^{-2}) 0.2500	0.1145	7	12
	15	$(2, 30, 37, 30, 23, 7, 3)$	(2^{-3}) 0.1250	0.0549	7	12
Decrease w (Decoding RGV Bound $k < r$)	16	$(2, 21, 34, 21, 8, 13, 13)$	(2^{1}) 2	0.7125	13	17
	17	$(2, 21, 34, 21, 8, 13, 12)$	(2^{0}) 1	0.4234	13	17
	18	$(2, 21, 34, 21, 8, 13, 11)$	(2^{-1}) 0.5000	0.2312	13	17
	19	$(2, 21, 34, 21, 8, 13, 10)$	(2^{-2}) 0.2500	0.1203	13	17
	20	$(2, 21, 34, 21, 8, 13, 9)$	(2^{-3}) 0.1250	0.0608	13	17

Theorem 12. *Assume that the* LR *problem defined in Definition 12 can be solved up to q-degree r. Let $f_i(x)$'s be fixed q-polynomials of q-degree $\leq k-1$. Let $\left(\boldsymbol{g}, \{\boldsymbol{y}_i\}_{i\in[N]}\right)$ be the instance of decoding IEG codes problem with $\boldsymbol{y}_i = f_i(\boldsymbol{g}) + \boldsymbol{e}_i$ and $\boldsymbol{e}_i$ of sharing w-dimensional support E. The solutions of the* $\mathsf{LR}\left(\boldsymbol{g}, \{\boldsymbol{y}_i\}_{i\in[N]}\right)$ *problem are included in a set $\mathcal{V}_w^{N,r}$:*

$$\mathcal{V}_w^{N,r} = \left\{ \left(v(x),\ \{-v(x) \circ f_i(x)\}_{i\in[N]}\right)\ :\ v(x) = z(x) \circ \mathcal{A}_E(x),\ z(x) \in \mathcal{L}_{\leq r-w}[x] \right\}.$$

Then the set $\mathcal{V}_w^{N,r}$ is isomorphic to a linear space over $\mathbb{F}_{q^m}$ of dimension $r-w+1$.

Theorem 13. *Let $f_i(x)$ be fixed q-polynomials of q-degree $\leq k-1$. Let $\boldsymbol{y}_i = f_i(\boldsymbol{g}) + \boldsymbol{e}_i$ and $\boldsymbol{e}_i$ of sharing w-dimensional support E. Let* $\mathsf{LR}\left(\boldsymbol{g}, \{\boldsymbol{y}_i\}_{i\in[N]}\right)$ *be the instance of the* LR *problem defined in Definition 12. If the dimension of the right kernel* $\mathsf{kernel}\left(\widehat{\boldsymbol{A}}\right)$ *of matrix $\widehat{\boldsymbol{A}}$ defined in Equation (17) is $r-w+1$, then* $\mathsf{kernel}\left(\widehat{\boldsymbol{A}}\right)$ *is isomorphic to $\mathcal{V}_w^{N,r}$ defined in Theorem 12.*

Proposition 13. *Assume that the LR problem defined in Definition 12 can be solved up to q-degree r. For any error $\boldsymbol{e}_i \in \mathbb{F}_{q^m}^n$ sharing w-dimensional support, if $k+r \leq t \leq \min\{n, m\}$ and $r+N(k+r) \leq nN$, then* $\mathsf{rank}\left(\widehat{\boldsymbol{B}}\right) \leq w+N(k+r)$.

Theorem 14. *For any N uniform errors $\boldsymbol{e}_i \in \mathbb{F}_{q^m}^n$ sharing w-dimensional support, if $k+r \leq t \leq \min\{n, m\}$ and $r+N(k+r) \leq nN$, then*

$$\Pr\left[\mathsf{rank}\left(\widehat{\boldsymbol{B}}\right) \leq w+N(k+r)\right] \leq \gamma_q \cdot q^{a(Nt+w-a-Nn)},$$

where $a = N(t-k-r)+1$.

D Public Key Encryption

Definition 16 (Public Key Encryption). *A Public Key Encryption (PKE) scheme* $\mathsf{PKE} = (\mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec})$ *consists of three polynomial-time algorithms:*

- KGen*: The key generation algorithm that takes the security parameter* λ *as input and outputs a public key* pk *and a private key* sk*. It is denoted as* $(pk, sk) \leftarrow \mathsf{KGen}(1^\lambda)$.
- Enc*: The encryption algorithm that takes* pk *and a plaintext* M *as inputs and outputs a ciphertext* C*. It is denoted as* $C \leftarrow \mathsf{Enc}(pk, M)$.
- Dec*: The decryption algorithm that takes* sk *and* C *as inputs and outputs a plaintext* M*. It is denoted as* $M \leftarrow \mathsf{Dec}(sk, C)$.

The correctness of PKE requires that for all $(pk, sk) \leftarrow \mathsf{KGen}(1^\lambda)$, any plaintext M, and any $C \leftarrow \mathsf{Enc}(pk, M)$, the equation $M = \mathsf{Dec}(sk, C)$ hold with overwhelming probability.

An IND-CPA secure PKE scheme is defined by the experiment $\mathsf{Exp}^{\mathsf{IND\text{-}CPA}}_{\mathsf{PKE},\mathcal{A}}(\lambda)$ between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}$:

1. $\mathcal{A}$ takes λ as inputs.
2. $\mathcal{C}$ computes $(pk, sk) \leftarrow \mathsf{KGen}(1^\lambda)$, gives pk to $\mathcal{A}$, and keeps sk to itself.
3. $\mathcal{A}$ outputs two equal length plaintexts M_0, M_1. $\mathcal{C}$ randomly chooses a bit $b^* \in \{0,1\}$ and returns the challenge ciphertext $C^* = \mathsf{Enc}(pk, M_{b^*})$ to $\mathcal{A}$.
4. $\mathcal{A}$ makes encryption queries for any polynomial times.
5. $\mathcal{A}$ outputs a guess $b \in \{0,1\}$. If $b = b^*$, $\mathcal{C}$ outputs 1, else outputs 0.

Definition 17 (IND-CPA Security). *A PKE scheme is Indistinguishability under Chosen Plaintext Attacks (IND-CPA) if for any* PPT *adversary* $\mathcal{A}$*, its advantage*

$$\mathsf{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{PKE},\mathcal{A}}(\lambda) = \left| \Pr\left[\mathsf{Exp}^{\mathsf{IND\text{-}CPA}}_{\mathsf{PKE},\mathcal{A}}(\lambda) = 1\right] - \frac{1}{2} \right| \leq \mathsf{negl}(\lambda).$$

E Blockwise Rank Decoding Problem

Rank-based cryptography is based on two types of well-known rank decoding assumptions which are expressed by the generator matrix and the parity-check matrix, respectively. We call them the Rank Decoding (RD) problem and the Rank Syndrome Decoding (RD) problem, respectively. Their computational versions are as follows.

Rank Decoding (RD) Problem: Let $\boldsymbol{G}$ be a generator matrix of a random $[n,k]_{q^m}$-linear code $\mathcal{C}$, $\boldsymbol{y} \in \mathbb{F}_{q^m}^n$, and $r \in \mathbb{N}$. The problem is to find $\boldsymbol{x} \in \mathbb{F}_{q^m}^k$ and $\boldsymbol{e} \in \mathcal{S}_r^n$ such that $\boldsymbol{y} = \boldsymbol{x}\boldsymbol{G} + \boldsymbol{e}$.

Rank Syndrome Decoding (RSD) Problem: Let $\boldsymbol{H}$ be a parity-check matrix of a random $[n,k]_{q^m}$-linear code $\mathcal{C}$, $\boldsymbol{s} \in \mathbb{F}_{q^m}^{n-k}$, and $r \in \mathbb{N}$. The problem is to

find $\boldsymbol{e} \in \mathcal{S}_r^n$ such that $\boldsymbol{s} = \boldsymbol{H}\boldsymbol{e}^\top$.

The hardness of two problems is equivalent given two forms. Even if the RD problem is not known to be NP-complete, there is a randomized reduction from the RD problem to an NP-complete problem [26], namely to decoding in the Hamming metric. An RD instance can also be seen as a structured version of the MinRank instance, more precisely, there exists a reduction from the RD problem to the MinRank problem [21]. The MinRank problem was proven NP-complete in [16] and is now ubiquitous in multivariate cryptography.

Let $\ell \in \mathbb{N}$. Let $\boldsymbol{n} = (n_1, n_2, \ldots, n_\ell) \in \mathbb{N}^\ell$ and $\boldsymbol{r} = (r_1, r_2, \ldots, r_\ell) \in \mathbb{N}^\ell$. Let $n = \sum_{i=1}^{\ell} n_i$ and $r = \sum_{i=1}^{\ell} r_i$.

Definition 18 (Blockwise Errors (ℓ-errors)). *Let $\boldsymbol{e}_i \in \mathbb{F}_{q^m}^{n_i}$ be a vector of weight r_i for $i \in [\ell]$. An error $\boldsymbol{e} = (\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots, \boldsymbol{e}_\ell) \in \mathbb{F}_{q^m}^{n}$ is called an ℓ-error if the supports of ℓ vectors $\boldsymbol{e}_i$'s are in direct sum.*

We denote the set of such ℓ-errors by $\mathcal{S}_{\boldsymbol{r}}^{\boldsymbol{n}}$:

$$\mathcal{S}_{\boldsymbol{r}}^{\boldsymbol{n}} = \Big\{ (\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots, \boldsymbol{e}_\ell) \in \mathbb{F}_{q^m}^{n} \ : \ \boldsymbol{e}_i \in \mathbb{F}_{q^m}^{n_i}, \mathsf{dim}(\mathrm{Supp}(\boldsymbol{e}_i)) = r_i, \text{ and } \mathsf{dim}\left(\sum_{i=1}^{\ell} \mathrm{Supp}(\boldsymbol{e}_i)\right) = \sum_{i=1}^{\ell} \mathsf{dim}\left(\mathrm{Supp}(\boldsymbol{e}_i)\right) \Big\}.$$

Since the supports of $\boldsymbol{e}_i$'s are in direct sum if and only if $\mathsf{dim}\left(\sum_{i=1}^{\ell} \mathrm{Supp}(\boldsymbol{e}_i)\right) = \sum_{i=1}^{\ell} \mathsf{dim}\left(\mathrm{Supp}(\boldsymbol{e}_i)\right)$, we have

$$\|\boldsymbol{e}\|_{\mathrm{R}} = \mathsf{dim}\left(\mathrm{Supp}(\boldsymbol{e})\right) = \mathsf{dim}\left(\sum_{i=1}^{\ell} \mathrm{Supp}(\boldsymbol{e}_i)\right) = \sum_{i=1}^{\ell} \mathsf{dim}\left(\mathrm{Supp}(\boldsymbol{e}_i)\right) = \sum_{i=1}^{\ell} r_i = r.$$

Remark 1. We note that in work [46] proposed initially ℓ-errors, the condition that the supports of $\boldsymbol{e}_i$'s are mutually disjoint is not sufficient to ensure the weight of ℓ-error $\boldsymbol{e}$ to be r. If considering the ℓ-error $\boldsymbol{e}$ with the mutually disjoint supports, the weight of ℓ-error $\boldsymbol{e}$ is less than or equal to r:

$$\|\boldsymbol{e}\|_{\mathrm{R}} = \mathsf{dim}\left(\mathrm{Supp}(\boldsymbol{e})\right) = \mathsf{dim}\left(\sum_{i=1}^{\ell} \mathrm{Supp}(\boldsymbol{e}_i)\right) \leq \sum_{i=1}^{\ell} \mathsf{dim}\left(\mathrm{Supp}(\boldsymbol{e}_i)\right) = \sum_{i=1}^{\ell} r_i = r.$$

We here consider the condition that the supports of $\boldsymbol{e}_i$'s are in direct sum, and the weight of ℓ-error $\boldsymbol{e}$ must be r. This does not negate the potential of the blockwise structure in optimizing rank-based cryptosystems. For the purpose of improving cryptosystems, the "mutually disjoint" condition is already sufficient.

Definition 19 (Blockwise RD (ℓ-RD) Problem). *Let $\boldsymbol{G}$ be a generator matrix of a random $[n,k]_{q^m}$-linear code $\mathcal{C}$ and $\boldsymbol{y} \in \mathbb{F}_{q^m}^n$. The problem is to find $\boldsymbol{x} \in \mathbb{F}_{q^m}^k$ and $\boldsymbol{e} \in \mathcal{S}_{\boldsymbol{r}}^{\boldsymbol{n}}$ such that $\boldsymbol{y} = \boldsymbol{x}\boldsymbol{G} + \boldsymbol{e}$.*

Definition 20 (Blockwise RSD (ℓ-RSD) Problem). *Let $\boldsymbol{H}$ be a parity-check matrix of a random $[n,k]_{q^m}$-linear code $\mathcal{C}$ and $\boldsymbol{s} \in \mathbb{F}_{q^m}^{n-k}$. The problem is to find $\boldsymbol{e} \in \mathcal{S}_{\boldsymbol{r}}^{\boldsymbol{n}}$ such that $\boldsymbol{s} = \boldsymbol{H}\boldsymbol{e}^\top$.*

The variants are exactly the RD and RSD problems when $\ell = 1$. We denote them by $\ell\text{-}\mathsf{RD}_{q,m,n,k}^{\boldsymbol{n},\boldsymbol{r}}$ and $\ell\text{-}\mathsf{RSD}_{q,m,n,k}^{\boldsymbol{n},\boldsymbol{r}}$, respectively. We use the decisional form of the latter.

Definition 21 (Decisional ℓ-RSD Problem). *Let $\boldsymbol{H} \in \mathbb{F}_{q^m}^{(n-k)\times n}$ be a parity-check matrix of a random $[n,k]_{q^m}$-linear code. The decisional $\ell\text{-}\mathsf{RSD}_{q,m,n,k}^{\boldsymbol{n},\boldsymbol{r}}$ problem is hard, if for any PPT adversary $\mathcal{B}$, the following advantage is negligible:*

$$\mathsf{Adv}_{\mathcal{B}}^{\ell\text{-}\mathsf{RSD}_{q,m,n,k}^{\boldsymbol{n},\boldsymbol{r}}}(\lambda) := \Big|\Pr\left[\mathcal{B}\left(\boldsymbol{H},\boldsymbol{s}\right)=1 \;:\; \boldsymbol{H} \xleftarrow{\$} \mathbb{F}_{q^m}^{(n-k)\times n}, \boldsymbol{s} \xleftarrow{\$} \mathbb{F}_{q^m}^{n-k}\right] - \Pr\left[\mathcal{B}\left(\boldsymbol{H},\boldsymbol{H}\boldsymbol{e}^\top\right)=1 \;:\; \boldsymbol{H} \xleftarrow{\$} \mathbb{F}_{q^m}^{(n-k)\times n}, \boldsymbol{e} \leftarrow \mathcal{S}_{\boldsymbol{r}}^{\boldsymbol{n}}\right]\Big|.$$

The best known attacks on the decisional version remain the direct attacks on the computational version.

F Ideal Codes

To present the ideal variants of the rank decoding problems, and improve RQC in Appendices I, in this section, we recall the definitions of ideal codes.

Let $P(X)$ be a polynomial of degree n in $\mathbb{F}_q[X]$ and $\mathcal{R} = \mathbb{F}_{q^m}[X]/\langle P(X)\rangle$. By the map $\psi : \mathbb{F}_{q^m}^n \to \mathcal{R}$, the element of $\mathbb{F}_{q^m}^n$ is viewed as one of $\mathcal{R}$ and vice versa. The polynomial associated the vector $\boldsymbol{u} = (u_0, u_1, \ldots, u_{n-1}) \in \mathbb{F}_{q^m}^n$ is defined as $\boldsymbol{u}(X) = \sum_{i=0}^{n-1} u_i X^i \in \mathcal{R}$. The vector $(1,0,\ldots,0) \in \mathbb{F}_{q^m}^n$ corresponds to the identity $\mathbf{1}$ of $\mathcal{R}$. For $\boldsymbol{u} \in \mathbb{F}_{q^m}^n$ and $\boldsymbol{v} \in \mathbb{F}_{q^m}^n$, the product $\boldsymbol{u}\cdot\boldsymbol{v}$ is defined as the vector of coefficients of $\boldsymbol{u}(X)\boldsymbol{v}(X) \mod P(X)$, i.e., $\psi^{-1}\big(\boldsymbol{u}(X)\boldsymbol{v}(X) \mod P(X)\big)$.

Definition 22 (Ideal Matrix). *The ideal matrix generated by $\boldsymbol{v} \in \mathbb{F}_{q^m}^n$ is defined as an $n \times n$ matrix:*

$$\mathcal{IM}(\boldsymbol{v}) = \begin{bmatrix} \psi^{-1}\big(\boldsymbol{v}(X)\big) \\ \psi^{-1}\big(X\boldsymbol{v}(X) \mod P(X)\big) \\ \vdots \\ \psi^{-1}\big(X^{n-1}\boldsymbol{v}(X) \mod P(X)\big) \end{bmatrix}.$$

With the ideal matrix, the product $\boldsymbol{u}\cdot\boldsymbol{v}$ is equivalent to the vector-matrix product

$$\boldsymbol{u}\cdot\boldsymbol{v} = \psi^{-1}\big(\boldsymbol{u}(X)\boldsymbol{v}(X) \mod P(X)\big) = \psi^{-1}\left(\sum_{i=0}^{n-1} u_i X^i \boldsymbol{v}(X) \mod P(X)\right)$$
$$= \psi^{-1}\left(\sum_{i=0}^{n-1} u_i \left(X^i \boldsymbol{v}(X) \mod P(X)\right)\right) = (u_0,\ldots,u_{n-1})\,\mathcal{IM}(\boldsymbol{v}) = \boldsymbol{u}\mathcal{IM}(\boldsymbol{v}).$$

Thus, $\boldsymbol{u} \cdot \boldsymbol{v} = \boldsymbol{u}\mathcal{IM}(\boldsymbol{v}) = \boldsymbol{v}\mathcal{IM}(\boldsymbol{u}) = \boldsymbol{v} \cdot \boldsymbol{u}$. It is clear that, for any $\boldsymbol{h}_1$, $\boldsymbol{h}_2$, $\boldsymbol{e}_1$, and $\boldsymbol{e}_2 \in \mathbb{F}_{q^m}^n$,

$$\begin{aligned} \boldsymbol{h}_1 \cdot \boldsymbol{e}_1 + \boldsymbol{h}_2 \cdot \boldsymbol{e}_2 = \boldsymbol{s} \iff & \begin{bmatrix} \boldsymbol{h}_1 & \boldsymbol{h}_2 \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{e}_1 \\ \boldsymbol{e}_2 \end{bmatrix} = \boldsymbol{s} \\ \iff & \begin{bmatrix} \mathcal{IM}(\boldsymbol{h}_1)^\top & \mathcal{IM}(\boldsymbol{h}_2)^\top \end{bmatrix} \begin{bmatrix} \boldsymbol{e}_1^\top \\ \boldsymbol{e}_2^\top \end{bmatrix} = \boldsymbol{s}^\top \end{aligned}$$

To reduce the size of rank-based cryptosystems, the family of ideal codes is introduced in rank-based cryptography. The ideal codes are codes with a systematic generator matrix consisting of blocks of ideal matrices.

Definition 23 (Ideal Codes [3,7]). *Let $P(X)$ be a polynomial of degree n in $\mathbb{F}_q[X]$. An $[n\ell, nt]_{q^m}$ code $\mathcal{C}$ is an (ℓ, t)-ideal code if its generator matrix under the systematic form is of the form*

$$\boldsymbol{G} = \begin{bmatrix} & \mathcal{IM}(\boldsymbol{g}_{1,1}) & \dots & \mathcal{IM}(\boldsymbol{g}_{1,\ell-t}) \\ \boldsymbol{I}_{tn} & \vdots & \ddots & \vdots \\ & \mathcal{IM}(\boldsymbol{g}_{t,1}) & \dots & \mathcal{IM}(\boldsymbol{g}_{t,\ell-t}) \end{bmatrix}$$

where $(\boldsymbol{g}_{i,j})_{i\in[t],j\in[\ell-t]}$ are vectors of $\mathbb{F}_{q^m}^n$.

It has been proven that if m and n are two different prime numbers and $P(X)$ is irreducible, then a non-zero ideal matrix is always non-singular [34]. In this case, the generator matrix of ideal codes can be always reduced to the systematic form. We only use $[\ell n, n]_{q^m}$-ideal codes with the systematic parity-check matrix

$$\boldsymbol{H} = \begin{bmatrix} & \mathcal{IM}(\boldsymbol{h}_1)^\top \\ \boldsymbol{I}_{(\ell-1)n} & \vdots \\ & \mathcal{IM}(\boldsymbol{h}_{\ell-1})^\top \end{bmatrix} \in \mathbb{F}_{q^m}^{(\ell-1)n\times\ell n}. \tag{36}$$

G Ideal Blockwise Rank Decoding Problem

Definition 24 (Ideal ℓ-RSD (ℓ-IRSD) Problem). *Let $\boldsymbol{H}$ be a systematic parity-check matrix of a random $[\ell n, n]_{q^m}$-ideal code and $\boldsymbol{s} \in \mathbb{F}_{q^m}^{(\ell-1)n}$. The problem is to find an ℓ-error $\boldsymbol{e} \in \mathcal{S}_{\boldsymbol{r}}^{\boldsymbol{n}}$ such that $\boldsymbol{s} = \boldsymbol{H}\boldsymbol{e}^\top$.*

This is called the computational ℓ-IRSD problem. In security proof of cryptosystems, the decisional version is often used.

Definition 25 (Decisional ℓ-IRSD Problem). *Let $\boldsymbol{H}$ be a systematic parity-check matrix of a random $[\ell n, n]_{q^m}$-ideal code. The decisional ℓ-$\mathsf{IRSD}_{q,m,\ell n,n}^{\boldsymbol{n},\boldsymbol{r}}$ problem is hard, if for any PPT adversary $\mathcal{B}$, the following advantage is negligible:*

$$\begin{aligned} \mathsf{Adv}_{\mathcal{B}}^{\ell\text{-}\mathsf{IRSD}_{q,m,\ell n,n}^{\boldsymbol{n},\boldsymbol{r}}}(\lambda) := \Big| & \Pr\left[\mathcal{B}(\boldsymbol{H}, \boldsymbol{s}) = 1 \mid \boldsymbol{H} \xleftarrow{\$} \mathbb{F}_{q^m}^{(\ell-1)n\times\ell n}, \boldsymbol{s} \xleftarrow{\$} \mathbb{F}_{q^m}^{(\ell-1)n}\right] \\ & - \Pr\left[\mathcal{B}\left(\boldsymbol{H}, \boldsymbol{H}\boldsymbol{e}^\top\right) = 1 \mid \boldsymbol{H} \xleftarrow{\$} \mathbb{F}_{q^m}^{(\ell-1)n\times\ell n}, \boldsymbol{e} \leftarrow \mathcal{S}_{\boldsymbol{r}}^{\boldsymbol{n}}\right]\Big|. \end{aligned}$$

As hardness assumptions [3,33] of the ideal variants of the rank decoding problem, we also argue that: (1) The ℓ-IRSD problem is as hard as the ℓ-RSD problem and there is no known strong improvement on the complexity of solving the ideal version, typically choosing a $P(X)$ with many small factors to resist the folding attack [29] and in our case, $P(X)$ is an irreducible polynomial; (2) The attacks on the decisional ℓ-IRSD problem remain the direct attacks on the computational ℓ-IRSD problem, thus decisional and computational versions have similar hardness.

H Best Attacks on Blockwise Rank Decoding Problem

The attacks on blockwise rank decoding problems are studied in [46,4,47]. The best attacks are the combinatorial attacks (the AGHT attack and the PRR attack) and the algebraic attacks (the MM modeling).

The best attacks start from solving the ℓ-RD problem (see Definition 19). Solving the ℓ-RD$_{q,n,k,r,\boldsymbol{n},\boldsymbol{r}}$ problem defined by a random $[n,k]_{q^m}$-linear code $\mathcal{C}$ is reduced to finding a blockwise codeword (i.e., an ℓ-error in $\mathcal{S}_{\boldsymbol{r}}^{\boldsymbol{n}}$) in the $[n,k+1]_{q^m}$ extended code of $\mathcal{C}$.

Once obtaining word $\boldsymbol{y}$, one adds $\boldsymbol{y}$ to $\mathcal{C}$ and obtains an $[n,k+1]_{q^m}$ extended code $\mathcal{C}_{\boldsymbol{y}} = \mathcal{C} + \langle \boldsymbol{y} \rangle$ with a $(k+1) \times n$ generator matrix $\begin{pmatrix} \boldsymbol{y} \\ \boldsymbol{G} \end{pmatrix}$. In this way, $\boldsymbol{e} = \begin{pmatrix} 1 & -\boldsymbol{m} \end{pmatrix} \begin{pmatrix} \boldsymbol{y} \\ \boldsymbol{G} \end{pmatrix} \in \mathcal{S}_{\boldsymbol{r}}^{\boldsymbol{n}}$ is exactly a codeword of $\mathcal{C}_{\boldsymbol{y}}$. Let $\boldsymbol{H}_{\boldsymbol{y}} \in \mathbb{F}_{q^m}^{(n-k-1)\times n}$ be a parity-check matrix of $\mathcal{C}_{\boldsymbol{y}}$. Then solving the ℓ-RD problem consists in finding an ℓ-error $\boldsymbol{e} \in \mathcal{S}_{\boldsymbol{r}}^{\boldsymbol{n}}$ such that

$$\boldsymbol{e}\boldsymbol{H}_{\boldsymbol{y}}^{\top} = \boldsymbol{0}_{n-k-1}. \tag{37}$$

Before recalling best attacks, we need to note that the Gaussian elimination of a $\mu \times \nu$ matrix of rank ρ over an $\mathbb{F}_q$ has a complexity of $\mathcal{O}(\rho^{\omega-2}\mu\nu)$ operations in $\mathbb{F}_q$, where ω is the exponent of matrix multiplication (or linear algebra constant) with $2 \leq \omega \leq 3$, a practical value is 2.81 when there are more than a few hundred rows and columns. While it is now common to conservatively take $\omega = 2$, which considers any algorithm that could take advantage of the structure of the matrices, we still take $\omega = 2.81$ for a fair comparison because this is the case of the previous RQC (NIST PQC [34], Asiacrypt [46], and TIT [15]).

The AGHT Attack: The AGHT attack [8] is proposed by Aragon, Gaborit, Hauteville, and Tillich and is an improvement of GRS combinatorial attack [25]. In [46], the authors adapted the idea of the AGHT attack to the ℓ-RD problem. The idea is that the solver tries to guess the whole subspace F that contains the support of the ℓ-error, then expresses the coordinates of the ℓ-error in a basis of F, finally checks if the choice is correct by solving the linear system obtained from the parity-check Equation (37): $\boldsymbol{e}\boldsymbol{H}_{\boldsymbol{y}}^{\top} = \boldsymbol{0}_{n-k-1}$.

- Guess randomly a t-dimensional subspace F such that F contains $\mathrm{Supp}(\boldsymbol{e})$ of dimension $r = \sum_{i=1}^{\ell} r_i$ of the ℓ-error $\boldsymbol{e} = (\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots, \boldsymbol{e}_\ell) = (e_1, e_2, \ldots, e_n)$.

– Let $(f_1, f_2, \ldots, f_t) \in \mathbb{F}_{q^m}^t$ be a basis of F. One expresses $\boldsymbol{e}$ under this basis $e_j = \sum_i^r f_i e_{i,j}$ for $j \in [n]$, and constructs a linear system

$$\boldsymbol{H_y} \boldsymbol{e}^\top = \sum_j^n h_{\lambda,j} \sum_i^r f_i e_{i,j} = 0, \quad \lambda \in [n-k-1] \tag{38}$$

– Unfold the linear system (38) over $\mathbb{F}_q$ and solve $e_{i,j}$. By unfolding over $\mathbb{F}_q$, a linear system with nt unknowns and $m(n-k-1)$ equations is obtained. The linear system has only one solution with overwhelming probability if $nt \leq m(n-k-1)$.
– Once all $e_{i,j}$ are solved, the solver computes $\boldsymbol{e}$.

The cost of attack depends on how to successfully guess such a subspace F. The probability of $F \supset \mathrm{Supp}(\boldsymbol{e})$ is estimated as $\frac{\begin{bmatrix} t \\ r \end{bmatrix}_q}{\begin{bmatrix} m \\ r \end{bmatrix}_q} \approx q^{-r(m-t)}$. In this way, the complexity is $\mathcal{O}\left(((n-k-1)m)^\omega q^{r\left\lceil \frac{(k+1)m}{n} \right\rceil}\right)$. Then one uses $\mathbb{F}_{q^m}$-linearity to reduce the cost. Since, for any $\lambda \in \mathbb{F}_{q^m}^*$, $\|\lambda \boldsymbol{e}\|_{\mathrm{R}} = r$ and all multiples $\lambda \boldsymbol{e}$ are solutions of Equation (37): $\boldsymbol{e}\boldsymbol{H_y}^\top = \boldsymbol{0}$, the complexity is divided by about q^m. As a result, this attack has a complexity of

$$\mathcal{O}\left(((n-k-1)m)^\omega q^{r\left\lceil \frac{(k+1)m}{n} \right\rceil - m}\right).$$

The PRR Attack: The PRR attack is proposed by Puchinger, Renner, and Rosenkilde in [42], where authors adapted the GRS attack [25] to the sum-rank decoding problem with equal block length. Here, we generalized the block length, adapted the ideas of the PRR attack [42] and the AGHT attack [8] to the ℓ-RD problem, and obtained the gain of m bits than only adapting the PRR attack.

– The solver *blockwisely* guesses the subspace that contains the support of $\boldsymbol{e}_i$.
– For $i \in \{1..\ell\}$, guess randomly t_i-dimensional subspace F_i such that F_i contains the support $E_i = \mathrm{Supp}(\boldsymbol{e}_i)$ of dimension r_i of $\boldsymbol{e}_i$.
– Let $\boldsymbol{f}_i \in \mathbb{F}_{q^m}^{t_i}$ be a basis of F_i, then there exists a matrix $\boldsymbol{E}_i \in \mathbb{F}_q^{t_i \times n_i}$ such that $\boldsymbol{e}_i = \boldsymbol{f}_i \boldsymbol{E}_i$. Further, the ℓ-error $\boldsymbol{e}$ is expressed as

$$\boldsymbol{e} = (\boldsymbol{f}_1, \boldsymbol{f}_2, ..., \boldsymbol{f}_\ell) \begin{pmatrix} \boldsymbol{E}_1 & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{E}_2 & \boldsymbol{0} & \boldsymbol{0} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{E}_\ell \end{pmatrix} := \boldsymbol{f}\boldsymbol{E} \in \mathbb{F}_{q^m}^n$$

where $\boldsymbol{f} = (\boldsymbol{f}_1, \boldsymbol{f}_2, ..., \boldsymbol{f}_\ell) \in \mathbb{F}_{q^m}^{\sum_{i=1}^\ell t_i}$.
– Construct a linear system $\boldsymbol{f}\boldsymbol{E}\boldsymbol{H_y}^\top = \boldsymbol{0}$ from Equation (37): $\boldsymbol{e}\boldsymbol{H_y}^\top = \boldsymbol{0}$ and unfold the linear system $\boldsymbol{f}\boldsymbol{E}\boldsymbol{H_y}^\top = \boldsymbol{0}$ over $\mathbb{F}_q$ to obtain a linear system with $\sum_{i=1}^\ell n_i t_i$ unknowns in the entries of $\boldsymbol{E}$ and $m(n-k-1)$ equations over $\mathbb{F}_q$. (The unfolding operation is similar to the third step of the AGHT attack.)

– Solve the obtained linear system for a single solution as long as $\sum_{i=1}^{\ell} n_i t_i \leq m(n-k-1)$.
– Once the entries of $\boldsymbol{E}$ are solved, the solver computes $\boldsymbol{fE}$ as $\boldsymbol{e}$.

The most costly part of the attack consists in successfully finding F_i containing E_i for $i \in [\ell]$. The probability that one successfully guesses $\mathbb{F}_q$-subspace F_i dimension respectively t_i of $\mathbb{F}_{q^m}$ such that $E_i \subset F_i$ is estimated as $q^{-\sum_{i=1}^{\ell} r_i(m-t_i)}$. In this way, the complexity of solving the ℓ-RD problem is estimated as

$$\mathcal{O}\left((m(n-k-1))^{\omega} q^{\sum_{i=1}^{\ell} r_i(m-t_i)}\right).$$

Finally, one takes advantage of the $\mathbb{F}_{q^m}$-linearity to reduce this cost: for any $\lambda \in \mathbb{F}_{q^m}^*$, $\|\lambda \boldsymbol{e}\|_{\mathrm{R}} = r$ and all multiples $\lambda \boldsymbol{e}$ are solutions of Equation (37): $\boldsymbol{H_y} \boldsymbol{e}^{\top} = \boldsymbol{0}$, hence the complexity is divided by about q^m. As a result, this attack strategy has a complexity of

$$\mathcal{O}\left((m(n-k-1))^{\omega} q^{\sum_{i=1}^{\ell} r_i(m-t_i)-m}\right)$$

where t_i is chosen to minimize $\sum_{i=1}^{\ell} r_i(m-t_i) - m$ under the constraints

$$\begin{cases} r_i \leq t_i \leq m, \quad for\ i \in [\ell]; \\ \sum_{i=1}^{\ell} t_i \leq m; \\ \sum_{i=1}^{\ell} n_i t_i \leq m(n-k-1). \end{cases}$$

The MaxMinors (MM) Modeling: The MM modeling is proposed in [10], improved in [12], and viewed as the most powerful algebraic attack for cryptographic parameters. In [46], the authors adapted the idea of the MM modeling to the ℓ-RD problem and obtained the gain of factor ℓ due to the block-diagonal form of the coefficient matrix of ℓ-error.

Equation $\boldsymbol{\varepsilon C H_y}^{\top} = \boldsymbol{0}_{n-k-1}$ ($\boldsymbol{e H_y}^{\top} = \boldsymbol{0}_{n-k-1}$ and $\boldsymbol{e} = \boldsymbol{\varepsilon C}$) implies that $\boldsymbol{C H_y}^{\top} \in \mathbb{F}_{q^m}^{r \times (n-k-1)}$ is not of row full rank because a non-zero vector $\boldsymbol{\varepsilon}$ belongs to its left kernel. Then all maximal minors $|\boldsymbol{C H_y}^{\top}|_{*,J}$ of $\boldsymbol{C H_y}^{\top}$ are equal to 0 for $J \subset \{1..n-k-1\}$ and $\#J = r$. By the Cauchy-Binet formula, each $|\boldsymbol{C H_y}^{\top}|_{*,J}$ can be viewed a non-zero linear combination about all maximal minors $c_T = |\boldsymbol{C}|_{*,T}$ for $T \subset \{1..n\}$ and $\#T = r$. Since $\boldsymbol{C}$ of ℓ-error has the block-diagonal form $\mathrm{diag}(\boldsymbol{C}_1, \boldsymbol{C}_2, \ldots, \boldsymbol{C}_{\ell})$, the number of non-zero c_T is $\prod_{i=1}^{\ell} \binom{n_i}{r_i}$. One views non-zero c_T as unknowns and solves c_T from a linear system with $\prod_{i=1}^{\ell} \binom{n_i}{r_i}$ unknowns and at most $\binom{n-k-1}{r}$ equations. However, this system has many solutions due to $\binom{n-k-1}{r} < \prod_{i=1}^{\ell} \binom{n_i}{r_i}$ whereas one wants more equations than unknowns for a unique solution. To obtain more equations, one unfolds the coefficients over $\mathbb{F}_q$ and obtains at most $m\binom{n-k-1}{r}$ equations. This builds the MM modeling. Once all c_T are solved by the resulting system, one can determine the entries of $\boldsymbol{C}$. Finally, one solves $\boldsymbol{\varepsilon}$ and determine $\boldsymbol{e} = \boldsymbol{\varepsilon C}$.

$$\left\{P_{i,J} = |\boldsymbol{C H_y}^{\top}|_{*,J} : J \subset \{1..n-k-1\}, \#J = r, i \in \{1..m\}\right\} \qquad \text{(MM-}\mathbb{F}_q\text{)}$$

Unknowns: $\prod_{i=1}^{\ell}\binom{n_i}{r_i}$ variables $c_T \in \mathbb{F}_q$,
Equations: At most $m\binom{n-k-1}{r}$ linear equations $P_{i,J}=0$ over $\mathbb{F}_q$ in c_T.

One must ensure $m\binom{n-k-1}{r} = \prod_{i=1}^{\ell}\binom{n_i}{r_i} - 1$ such that the linear system has a unique solution.

- When the system is still underdetermined: $m\binom{n-k-1}{r} < \prod_{i=1}^{\ell}\binom{n_i}{r_i}$, one uses the hybrid approach to guess the last a_i columns of $\boldsymbol{C}_i$, and reduces the number of unknowns to $\prod_{i=1}^{\ell}\binom{n_i-a_i}{r_i}$ such that the system has almost the same number of equations than unknowns.
- When the system is very overdetermined: $m\binom{n-k-1}{r} \geq \prod_{i=1}^{\ell}\binom{n_i}{r_i}$, one punctures $\mathcal{C}_{\boldsymbol{y}}$ at last p coordinates, and applies the MM modeling on the punctured code $\mathcal{C}_{\boldsymbol{y}}$ such that the system has almost the same number of equations than unknowns.

The complexity of solving the ℓ-$\mathrm{RD}^{\boldsymbol{n},\boldsymbol{r}}_{q,m,n,k}$ problem by the MM modeling is estimated as

$$\begin{cases} \mathcal{O}\left(m\binom{n-p-k-1}{r}\left(\binom{n_\ell-p}{r_\ell}\prod_{i=1}^{\ell-1}\binom{n_i}{r_i}\right)^{\omega-1}\right), & m\binom{n-k-1}{r} \geq \prod_{i=1}^{\ell}\binom{n_i}{r_i}; \\ \mathcal{O}\left(q^{\sum_{i=1}^{\ell}a_i r_i} m\binom{n-k-1}{r}\left(\prod_{i=1}^{\ell}\binom{n_i-a_i}{r_i}\right)^{\omega-1}\right), & m\binom{n-k-1}{r} < \prod_{i=1}^{\ell}\binom{n_i}{r_i}. \end{cases}$$

where $p = \max\left\{ i \;\middle|\; m\binom{n-i-k-1}{r} \geq \binom{n_\ell-i}{r_\ell}\prod_{i=1}^{\ell-1}\binom{n_i}{r_i} - 1 \right\}$ such that $m\binom{n-p-k-1}{r} \geq \binom{n_\ell-p}{r_\ell}\prod_{i=1}^{\ell-1}\binom{n_i}{r_i} - 1$ holds and $(a_1, a_2, \ldots, a_\ell)$ is an integers sequence such that $m\binom{n-k-1}{r} \geq \prod_{i=1}^{\ell}\binom{n_i-a_i}{r_i} - 1$ exactly holds.

I Application to RQC

In this section, we improve RQC PKE [34] submitted to the NIST PQC competition by using the EG codes and the ℓ-IRSD problem.

I.1 Our RQC

Our RQC uses three codes:

- A public $[n,k]_{q^m}$-EG code with generator matrix $\boldsymbol{G} \in \mathbb{F}_{q^m}^{k\times n}$ generated by $\boldsymbol{g} \in \mathbb{F}_{q^m}^{n}$ of weight t. This EG code can correct up to weight $\min\left\{\left\lfloor\frac{n-k}{2}\right\rfloor, t-k\right\}$ by a probabilistic decoding algorithm denoted by $\mathsf{EG.Decode}$.
- A random $[2n,n]_{q^m}$-ideal code with parity-check matrix $\begin{bmatrix}\boldsymbol{I}_n & \mathcal{IM}(\boldsymbol{h})^\top\end{bmatrix}$.
- A random $[3n,n]_{q^m}$-ideal code with parity-check matrix $\begin{bmatrix}\boldsymbol{I}_n & \boldsymbol{0} & \mathcal{IM}(\boldsymbol{h})^\top \\ \boldsymbol{0} & \boldsymbol{I}_n & \mathcal{IM}(\boldsymbol{s})^\top\end{bmatrix}$.

The EG code is used for the decryption step. Two random ideal codes are used to ensure the security of the scheme. Our RQC scheme is described in Figure 1.

$\underline{\mathsf{RQC.KGen}(\lambda)}$:

- Sample $\boldsymbol{g} \xleftarrow{\$} \mathcal{S}_t^n$ and $\boldsymbol{h} \xleftarrow{\$} \mathbb{F}_{q^m}^n$ from a seed seed_1 of 40 bytes
- Sample $(\boldsymbol{x}, \boldsymbol{y}) \xleftarrow{\$} \mathcal{S}^{(n,n)}_{(w_{\boldsymbol{x}}, w_{\boldsymbol{y}})}$ from a seed seed_2 of 40 bytes
- Compute $\boldsymbol{s} = \boldsymbol{x} + \boldsymbol{h} \cdot \boldsymbol{y}$
- Output the public key $pk = (\boldsymbol{g}, \boldsymbol{h}, \boldsymbol{s})$ and the private key $sk = (\boldsymbol{x}, \boldsymbol{y})$.

$\underline{\mathsf{RQC.Enc}(pk, \boldsymbol{m})}$:

- Compute the generator matrix $\boldsymbol{G} \in \mathbb{F}_{q^m}^{k \times n}$ of $[n,k]_{q^m}$-EG code by $\boldsymbol{g}$
- Sample $(\boldsymbol{r}_1, \boldsymbol{r}_2, \boldsymbol{e}) \xleftarrow{\$} \mathcal{S}^{(n,n,n)}_{(w_{\boldsymbol{r}_1}, w_{\boldsymbol{r}_1}, w_{\boldsymbol{e}})}$
- Compute $\boldsymbol{u} = \boldsymbol{r}_1 + \boldsymbol{h} \cdot \boldsymbol{r}_2$ and $\boldsymbol{v} = \boldsymbol{m}\boldsymbol{G} + \boldsymbol{s} \cdot \boldsymbol{r}_2 + \boldsymbol{e}$
- Output $\boldsymbol{c} = (\boldsymbol{u}, \boldsymbol{v})$.

$\underline{\mathsf{RQC.Dec}(sk, \boldsymbol{c})}$: Output $\mathsf{EG.Decode}(\boldsymbol{v} - \boldsymbol{y} \cdot \boldsymbol{u})$.

Fig. 1. Description of our RQC PKE scheme.

Correctness. In the decryption step, $\boldsymbol{v} - \boldsymbol{u} \cdot \boldsymbol{y} = \boldsymbol{m}\boldsymbol{G} + \boldsymbol{x} \cdot \boldsymbol{r}_2 + \boldsymbol{e} - \boldsymbol{r}_1 \cdot \boldsymbol{y}$ and $\|\boldsymbol{x} \cdot \boldsymbol{r}_2 + \boldsymbol{e} - \boldsymbol{r}_1 \cdot \boldsymbol{y}\|_{\mathrm{R}} \leq w_{\boldsymbol{x}} w_{\boldsymbol{r}_2} + w_{\boldsymbol{y}} w_{\boldsymbol{r}_1} + w_{\boldsymbol{e}}$. Let $r = w_{\boldsymbol{x}} w_{\boldsymbol{r}_2} + w_{\boldsymbol{y}} w_{\boldsymbol{r}_1} + w_{\boldsymbol{e}}$. Thus, to obtain $\boldsymbol{m}$, one needs to ensure that the error $\boldsymbol{x} \cdot \boldsymbol{r}_2 + \boldsymbol{e} - \boldsymbol{r}_1 \cdot \boldsymbol{y}$ of weight $\leq r$ can be decoded and ensure that the DFR (Theorem 3) decoding errors of weight r is small enough.

Theorem 15. *Under the decisional* $2\text{-}\mathsf{IRSD}^{(n,n),(w_{\boldsymbol{x}},w_{\boldsymbol{y}})}_{q,m,2n,n}$ *and* $3\text{-}\mathsf{IRSD}^{(n,n,n),(w_{\boldsymbol{x}},w_{\boldsymbol{y}},w_{\boldsymbol{e}})}_{q,m,3n,n}$ *problems (Appendix G), our RQC in Figure 1 is IND-CPA secure. Concretely, for any PPT algorithm* $\mathcal{A}$*, there is a PPT algorithm* $\mathcal{B}$ *such that the advantage that* $\mathcal{A}$ *against the* $\mathsf{IND\text{-}CPA}$ *experiment defined by our RQC is bounded as*

$$\mathsf{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{RQC},\mathcal{A}}(\lambda) \leq 2\left(\mathsf{Adv}^{2\text{-}\mathsf{IRSD}^{(n,n),(w_{\boldsymbol{x}},w_{\boldsymbol{y}})}_{q,m,2n,n}}_{\mathcal{B}}(\lambda) + \mathsf{Adv}^{3\text{-}\mathsf{IRSD}^{(n,n,n),(w_{\boldsymbol{x}},w_{\boldsymbol{y}},w_{\boldsymbol{e}})}_{q,m,3n,n}}_{\mathcal{B}}(\lambda)\right).$$

Proof. The proof is similar to [34] with 2-IRSD and 3-IRSD instances. Two instances are defined as

$$\boldsymbol{x} + \boldsymbol{h} \cdot \boldsymbol{y} = \boldsymbol{s} \iff \begin{bmatrix} \boldsymbol{I}_n & \mathcal{IM}(\boldsymbol{h})^\top \end{bmatrix} \begin{bmatrix} \boldsymbol{x}^\top \\ \boldsymbol{y}^\top \end{bmatrix} = \boldsymbol{s}^\top,$$

$$\begin{matrix} \boldsymbol{r}_1 + \boldsymbol{h} \cdot \boldsymbol{r}_2 = \boldsymbol{u} \\ \boldsymbol{m}\boldsymbol{G} + \boldsymbol{s} \cdot \boldsymbol{r}_2 + \boldsymbol{e} = \boldsymbol{v} \end{matrix} \iff \begin{bmatrix} \boldsymbol{I}_n & \boldsymbol{0} & \mathcal{IM}(\boldsymbol{h})^\top \\ \boldsymbol{0} & \boldsymbol{I}_n & \mathcal{IM}(\boldsymbol{s})^\top \end{bmatrix} \begin{bmatrix} \boldsymbol{r}_1^\top \\ \boldsymbol{e}^\top \\ \boldsymbol{r}_2^\top \end{bmatrix} = \begin{bmatrix} \boldsymbol{u}^\top \\ \boldsymbol{v}^\top - (\boldsymbol{m}\boldsymbol{G})^\top \end{bmatrix}.$$

□

I.2 Parameters and Performance of Our (Conservative) RQC

The parameters are chosen in three principles. First, the complexity of solving decoding problems is ensured to reach the target security level 2^λ for the target

security level parameter λ. Secondly, the decoding capacity of the EG codes is ensured to satisfy the decryption correctness condition. The EG codes must be designed to correct errors of weight up to r. Third, one must ensure that the decoding failure rate (DFR) correcting errors of weight up to r (Theorem 3 for DFR) is less than $2^{-\lambda}$. The specific parameters are given in Table 7.

In Table 8, we list the complexity of best attacks on 2-$\mathsf{IRSD}^{(n,n),(w_{\boldsymbol{x}},w_{\boldsymbol{y}})}_{q,m,2n,n}$ and 3-$\mathsf{IRSD}^{(n,n,n),(w_{\boldsymbol{x}},w_{\boldsymbol{y}},w_{\boldsymbol{e}})}_{q,m,3n,n}$ problems with security parameters in Table 7. The best attack is the MM attack. "$\mathbb{C}$" represents the complexity of solving decoding problems. "MM-$2n$" (resp. "MM-$3n$") represents MM attacks on the 2-IRSD (resp. 3-IRSD) instances. One can verify the complexity of the MM attack by the optimal (a_1, a_2) (or (a_1, a_2, a_3)) and p. Based on the hardness assumptions of the ℓ-IRSD problem, we estimate its complexity by the ℓ-RSD problem. We refer to Appendix H for the best attacks on the ℓ-RSD problem.

From Table 9, we can observe that our parameters are smaller than that of the original one [46], which leads to a smaller size. The improvement benefits from the gain of the EG codes in decoding capacity. Moreover, the EG codes allow RQC to work over a small finite field, which enables our RQC to mitigate the advantage of the MM attack and also brings a gain of parameter size. Please see the MM attack for a detailed explanation in Appendix H.

In Table 10, we provide reference timings for our RQC on SageMath 9.5. These timings are roughly 1000 times greater than when running on C language. Thus, the efficiency is practical.

Table 7. Parameters of our (Conservative) RQC.

Our RQC	q	m	n	t	k	$w_{\boldsymbol{x}}$	$w_{\boldsymbol{y}}$	$w_{\boldsymbol{r}_1}$	$w_{\boldsymbol{r}_2}$	$w_{\boldsymbol{e}}$	r	$P(X)$	λ
RQC-128	2	53	83	53	3	4	4	4	4	4	36	$X^{83}+X^7+X^4+X^2+1$	128
RQC-192	2	59	108	59	4	4	5	4	5	4	44	$X^{108}+X^{17}+1$	192
RQC-256	2	73	137	73	4	5	5	5	5	7	57	$X^{137}+X^{21}+1$	256

Our RQC (Conservative)	q	m	n	t	k	$w_{\boldsymbol{x}}$	$w_{\boldsymbol{y}}$	$w_{\boldsymbol{r}_1}$	$w_{\boldsymbol{r}_2}$	$w_{\boldsymbol{e}}$	r	$P(X)$	λ
RQC-128	2	57	106	57	3	4	4	5	5	5	45	$X^{106}+X^{15}+1$	128
RQC-192	2	83	161	83	3	4	5	7	7	7	70	$X^{161}+X^{18}+1$	192
RQC-256	2	113	223	113	3	5	5	9	9	9	99	$X^{223}+X^{33}+1$	256

We set the weight t of generator $\boldsymbol{g}$ of the EG codes as m. $P(X)$ is a polynomial of degree n in $\mathbb{F}_q[X]$, and is used to build the ring $\mathcal{R} = \mathbb{F}_{q^m}[X]/\langle P(X)\rangle$ and construct the ideal codes (see Appendix F). $P(X)$ is set to be an irreducible polynomial for resisting the folding attack [29]. Moreover, to decrease the computational costs, $P(X)$ is suggested to be sparse. We choose minimal weight $P(X)$ with the SageMath 9.5 software.

Table 8. Complexity of the current best attacks on parameters in Table 7.

Our RQC	MM-2n $(\mathbb{C}, a_1, a_2, p)$	MM-3n $(\mathbb{C}, a_1, a_2, a_3, p)$	Security Level
RQC-128	$(2^{163}, 6, 6, 0)$	$(2^{165}, 0, 0, 0, 42)$	2^{163}
RQC-192	$(2^{235}, 3, 18, 0)$	$(2^{192}, 0, 0, 0, 41)$	2^{192}
RQC-256	$(2^{339}, 9, 28, 0)$	$(2^{262}, 0, 0, 0, 14)$	2^{262}

Our RQC (Conservative)	MM-2n $(\mathbb{C}, a_1, a_2, p)$	MM-3n $(\mathbb{C}, a_1, a_2, a_3, p)$	Security Level
RQC-128	$(2^{167}, 1, 10, 0)$	$(2^{218}, 0, 0, 0, 26)$	2^{167}
RQC-192	$(2^{243}, 7, 13, 0)$	$(2^{497}, 0, 6, 19, 0)$	2^{243}
RQC-256	$(2^{381}, 20, 21, 0)$	$(2^{1051}, 12, 20, 38, 0)$	2^{381}

For all estimations of complexity, the exponent of matrix multiplication (or linear algebra constant ω) is set as 2.81.

Table 9. Sizes and DFR of our (Conservative) RQC.

Our RQC	m	n	k	$w_{\boldsymbol{x}}$	$w_{\boldsymbol{y}}$	$w_{\boldsymbol{r}_1}$	$w_{\boldsymbol{r}_2}$	$w_{\boldsymbol{e}}$	r	pt	sk	pk	ct	total	DFR
RQC-128	53	83	3	4	4	4	4	4	36	159	40	590	1100	1690	2^{-133}
RQC-192	59	108	4	4	5	4	5	4	44	236	40	837	1594	2431	2^{-202}
RQC-256	73	137	4	5	5	5	5	7	57	292	40	1291	2566	3857	2^{-258}

Our RQC (Conservative)	m	n	k	$w_{\boldsymbol{x}}$	$w_{\boldsymbol{y}}$	$w_{\boldsymbol{r}_1}$	$w_{\boldsymbol{r}_2}$	$w_{\boldsymbol{e}}$	r	pt	sk	pk	ct	total	DFR
RQC-128	57	106	3	4	4	5	5	5	45	171	40	796	1512	2308	2^{-138}
RQC-192	83	161	3	4	5	7	7	7	70	249	40	1711	3342	5053	2^{-207}
RQC-256	113	223	3	5	5	9	9	9	99	339	40	3190	6300	9490	2^{-274}

Plaintext size (pt): mk bits; private key size (sk): a seed of 40 bytes; public key size (pk): $\left(\lceil \frac{mn}{8} \rceil + 40\right)$ bytes; ciphertext size (ct): $2\lceil \frac{mn}{8} \rceil$ bytes; bandwidth (total): pk + ct. The DFR is estimated by Equation (14) with $\gamma_q = 4$.

Table 10. Timings for our RQC.

RQC	KGen (ms)	Enc (ms)	Dec (ms)	Total (ms)
RQC-128	90	98	201	389
RQC-192	129	136	464	729
RQC-256	195	201	825	1221

Our RQC (Conservative)	KGen (ms)	Enc (ms)	Dec (ms)	Total (ms)
RQC-128	109	116	448	673
RQC-192	235	250	1080	1565
RQC-256	458	481	3480	4419

The schemes are implemented on SageMath 9.5. The benchmark is Ubuntu-22.04 + WSL + Intel(R) Core(TM) i5-8265U CPU @ 1.60GHz with SageMath 9.5. The test scripts are available online at https://github.com/RQCPKE/EGCodesRQC. Total: KGen + Enc + Dec.