List-decoding of Subspace Codes and Rank-Metric Codes up to Singleton Bound

Hessam Mahdavifar and Alexander Vardy
University of California San Diego, La Jolla, CA 92093, USA
{hessam@ucsd.edu, avardy@ucsd.edu}

Abstract—Subspace codes and rank-metric codes can be used to correct errors and erasures in networks, with linear network coding. Both types of codes have been extensively studied in the past five years.

Subspace codes were introduced by Koetter and Kschischang to correct errors and erasures in networks where topology is unknown (the noncoherent case). In this model, the codewords are vector subspaces of a fixed ambient space; thus codes for this model are collections of such subspaces. In a previous work, we have developed a family of subspace codes, based upon the Koetter-Kschischang construction, which are efficiently list decodable. Using these codes, we achieved a better decoding radius than Koetter-Kschischang codes at low rates. Herein, we introduce a new family of subspace codes based upon a different approach which leads to a linear-algebraic list-decoding algorithm. The resulting error correction radius can be expressed as follows: for any integer \( s \), our list-decoder using \( s + 1 \)-interpolation polynomials guarantees successful recovery of the message subspace provided the normalized dimension of errors is at most \( s(1 - sR) \). The same list-decoding algorithm can be used to correct erasures as well as errors. The size of output list is at most \( Q^{s+1} \), where \( Q \) is the size of the field that message symbols are chosen from.

Rank-metric codes are suitable for error correction in the case where the network topology and the underlying network code are known (the coherent case). Gabidulin codes are a well-known class of algebraic rank-metric codes that meet the Singleton bound on the minimum rank metric of a code. In this paper, we introduce a folded version of Gabidulin codes analogous to the folded Reed-Solomon codes of Guruswami and Rudra along with a list-decoding algorithm for such codes. Our list-decoding algorithm makes it possible to recover the message provided that the normalized rank of erasure is at most \( 1 - R - \epsilon \), for any \( \epsilon > 0 \). Notably this achieves the information theoretic bound on the decoding radius of a rank-metric code.

Index Terms—list-decoding, subspace codes, rank-metric codes, linearized polynomial, Gabidulin codes

I. INTRODUCTION

Subspace codes and rank metric codes are two closely related family of codes used for reliable communication of messages in linear network coding [9] [18]. Network coding, in general, is a technique where nodes of the network take several packets and combine them together for transmission instead of simply relaying the packets of information they receive [1]. It is known that in multicast setting, where one transmitter communicates with several receivers in the network simultaneously, linear network coding, wherein all the operations performed at network nodes are linear operations, is sufficient to achieve the individual max-flow bound on the rate of communication between the transmitter and each of the receivers [10].

In random linear network coding, each intermediate node of the network creates a random linear combination of the packets it receives and sends it through its output links [7]. It is proved that random linear network coding is as good as linear network coding, in the context of multicast setting, with high probability if the size of the field the message symbols are chosen from is large enough [8]. Subspace codes have been recently introduced in order to enable reliable communication of messages in random linear network coding [9]. Let the ambient space \( W \) be a vector space over a finite field \( \mathbb{F}_q \). A subspace code in \( W \) is a non-empty subset of all the subspaces of \( W \). Koetter-Kschischang algebraic construction of subspace codes, originally called Reed-Solomon-like codes in [9], is analogous to Reed-Solomon codes in classical block codes wherein symbols are replaced by vectors, regular polynomials with linearized polynomials, and sequences of symbols with \( \mathbb{F}_q \)-linear span of the corresponding vectors.

In a previous work, we proposed a new family of subspace codes that enables list-decoding, hence achieving a better tradeoff between rate and error correction capability [12], [13]. The idea was to evaluate all the powers of the linearized message polynomial, up to some power \( L \), in order to list-decode with list size at most \( L \). In a sense our algorithm can be regarded as an analogous to Sudan list-decoding algorithm of Reed-Solomon codes [19]. We further improved this result by introducing multiplicity in the ring of linearized polynomials [14]. This was motivated by Guruswami-Sudan list-decoding algorithm of Reed-Solomon codes [6]. We were able to list-decode for a wider range of rates and also to achieve a better tradeoff between the rate and decoding radius by enforcing multiple roots for the interpolation polynomial [14].

In this paper, we introduce a new family of subspace codes that allows a simple linear-algebraic list-decoding by using \( s + 1 \)-variate interpolation polynomials, where \( s \) is a design parameter. The entire list-decoding algorithm is linear-algebraic. A system of linear equations is solved for the interpolation step and another linear system is solved to compute the set of all the possible solutions which indeed is a linear space. This is motivated by the recent work of Vadhan [20, Ch. 5] and Guruswami [4] which suggested a simplified version, with no need of multiplicity, of previously proposed list-decoding algorithm of folded Reed-Solomon codes by Guruswami and Rudra in [5]. The later was built upon the work of Parvaresh and Vardy on list-decoding of Reed-Solomon codes by proposing multivariate interpolation [15].

In the coherent system of network coding, the network topology and the particular network coding operations done at intermediate nodes are known to both the transmitter and the receiver. In this setting, rank-metric codes turned out to be the suitable tool to deal with possible injected errors into the network [18]. Each codeword in a rank-metric code is a matrix
with fixed dimensions whose entries are taken from a finite field \( \mathbb{F}_q \). The distance between two matrices is simply the rank of their difference. Gabidulin codes were introduced as a class of MRD (maximum rank distance) codes. They achieve the Singleton bound on the minimum rank distance of a rank metric code. In this paper, we define a folded version of Gabidulin codes. Then we propose a list-decoding algorithm that can correct the fraction of errors up to the Singleton bound which is the information theoretic upper bound on the error correction capability of a code.

The rest of this paper is organized as follows. We start with a brief overview of linearized polynomials, subspace codes, rank-metric codes and Gabidulin codes in Section II. In Section III we first discuss our new construction of subspace codes, then we propose a list-decoding algorithm. After that, we establish the correctness of the algorithm and provide the decoding radius and other parameters of our code. In Section IV we introduce the folded version of Gabidulin codes and provide the list-decoding algorithm. Then we show that we are able to correct the fraction of errors up to \( 1 - R \), \( R \) being the rate of the code, hence achieving the Singleton upper bound on the error correction capability of rank-metric codes.

II. Background and Prior Work

In this section, we first explain the ring of linearized polynomials. We establish some relevant terminology for subspace codes and explain the Koetter-Kschischang construction of subspace codes. Then we briefly recap the results of [12] and [13] which provide a new construction of subspace codes. In [12] and [13] we suitably modified and extended Koetter-Kschischang construction in many important respects in order to enable list-decoding. Then we briefly discuss the results of the follow-up work in [14] wherein we introduced multiple roots for the interpolation polynomial. At the end, we briefly review rank-metric codes and Gabidulin codes as a class of maximum rank distance codes.

A polynomial over some extension field \( \mathbb{F}_{q^m} \) of \( \mathbb{F}_q \) is called linearized if it has the following form:

\[
    f(X) = \sum_{i=0}^{s} a_i X^{q^i},
\]

where \( a_i \in \mathbb{F}_q \), for \( i = 0, 1, \ldots, s \). Assuming that \( a_s \neq 0 \) we say that the polynomial \( f(X) \) has \( q \)-degree \( s \) which means that its actual degree is \( q^s \). When \( q \) is fixed under discussion, we will let \( X^{q^i} \) denote \( X^{q^i} \). The main property of linearized polynomials from which they receive their name is that they act as linear maps with respect to \( \mathbb{F}_q \). The set of linearized polynomials forms a non-commutative ring under addition + and composition operation \( \otimes \). For any two linearized polynomials \( f_1(X) \) and \( f_2(X) \), the composition operation \( f_1(X) \otimes f_2(X) \) is defined to be the composition \( f_1(f_2(X)) \) which is always a linearized polynomial. The ring of linearized polynomials over \( \mathbb{F}_{q^m} \) is denoted by \( \mathcal{L}_{q^m} \).

A. Subspace Codes

Let \( W \) be a fixed \( N \)-dimensional vector space over \( \mathbb{F}_q \) and \( \mathcal{G}(W) \) denote the set of all subspaces of \( W \). For any \( V \in \mathcal{G}(W) \), the dimension of \( V \) is denoted by \( \dim(V) \). For any \( A, B \in \mathcal{G}(W) \), the distance between \( A \) and \( B \) is defined as follows:

\[
    d(A, B) \overset{\text{def}}{=} \dim(A + B) - \dim(A \cap B)
\]

\( \mathcal{G}(W) \) is indeed a metric space under this metric. Let \( \mathcal{G}(W, n) \) denote the set of all \( n \)-dimensional subspaces of \( W \). A code \( C \) in the ambient space \( W \) is a non-empty subset of \( \mathcal{G}(W) \). A code-word is an element of \( C \) which is in fact a subspace of \( W \).

Definition. [9] Let \( C \) be a code associated with the ambient space \( W \) of dimension \( N \) over \( \mathbb{F}_q \). Suppose that the dimension of any \( V \in C \) is at most \( n \). Then the rate of the code \( R \) is defined as follows.

\[
    R \overset{\text{def}}{=} \frac{\log_q |C|}{nN}
\]

In [14], we defined a new parameter, called the packet rate of the code. The packet rate \( R^* \) is defined as follows:

\[
    R^* \overset{\text{def}}{=} \frac{\log_q |C|}{nm}
\]

where \( q^m \) is the size of the underlying extension field.

Koetter-Kschischang construction [9] of subspace codes can be regarded as an analogous to Reed-Solomon codes wherein symbols are replaced by vectors, polynomials with linearized polynomials and sequences of symbols with \( \mathbb{F}_q \)-linear span of the corresponding vectors. Fix \( m \) and an extension field \( \mathbb{F}_{q^m} \) of \( \mathbb{F}_q \). \( \mathbb{F}_{q^m} \) can be also regarded as a vector space of dimension \( m \) over \( \mathbb{F}_q \). Fix a set \( A = \{ \alpha_1, \ldots, \alpha_n \} \) of \( n \) linearly independent vectors in \( \mathbb{F}_{q^m} \). Let \( u = (u_0, \ldots, u_{k-1}) \) be the message vector and \( f_u(X) = \sum_{i=0}^{k-1} u_i X^{q^i} \) be the corresponding linearized message polynomial. Then the corresponding codeword \( V \) is the \( \mathbb{F}_q \)-linear span of the set \( \{ (\alpha_1, f(\alpha_1)), \ldots, (\alpha_n, f(\alpha_n)) \} \) which is an \( n \)-dimensional vector space. The ambient space \( W \) is equal to \( \langle A \rangle \otimes \mathbb{F}_{q^m}^n \) which is an \( (n \times n) \)-dimensional vector space over \( \mathbb{F}_q \). The codewords of Koetter-Kschischang code, which we simply call KK code, are in fact \( n \) dimensional subspaces of the ambient space \( W \). Each element of \( W \) is represented as a vector \( (x, y) \) where \( x \) belongs to the span of \( \alpha_i \)'s and \( y \) is an element of \( \mathbb{F}_{q^m}^n \).

At the decoder, a nonzero bivariate linearized polynomial \( Q(X, Y) \) of the form

\[
    Q(X, Y) = Q_0(X) + Q_1(Y),
\]

is constructed, where \( Q_0 \) and \( Q_1 \) are subject to some degree constraints such that \( Q(x_i, y_i) = 0 \) for all the basis elements of the received subspace. Then the equation \( Q(X, f(X)) = 0 \) is solved to recover the message polynomial. It is proved in [9] that if not too many errors and erasures happen, then \( f_u(X) \) is the unique solution to this equation. Koetter and Kschischang give the normalized decoding radius of this scheme as

\[
    R^* = 1 - \frac{k}{n} \approx 1 - \left( 1 + \frac{n}{m} \right)^{R^*} = 1 - R^*
\]

The main obstacle in the list-decoding of KK codes is that the ring of linearized polynomials is non-commutative. Because of that, an equation of certain degree over the ring of linearized polynomials may have exponentially many roots, while one has to guarantee a bounded list-size at the output of the decoder.
In order to enable list-decoding, we modified the KK construction in many important ways. Our work in [12] and [13] basically leads to a new construction of subspace codes which is list-decodable.

Next, we turn to describe the encoding and decoding of this construction of subspace codes [12], [13]. Recall from [11, Ch. 4.9] that any finite extension $\mathbb{F}_q$ of $\mathbb{F}_q$ contains a primitive element $\gamma$ such that $\gamma, \gamma^q, \ldots, \gamma^{q^m-1}$ forms a basis for $\mathbb{F}_q$ as a vector space over $\mathbb{F}_n$. This is called a normal basis for $\mathbb{F}_q$. Fix a finite field $\mathbb{F}_q$ and let $n$ divides $q-1$. Then the equation $x^n - 1 = 0$ has $n$ distinct solutions in $\mathbb{F}_q$. Let $e_1 = 1, e_2, e_3, \ldots, e_n$ be these solutions. Let $\mathcal{F} = G(\mathbb{F}^{q^m})$ and $\gamma$ be a generator of a normal basis for $\mathcal{F}$. Then define

$$\alpha_i = \gamma + e_i^{-1} \gamma^m + e_i^{-2} \gamma^{2m} + \cdots + e_i^{-(n-1)} \gamma^{(n-1)m} \quad (4)$$

for $i = 1, 2, \ldots, n$. For a given message polynomial $f_u(X)$, our encoder constructs the vectors $v_i$'s as follows:

$$v_i = (\alpha_i, f_u(\alpha_i), f_u^{\otimes 2}(\alpha_i), \ldots, f_u^{\otimes L}(\alpha_i))$$

for $i = 1, 2, \ldots, n$. Then it outputs the $n$-dimensional vector space spanned by $v_1, v_2, \ldots, v_n$. In this construction, the ambient space $W$ has dimension equal to $n + nmL$ and each element in $W$ is represented as a vector with $L + 1$ coordinates such as $(x, y_1, y_2, \ldots, y_L)$, where $x$ belongs to the vector space spanned by $e_1, e_2, \ldots, e_n$ and $y_i \in \mathbb{F}^{q^m}$, for $i = 1, 2, \ldots, L$. The decoding algorithm consists of three steps. In the first step, it computes the interpolation points. In the second step, a multivariate linearized polynomial $Q(X, Y_1, Y_2, \ldots, Y_L)$ of the form

$$Q_0(X) + Q_1(Y_1) + Q_2(Y_2) + \cdots + Q_L(Y_L)$$

is constructed, where each $Q_i$ is subject to a degree constraint, such that $Q_i(x, y_1, y_2, \ldots, y_L) = 0$ for all the interpolation points $(x, y_1, y_2, \ldots, y_L)$. Then in the factorization step, we compute all the roots $f(X) \in \mathcal{L}_q[X]$, with degree at most $k - 1$, of the equation:

$$Q(X, f(X), \ldots, f^{\otimes L}(X)) = 0$$

To solve this equation efficiently, we propose a linearized version of Roth-Ruckenstein algorithm which was designed to solve equations over the ring of polynomials [17]. We also show in [13] that there are at most $L$ solutions for $f(X) \in \mathcal{L}_q[X]$. Each solution corresponds to one possible output message.

We prove in [13] that the normalized decoding radius of this list-decoding algorithm in terms of list size $L$ and packet rate $R^*$ is given by

$$L - \frac{1}{2} L(L + 1) R^*$$

(5)

We further improve this result by introducing multiplicities for the interpolation polynomial in [14]. First, we establish the notion of multiplicity for linearized polynomials in this context. Then by enforcing multiple roots for the interpolation polynomial we achieve a better decoding radius. We are also able to list-decode at higher rates. For every positive integers $L$ and $c$, our list-$L$ decoder with multiplicity $c$ guarantees successful recovery of the message subspace provided that the normalized dimension of the error is at most

$$\frac{2(L + 1)}{c + 1} - 1 - \frac{L(L + 1)}{c(c + 1)} R^*$$

This improves the normalized decoding radius upon the previous results, given in (3) and (5), for a wide range of rates. The parameter $c$ is independent of the code construction and can be chosen at the decoder in such a way that the decoding radius is maximized. As $L$ tends to infinity, the decoding radius of our construction with appropriate choice of $c$ approaches $\frac{1}{R^*} - 1$.

B. Rank-Metric Codes

Rank-metric codes are suitable for the coherent system of network coding, where the network topology and the underlying network code are known to both the transmitter and the receiver [18]. In [18], Silva et al. also show that subspace codes and rank-metric are closely related. Indeed, there is an injective mapping between rank-metric codes and subspace codes through a lifting operation.

Let $\mathbb{F}_{q}^{n \times m}$ denote the set of all $n \times m$ matrices over $\mathbb{F}_q$. For any $X \in \mathbb{F}_{q}^{n \times m}$, let $\langle X \rangle$ denote the row space of the matrix $X$. A rank-metric code is just a subset of $\mathbb{F}_{q}^{n \times m}$ which is called an array code in [16]. The distance between $X, Y \in \mathbb{F}_{q}^{n \times m}$ is defined as rank($X - Y$). We define the rate $R$ of a rank-metric code $C \subseteq \mathbb{F}_{q}^{n \times m}$ as follows:

$$R \equiv \frac{\log_q|C|}{nm}$$

The minimum (rank) distance of $C$ is the minimum distance between distinct elements of $C$. The Singleton bound is established in the context of rank-metric codes by Gabidulin in [3]. It states that the minimum distance of a code $C$ with rate $R$, normalized by the number of rows $n$, is at most $1 - R$. A rank-metric code that meets the Singleton bound on the minimum distance is called a maximum rank distance (MRD) code. Gabidulin codes are a class of MRD codes proposed in [3].

A Gabidulin code in $\mathbb{F}_{q}^{n \times m}$ is indeed a linear $(n, k)$ code over $\mathbb{F}_{q^m}$ whose generator matrix $G$ has the following form:

$$
\begin{bmatrix}
\alpha_1^{[0]} & \alpha_2^{[0]} & \cdots & \alpha_n^{[0]} \\
\alpha_1^{[1]} & \alpha_2^{[1]} & \cdots & \alpha_n^{[1]} \\
\vdots & \vdots & \ddots & \vdots \\
\alpha_1^{[k-1]} & \alpha_2^{[k-1]} & \cdots & \alpha_n^{[k-1]}
\end{bmatrix}
$$

where the elements $\alpha_1, \alpha_2, \ldots, \alpha_n \in \mathbb{F}_{q^m}$ are linearly independent over $\mathbb{F}_q$. Each codeword is a column vector of length $n$ over $\mathbb{F}_{q^m}$ which can be also regarded as a matrix in $\mathbb{F}_{q}^{n \times m}$. Note that the condition $n \leq m$ is required. The rate of the code is $R = \frac{k}{n}$. The minimum rank distance of a Gabidulin code is $d = n - k + 1$ which satisfies the Singleton bound in the rank metric [3]. The minimum rank distance can be normalized to $1 - R$. The unique decoding radius bound then becomes equal to $(1 - R)/2$. A decoding algorithm which can correct errors, as long as the rank of error is less than $(d - 1)/2$, is proposed in
Suppose that the input to the Gabidulin encoder is a message vector \( u = (u_0, u_1 \ldots u_{k-1}) \) which consists of \( k \) message symbols in \( \mathbb{F}_{q^m} \). Let \( f_u(X) \) denote the corresponding linearized message polynomial \( \sum_{i=0}^{k-1} u_iX^i \). Then the corresponding codeword \( V = (uG)^T \) is indeed equal to
\[
\begin{bmatrix}
    f_u(\alpha_1) & f_u(\alpha_2) & \cdots & f_u(\alpha_n)
\end{bmatrix}^T
\]
which can be also regarded as a matrix in \( \mathbb{F}_{q}^{n \times m} \).

Now, the close relation between Koetter-Kschischang construction of subspace codes and Gabidulin codes becomes clear. Intuitively KK codes can be thought as a modification of Gabidulin codes where there is no ordering for the coordinates \( f_u(\alpha_i) \)'s. Each \( \alpha_i \) is appended to the corresponding \( f_u(\alpha_i) \), as a vector in \( n \)-dimensional vector space spanned by all \( \alpha_i \)'s, in order to keep track of evaluation points of the linearized polynomial \( f_u(X) \). More rigorously, the lifting mapping, defined in [18], translates Gabidulin codes into KK codes.

### III. NEW SUBSPACE CODES AND ALGEBRAIC LIST-DECODING THEREOF

In this section, we present a new construction of subspace codes and a list-decoding algorithm capable of correcting both errors and erasures. Our results in this section are motivated by the recent work of Vadhan [20, Ch. p] and Guruswami [4]. Then we establish the correctness of our algorithm and compute the error correction capability of the proposed construction.

#### A. Code Construction and List-decoding Algorithm

The following parameters of the construction are fixed: the finite field \( \mathbb{F}_q \) and an extension \( \mathbb{F}_{q^m} \), the number of information symbols \( k \), the dimension of code \( n \) and the parameter \( s \) which is related to the list size. Let \( k \leq n \leq m \). A set \( A = \{ \alpha_1, \alpha_2, \ldots, \alpha_n \} \) of linearly independent elements of \( \mathbb{F}_{q^m} \) is also fixed. In this construction, the ambient space \( W \) is an \( n + sm \)-dimensional vector space over \( \mathbb{F}_q \). Let \( \gamma \) be an element of \( \mathbb{F}_{q^m} \) which is not contained in any subfield of \( \mathbb{F}_{q^m} \), i.e. \( \gamma, \gamma^n, \ldots, \gamma^{n-1} \) are all distinct.

**Encoding Algorithm:**

Formally, the encoder is a function \( E: \mathbb{F}_{q^m}^k \to \mathcal{G}(W, n) \). It accepts as input a message \( u = (u_0, u_1 \ldots, u_{k-1}) \in \mathbb{F}_{q^m}^k \). The corresponding message polynomial is \( f_u(X) = \sum_{i=0}^{k-1} u_iX^i \). Then the corresponding codeword \( V \) is the \( \mathbb{F}_q \)-linear span of the set \( \{ f_u(\alpha_i), f(\gamma\alpha_i), \ldots, f(\gamma^{n-1}\alpha_i) \} \) : \( i \in [n] \).

Notice that, KK code is a special case of this for \( s = 1 \). Since \( \alpha_i \)'s are linearly independent, each codeword is an \( n \)-dimensional vector space which is a subspace of
\[
W = \langle \alpha_1, \alpha_2, \ldots, \alpha_n \rangle \oplus \mathbb{F}_{q^m}^n \oplus \cdots \oplus \mathbb{F}_{q^m}^n \quad \text{(6)}
\]
The dimension of \( W \) is equal to \( n + sm \), as mentioned before. Each vector in \( W \) is represented as a vector with \( s + 1 \) coordinates such as \( (x, y_1, \ldots, y_s) \), where \( x \) is an element of the vector space spanned by \( \alpha_1, \alpha_2, \ldots, \alpha_n \) and all \( y_i \)'s belong to \( \mathbb{F}_{q^m} \).

Now, we turn to explain the list-decoding algorithm. Suppose that \( V \) is transmitted and a subspace \( U \) of \( W \) of dimension \( r \) is received. We need another parameter \( d \) at the decoder which is computed as follows:
\[
d = \left\lceil \frac{r + s(k - 1) + 1}{s + 1} \right\rceil \quad \text{(7)}
\]
As we will see, \( d \) is chosen in such a way that existence of the interpolation polynomial is guaranteed at the decoder.

**List-decoding Algorithm:**

The decoder accepts as input a vector space \( U \) which is a subspace of \( W \). It then outputs a list of size at most \( q^{s(n-1)} \) of vectors in \( \mathbb{F}_{q^m}^k \) in three steps:

1. **Computing the interpolation points:** Find a basis \( (x_1, y_{11}, y_{12}, \ldots, y_{1s}), i = 1, 2, \ldots, r \), for \( U \). This is the set of interpolation points.
2. **Interpolation:** Construct a nonzero multivariate linearized polynomial \( Q(X, Y_1, Y_2, \ldots, Y_s) \) of the form
   \[
   Q(X, Y) = Q_0(X) + Q_1(Y) + Q_2(Y_2) + \cdots + Q_s(Y_s)
   \]
   where \( Q_i \)'s are linearized polynomials over \( \mathbb{F}_{q^m} \), \( Q_0 \) has \( q \)-degree at most \( d - 1 \) and \( Q_i \) has \( q \)-degree at most \( d - k \), for \( i = 1, 2, \ldots, s \), subject to the constraint that
   \[
   Q(x_i, y_{i1}, y_{i2}, \ldots, y_{is}) = 0 \quad \text{for } i = 1, 2, \ldots, r \quad \text{(8)}
   \]
3. **Message recovery:** Find all polynomials \( f(X) \in \mathcal{L}_{q^m}(X) \) of degree at most \( k - 1 \) that satisfy the following equation
   \[
   Q(X, f(\gamma X), f(\gamma^2 X), \ldots, f(\gamma^{s-1} X)) = 0
   \]
   The decoder outputs coefficients of each solution \( f(X) \) as a vector of length \( k \).

The first step of this list-decoding algorithm can be done using elementary linear algebraic operations. The second step is basically solving a linear system of equations. There are several ways for doing that. The most straightforward way is the Gaussian elimination method. However, this method does not take advantage of the certain structure of this system of equations and therefore, it is not efficient. Efficient interpolation algorithms in the ring of linearized polynomials are presented in [21]. In this case, the complexity of corresponding interpolation algorithm is given as \( O(n^2 s^3) \) field operations over \( \mathbb{F}_{q^m} \). The parameter \( s \) is in fact a design parameter and can be regarded as a constant. Indeed, the interpolation step is quadratic in terms of \( n \). In the next subsection, we explain how the message recovery step can be done using a linear algebraic method. The complexity of the message recovery step is also quadratic. Hence, the total complexity of our algorithm is quadratic in terms of \( n \), the dimension of the code.

**B. Recovering the Message Polynomial**

As discussed in the foregoing section, in the last step of the list-decoding algorithm we need to find all polynomials \( f(X) \in \mathcal{L}_{q^m}(X) \) of degree at most \( k - 1 \) that satisfy
\[
Q_0(X) + Q_1(f(X)) + Q_2(f(\gamma X)) + \cdots + Q_s(f(\gamma^{s-1} X)) = 0
\]
(9)
Remark. Suppose that \( f, g \in \mathbb{L}_q^m[X] \) are two solutions to the equation (9). Since \( Q_i \)'s are linearized polynomials, for any \( \alpha \in \mathbb{F}_q \), \( \alpha f + (1 - \alpha)g \) is also a solution to (9). Therefore, the set of solutions, which can be regarded as vectors of length \( k \) over \( \mathbb{F}_q^m \), forms an affine subspace of \( \mathbb{F}_q^m \) as a vector space over \( \mathbb{F}_q \).

In the next lemma, we establish an upper bound on the number of solutions to (9). The proof of lemma also clarifies how the affine space of solutions can be computed with quadratic complexity.

Lemma 1. The dimension of the affine space of solutions \( f(X) \in \mathbb{L}_q^m[X], \) of degree at most \( k - 1 \), to (9) is at most \( m(s - 1) \).

Proof. For \( i = 0, 1, 2, \ldots, s \), let
\[
Q_i(X) = \sum_{j=0}^{d} q_{i,j} X^{q_{i,j}}
\]
If \( q_{i,0} = 0 \) for \( i = 0, 1, 2, \ldots, s \), then we replace \( Q_i \) with \( Q_i' \) where \( Q_i'(X) = Q_i(X)^{q_i} \), in (9) and the space of solutions remains unchanged. Therefore, one can assume that at least one \( q_{i^*,0} \) is non-zero for some \( i^* \in \{0, 1, 2, \ldots, s\} \). Furthermore, if \( q_{1,0}, q_{2,0}, \ldots, q_{s,0} \) are all zero, then so is \( q_{0,0} \), otherwise there is no solution to (9). Thus, we can take \( i^* \) from the set \( \{1, 2, \ldots, s\} \).

Let us define the linearized polynomial \( P(X) \) as
\[
P(X) = Q_0(X) + \sum_{i=1}^{s} Q_i(f(\gamma^{i-1}X))
\]
and the polynomial \( A(X) \) as
\[
A(X) = q_{1,0} + q_{2,0}X + \cdots + q_{s,0}X^{s-1}
\]
Then the coefficient of \( X^{s-1} \) in \( P(X) \), for \( i = 0, 1, \ldots, k - 1 \), is equal to
\[
q_{0,i} + u_i(q_{1,0} + q_{2,0}\gamma^{q_i} + \cdots + q_{s,0}\gamma^{(s-1)q_i})
+ u_{i-1}^{q_i}(q_{1,1} + q_{2,1}\gamma^{q_i} + \cdots + q_{s,1}\gamma^{(s-1)q_i})
+ \cdots + u_{0}^{q_i}(q_{1,i} + q_{2,i}\gamma^{q_i} + \cdots + q_{s,i}\gamma^{(s-1)q_i})
\]
which can be simply expressed as
\[
q_{0,i} + A(\gamma^{q_i})u_i + \sum_{j=0}^{i-1} a_j^{(i)} u_j^{q_i-j} \quad \text{(10)}
\]
for some elements \( a_j^{(i)} \in \mathbb{F}_q^m \). Now, suppose we want to find all possible solutions for \( f(X) \) in (9). Then all the coefficients of \( P(X) \) have to be equal to zero. In particular, for the coefficient of \( X \) in \( P(X) \):
\[
A(\gamma)u_0 + q_{0,0} = 0
\]
If \( A(\gamma) \) is non-zero, then \( u_0 = -\frac{q_{0,0}}{A(\gamma)} \). If \( A(\gamma) \) is zero but \( q_{0,0} \) is not zero, then there is no solution for \( u_0 \) and consequently for \( f(X) \). If both \( A(\gamma) \) and \( q_{0,0} \) are zero, then we can set \( u_0 \) to any element of \( \mathbb{F}_q^m \). Then we find the solutions to \( u_i \)'s iteratively. For each \( i \), suppose that \( u_0, u_1, \ldots, u_{i-1} \) are already computed. If \( A(\gamma^{q_i}) \) is non-zero, then \( u_i \) can be uniquely determined by (10). Otherwise, we take all the elements of \( \mathbb{F}_q^m \) as possible solutions to \( u_i \) and keep going for each of them separately. Notice that \( A(X) \) is a non-zero polynomial of degree \( s - 1 \) and \( \gamma, \gamma^q, \ldots, \gamma^{q^{s-1}} \) are all distinct elements of \( \mathbb{F}_q^m \). Therefore, \( A(\gamma^q) \) is equal to zero for at most \( s - 1 \) possible values of \( i \). This implies that the total number of solutions for \( f(X) \) to (9) is at most \( q^{m(s-1)} \) which proves the lemma. □

Corollary 2. The affine space of solutions to (9) can be computed with quadratic complexity in terms of dimension \( n \).

C. Correctness of the Algorithm and Code Parameters

In this subsection, we first establish the correctness of our list-decoding algorithm. Then we compute the corresponding decoding radius.

Lemma 3. The particular choice of \( d \) in (7) guarantees existence of a non-zero solution for interpolation polynomial \( Q \) that satisfies (8).

Proof. (6) defines a homogeneous system of \( r \) linear equations. The number of unknown coefficients is equal to
\[
d + (d - k + 1)s = d(s + 1) - s(k - 1)
\]
A non-zero solution for this homogeneous system of linear equations is guaranteed if the number of equations is strictly less than the number of variables, i.e.
\[
r \leq d(s + 1) - s(k - 1) - 1 \iff d \geq \frac{r + s(k - 1) - 1}{s + 1}
\]
This is guaranteed by the choice of \( d \) in (7). □

We form the following linearized polynomial \( E(X) \) wherein \( f_u(X) \) is the message polynomial and \( Q(X, Y_1, \ldots, Y_t) \) is the interpolation polynomial provided by the list-decoding algorithm.
\[
E(X) = Q(X, f_u(X), f_u(\gamma X), \ldots, f_u(\gamma^{s-1}X)) = Q_0(X) + \sum_{i=1}^{s} Q_i \otimes f_u(\gamma^{i-1}X)
\]
Let \( \rho \) and \( t \) denote the number of erasures and errors in the received subspace \( U \), respectively. Hence, the dimension of \( U \) is in fact equal to \( r = n - \rho + t \).

Lemma 4. The linearized polynomial \( E(X) \) has at least \( n - \rho \) linearly independent roots in \( \mathbb{F}_q^m \).

Proof. Let \( U' \) denote the intersection of the transmitted codeword \( V \) and the received subspace \( U \). Then \( U' \) is a subspace of the received vector space \( U \) with dimension \( n - \rho \). Since \( Q \) is a linearized polynomial
\[
Q(x, y_1, \ldots, y_s) = 0
\]
for any \( (x, y_1, \ldots, y_s) \in U' \). On the other hand, \( (x, y_1, \ldots, y_s) \) is also an element of the transmitted codeword \( V \). Therefore,
\[
(x, y_1, \ldots, y_s) = (\beta, f_u(\beta), f_u(\gamma \beta), \ldots, f_u(\gamma^{s-1} \beta))
\]
for some \( \beta \) in the linear span of \( \alpha_1, \alpha_2, \ldots, \alpha_n \). Therefore, \( \beta \) is a root for the polynomial \( E(X) \). Hence, there are at least \( n - \rho \) linearly independent roots for \( E(X) \). □
Corollary 5. If \( d \leq n - \rho \), then the linearized polynomial \( E(X) \) is identically zero.

Proof. The \( q \)-degree of \( f_u(X) \) is at most \( k-1 \). Therefore, the \( q \)-degree of \( Q X_i \otimes f_u(\gamma^{-1} X) \) is at most
\[
d - k + k - 1 = d - 1
\]
for \( i = 1, \ldots, L \). Also, the \( q \)-degree of \( Q X_0(\gamma) \) is at most \( d - 1 \). Thus the \( q \)-degree of \( E(X) \) is at most \( d - 1 \). On the other hand, \( E(X) \) has at least \( n - \rho \) linearly independent roots by Lemma 3. Therefore, \( E(X) \) must be the all zero polynomial. ■

Theorem 6. The output of our list-decoding algorithm is a list of size at most \( q^{m(s-1)} \) which includes the transmitted message \( u \) provided that
\[
s \rho + t < ns - s(k-1)
\]
Proof. The existence of non-zero interpolation polynomial \( Q \) that satisfies (8) is guaranteed by Lemma 3. Then by Corollary 5, \( E(X) \) is the all zero polynomial provided that
\[
\left[ \frac{r + s(k-1) + 1}{s + 1} \right] \leq (n - \rho)
\]
where we have used the expression for \( d \) from (7). We plug in \( r = n - \rho + t \) into (12). Then observe that (12) is in fact equivalent to
\[
s \rho + t < ns - s(k-1)
\]
Thus this condition on the number of errors and erasures implies that \( E(X) \) is identically zero. Therefore, the message polynomial \( f_u(X) \) is a solution to (9). There are at most \( q^{m(s-1)} \) solutions to (9) by Lemma 1. Therefore, the list size is at most \( q^{m(s-1)} \). ■

Now, we turn to the parameters of the proposed construction. The ambient space \( W \) is given in (6) which has dimension equal to \( n + sm \). The symbol rate \( R \) and the packet rate \( R^* \) of the code can be computed as defined in (1) and (2):
\[
R = \frac{\log_q (\text{size of the code})}{n(\dim(W))} = \frac{km}{n(n + sm)}
\]
\[
R^* = \frac{\log_q (\text{size of the code})}{n} = \frac{k}{n}
\]
The normalized decoding radius, in which erasures have weight \( s \), is given by Theorem 5 as
\[
s - s(k-1) \approx s - s^2(1 + \frac{n}{ms})R
\]
In the regime where \( n \) is much smaller than \( ms \), the error decoding radius can be approximated as \( s - s^2R \).

The normalized decoding radius in terms of the packet rate \( R^* \) can be approximated as \( s(1 - R^*) \). It implies that, for any packet rate \( R^* \), we basically achieve any decoding radius by letting the list size to be large enough.

IV. LIST-DECODING OF GABIDULIN CODES

In this section, we first introduce a folded version of Gabidulin codes. Then, we propose a list-decoding algorithm which provides decoding radius up to the Singleton bound \( 1 - R \), the best possible trade-off between the rate and error-correction radius.

Let \( \gamma \) be a primitive element of \( \mathbb{F}_{q^n} \). Let \( C \) denote the Gabidulin code constructed with parameters \( \alpha_i = \gamma^{(i-1)} \) as discussed in Section 3.3. Let also \( h \) be a positive integer that divides \( n \) and let \( q = n/h \).

Definition. (Folded Gabidulin Code) The \( h \)-folded version of Gabidulin code \( C \) is a code whose codewords are elements of \( \mathbb{F}_q^{g \times hm} \). The encoding of a message polynomial \( f_u(X) \) of \( q \)-degree at most \( k - 1 \) has as its \( i \)-th row, for \( 0 \leq i < g \), the \( h \)-tuple \( \{ f_u(\gamma^{ih}), f_u(\gamma^{ih+1}), \ldots, f_u(\gamma^{(i+1)h-1}) \} \), which can be regarded as an element in \( \mathbb{F}_q^{hm} \).

Notice that folding does not change the rate. The rate of folded version of code \( C \) is equal to the rate of \( C \) which is equal to \( k/n \).

Before going into the details of list-decoding algorithm, we would like to clarify the difference between the notion of "error" in subspace codes and rank-metric codes. Suppose that a codeword \( X \) in code \( C \) is transmitted and a word \( Y \) received i.e. \( \text{rank}(X - Y) = t \). Now consider \( X \) and \( Y \) in the context of subspace codes. Then \( Y \) is corrupted with \( t \) errors and \( t \) erasures with respect to \( X \). In fact, in rank-metric codes, there is no notion of "erasure" and each error is corresponding to one error and one erasure in the context of subspace codes.

For \( 0 \leq i \leq g \) and \( 0 \leq j \leq m \), let \( y_{i,j} \in \mathbb{F}_q \) denote the \((i,j)\)-th coordinate of received word \( Y \) regarded as a matrix in \( \mathbb{F}_q^{g \times hm} \). Let \( s \) be a positive integer less than or equal to \( h \). We propose a decoding algorithm based on interpolating an \( s + 1 \)-variate linearized polynomial \( Q(X, Y_1, \ldots, Y_s) \). The \( q \)-degree of \( Q \) is characterized in terms of interpolating polynomial.

List-decoding algorithm of folded Gabidulin codes

1) Interpolation: Construct a nonzero multivariate linearized polynomial \( Q(X, Y_1, Y_2, \ldots, Y_s) \) of the form
\[
Q(X, Y) = Q_0(X) + Q_1(Y) + Q_2(Y_2) + \cdots + Q_s(Y_s)
\]
where \( Q_i \)'s are linearized polynomials over \( \mathbb{F}_q \), \( Q_0 \) has \( q \)-degree at most \( d - 1 \) and the \( q \)-degree of all other \( Q_i \)'s is at most \( d - k \) subject to the constraint that
\[
Q(\gamma^{ih+j}, y_{i,j}, y_{i,j+1}, \ldots, y_{i,j+s-1}) = 0
\]
for \( i = 0, 1, \ldots, g-1 \) and \( j = 0, 1, \ldots, h-s \).

2) Message recovery: Find all the solutions \( f(X) \in \mathbb{L}_q[X] \) to the following equation:
\[
Q(X, f(\gamma X), f(\gamma^2 X), \ldots, f(\gamma^{s-1} X)) = 0
\]
The decoder outputs coefficients of each solution $f(X)$ as a vector of length $k$.

The interpolation step is very similar to the interpolation step of the list-decoding algorithm discussed in Section II-A. It can be done using either the straightforward Gaussian elimination method or an efficient interpolation algorithm in the ring of linearized polynomials as presented in [21], similar to the algorithm in Section II-A. The message recovery step is exactly similar to that of list-decoding algorithm in Section II-A. It can be also done as discussed in Section II-B. The total complexity of our list-decoding algorithm is then quadratic in terms of dimension $n$.

Next, we establish correctness of the proposed list-decoding algorithm and compute the decoding radius of the code.

**Lemma 7.** The particular choice of $d$ in (14) guarantees existence of a non-zero solution for interpolation polynomial $Q$ that satisfies (15).

**Proof.** (15) is in fact a homogeneous system of $g(h - s + 1)$ linear equations. The number of unknown coefficients is given by

$$d + (d - k + 1)s = d(s + 1) - s(k - 1)$$

If the number of equations is strictly less than the number of variables in a homogeneous system of linear equations, then a non-zero solution is guaranteed to exist. i.e.

$$g(h - s + 1) < d(s + 1) - s(k - 1) - 1 \iff \begin{equation}
d \geq \frac{g(h - s + 1) + s(k - 1) + 1}{s + 1}
\end{equation}$$

This is guaranteed by the choice of $d$ in (14). $\blacksquare$

Let $U \in \mathbb{F}_q^{R \times h m}$ denote the codeword corresponding to the message polynomial $f_0(X)$. Then $\langle U \rangle \cap \langle Y \rangle$, the intersection of the row spaces of matrices $U$ and $Y$, has dimension $g - t$, where $t$ is the rank of error. We also define the linearized polynomial $E(X)$ as follows:

$$E(X) = Q(X, f_0(X), f_0(gX), \ldots, f_0(g^{s - 1}X)) = Q_0(X) + \sum_{i=1}^{s} Q_i \otimes f_0(g^{i-1}X)$$

**Lemma 8.** There are at least $(g - t)(h - s + 1)$ linearly independent roots in $\mathbb{F}_{q^m}$ for the linearized polynomial $E(X)$.

**Proof.** Notice that any element in the row space of $U$ can be represented as

$$(f_0(\beta), f_0(g \beta), \ldots, f_0(g^{h-1} \beta))$$

for some $\beta \in \mathbb{F}_{q^m}$. Now consider a basis for $\langle U \rangle \cap \langle Y \rangle$. Indeed, the basis can be represented as

$$\{(f_0(\beta_i), f_0(g \beta_i), \ldots, f_0(g^{h-1} \beta_i)) : i = 1, 2, \ldots, n-t \}$$

where $\beta_1, \ldots, \beta_{g-t}$ are $g - t$ linearly independent elements of $\mathbb{F}_{q^m}$. In fact, they are taken from the subspace spanned by $1, \gamma, \ldots, h^{(g-1)}$. Then linearity of the interpolation $Q$ and (15) together imply that

$$Q(\gamma^j \beta_1, f_0(\gamma^j \beta_1), f_0(\gamma^j+1 \beta_1), \ldots, f_0(\gamma^j+s-1 \beta_1)) = 0$$

for $i = 1, 2, \ldots, g - t$ and $j = 0, 1, \ldots, h - s$. It is indeed equivalent to $\gamma^j \beta_1$ being a root for $E(X)$. We claim that $\gamma^j \beta_i$, for $i = 1, 2, \ldots, g - t$ and $j = 0, 1, \ldots, h - s$ are all linearly independent elements of $\mathbb{F}_{q^m}$. Let $P_j$ denote the subspace spanned by $\gamma^j, \gamma^j + h, \ldots, \gamma^j + h^{(g-1)}$. Since $1, \gamma, \ldots, \gamma^{n-1}$ are all linearly independent, $P_j$‘s are all disjoint. Also, $\gamma^j \beta_i$, for $i = 1, 2, \ldots, g - t$, are $g - t$ linearly independent elements of $P_j$. This completes the proof of claim. Therefore, $\gamma^j \beta_i$, for $i = 1, 2, \ldots, g - t$ and $j = 0, 1, \ldots, h - s$, are $(g - t)(h - s + 1)$ linearly independent roots for $E(X)$. $\blacksquare$

**Corollary 9.** If $d \leq (g - t)(h - s + 1)$, then $E(X)$ is identically equal to zero.

**Proof.** The proof is very similar to the proof of Corollary 5. The $q$-degree of $E(X)$ is at most $d - 1$ by the same argument. $E(X)$ has at least $(g - t)(h - s + 1)$ linearly independent roots by Lemma 8. Thus, $E(X)$ must be the all zero polynomial. $\blacksquare$

**Theorem 10.** If the number of errors, $t$, is bounded as

$$t < \frac{gs}{s + 1} \left(1 - \frac{h}{h - s + 1}R\right)$$

Then the proposed list-decoding algorithm of folded Gabidulin codes is correct i.e. it outputs a list of size at most $q^{m(s-1)}$ which includes the transmitted message $u$.

**Proof.** The interpolation polynomial $Q$ that satisfies (15) is guaranteed to exist by Lemma 7.

$$\left|\frac{g(h - s + 1) + s(k - 1) + 1}{s + 1}\right| \leq (g - t)(h - s + 1) \quad (18)$$

then by Corollary 9 and using the expression for $d$ from (14). $E(X)$ is the all zero polynomial. (18) is equivalent to

$$g(h - s + 1) + s(k - 1) < (g - t)(h - s + 1)(s + 1)$$

which can be simplified to (17) by using the approximation

$$R \approx \frac{k - 1}{n}$$

Therefore, the message polynomial $f_0(X)$ is a solution to (16).

There are at most $q^{m(s-1)}$ solutions to (16) by Lemma 11. Therefore, the list size is at most $q^{m(s-1)}$. $\blacksquare$

**Corollary 11.** The normalized decoding radius of folded Gabidulin code using the proposed list-decoding algorithm is equal to

$$\frac{s}{s + 1} \left(1 - \frac{h}{h - s + 1}R\right)$$

If we let both $s$ and $h$ grow large while $s$ is much smaller that $h$, we get decoding radius arbitrary close to $1 - R$. Notice that $1 - R$ is indeed equal to the normalized minimum rank distance of the code. This means that we are able to achieve the ultimate error-correction radius for rank-metric codes. This result is stated in the following theorem.

**Theorem 12.** For every $\varepsilon > 0$ and $0 < R < 1$, there is a family of folded Gabidulin codes with rate $R$ that can be list-decoded up to normalized number of errors $1 - R - \varepsilon$. The size of output list is at most $Q^{O(1/\varepsilon)}$, where $Q$ is the size of the field that message symbols are chosen from.

**Proof.** Given $R$ and $\varepsilon$, we can apply the results of Theorem 10 and Corollary 11 with the choice $s = 1/2\varepsilon$ and $h = 1/4\varepsilon^2$. $\blacksquare$
REFERENCES

[1] R. Ahlswede, N. Cai, Sh. Y. R. Li and R. W. Yeung, “Network information flow,” IEEE Transactions on Information Theory, vol. 46, pp. 1204–1216, Jul. 2000.

[2] P. Delsarte, “Bilinear forms over a finite field, with applications to coding theory,” J. Comb. Theory. Ser. A, vol. 25, pp. 226-241, 1978.

[3] E. M. Gabidulin, “Theory of codes with maximum rank distance,” Probl. Inf. Transm., vol. 21, no. 1, pp. 1-12, 1985.

[4] V. Guruswami, “Linear-algebraic list decoding of folded Reed-Solomon codes,” Proc. of the 26th IEEE Conference on Computational Complexity (CCC), pp. 77-85, 2011.

[5] V. Guruswami and A. Rudra, “Explicit codes achieving list decoding capacity: Error-correction up to the Singleton bound.” IEEE Transactions on Information Theory, vol. 54, pp.135150, Jan. 2008.

[6] V. Guruswami and M. Sudan, “Improved decoding of Reed-Solomon and algebraic-geometric codes,” IEEE Trans. on Inf. Theory, vol.45, pp.1757–1767, Sept. 1999.

[7] T. Ho, R. Koetter, M. Medard, D. Karger and M. Effros, “The benefits of coding over routing in a randomized setting,” in Proc. 2003 IEEE Int. Symp. on Inf. Theory, p. 442 Yokohama, Japan, June 29-July 4 2003.

[8] T. Ho, M. Medard, R. Koetter, D. Karger, M. Effros, J. Shi and B. Leong, “A random linear network coding approach to multicast,” IEEE Transactions on Information Theory, vol. 52, pp. 4413–4430, Oct. 2006.

[9] R. Koetter and F.R. Kschischang, “Coding for errors and erasures in random network coding,” IEEE Transactions on Information Theory, vol. 54, pp. 3579–3591, Aug. 2008.

[10] Sh. Y. R. Li, R. W. Yeung and N. Cai, “Linear network coding,” IEEE Transactions on Information Theory, vol.49, pp. 371–381, Feb. 2003.

[11] F. J. MacWilliams and N. J. A. Sloane, “The theory of error-correcting codes.” New York: North Holland, 1977

[12] H. Mahdavifar and A. Vardy “Algebraic list-decoding on the operator channel,” Proc. IEEE Intern. Symp. Inf. Theory, pp. 1193–1197, Austin. TX., June 2010

[13] H. Mahdavifar and A. Vardy “Algebraic list-decoding of subspace codes,” IEEE Transactions on Information Theory, accepted for publication, available at [http://arxiv.org/abs/1202.0338](http://arxiv.org/abs/1202.0338).

[14] H. Mahdavifar and A. Vardy, “Algebraic list-decoding of subspace codes with multiplicities”, Proceedings of the 49th annual Allerton Conference on Communications, Control and Computing, pp. 1430–1437, Sep. 2011.

[15] F. Parvaresh and A. Vardy, “Correcting errors beyond the Guruswami-Sudan radius in polynomial time,” Proc. of the 46th Annual IEEE Symposium on Foundations of Computer Science, pp. 285294, 2005.

[16] R. M. Roth, “Maximum-rank array codes and their application to crisscross error correction,” IEEE Transactions on Information Theory, vol. 37, no. 2, pp. 328-336, Mar. 1991.

[17] R. M. Roth, G. Ruckenstein, “Efficient decoding of Reed-Solomon codes beyond half the minimum distance,” IEEE Transactions on Information Theory, vol. 46, pp. 246–257, Jan. 2000.

[18] D. Silva, F. R. Kschischang, R. Koetter, “A rank-metric approach to error control in random network coding,” IEEE Transactions on Information Theory, vol. 54, no. 9, pp. 3951–3967, Sep. 2008.

[19] M. Sudan, “Decoding of Reed-Solomon codes beyond the error-correction bound.” J. Complexity, 12: 180–193, Mar. 1997

[20] S. Vadhan, “Pseudorandomness.” Foundations and Trends in Theoretical Computer Science (FnT-TCS). NOW publishers, 2010. To appear. Draft available at [http://people.seas.harvard.edu/salu/pseudorandomness](http://people.seas.harvard.edu/salu/pseudorandomness).

[21] H. Xie, Z. Yan, B. W. Suter, “General linearized polynomial interpolation and its applications,” 2011 Int. Symp. on Network Coding, Beijing, China, July 25-27 2011.