Encryption scheme based on expanded Reed-Solomon codes

Khathuria, Karan ; Rosenthal, Joachim ; Weger, Violetta

DOI: https://doi.org/10.3934/amc.2020053

Advances in Mathematics of Communications
Volume 15, No. 2, 2021, 207–218
doi:10.3934/amc.2020053

ENCRYPTION SCHEME BASED ON EXPANDED REED-SOLOMON CODES

Karan Khathuria, Joachim Rosenthal and Violetta Weger

Institute of Mathematics
University of Zurich
Winterthurerstrasse 190
8057 Zurich, Switzerland

(Communicated by Sihem Mesnager)

Abstract. We present a code-based public-key cryptosystem, in which we use Reed-Solomon codes over an extension field as secret codes and disguise it by considering its shortened expanded code over the base field. Considering shortened expanded codes provides a safeguard against distinguisher attacks based on the Schur product. Moreover, without using a cyclic or a quasi-cyclic structure we obtain a key size reduction of nearly 45% compared to the classic McEliece cryptosystem proposed by Bernstein *et al.*

1. Introduction

In 1978 McEliece [31] presented the first code-based public key cryptosystem. It belongs to the family of very few public-key cryptosystems which are unbroken since decades. The hard problem the McEliece system relies on, is the difficulty of decoding a random (-like) linear code having no visible structure. McEliece proposed to use binary Goppa codes for the encryption scheme. Due to the low error-correcting capacity of Goppa codes, the cryptosystem results in large public key sizes. Several alternative families of codes have been proposed with the aim of reducing the key sizes. Some of the famous families of codes considered are: generalized Reed-Solomon codes [5, 6, 8, 10, 14, 25, 36], non-binary Goppa codes [12], algebraic geometric codes [24], LDPC and MDPC codes [7, 34], Reed-Muller codes [41] and convolutional codes [29]. Most of them were unsuccessful in hiding the structure of the private code [15, 16, 17, 18, 26, 33, 37, 42, 46].

The motivation to quest for better code-based cryptosystems is mainly due to the advent of quantum computers. In 1994 Peter Shor [40] developed a polynomial time quantum algorithm for factoring integers and solving discrete logarithm problems. This means that most of the currently popular cryptosystems, such as RSA and ECC, will be broken in an era of quantum computers. In the ongoing process of the standardization of quantum-resistant public-key cryptographic algorithms by the National Institute of Standards and Technology (NIST), code-based cryptosystems are one of the most promising candidates. At the time of this writing there are seven code-based cryptosystems included in NIST's standardization process: BIKE [3] based on quasi-cyclic MDPC codes, classic McEliece [11] based on binary Goppa codes, ROLLO [32] based on quasi-cyclic LRPC codes, RQC [1] based on rank

2020 *Mathematics Subject Classification:* 14G50, 94A60, 11T71.
Key words and phrases: Code-based cryptography, McEliece cryptosystem, Reed-Solomon codes, expanded codes, shortened codes.

metric quasi-cyclic codes, HQC [1] based on Hamming metric quasi-cyclic codes, LEDAcrypt [4] based on quasi-cyclic LDPC codes and NTS-KEM [2] based on binary Goppa codes.

In this paper we present a new variant of the McEliece scheme using expanded Reed-Solomon codes. A linear $[n, k]$ code defined over an extension field $\mathbb{F}_{q^m}$ can be expanded, over the base field $\mathbb{F}_q$, to a $[mn, mk]$ linear code by expanding each codeword with respect to a fixed $\mathbb{F}_q$-linear isomorphism from $\mathbb{F}_{q^m}$ to $\mathbb{F}_q^m$. In the proposed cryptosystem we hide the structure of an expanded GRS code by puncturing and permuting the columns of its parity check matrix and multiplying by an invertible block diagonal matrix. In order to decode a large number of non-codewords, we use a burst of errors during the encryption step, i.e. we consider error vectors having support in sub-vectors of size λ. This error pattern comes with a disadvantage: it can be used to speed up the information set decoding (ISD) algorithms. However, for a small degree of extension m, the key sizes turn out to be remarkably competitive.

The paper is organized as follows. In Section 2, we give the preliminaries regarding the expanded codes. In Section 3, we describe the proposed cryptosystem which is based on the shortening of an expanded generalized Reed-Solomon code. In Section 4, we provide security arguments for the proposed cryptosystem against the known structural and non-structural attacks. In Section 5, we provide parameters of the proposed cryptosystem that achieve a security level of 256-bits against the ISD algorithm.

2. Background

2.1. Expanded codes.

Let q be a prime power and let m be an integer. Let γ be a primitive element of the field $\mathbb{F}_{q^m}$, i.e. $\mathbb{F}_{q^m} \cong \mathbb{F}_q(\gamma)$. The field $\mathbb{F}_{q^m}$ can also be seen as an $\mathbb{F}_q$- vector space of dimension m via the following $\mathbb{F}_q$-linear isomorphism

$$\begin{aligned} \phi : \mathbb{F}_{q^m} &\longrightarrow \mathbb{F}_q^m, \\ a_0 + a_1\gamma + \cdots + a_{m-1}\gamma^{m-1} &\longmapsto (a_0, a_1, \ldots, a_{m-1}). \end{aligned}$$

We extend this isomorphism for vectors over $\mathbb{F}_{q^m}$ in the following way:

$$\begin{aligned} \phi_n : \mathbb{F}_{q^m}^n &\longrightarrow \mathbb{F}_q^{mn}, \\ (\alpha_0, \alpha_1, \ldots, \alpha_{n-1}) &\longmapsto (\phi(\alpha_0), \phi(\alpha_1), \ldots, \phi(\alpha_{n-1})) \,. \end{aligned}$$

This is clearly an $\mathbb{F}_q$-linear isomorphism. Hence this gives us a way to obtain a linear code over $\mathbb{F}_q$ from a linear code over $\mathbb{F}_{q^m}$.

Definition 2.1 (Expanded Codes)**.** Let n, k be positive integers with $k \leq n$, let q be a prime power and m be an integer. Let $\mathcal{C}$ be a linear code of length n and dimension k over $\mathbb{F}_{q^m}$. The expanded code of $\mathcal{C}$ with respect to a primitive element $\gamma \in \mathbb{F}_{q^m}$ is a linear code over the base field $\mathbb{F}_q$ defined as

$$\widehat{\mathcal{C}} := \{\phi_n(c) : c \in \mathcal{C}\},$$

where ϕ_n is the $\mathbb{F}_q$-linear isomorphism defined by γ as above.

Remark 1. It is easy to see that the expanded code $\widehat{\mathcal{C}}$ is a linear code of length mn and dimension mk, because ϕ_n is an $\mathbb{F}_q$-linear isomorphism and $|\widehat{\mathcal{C}}| = |\mathcal{C}| = (q^m)^k = q^{mk}$.

Given a code $\mathcal{C}$ with its generator matrix and parity check matrix, the following lemma gives a way to construct a generator matrix and a parity check matrix of the expanded code $\widehat{\mathcal{C}}$.

Lemma 2.2. *Let $\mathcal{C}$ be a linear code in $\mathbb{F}_{q^m}^n$.*

1. *Let $\mathcal{C}$ have a generator matrix $G = [g_1, g_2, \ldots, g_k]^\intercal$, where $g_1, g_2, \ldots, g_k$ are vectors in $\mathbb{F}_{q^m}^n$. Then the expanded code of $\mathcal{C}$ over $\mathbb{F}_q$ with respect to a primitive element $\gamma \in \mathbb{F}_{q^m}$ has the expanded generator matrix*

$$\widehat{G} := [\phi_n(g_1), \phi_n(\gamma g_1), \ldots, \phi_n(\gamma^{m-1} g_1), \phi_n(g_2), \phi_n(\gamma g_2), \ldots, \phi_n(\gamma^{m-1} g_2), \ldots, \phi_n(g_k), \phi_n(\gamma g_k) \ldots, \phi_n(\gamma^{m-1} g_k)]^\intercal.$$

2. *Let $\mathcal{C}$ have a parity check matrix $H = [h_1^\intercal, h_2^\intercal, \ldots, h_n^\intercal]$, where $h_1, h_2, \ldots, h_n$ are vectors in $\mathbb{F}_{q^m}^{n-k}$. Then the expanded code of $\mathcal{C}$ over $\mathbb{F}_q$ with respect to a primitive element $\gamma \in \mathbb{F}_{q^m}$ has the expanded parity check matrix*

$$\widehat{H} := [\phi_{n-k}(h_1)^\intercal, \phi_{n-k}(\gamma h_1)^\intercal, \ldots, \phi_{n-k}(\gamma^{m-1} h_1)^\intercal, \phi_{n-k}(h_2)^\intercal, \phi_{n-k}(\gamma h_2)^\intercal, \ldots, \phi_{n-k}(\gamma^{m-1} h_2)^\intercal, \ldots, \phi_{n-k}(h_n)^\intercal, \phi_{n-k}(\gamma h_n)^\intercal \ldots, \phi_{n-k}(\gamma^{m-1} h_n)^\intercal].$$

Proof. See [47, Theorem 1]. □

Proposition 1. *Let $\mathcal{C}$ be a linear code in $\mathbb{F}_{q^m}^n$ having a generator matrix $G = [g_1, g_2, \ldots, g_k]^\intercal$ and a parity check matrix $H = [h_1^\intercal, h_2^\intercal, \ldots, h_n^\intercal]$. Let $\widehat{G}$ and $\widehat{H}$ be the expanded generator matrix and expanded parity check matrix of $\widehat{\mathcal{C}}$, respectively. Then*

1. *$\phi_n(xG) = \phi_k(x)\widehat{G}$ for all $x \in \mathbb{F}_{q^m}^k$,*
2. *$\phi_{n-k}(Hy^\intercal) = \widehat{H}(\phi_n(y))^\intercal$ for all $y \in \mathbb{F}_{q^m}^n$.*

Proof. Let $x = (x_1, x_2, \ldots, x_k) \in \mathbb{F}_{q^m}^k$ and let $x_i = \sum_{j=0}^{m-1} x_{ij}\gamma^j$ for all $i \in \{1, 2, \ldots, k\}$. Then

$$\begin{aligned}
\phi_k(x)\widehat{G} &= \sum_{i=1}^{k} \sum_{j=0}^{m-1} x_{ij} \phi_n(\gamma^j g_i) \\
&= \sum_{i=1}^{k} \phi_n \left(\sum_{j=0}^{m-1} x_{ij} \gamma^j g_i \right) \\
&= \sum_{i=1}^{k} \phi_n(x_i g_i) \\
&= \phi_n \left(\sum_{i=1}^{k} x_i g_i \right) \\
&= \phi_n(xG).
\end{aligned}$$

Similarly, $\phi_{n-k}(Hy^\intercal) = \widehat{H}(\phi_n(y))^\intercal$ for all $y \in \mathbb{F}_{q^m}^n$. □

Remark 2. $\widehat{\mathcal{C}}$ can also be determined by the commutativity of the following diagram (as $\mathbb{F}_q$-linear maps):

$$\begin{array}{ccccccccc} 0 & \longrightarrow & \mathbb{F}_{q^m}^k & \xrightarrow{G} & \mathbb{F}_{q^m}^n & \xrightarrow{H^\intercal} & \mathbb{F}_{q^m}^{n-k} & \longrightarrow & 0 \\ & & \downarrow{\scriptstyle \phi_k} & & \downarrow{\scriptstyle \phi_n} & & \downarrow{\scriptstyle \phi_{n-k}} & & \\ 0 & \longrightarrow & \mathbb{F}_q^{mk} & \xrightarrow{\widehat{G}} & \mathbb{F}_q^{mn} & \xrightarrow{\widehat{H}^\intercal} & \mathbb{F}_q^{m(n-k)} & \longrightarrow & 0 \end{array}$$

3. The Cryptosystem

In this section we will present the proposed cryptosystem in the Niederreiter version. We consider an expanded GRS code whose parity check matrix can be viewed as n blocks, where each block is of size m. In order to destroy the algebraic structure of the code, we choose $2 \leq \lambda \leq m-1$ and shorten it on randomly chosen $m-\lambda$ columns in each block. We then hide the shortened code by multiplying it with an invertible matrix, which preserves the weight of a vector over the extension field $\mathbb{F}_{q^m}$.

Key generation: Let q be a prime power, $2 \leq \lambda < m$ be positive integers and $k < n \leq q^m$ be positive integers, satisfying $R := k/n > (1-\lambda/m)$. Consider a GRS code $\mathcal{C} = \mathrm{GRS}_{n,k}(\alpha, \beta)$ of dimension k and length n over the finite field $\mathbb{F}_{q^m}$ and choose a parity check matrix H of $\mathcal{C}$. Let t be the error correction capacity of $\mathcal{C}$.

Let $\widehat{H}$ be the expanded parity check matrix of the expanded code $\widehat{\mathcal{C}}$ of $\mathcal{C}$ with respect to a primitive element $\gamma \in \mathbb{F}_{q^m}$. $\widehat{H}$ is an $m(n-k) \times mn$ matrix over $\mathbb{F}_q$.

Shortening $\widehat{\mathcal{C}}$.

- For each $1 \leq i \leq n$, let S_i be a randomly chosen subset of $\{(i-1)m+1, (i-1)m+2, \ldots, im\}$ of size $m-\lambda$ and define $S = \bigcup_{i=1}^{n} S_i$.
- We puncture $\widehat{H}$ on columns indexed by S. Let $\widehat{H}_S$ be the resulting $m(n-k) \times \lambda n$ parity check matrix and let $\widehat{\mathcal{C}}_S$ be the shortened code.

Hiding $\widehat{\mathcal{C}}_S$.

- Choose n random $\lambda \times \lambda$ invertible matrices $T_1, T_2, \ldots, T_n$ over $\mathbb{F}_q$. Define T to be the block diagonal matrix having $T_1, T_2, \ldots, T_n$ as diagonal blocks.
- Now choose a random permutation σ of length n and define P_σ to be the block permutation matrix of size $\lambda n \times \lambda n$. It can also be seen as Kronecker product of the $n \times n$ permutation matrix corresponding to σ and the identity matrix of size λ.
- Define $Q := TP_\sigma$ and compute $H' = \widehat{H}_S Q$.

The private key is then (H, Q, γ) and the public key is (H', t, λ).

Encryption: Let $y \in \mathbb{F}_q^{\lambda n}$ be a message having support in t sub-vectors each of length λ, in particular

$$\begin{aligned} \mathrm{support}(y) \subseteq \{&\lambda(i_1-1)+1, \lambda(i_1-1)+2, \ldots, \lambda(i_1), \lambda(i_2-1)+1, \lambda(i_2-1)+2, \\ &\ldots, \lambda(i_2), \ldots, \lambda(i_t-1)+1, \lambda(i_t-1)+2, \ldots, \lambda(i_t)\}, \end{aligned}$$

for some distinct $i_1, i_2, \ldots, i_t \in \{1, 2, \ldots, n\}$. Then compute the cipher text

$$c = H' y^\intercal.$$

Decryption: For the decryption we apply ϕ_{n-k}^{-1} on c, i.e.

$$\phi_{n-k}^{-1}(c) \quad = \quad \phi_{n-k}^{-1}\left(\widehat{H}_S Q y^{\intercal}\right).$$

Observe that $\widehat{H}_S Q y^{\intercal} = \widehat{H}\bar{y}^{\intercal}$, where $\bar{y}$ is the embedding of $yQ^{\intercal}$ to $\mathbb{F}_{q^m}$, by introducing zeros on the positions indexed by S. From Proposition 1 we get

$$\phi_{n-k}^{-1}\left(\widehat{H}\bar{y}^{\intercal}\right) \quad = \quad H\left(\phi_n^{-1}(\bar{y})\right)^{\intercal}.$$

Due to the block structure of the matrix Q, the vector of $Qy^{\intercal}$ has support in t sub-vectors each of length λ, thus $\bar{y}$ has support in t sub-vectors each of length m. Henceforth $\mathrm{wt}(\phi_n^{-1}(\bar{y})) \leq t$, and we can decode $\phi_{n-k}^{-1}(c)$ to get $\phi_n^{-1}(\bar{y})$. By applying ϕ_n we get $\bar{y}$ and by projecting on positions not indexed by S, we get $Qy^{\intercal}$ and therefore after multiplying by Q^{-1}, we recover the message y.

Choice of parameters. For low key sizes it is desirable to use a small degree of extension m and small λ.

In the case of quadratic extension and in the case of $\lambda = 1$, puncturing all but one column from each block results in an alternant code (subfield subcode of a GRS code). Alternant codes are known to be vulnerable to square code attacks [17, 19]. Hence, we do not propose to use quadratic extensions or $\lambda = 1$.

We therefore propose to use $m = 3$ and $m = 4$ with $\lambda = 2$.

4. Security

In this section we discuss the security of the proposed cryptosystem. We focus on the three main attacks on cryptosystems based on GRS codes. Two of them are structural (or key recovery) attacks, namely the Sidelnikov-Shestakov attack and the distinguisher attack based on the Schur product of the public code. The third one is the best known non-structural attack called information set decoding (ISD).

4.1. Sidelnikov and Shestakov attack.

The first code-based cryptosystem using GRS codes as secret codes was proposed by Niederreiter in the same article [36] as the famous Niederreiter cryptosystem. This proposal was then attacked by Sidelnikov and Shestakov in [43], where they used the fact, that the public matrix is still a generator matrix of a GRS code and they were able to recover the evaluation points and hence the GRS structure of the public matrix.

In the cryptosystem proposed in Section 3, the secret GRS parity check matrix H over $\mathbb{F}_{q^m}$ is hidden in two ways: first by puncturing its expanded parity check matrix $\widehat{H}$ over $\mathbb{F}_q$ and then by scrambling the columns of the punctured matrix $\widehat{H}_S$. Due to multiplying $\widehat{H}_S$ with a block diagonal matrix it is clear that the resulting code is no more equivalent to an evaluation code (or an expanded evaluation code). Hence evaluations (or expanded evaluation column vectors) can not be exploited using the Sidelnikov-Shestakov attack.

4.2. Distinguisher attack based on the Schur product.

For the attack based on the Schur product we need to introduce some definitions and notations.

Definition 4.1 (Schur product)**.** Let $x, y \in \mathbb{F}_q^n$. We denote by the Schur product of x and y their component-wise product

$$x \star y = (x_1 y_1, \ldots, x_n y_n).$$

Remark 3. The Schur product is symmetric and bilinear.

Definition 4.2 (Schur product of codes and square code)**.** Let $\mathcal{A}, \mathcal{B}$ be two codes of length n. The Schur product of two codes is the vector space spanned by all $a \star b$ with $a \in \mathcal{A}$ and $b \in \mathcal{B}$:

$$\langle \mathcal{A} \star \mathcal{B} \rangle = \langle \{ a \star b \mid a \in \mathcal{A}, b \in \mathcal{B} \} \rangle.$$

If $\mathcal{A} = \mathcal{B}$, then we call $\langle \mathcal{A} \star \mathcal{A} \rangle$ the square code of $\mathcal{A}$ and denote it by $\langle \mathcal{A}^2 \rangle$.

Definition 4.3 (Schur matrix)**.** Let G be a $k \times n$ matrix, with rows $(g_i)_{1 \leq i \leq k}$. The Schur matrix of G, denoted by $S(G)$, consists of the rows $g_i \star g_j$ for $1 \leq i \leq j \leq k$.

We observe by Remark 3, that if G is a generator matrix of a code $\mathcal{C}$ then its Schur matrix $S(G)$ is a generator matrix of the square code of $\mathcal{C}$. Let s be the following map

$$\begin{array}{rcl} s : \mathbb{N} & \to & \mathbb{N} \\ k & \mapsto & \frac{1}{2}\left(k^2 + k\right). \end{array}$$

For a $k \times n$ matrix A, we observe that $S(A)$ has the size $s(k) \times n$.

Various McEliece cryptosystems based on modifications of GRS codes have been proved to be insecure [15, 18, 20]. This is because the dimension of the square code of GRS codes is very low compared to a random linear code of the same dimension. Moreover, other families of codes have also been shown to be vulnerable against the attacks based on Schur products. In [16], Couvreur *et al.* presented a general attack against cryptosystems based on algebraic geometric codes and their subcodes. In [19] Faugère *et al.* showed that high rate binary Goppa codes can be distinguished from a random code. In [17], Couvreur *et al.* presented a polynomial time attack against cryptosystems based on non-binary Goppa codes defined over quadratic extensions.

The distinguisher attack is based on the low dimensional square code of the public code (or of the shortened public code). In the following, based on experimental observations, we infer that the public code of the proposed cryptosystem cannot be distinguished using square code techniques.

Let $\widehat{\mathcal{C}}_S$ be the public code of the proposed cryptosystem. Note that $\widehat{\mathcal{C}}_S$ is a shortening of an expanded GRS code $\widehat{\mathcal{C}}$.

1. *Squares of expanded GRS codes*: Like in the case of Reed-Solomon codes and their subfield subcodes, the expanded GRS codes also have low square code dimension. To see this, we visualize expanded GRS codes as subfield subcodes of GRS-like codes. Let $\mathcal{C}$ be a GRS code of length n and dimension k over $\mathbb{F}_{q^m}$ having the following parity check matrix

$$H = V_r(x, y) := \begin{pmatrix} y_1 & y_2 & \cdots & y_n \\ y_1 x_1 & y_2 x_2 & \cdots & y_n x_n \\ \vdots & \vdots & \ddots & \vdots \\ y_1 x_1^{r-1} & y_2 x_2^{r-1} & \cdots & y_n x_n^{r-1} \end{pmatrix},$$

where $x = (x_1, \ldots, x_n)$ is a vector of distinct elements in $\mathbb{F}_{q^m}$, $y = (y_1, \ldots, y_n)$ is a vector over $\mathbb{F}_{q^m}^*$ and $r := n - k$. Let γ be a primitive element in $\mathbb{F}_{q^m}$. We define a new code $\mathcal{B}$ of length mn over $\mathbb{F}_{q^m}$ given by the kernel of the

following parity check matrix

$$H' = \begin{pmatrix} V_r(x,y) & | & V_r(x,\gamma y) & | & \cdots & | & V_r(x,\gamma^{m-1}y) \end{pmatrix}.$$

Using Lemma 2.2, it is easy to observe, that the expanded code $\widehat{\mathcal{C}}$ of $\mathcal{C}$ with respect to γ is permutation equivalent to the $\mathbb{F}_q$-kernel of H'. In other words $\widehat{\mathcal{C}}$ is permutation equivalent to the subfield subcode of $\mathcal{B}$ over $\mathbb{F}_q$. Observe that a generator matrix G' of $\mathcal{B}$ is given by

$$\left(\begin{array}{ccccc} V_k(x,y') & 0 & \dots & 0 & 0 \\ 0 & V_k(x,\gamma^{-1}y') & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & V_k(x,\gamma^{-(m-2)}y') & 0 \\ 0 & 0 & \dots & 0 & V_k(x,\gamma^{-(m-1)}y') \\ \hline V_r(x,y'') & 0 & \dots & 0 & -V_r(x,\gamma^{-(m-1)}y'') \\ 0 & V_r(x,\gamma^{-1}y'') & \dots & 0 & -V_r(x,\gamma^{1-(m-1)}y'') \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & V_r(x,\gamma^{-(m-2)}y'') & -V_r(x,\gamma^{(m-2)-(m-1)}y'') \end{array}\right),$$

where y' is such that $V_k(x,y')V_r(x,y)^{\intercal} = 0$, and $y'' = (x_1^k, x_2^k, \dots, x_n^k) \star y'$. One can verify that $G'(H')^{\intercal} = 0$. Observe that a generator matrix of $\widehat{\mathcal{C}}$ is permutation equivalent to

$$\widehat{G} = \left(\begin{array}{cccc} G_1 & 0 & \dots & 0 \\ 0 & G_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & G_m \\ \hline \multicolumn{4}{c}{G_{gv}} \end{array}\right),$$

where G_i is a generator matrix of the subfield subcode of $V_k(x,\gamma^{-(i-1)}y')$ over $\mathbb{F}_q$, and G_{gv} is a generator matrix of the $\mathbb{F}_q$-subfield subcode of the bottom $(m-1)r$ rows of G'. The matrix G_{gv} is also known as the glue-vector generator matrix, as in [45]. Due to the block structure of $\widehat{G}$ the Schur matrix of $\widehat{G}$ will have many zero rows. As a result the dimension of the square code is not full, given large enough n. This may lead to vulnerabilities when using expanded GRS codes directly in the cryptosystem.

2. *Effect of Shortening*: Consider the parity check matrix $\widehat{H}$ of an expanded GRS code as shown in Lemma 2.2. We partition the columns of $\widehat{H}$ into n blocks, each of size m. By the definition of $\widehat{H}$, each of these blocks corresponds to a unique column vector of the parity check matrix of the parent GRS code. In order to weaken this correspondence, we puncture (randomly chosen) $m - \lambda$ of the columns from each block of $\widehat{H}$. As a result the correspondence of each block to the parent column vector is inconsistent. In addition we multiply the punctured parity check matrix by an invertible block diagonal matrix T. This further destroys the algebraic structure inherited from the parent GRS code. This was evident in our computations of the square code dimension of such shortened codes. Even in the case of $m = 3$ we observed that puncturing one column from each block of $\widehat{H}$ results in a full square code dimension.

4.3. INFORMATION SET DECODING. Information set decoding (ISD) algorithms are the best known algorithms for decoding a general linear code. ISD algorithms were introduced by Prange [39] in 1962. Since then several improvements have been proposed for codes over the binary field by Lee-Brickel [27], Leon [28], Stern [44] and more recently by Bernstein *et al.* [13], Becker *et al.* [9], May-Ozerov [30]. Several of these algorithms have been generalized to the case of codes over general finite fields, see [21, 22, 23, 35, 38].

An ISD algorithm in its simplest form first chooses an information set I, which is a size k subset of $\{1, 2, \ldots, n\}$ such that the restriction of the parity check matrix on the columns indexed by the complement of I is non-singular. Then Gaussian elimination brings the parity check matrix in a standard form and assuming that the errors are outside of the information set, these row operations on the syndrome will exploit the error vector, if the weight does not exceed the given error correction capacity.

ISD for the proposed cryptosystem: In the proposed cryptosystem we introduce a burst pattern in the error vector, in particular the error vector has support in t sub-vectors each of length λ. Henceforth, we modify Stern's ISD algorithm to incorporate such pattern in the error vector.

We first recall the Stern's algorithm. The algorithm partitions the information set I into two equal-sized subsets X and Y, and chooses uniformly at random a subset Z of size ℓ outside of I. Then it looks for vectors having exactly weight p among the columns indexed by X, exactly weight p among the columns indexed by Y, and exactly weight 0 in columns indexed by Z and the missing weight $t - 2p$ in the remaining indices.

In the proposed cryptosystem we have been given a public code $\widehat{\mathcal{C}_S}$ of length λn and dimension $k' := mk - (m-\lambda)n$ over $\mathbb{F}_q$. We also know that the error vector has support in t sub-vectors of length λ. Hence we use Stern's algorithm on the blocks of size λ. We consider the information set I to have $\lfloor k'/\lambda \rfloor$ blocks. We partition I into two equal-sized subsets X and Y, and choose uniformly at random a subset Z of ℓ blocks outside of I. Then we look for vectors having support in exactly p blocks in X, exactly p blocks in Y, and exactly 0 blocks in Z.

In Section 5 we compute the key sizes of the proposed cryptosystem having 256-bit security against this modified ISD algorithm.

5. KEY SIZE

In this section we compute the key sizes of the proposed cryptosystem having 256-bit security against the ISD algorithm discussed in Section 4.3. Later we compare these key sizes with the key sizes of the McEliece cryptosystem using binary Goppa codes [11] and some recently proposed cryptosystems that are using Reed-Solomon codes as secret codes. These are based on the idea of [5, 6] (BBCRS), where the authors proposed to hide the structure of the code using as transformation matrix the sum of a rank z matrix and a weight w matrix. The proposed parameters in [5, 6] with $z = 1$ and $w \leq 1 + R$ were broken by the square code attack [15, 18], where R denotes the rate of the code. Two countermeasures were recently proposed in [8, 25]. In order to hide the structure of the Reed-Solomon code the authors of [8] use $w > 1 + R$ and $z = 1$ or $w < 1 + R$ and $z > 1$. Whereas in [25] the transformation matrix has weight $w = 2$ and rank $z = 0$.

In the proposed cryptosystem, the public key is a parity check matrix of a linear code over $\mathbb{F}_q$ having length λn and dimension $mk - (m-\lambda)n$. Hence the public key

size is $(\lambda n - m(n-k)) \cdot m(n-k) \cdot \log_2(q)$ bits. For a degree of extension m, let $\mathcal{C}_m$ be the public code.

In Table 1, we provide the key sizes for different rates of the public code $\mathcal{C}_3$ achieving a 256-bit security level against the modified ISD algorithm discussed in Section 4.3. Observe that the smallest key size is achieved at rate 0.82.

Rate	q	n	k	t	Key Size (bits)
0.60	13	1382	829	277	6783627
0.65	13	1270	825	223	5952804
0.70	13	1207	844	182	5339456
0.75	13	1192	894	149	4929077
0.80	13	1230	984	123	4702652
0.82	13	1258	1031	114	4624198
0.85	13	1340	1139	101	4634545
0.87	13	1420	1235	93	4692805
0.90	13	1602	1441	81	4863276

TABLE 1. Comparing key sizes of the proposed cryptosystem with $m = 3$ and $\lambda = 2$ reaching a 256-bit security level against the modified ISD algorithm.

In Table 2, we provide the key sizes for different rates of the public code $\mathcal{C}_4$ achieving a 256-bit security level against the modified ISD algorithm discussed in Section 4.3. In this case the smallest key size is achieved at rate 0.89.

Rate	q	n	k	t	Key Size (bits)
0.65	7	2360	1534	413	13134108
0.70	7	1945	1361	292	10191102
0.75	7	1738	1303	218	8480009
0.80	7	1662	1329	167	7448878
0.85	7	1700	1445	128	6815134
0.87	7	1770	1539	116	6785893
0.89	7	1872	1666	103	6754721
0.91	7	2024	1841	92	6814326

TABLE 2. Comparing key sizes of the proposed cryptosystem with $m = 4$ and $\lambda = 2$ reaching a 256-bit security level against the modified ISD algorithm.

In conclusion, for a 256 bit security level we propose to use the cryptosystem with the two sets of parameters $(q = 13, m = 3, \lambda = 2, n = 1258, k = 1031)$ and $(q = 7, m = 4, \lambda = 2, n = 1872, k = 1666)$, see Table 3.

The proposed parameters for the classic McEliece system using binary Goppa codes by Bernstein *et al.* in [11] are $q = 2, m = 13, n = 6960, k = 5413$, which gives a key size of 8373911 bits. It achieves a security level of 260-bits with respect to the ball-collision algorithm [13].

In comparison to the classic McEliece system, the Type I set of parameters reduces the key size by 44.8% and the Type II set of parameters reduces the key size by 19.3%.

		q	m	n	k	Key Size (in bits)
Proposed system	Type I	13	3	1258	1031	4624198
	Type II	7	4	1872	1666	6754721
classical McEliece		2	13	6960	5413	8373911
BBCRS based schemes	$w = 1.708$ and $z = 1$	1423	1	1422	786	5113520
	$w = 1.2$ and $z = 10$	1163	1	1162	928	2274160
	$w = 2$ and $z = 0$	1993	1	1992	1593	6966714

TABLE 3. Comparing the key sizes of the proposed parameters against different cryptosystems.

ACKNOWLEDGMENTS

The authors would like to thank Matthieu Lequesne and Jean-Pierre Tillich for pointing out the square code vulnerability in the case of quadratic extensions. This work has been supported by the Swiss National Science Foundation under grant no. 169510.

REFERENCES

[1] C. Aguilar-Melchor, O. Blazy, J.-C. Deneuville, P. Gaborit and G. Zémor, Efficient encryption from random quasi-cyclic codes, *IEEE Transactions on Information Theory*, **64** (2018), 3927–3943.

[2] M. Albrecht, C. Cid, K. G. Paterson, C. J. Tjhai and M. Tomlinson, *NTS-KEM*, 2018.

[3] N. Aragon, P. S. L. M. Barreto, S. Bettaieb, Lo. Bidoux, O. Blazy, J.-C. Deneuville, P. Gaborit, S. Gueron, T. Guneysu, C. A. Melchor, R. Misoczki, E. Persichetti, N. Sendrier, J.-P. Tillich and G. Zémor, *Bike: Bit Flipping Key Encapsulation*, 2017.

[4] M. Baldi, A. Barenghi, F. Chiaraluce, G. Pelosi and P. Santini, LEDAkem: A post-quantum key encapsulation mechanism based on QC-LDPC codes, *Post-Quantum Cryptography, Lecture Notes in Comput. Sci., Springer, Cham*, **10786** (2018), 3–24.

[5] M. Baldi, M. Bianchi, F. Chiaraluce, J. Rosenthal and D. Schipani, A variant of the McEliece cryptosystem with increased public key security, *Proceedings of the Seventh International Workshop on Coding and Cryptography (WCC) 2011*, (2011), 173–182.

[6] M. Baldi, M. Bianchi, F. Chiaraluce, J. Rosenthal and D. Schipani, *Method and Apparatus for Public-Key Cryptography Based on Error Correcting Codes*, 2015, US Patent 9,191,199.

[7] M. Baldi, M. Bodrato and F. Chiaraluce, A new analysis of the McEliece cryptosystem based on QC-LDPC codes, *International Conference on Security and Cryptography for Networks, Springer Berlin Heidelberg*, (2008), 246–262.

[8] M. Baldi, F. Chiaraluce, J. Rosenthal, P. Santini and D. Schipani, On the security of generalized Reed-Solomon code-based cryptosystems, *IET Information Security*, (2019).

[9] A. Becker, A. Joux, A. May and A. Meurer, Decoding random binary linear codes in $2^{n/20}$: How $1 + 1 = 0$ improves information set decoding, *Advances in Cryptology—EUROCRYPT 2012, Lecture Notes in Comput. Sci., Springer, Heidelberg*, **7237** (2012), 520–536.

[10] T. P. Berger and P. Loidreau, How to mask the structure of codes for a cryptographic use, *Des. Codes Cryptogr.*, **35** (2005), 63–79.

[11] D. J. Bernstein, T. Lange and C. Peters, Attacking and defending the McEliece cryptosystem, *Post-Quantum Cryptography, Lecture Notes in Comput. Sci., Springer, Berlin*, **5299** (2008), 31–46.

[12] D. J. Bernstein, T. Lange and C. Peters, Wild McEliece, *Selected Areas in Cryptography, Lecture Notes in Comput. Sci., Springer, Heidelberg*, **6544** (2011), 143–158.

[13] D. J. Bernstein, T. Lange and C. Peters, Smaller decoding exponents: Ball-collision decoding, *Advances in Cryptology—CRYPTO 2011, Lecture Notes in Comput. Sci., Springer, Heidelberg*, **6841** (2011), 743–760.

[14] J. Bolkema, H. Gluesing-Luerssen, C. A. Kelley, K. E. Lauter, B. Malmskog and J. Rosenthal, Variations of the McEliece cryptosystem, *Algebraic Geometry for Coding Theory and Cryptography, Assoc. Women Math. Ser., Springer, Cham*, **9** (2017), 129–150.

[15] A. Couvreur, P. Gaborit, V. Gauthier-Umaña, A. Otmani and J.-P. Tillich, Distinguisher-based attacks on public-key cryptosystems using reed-solomon codes, *Designs, Codes and Cryptography*, **73** (2014), 641–666.

[16] A. Couvreur, I. Márquez-Corbella and R. Pellikaan, Cryptanalysis of McEliece cryptosystem based on algebraic geometry codes and their subcodes, *IEEE Trans. Inform. Theory*, **63** (2017), 5404–5418.

[17] A. Couvreur, A. Otmani and J.-P. Tillich, Polynomial time attack on wild McEliece over quadratic extensions, *IEEE Transactions on Information Theory*, **63** (2017), 404–427.

[18] A. Couvreur, A. Otmani, J.-P. Tillich and V. Gauthier-Umaña, A polynomial-time attack on the BBCRS scheme, *Public-key Cryptography-PKC 2015, Lecture Notes in Comput. Sci., Springer, Heidelberg*, **9020** (2015), 175–193.

[19] J.-C. Faugère, V. Gauthier-Umaña, A. Otmani, L. Perret and J.-P. Tillich, A distinguisher for high-rate McEliece cryptosystems, *IEEE Transactions on Information Theory*, **59** (2013), 6830–6844.

[20] V. Gauthier-Umaña, A. Otmani and J.-P. Tillich, A distinguisher-based attack on a variant of McEliece's cryptosystem based on reed-solomon codes, Preprint, (2012), arXiv:1204.6459.

[21] C. T. Gueye, J. B. Klamti and S. Hirose, Generalization of BJMM-ISD using May-Ozerov nearest neighbor algorithm over an arbitrary finite field $\mathbb{F}_q$, *Codes, Cryptology and Information Security, Lecture Notes in Comput. Sci., Springer, Cham*, **10194** (2017), 96–109.

[22] S. Hirose, May-Ozerov algorithm for nearest-neighbor problem over $\mathbb{F}_q$ and its application to information set decoding, *International Conference for Information Technology and Communications, Springer*, (2016), 115–126.

[23] C. Interlando, K. Khathuria, N. Rohrer, J. Rosenthal and V. Weger, Generalization of the ball-collision algorithm, Preprint, (2018), arXiv:1812.10955.

[24] H. Janwa and O. Moreno, McEliece public key cryptosystems using algebraic-geometric codes, *Designs, Codes and Cryptography*, **8** (1996), 293–307.

[25] K. Khathuria, J. Rosenthal and V. Weger, Weight two masking of the reed-solomon structure in conjugation with list decoding, *Proceedings of 23rd International Symposium on Mathematical Theory of Networks and Systems, Hong Kong University of Science and Technology, Hong Kong*, (2018), 309–314.

[26] G. Landais and J.-P. Tillich, An efficient attack of a McEliece cryptosystem variant based on convolutional codes, *International Workshop on Post-Quantum Cryptography, Springer*, (2013), 102–117.

[27] P. J. Lee and E. F. Brickell, An observation on the security of McEliece's public-key cryptosystem, *Advances in Cryptology—EUROCRYPT '88 (Davos, 1988), Lecture Notes in Comput. Sci., Springer, Berlin*, **330** (1988), 275–280.

[28] J. S. Leon, A probabilistic algorithm for computing minimum weights of large error-correcting codes. Coding techniques and coding theory, *IEEE Transactions on Information Theory*, **34** (1988), 1354–1359.

[29] C. Löndahl and T. Johansson, A new version of McEliece PKC based on convolutional codes, *International Conference on Information and Communications Security, Springer*, (2012), 461–470.

[30] A. May and I. Ozerov, On computing nearest neighbors with applications to decoding of binary linear codes, *Advances in Cryptology—EUROCRYPT 2015. Part I, Lecture Notes in Comput. Sci., Springer, Heidelberg*, **9056** (2015), 203–228.

[31] R. J. McEliece, *A Public-Key Cryptosystem Based on Algebraic Coding Theory*, Technical report, DSN Progress report, Jet Propulsion Laboratory, Pasadena, 1978.

[32] C. A. Melchor, N. Aragon, M. Bardet, S. Bettaieb, L. Bidoux, O. Blazy, J.-C. Deneuville, P. Gaborit, A. Hauteville, A. Otmani, O. Ruatta, J.-P. Tillich and G. Zémor, *ROLLO-Rank-Ouroboros*, LAKE & LOCKER, 2018.

[33] L. Minder and A. Shokrollahi, Cryptanalysis of the Sidelnikov cryptosystem, *Advances in Cryptology—EUROCRYPT 2007, Lecture Notes in Comput. Sci., Springer, Berlin*, **4515** (2007), 347–360.

[34] R. Misoczki, J.-P. Tillich, N. Sendrier and P. S. L. M. Barreto, MDPC-McEliece: New McEliece variants from moderate density parity-check codes, *2013 IEEE International Symposium on Information Theory*, (2013), 2069–2073.

[35] R. Niebuhr, E. Persichetti, P.-L. Cayrel, S. Bulygin and J. Buchmann, On lower bounds for information set decoding over $\mathbb{F}_q$ and on the effect of partial knowledge, *Int. J. Inf. Coding Theory*, **4** (2017), 47–78.

[36] H. Niederreiter, Knapsack-type cryptosystems and algebraic coding theory, *Problems Control Inform. Theory/Problemy Upravlen. Teor. Inform.*, **15** (1986), 159–166.
[37] A. Otmani, J.-P. Tillich and L. Dallot, Cryptanalysis of two McEliece cryptosystems based on quasi-cyclic codes, *Mathematics in Computer Science*, **3** (2010), 129–140.
[38] C. Peters, Information-set decoding for linear codes over $\mathbb{F}_q$, *Post-Quantum Cryptography, Lecture Notes in Comput. Sci., Springer, Berlin*, **6061** (2010), 81–94, https://bitbucket.org/cbcrypto/isdfq/src/master/.
[39] E. Prange, The use of information sets in decoding cyclic codes, *IRE Transactions on Information Theory*, **8** (1962), S5–S9.
[40] P. W. Shor, Algorithms for quantum computation: Discrete logarithms and factoring, *35th Annual Symposium on Foundations of Computer Science (Santa Fe, NM, 1994)*, IEEE Comput. Soc. Press, Los Alamitos, CA, (1994), 124–134.
[41] V. M. Sidelnikov, A public key cryptosystem based on Reed-Muller binary codes, *Discrete Math. Appl.*, **4** (1994), 191–207.
[42] V. M. Sidelnikov and S. O. Shestakov, On an encoding system constructed on the basis of generalized Reed-Solomon codes, *Diskret. Mat.*, **4** (1992), 57–63.
[43] V. M. Sidelnikov and S. O. Shestakov, On insecurity of cryptosystems based on generalized Reed-Solomon codes, *Discrete Mathematics and Applications*, **2** (1992), 439–444.
[44] J. Stern, A method for finding codewords of small weight, *Coding Theory and Applications, Lecture Notes in Comput. Sci., Springer, New Yorkpages*, **388** (1989), 106–113.
[45] A. Vardy and Y. Be′ery, Bit-level soft-decision decoding of Reed-Solomon codes, *IEEE Transactions on Communications*, **39** (1991), 440–444.
[46] C. Wieschebrink, Cryptanalysis of the Niederreiter public key scheme based on GRS subcodes, *Post-Quantum Cryptography, Lecture Notes in Comput. Sci., Springer, Berlin*, **6061** (2010), 61–72.
[47] Y. Q. Wu, On expanded cyclic and Reed-Solomon codes, *IEEE Transactions on Information Theory*, **57** (2011), 601–620.

Received June 2019; revised October 2019.

E-mail address: karan.khathuria@math.uzh.ch
E-mail address: rosenthal@math.uzh.ch
E-mail address: violetta.weger@math.uzh.ch