21st International Symposium on
Mathematical Theory of Networks and Systems
July 7-11, 2014. Groningen, The Netherlands

Efficient Burst Error Correction by Hermitian Codes

Wenhui Li, Xiaozhou Wang, and Vladimir Sidorenko

***Abstract*— Hermitian codes belong to a powerful class of algebraic geometry codes, which allow to correct many independent errors. We show that Hermitian codes can correct many bursts of errors as well. Decoding of a Hermitian (N, K) code over $_{q^2}$ can be reduced to decoding of interleaved extended Reed-Solomon codes. Using this fact, we propose an efficient unique decoding algorithm correcting up to $(N-K)/(q+1)$ phased bursts of length q. Decoding failure probability is upper bounded by $1/q^2$ and exponentially decreases with the number of bursts. It is also shown that low rate Hermitian codes can correct even more bursts of errors using "power" and "mixed" decoding. Time complexity of the algorithms is $\mathcal{O}(N^{5/3})$ field operations.**

I. Introduction

Hermitian codes are one of the most studied families of algebraic geometry codes [1]. Hamming distance d of (N, K) Hermitian code over the field $_Q = _{q^2}$ is not far from the Singleton bound $d \le N - K + 1$, but the code length N can be much longer than the order Q of the field, in contrast to very popular Reed-Solomon (RS) codes. This fact makes Hermitian codes potentially very interesting for applications and a number of efficient unique decoding algorithms were suggested, correcting up to $d/2$ independent errors [2], [3] and list decoders, correcting more than $d/2$ errors [4].

In this paper we consider Hermitian codes having rate $R = K/N \ge 1/2q$. For these codes we propose an efficient algorithm, correcting phased bursts of errors of length q, later called simply bursts. This means that we decode Hermitian codes in the burst metric, which is equivalent to decoding q-folded Hermitian codes in Hamming metric. Our algorithm is based on the fact, that decoding a Hermitian code can be reduced to decoding interleaved extended Reed-Solomon (IERS) codes [5], [6].

Denote by d_B the code distance of a Hermitian code in the burst metric. We consider unique decoding algorithms correcting more than $d_B/2$ bursts. These algorithms may fail in some cases. By $P_f(t_{\max})$ we denote the failure probability of a given decoding algorithm in presence of t bursts in the channel. The function $P_f(t)$ describes performance of the decoding algorithm. Usually $P_f(t) = 0$ if $t < d_B/2$, then $P_f(t) \ll 1$ for $d_B/2 \le t \le t_{\max}$ for some integer $t_{\max}$, which is called the decoding radius, and then $P_f(t) \approx 1$ for $t > t_{\max}$. This definition of $t_{\max}$ is not precise, and it depends on what $P_f(t_{\max}) \ll 1$ means. For the proposed algorithm we will give the function $P_f(t)$. Performance of a decoding algorithm can be approximately described by $t_{\max}$ and $P_f(t_{\max})$.

Known results. It was shown by Yaghoobian and Blake [5] and by Ren [6] that every Hermitian code can be represented as concatenation of IERS outer code and trivial (q, q) inner codes. Hence decoding of a Hermitian code can be reduced to decoding IERS codes [6]. To decode IERS code, Ren [6] suggested to decode RS codes individually, where every next decoder erases positions, corrected by previous RS decoders. It was shown by examples, that this decoder can correct more than $d/2$ errors, if the eroors are in bursts. However, burst error correcting radius and decoding-failure probability were not obtained. Time complexity of the algorithm is $\mathcal{O}(N^{5/3})$ operations in $_Q$.

It is known that for decoding interleaved RS (IRS) codes *joint* decoding [7], [8] is more effective than Ren's [6]. Özbudak and Yayla [9] suggested an algorithm for joint decoding IERS codes having cubic in length complexity, and applied it to decode Hermitian codes resulting in correcting up to $t_{\max} = (N-K)/(q+1)$ bursts with failure probability $P_f(t_{\max}) < ((1-R)q/(q+1)$ and complexity $\mathcal{O}(N^3)$ field operations.

Our contribution. We propose joint decoding algorithm for interleaved *extended* RS codes having quadratic in length complexity. This result has interest itself.

Then we apply this algorithm to decode Hermitian codes resulting in correcting up to $t_{\max} = (N-K)/(q+1)$ bursts with complexity $\mathcal{O}(N^{1\frac{2}{3}})$ field operations. For the failure probability $P_f(t)$ we give an upper bound, which is at most $1/Q$ when $t = t_{\max}$, and exponentially decreases when t decreases, see Theorem 5 for details. Simulations show that the bound is precise. As a result, our algorithm has less failure probability than the one of Ren [6], and lower complexity and better bound on decoding failure probability in comparison with Özbudak and Yayla [9].

We also show that low rate Hermitian codes can correct even more bursts of errors using "power" and "mixed" decoding [10], [11].

The rest of the paper is organized as follows. In Section II we give a simple definition of Hermitian codes and phased bursts. In Section III we reduce decoding Hermitian codes to decoding IERS codes. Section IV reminds an idea of decoding a single *extended* RS code. This idea is generalized in Section VI for *interleaved extended* RS codes. The proposed algorithm is based on decoding interleaved non extended codes described in the previous Section V. In Section VII we shortly explain how power and mixed decoding can increase

The work is supported by German Research Council (Deutsche Forschungsgemeinschaft DFG) under projects Bo 867/22.

The authors are with Institute of Communications Engineering, University of Ulm, Ulm, Germany. V. Sidorenko is on leave from the Institute for Information Transmission Problems, Russian Academy of Sciences. wenhui.li, xiaozhou.wang, vladimir.sidorenko @uni-ulm.de.

the decoding radius and/or decrease the failure probability for low rate IERS codes. Finally, in Section VIII we describe and analyse the decoding algorithm for Hermitian codes.

II. Hermitian codes and burst-errors

Consider an extension field $_Q$ where $Q = q^2$ and q is a power of prime. A Hermitian curve $\mathcal{H}(q)$ over $_Q$ is defined by (see [12]), [13], [5])

$$y^q + y = x^{q+1}. \tag{1}$$

There are q^3 points $p = (x, y) \in {}^2_Q$ that satisfy (1) and hence lie on the Hermitian curve $\mathcal{H}(q)$. In order to list this points denote by α a primitive element of $_Q$ and define the elements β_i as follows

$$(\beta_1, \beta_2, \ldots, \beta_q) = \left(0, 1, \alpha^{(q+1)}, \alpha^{2(q+1)}, \ldots, \alpha^{(q-2)(q+1)}\right). \tag{2}$$

Let $(1, \gamma)$ be a solution to (1), then q^3 points $p_{i,j} = (x_j, y_{i,j})$ on the Hermitian curve $\mathcal{H}(q)$ can be written as elements of $q \times q^2$ matrix $P = (p_{i,j})$, $i = 1, \ldots, q, j = 1, \ldots, Q$ as follows [5], [6]

$$\begin{array}{ll} p_{i,j} = (\alpha^{j-1}, \gamma\alpha^{j-1} + \beta_i), & i = 1, \ldots, q, j = 1, \ldots, Q-1, \\ p_{i,q^2} = (0, \beta_i), & i = 1, \ldots, q. \end{array} \tag{3}$$

We pay attention that the x coordinate of the point $p_{i,j}$ does not depend on i, hence is denoted by x_j, and x_j runs through all Q elements of $_Q$ when $j = 1, \ldots, Q$.

To define Hermitian codes we consider bivariate (information) polynomials of the form

$$h(x, y) = f^{(1)}(x) + yf^{(2)}(x) + \cdots + y^{q-1}f^{(q)}(x), \tag{4}$$

where $f^{(i)}(x)$ is a polynomial over $_Q$ of restricted degree. Given a point $p = (x, y)$, by $h(p)$ we mean the evaluation $h(p) = h(x, y)$, and for the matrix $P = (p_{i,j})$ we agree that $h(P) = (h(p_{i,j}))$. For an integer $m \geq q^2 - 1$ we define the Hermitian code $\mathcal{H}_m$ as follows.

Definition 1 (Hermitian code): For an integer $m \geq q^2 - 1$ the Hermitian code $\mathcal{H}_m$ of length $N = q^3$ over the field $_{q^2}$ is the set of $q \times q^2$ matrices W

$$\mathcal{H}_m = \{\ W = h(P)\ \}, \tag{5}$$

where the matrix P is defined by (3), and h are all possible polynomials defined by (4) with degree constraints for $i = 1, \ldots, q$

$$\deg f^{(i)}(x) < k^{(i)} = \left\lfloor \frac{m - (i-1)(q+1)}{q} \right\rfloor + 1. \tag{6}$$

Hermitian code $\mathcal{H}_m$ is a linear (N, K) code over $_Q$. The dimension of the code is

$$K = \sum_{i=1}^{q} k^{(i)} = m - g + 1, \tag{7}$$

where

$$g = (q^2 - q)/2 \tag{8}$$

is the genus of the Hermitian curve (1). The code distance $\mathrm{dist}(\mathcal{H}_m)$ in the Hamming metric is lower bounded by the designed distance [2], [3]

$$\mathrm{dist}(\mathcal{H}_m) \geq d_H = q^3 - m = N - K + 1 - g \tag{9}$$

and is upper bounded by $\mathrm{dist}(\mathcal{H}_m) \leq q^3 - q\lfloor m/q \rfloor$ [6]. So Hermitian codes almost reach the Singleton upper bound $N - K + 1$ on the code distance, but they are longer by factor q in comparison with Reed-Solomon codes, which is an advantage in many applications.

There are effective algebraic decoding algorithms, which allow to correct up to $(d_H - 1)/2$ independent errors [2], [3]. However in this paper we are focused on correction of phased bursts of errors of length q, i.e., q-bursts. One burst-error can corrupt one column in the code matrix W, i.e., at least one element of the column is wrong.

Definition 2 (q-burst metric): The *burst weight* $w_B(V)$ of a $q \times q^2$ matrix V is the minimum number of columns that contain all non zero components of V. The *burst distance* $d_B(V, W)$ between matrices V and W of the same size is the weight of their difference $d_B(V, W) = w_B(V - W)$.

It follows from (9) and from [14] that the code distance d_B of (n, k) Hermitian code in the q-burst metric is bounded by

$$(N - K + 1 - g)/q \leq d_B \leq (N - K)/q + 1 = \widehat{d}_B. \tag{10}$$

The code $\mathcal{H}_m$ with burst distance d_B guarantees correction of $(d_B - 1)/2$ bursts of errors. In Section VIII we propose an algorithm correcting with high probability $\frac{q}{q+1}(\widehat{d}_B - 1)$ bursts of errors, which is almost twice the guaranteed correcting radius.

III. From Hermitian to Reed-Solomon codes

Let the received matrix V be obtained from a code matrix W of Hermitian code $\mathcal{H}_m$ by changing some elements in t columns of W. We can write $V = W + E$, where E has t non-zero columns and we assume that every non-zero column in E is equiprobable. For the error free case, $V = W$, from definition of the code we have for all i, j

$$v_{i,j} = f^{(1)}(x_j) + y_{i,j}f^{(2)}(x_j) + \cdots + y_{i,j}^{q-1}f^{(q)}(x_j). \tag{11}$$

For the jth column r_j, $j = 1, \ldots, Q$, of the received matrix V we get the system of equations for unknowns $f^{(i)}(x_j)$

$$\left\{ \begin{array}{rcl} f^{(1)}(x_j) + y_{1,j}f^{(2)}(x_j) + \cdots + y_{1,j}^{q-1}f^{(q)}(x_j) & = & v_{1,j} \\ f^{(1)}(x_j) + y_{2,j}f^{(2)}(x_j) + \cdots + y_{2,j}^{q-1}f^{(q)}(x_j) & = & v_{2,j} \\ \cdots & & \\ f^{(1)}(x_j) + y_{q,j}f^{(2)}(x_j) + \cdots + y_{q,j}^{q-1}f^{(q)}(x_j) & = & v_{q,j} \end{array} \right. . \tag{12}$$

It follows from (3) that for every fixed j, elements $y_{i,j}$ are different. Hence the Vandermonde matrix of the system (12) is non-singular and we can uniquely compute the unknowns $f^{(i)}(x_j)$. By solving the system (12) for each of Q columns of V we will find $f^{(i)}(x_j)$ for all i and j. Solving the system (12) with q unknowns by Gaussian elimination requires $\mathcal{O}(q^3)$ filed operations. Solving this system $Q = q^2$ times

requires $\mathcal{O}(Qq^3) = \mathcal{O}(N^{5/3})$ operations in $_Q$, since $N = q^3$.

Definition 3 (Reed-Solomon code): Given n pairwise different elements (called locators) α_i of the field $_Q$, the $\mathcal{RS}(n,k)$ code consists of the following codewords

$$\mathcal{C} = \{(f(\alpha_1), \ldots, f(\alpha_n)) : \deg f(x) < k\}, \qquad (13)$$

where $f(x)$ is a polynomial over $_Q$. The code is called extended RS code $\mathcal{ERS}(n,k)$ if $\alpha_i = 0$ for some i, i.e., if zero locator is used, otherwise the code is called classical. A classical RS code is called primitive if it has all non-zero field elements as locators and hence has length $n = Q - 1$.

Code distance of an (n,k) RS code is $n-k+1$. An extended primitive RS code has all Q field elements as locators, the length of this code we will denote by $n+1 = Q$.

Definition 4 (Interleaved RS codes): Given ℓ extended Reed–Solomon codes $\mathcal{ERS}(n+1, k^{(l)})$, $l = 1, 2, \ldots, \ell$, over $_Q$ of length $n+1$ and dimensions $k^{(l)}$, the interleaved ERS (IERS) code $\mathcal{IERS}(n+1, k^{(1)}, \ldots, k^{(\ell)})$ consists of all $q \times (n+1)$ matrices C

$$C = \begin{pmatrix} c^{(1)} \\ c^{(2)} \\ \vdots \\ c^{(\ell)} \end{pmatrix} = \begin{pmatrix} f^{(1)}(x_1) & \ldots & f^{(1)}(x_{n+1}) \\ f^{(2)}(x_1) & \ldots & f^{(2)}(x_{n+1}) \\ \vdots & \vdots & \vdots \\ f^{(\ell)}(x_1) & \ldots & f^{(\ell)}(x_{n+1}) \end{pmatrix}, \qquad (14)$$

where $c^{(l)} \in \mathcal{ERS}(n+1, k^{(l)})$ and $x_1, \ldots, x_{n+1}$ are the code locators.

Now let us return to decoding Hermitian codes. By solving Q systems (12) we will compute $f^{(i)}(x_j)$ for all i, j and obtain for $n+1 = Q$ a code matrix C (14) of the interleaved extended RS codes in the error free case.

If there was at least one error in jth column v_j of the Hermitian codeword, then solution $f^{(1)}(x_j), \ldots, f^{(q)}(x_j)$ of the system (12) will be wrong, i.e., components of jth column in matrix C will be replaced by another field elements, and we will get from C the received matrix R of the IERS code with wrong jth column. If t columns of the Hermitian matrix W were corrupted, then the same t columns of received matrix R of the IERS code will be corrupted.

If we know how to decode IERS codes then we get the following algorithm correcting burst errors in the Hermitian code. Given a received matrix V of the Hermitian code with t erroneous columns, i.e., with t bursts of errors, first we compute the matrix R of the IERS code with t erroneous columns. Second, by decoding the IERS code find a codeword C nearest to R in the burst metric and obtain information polynomials $f^{(i)}(x)$ for all i. Since information polynomials for IERS and Hermitian codes coincide, we are done.

In the next sections we show how to decode IERS codes.

IV. Decoding Extended Reed–Solomon Codes

Lemma 1: Consider an extended primitive $(n+1, k, d+1)$ RS code $\mathcal{ERS}(n+1,k)$ from Definition 3 of length $n+1 = Q$ over $_Q$, having all nonzero elements α_i, $i = 1, \ldots, n$, of $_Q$ as non-zero code locators and the last zero locator $\alpha_{n+1} = 0$. Then the following matrix H is a parity check matrix of the code

$$H = \begin{pmatrix} \alpha_1^{(d-1)} & \alpha_2^{(d-1)} & \ldots & \alpha_n^{(d-1)} & 0 \\ \vdots & \vdots & \ldots & \vdots & \vdots \\ \alpha_1^2 & \alpha_2^2 & \ldots & \alpha_n^2 & 0 \\ \alpha_1 & \alpha_2 & \ldots & \alpha_n & 0 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}, \qquad (15)$$

where $d = n - k + 1$.

Proof: Since $\alpha_{n+1}^i = 0$ for $i \neq 0$ and $\alpha_{n+1}^0 = 1$, it follows from Definition 3 that the code can be generated by the following matrix of full rank k

$$G = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ \alpha_1 & \alpha_2 & \ldots & \alpha_n & 0 \\ \vdots & \vdots & \ldots & \vdots & \vdots \\ \alpha_1^{(k-1)} & \alpha_2^{(k-1)} & \ldots & \alpha_n^{(k-1)} & 0 \end{pmatrix}. \qquad (16)$$

The statement of the lemma follows, since H has full rank $n-k+1$ and it is orthogonal to G. ■

By $\hat{\mathcal{C}}$ denote $(n, k-1, d+1)$ RS code obtained by *shortening* the code $\mathcal{C}$ in the last position. The shortened code $\hat{\mathcal{C}}$ is defined by the parity check matrix

$$\hat{H} = \begin{pmatrix} \alpha_1^{(d-1)} & \alpha_2^{(d-1)} & \ldots & \alpha_n^{(d-1)} \\ \vdots & \vdots & \ldots & \vdots \\ \alpha_1 & \alpha_2 & \ldots & \alpha_n \\ 1 & 1 & 1 & 1 \end{pmatrix}, \qquad (17)$$

obtained from H by deleting the last column. The code $\hat{\mathcal{C}}$ of length $Q-1$ has nonzero locators only and hence is a primitive RS code.

By $\tilde{\mathcal{C}}$ denote the primitive (n,k,d) RS code obtained by *puncturing* the code $\mathcal{C}$ in the last position. The punctured code $\tilde{\mathcal{C}}$ is defined by the parity check matrix

$$\tilde{H} = \begin{pmatrix} \alpha_1^{(d-1)} & \alpha_2^{(d-1)} & \ldots & \alpha_n^{(d-1)} \\ \vdots & \vdots & \ldots & \vdots \\ \alpha_1^2 & \alpha_2^2 & \ldots & \alpha_n^2 \\ \alpha_1 & \alpha_2 & \ldots & \alpha_n \end{pmatrix}, \qquad (18)$$

obtained from H by deleting both the last column and the last row.

The extended RS code $\mathcal{C}$ can correct up to $t = \lfloor d/2 \rfloor$ errors as follows (see e.g. [15]). Assume a codeword $c \in \mathcal{C}$ was transmitted and a word $r = (r_1, r_2, \ldots, r_{n+1})$ over $_Q$ containing up to t errors was received.

If the distance $d+1$ of the extended code is even, then the punctured code $\tilde{\mathcal{C}}$ has odd distance d and can correct t errors. Hence, we can correct up to t errors in the punctured word $r = (r_1, r_2, \ldots, r_n)$ using a bounded minimum distance (BMD) decoder of the classical RS code $\tilde{\mathcal{C}}$. After decoding, the last symbol r_{n+1} can be corrected using the last parity check $c_1 + c_2 + \cdots + c_{n+1} = 0$ from (15). Let us consider the nontrivial case of odd $d+1$.

First we compute the syndrome

$$s = (s_1, s_2, \ldots, s_{d-1}, s_d) = rH^T = eH^T, \qquad (19)$$

where e is the error vector. We distinguish two cases: the last received symbol r_{n+1} is wrong or correct.

1. Assume that r_{n+1} is wrong and consider the received word r punctured in the last position, $r' = (r_1, \ldots, r_n)$ as a received word of the punctured code $\tilde{\mathcal{C}}$. The syndrome $\tilde{s} = (s_1, s_2, \ldots, s_{d-1}) = r'\tilde{H}$ of the punctured code $\tilde{\mathcal{C}}$ can be obtained directly from known syndrome s. By assumption, r' contains $t-1$ errors at most. The punctured code $\tilde{\mathcal{C}}$ has even distance $\tilde{d} = d$. It can correct up to $t-1$ errors using a BMD decoder of classical RS code and can detect if there were t errors. If t errors were detected, then we go to the next step, otherwise all errors in r' were corrected and the last symbol can be obtained using the last parity check $c_1 + c_2 + \cdots + c_{n+1} = 0$ from (15).

2. Since t errors were detected in r', the last symbol r_{n+1} in word r is error free by assumption, $e_{n+1} = 0$, and we should correct t errors in the word r'. To do this we consider the word r' as a received word of the shortened code $\hat{\mathcal{C}}$ having distance $d+1$ and hence correcting t errors. It follows from (15) and (17) that the syndrome of r' in the shortened code $\hat{\mathcal{C}}$ is $\hat{s} = r'\hat{H}^T = (e_1, e_2, \ldots, e_n)\hat{H}^T = eH^T = s$ since $e_{n+1} = 0$. Using a BMD decoder of classical RS code $\hat{\mathcal{C}}$ we will correct t errors in the word r' and obtain correct codeword, since r_{n+1} is error free.

The drawback of the algorithm is that we use BMD decoders of RS codes twice. However, it was shown in [15], Section 9.3, how to use results of the first decoding step in the second step. This method gives an algorithm for the extended code having the same complexity as a BMD decoder of a classical RS code. In the next sections we show how to extend these ideas for interleaved extended RS codes.

V. Interleaving of classical Reed–Solomon codes

Given ℓ Reed–Solomon codes $\mathcal{RS}(n, k^{(l)})$, $l = 1, 2, \ldots, \ell$, over $= Q$ of length n and dimensions $k^{(l)}$ defined by parity check matrices $H^{(l)}$ of the form (17) or (18), the interleaved RS (IRS) code $\mathcal{IRS}(n, k^{(1)}, \ldots, k^{(\ell)})$ consists of all $\ell \times n$ matrices C

$$C = \begin{pmatrix} c^{(1)} \\ c^{(2)} \\ \vdots \\ c^{(\ell)} \end{pmatrix} = \begin{pmatrix} c_1^{(1)} & c_2^{(1)} & \ldots & c_n^{(1)} \\ c_1^{(2)} & c_2^{(2)} & \ldots & c_n^{(2)} \\ \vdots & \vdots & \vdots & \vdots \\ c_1^{(\ell)} & c_2^{(\ell)} & \ldots & c_n^{(\ell)} \end{pmatrix}, \qquad (20)$$

where $c^{(l)} \in \mathcal{RS}(n, k^{(l)})$.

We assume the following *channel model*. When we transmit a code matrix C and receive a $q \times n$ matrix R over $_q$, we say that the error matrix was $E = R - C$. The number of nonzero columns in the matrix E is called the error weight. Later we will assume a Q-ary symmetric channel, i.e., all error matrices of weight t are equiprobable.

The IRS code can be efficiently decoded as follows. Let us agree that the index l always runs from 1 to ℓ. Given a received matrix R, denote by $r^{(l)}$ rows of R and compute the syndrome vectors $s^{(l)}$ and polynomials $s^{(l)}(x)$ for every component RS code as follows

$$s^{(l)} = (s_1^{(l)}, s_2^{(l)}, \ldots, s_{d-1}^{(l)}) = r^{(l)}H^{(l)T} \qquad (21)$$

and

$$s^{(l)}(x) = \sum_{i=1}^{d-1} s_i^{(l)} x^{i-1}. \qquad (22)$$

Assume that t erroneous columns are at positions $\mathcal{E} = \{i_1, \ldots, i_t\}$ and define the error locator polynomial as follows

$$\sigma(x) = 1 + \sigma_1 x + \cdots + \sigma_t x^t = \prod_{i \in \mathcal{E}} (x - \alpha_i). \qquad (23)$$

Then the syndromes and the error locator polynomial for the IRS code satisfy the following system of key equations

$$s^{(l)}(x)\sigma(x) \equiv \omega^{(l)}(x) \mod x^{d^{(l)}-1} \text{ for } l = 1, \ldots, \ell, \qquad (24)$$

where $\omega^{(l)}(x)$ is a polynomial with $\deg \omega^{(l)}(x) < \deg \sigma(x)$. Solving the system of key equations is equivalent to solving the following problem of multi-sequence linear feedback shift register synthesis.

Problem 1: Let $\mathrm{S}^{(1)}, \mathrm{S}^{(2)}, \ldots, \mathrm{S}^{(\ell)}$ be sequences over a field with lengths $N^{(1)}, N^{(2)}, \ldots, N^{(\ell)}$, respectively. Find the smallest nonnegative integer λ for which there is a vector of coefficients $\sigma = (\sigma_1, \sigma_2, \ldots, \sigma_\lambda)$ over such that for $l = 1, 2, \ldots, \ell$ and for $n = \lambda + 1, \ldots, N^{(l)}$

$$s_n^{(l)} = -\sigma_1 s_{n-1}^{(l)} - \sigma_2 s_{n-2}^{(l)} - \cdots - \sigma_\lambda s_{n-\lambda}^{(l)}. \qquad (25)$$

Moreover, find a vector of coefficients σ which fulfills (25). ■

An efficient solution [16] of Problem 1 is given by Algorithm 1.

Theorem 2 ([16]): If the output of Algorithm 1 is λ, $\sigma(x)$, N; and $\sigma^{(l)}(x)$, $n^{(l)}$, $\lambda^{(l)}$ for $l = 1, \ldots, \ell$, then λ is the length of a shortest shift-register that generates ℓ sequences $\mathrm{S}^{(l)}$. The connection polynomial $\sigma(x)$ is unique if and only if $\varepsilon = 0$, where

$$\varepsilon = \sum_{l=1}^{\ell} \varepsilon^{(l)},$$

$$\varepsilon^{(l)} = \max\{0, n^{(l)} - \lambda^{(l)} - z^{(l)} - (N - \lambda)\},$$

$$z^{(l)} = \max\{0, \lambda - N^{(l)}\}.$$

Time complexity of Algorithm 1 is $\mathcal{O}(\ell N^2)$ operations in .

After the unique error locator polynomial is found, position of errors can be obtained by computing roots of the polynomial. When the error positions are known, the errors are transformed to erasures and can be corrected independently in every component code. As a result the IRS code can be decoded by Algorithm 2.

Theorem 3 ([8]): For an IRS code $\mathcal{IRS}(n, k^{(1)}, \ldots, k^{(\ell)})$, Algorithm 2 corrects errors of

Algorithm 1: Multisequence Shift-Register Synthesis (solution of Problem 1)

input: ℓ; $\mathrm{s}^{(l)} = s_1^{(l)}, \ldots, s_{N^{(l)}}^{(l)}$ and $N^{(l)}$ for $l = 1, \ldots, \ell$
begin
  $\lambda \leftarrow 0$, $\sigma(x) \leftarrow 1$, $N \leftarrow \max_l\{N^{(l)}\}$
  $\delta^{(l)} \leftarrow N - N^{(l)}$, $n^{(l)} \leftarrow \delta^{(l)}$, $\lambda^{(l)} \leftarrow 0$, $d^{(l)} \leftarrow 1$, $\sigma^{(l)}(x) \leftarrow 0$ for $l = 1, \ldots, \ell$
  for *each* n *from* 1 *to* N **do**
    for *each* l *from* 1 *to* ℓ **do**
      if $n - \lambda > \delta^{(l)}$ **then**
        $d \leftarrow \sum_{j=0}^{\lambda} \sigma_j s_{n-j-\delta^{(l)}}^{(l)}$
        if $d \neq 0$ **then**
          if $n - \lambda \leq n^{(l)} - \lambda^{(l)}$ **then**
            $\sigma(x) \leftarrow \sigma(x) - \frac{d}{d^{(l)}} \sigma^{(l)}(x) x^{n-n^{(l)}}$
          else
            $\tilde{\lambda} \leftarrow \lambda$, $\tilde{\sigma}(x) \leftarrow \sigma(x)$
            $\sigma(x) \leftarrow \sigma(x) - \frac{d}{d^{(l)}} \sigma^{(l)}(x) x^{n-n^{(l)}}$
            $\lambda \leftarrow \lambda^{(l)} + n - n^{(l)}$
            $\lambda^{(l)} \leftarrow \tilde{\lambda}$, $\sigma^{(l)}(x) \leftarrow \tilde{\sigma}(x)$,
            $d^{(l)} \leftarrow d$, $n^{(l)} \leftarrow n$
end
output: λ, $\sigma(x)$, N; and $\sigma^{(l)}(x)$, $n^{(l)}$, $\lambda^{(l)}$ for $l = 1, \ldots, \ell$

weight t up to decoding radius $t \leq \min\{t_{\max}, n - k_{\max}\}$ with failure probability $P_f(t)$, where

$$t_{\max} = \frac{\ell}{\ell+1}\left(n - \overline{k}\right), \tag{26}$$

$$\overline{k} = \frac{1}{\ell}\sum_{l=1}^{\ell} k^{(l)}, \quad k_{\max} = \max\{k^{(l)}\}$$

are the average and the maximum dimension of the ℓ Reed–Solomon codes respectively, and

$$P_f(t) \leq \widehat{P}_f(t) = \gamma Q^{-(l+1)(t_{\max}-t)-1}, \tag{27}$$

$$\gamma = \left(\frac{Q^l - \frac{1}{Q}}{Q^l - 1}\right)^t \frac{Q}{Q-1} \approx 1.$$

The bound (27) can be also written as

$$P_f(t) \leq \widehat{P}_f(t) = \gamma Q^{-(\#\text{ of equations}-t)-1}, \tag{28}$$

using the numbers of equations in the system (25).

VI. Interleaving of Extended Reed–Solomon codes

Consider ℓ extended Reed–Solomon codes $\mathcal{ERS}(n+1, k^{(l)})$, $l = 1, 2, \ldots, \ell$, over $= Q$ of length $n+1$ and dimensions $k^{(l)}$ defined by the parity check matrices $H^{(l)}$ obtained from (15) by replacing d with $d^{(l)} = n - k^{(l)} + 1$.

Algorithm 2: Decoding IRS code

input: Received words $r^{(1)}, \ldots, r^{(\ell)}$
begin
  Compute syndromes $s^{(l)} = r^{(l)} H^{(l)T}$ for $l = 1, \ldots, \ell$
  Run Algorithm 1 for ℓ sequences $\mathrm{s}^{(l)} = s^{(l)}$, $l = 1, \ldots, \ell$ of length $N^{(l)} = n - k^{(l)}$, and get $t = \lambda$ and $\sigma(x)$
  if $\varepsilon \neq 0$ **then**
    output decoding failure and stop
  Find roots $\alpha^{i_1}, \ldots, \alpha^{i_\tau}$ of $\sigma(x)$ in
  if *number of roots not equal* t **then**
    output decoding failure and stop
  for $l = 1, \ldots, \ell$ **do**
    Compute $c^{(l)}$ by correcting t erasures in positions $i_1, \ldots, i_t$ of $r^{(l)}$
end
output: Codewords $c^{(1)}, \ldots, c^{(\ell)}$ or decoding failure

The IERS code $\mathcal{IERS}(n+1, k^{(1)}, \ldots, k^{(\ell)})$ consists of all $\ell \times (n+1)$ matrices C

$$C = \begin{pmatrix} c^{(1)} \\ c^{(2)} \\ \vdots \\ c^{(\ell)} \end{pmatrix} = \begin{pmatrix} c_1^{(1)} & c_2^{(1)} & \ldots & c_{n+1}^{(1)} \\ c_1^{(2)} & c_2^{(2)} & \ldots & c_{n+1}^{(2)} \\ \vdots & \vdots & \vdots & \vdots \\ c_1^{(\ell)} & c_2^{(\ell)} & \ldots & c_{n+1}^{(\ell)} \end{pmatrix}, \tag{29}$$

where $c^{(l)} \in \mathcal{ERS}(n+1, k^{(l)})$. Denote the received matrix R with t erroneous columns as follows

$$R = (R_1, \ldots, R_{n+1}) = (r^{(1)}, \ldots, r^{(\ell)})^T$$

and compute ℓ syndrome vectors $s^{(l)} = r^{(l)} H^{(l)T}$. Our goal will be to correct up to $t_{\max}^+$ columns in R, where decoding radius $t_{\max}^+$ is given by (32) obtained from (26) using parameters of the IERS code. Let us reduce decoding of IERS codes to decoding of IRS codes. If we delete the last column R_{n+1} from the received matrix R we obtain interleaving of classical codes $\mathcal{RS}(n, k^{(l)}, d^{(l)})$, which can be efficiently decoded. Let us consider again two cases (implemented by two steps): the last column R_{n+1} is wrong or correct.

Step 1. Assume that R_{n+1} is wrong. Denote by $R' = (R_1, \ldots, R_n)$ the received matrix punctured in the last position, rows of the matrix denote by $r^{(l)'}$. By assumption, there are $t-1$ erroneous columns in R'. Syndromes $s^{(l)'}$ of punctured codes can be obtained by puncturing the syndromes $s^{(l)}$ in the last position. We will decode the matrix R' using interleaving of ℓ punctured codes $\tilde{\mathcal{C}}^{(l)}$, which are classical RS codes defined by parity check matrices $\tilde{H}^{(l)}$ (18), where d should be replaced by $d^{(l)} = n - k^{(l)} + 1$ to get $\tilde{H}^{(l)}$ from $\tilde{H}$. By decoding interleaving of ℓ codes $\tilde{\mathcal{C}}^{(l)}$ we are able to

correct up to $t_{\max}$ columns in R' according to (26), where from (32)

$$t_{\max} = t^+_{\max} - \ell/(\ell+1) \geq t^+_{\max} - 1.$$

Hence, from (27) decoding failure probability when correcting $t-1$ columns is

$$P_f(t) \leq \gamma Q^{-(l+1)(t_{\max}-(t-1))-1} \leq \gamma Q^{-(l+1)(t^+_{\max}-t)-1} \tag{30}$$

After this, the last column in R can be corrected using the last parity check $C_1 + \cdots + C_{n+1} = 0$ in (15). Since we are correcting errors beyond half the code distance, we can not detect the case when $t^+_{\max}$ columns of $\tilde{R}$ are in error and hence we should go to the next step.

Step 2. Assume that R_{n+1} is correct. By decoding R' using an interleaving of ℓ shortened codes $\hat{\mathcal{C}}^{(l)}$ and full syndromes $s^{(l)}$, we are able to correct up to $t^+_{\max}$ columns in R' according to Theorem 3, and get a correct code matrix C since the last column R_{n+1} was error free by assumption. By Theorem 3, decoding failure probability when correcting t columns is

$$P_f(t) \leq \gamma Q^{-(l+1)(t^+_{\max}-t)-1}, \tag{31}$$

hence (31) gives decoding failure probability of the complete decoder, since (31) coincides with the upper bound (30).

The proposed solution calls twice the decoder of IRS codes and hence extension of IRS codes by one symbol is twice more complex in comparison with classical RS codes. Fortunately, we are able to localize erroneous columns executing the following modification of Algorithm 1 only once.

The syndrome $s^{(l)}$ can be obtained from $s^{(l)'}$ by adding one more $d^{(l)}$th symbol. Berlekamp–Massey (BM) type algorithm process elements of a syndrome sequence sequentially to compute the polynomial $\sigma(x)$. As a result, for the case of a single sequence, when the sequence $s^{(l)'}$ was processed, to process the sequence $s^{(l)}$ one should just continue the BM algorithm and make one step for the last element of $s^{(l)}$.

In case of multiple sequences of different length application of this idea depends on the order of processing the elements of the sequences. Fortunately, the processing order in Agorithm 1 allows to apply this idea. To explain the processing order in Agorithm 1 consider the following example of two syndrome sequences

$$s^{(1)} = (s^{(1)}_1, s^{(1)}_2, s^{(1)}_3)$$

and

$$s^{(2)} = (s^{(2)}_1, s^{(2)}_2, s^{(2)}_3, s^{(2)}_4, s^{(2)}_5)$$

of lengths $N^{(1)} = 3$ and $N^{(2)} = 5$ respectively. Then we obtain the maximum sequence length $N = 5$ and compose the syndrome matrix as follows

$$S = \begin{pmatrix} & & s^{(1)}_1 & s^{(1)}_2 & s^{(1)}_3 \\ s^{(2)}_1 & s^{(2)}_2 & s^{(2)}_3 & s^{(2)}_4 & s^{(2)}_5 \end{pmatrix}.$$

This means that the sequences are aligned to the *right*. In Algorithm 1 the elements $s^{(l)}_n$ in the array S are processed one-by-one and downwards by columns in the following order

$$s^{(2)}_1, s^{(2)}_2, s^{(1)}_1, s^{(2)}_3, s^{(1)}_2, s^{(2)}_4, s^{(1)}_3 s^{(2)}_5.$$

This processing order in Algorithm 1 is implemented in Lines 7–8. Last elements of the syndrome sequences are in the last column of the syndrome matrix S which is processed column wise. This allows to simplify our decoding as it is shown by Algorithm 3, where Step 1 is implemented in Line 6 and Step 2 in Line 6. One of the steps or both can fail. By t_i a nd $\sigma_i(x)$ we denote the number of errors and the error locator polynomial respectively found in the ith step, $i = 1, 2$. For every found $\sigma_i(x)$ we compute positions of errors in Line 8, correct them separately in each interleaved code in Line 10, and compute the code matrix in Line 11.

Algorithm 3: Decoding interleaved extended RS codes

1 **input:** Receive $R = (R_1, \ldots, R_{n+1}) = (r^{(1)}, \ldots, r^{(\ell)})^T$
2 **begin**
3 Compute syndromes $s^{(l)} = r^{(l)} H^{(l)T}$, $l = 1, \ldots, \ell$
4 Run Algorithm 1 for ℓ sequences $\mathrm{S}^{(l)} = s^{(l)}$ of length $N^{(l)} = n - k^{(l)}$ as follows:
5 First run Algorithm 1 for $n = 1 \ldots, N-1$ and get $t_1 = \lambda_1$ and $\sigma_1(x)$. If $\varepsilon \neq 0$ skip $\sigma_1(x)$
6 Second, run Lines 6-16 of Algorithm 1 for the last syndrome elements, and get $t_2 = \lambda$ and $\sigma_2(x)$. If $\varepsilon \neq 0$ skip $\sigma_2(x)$. If both $\sigma_i(x)$ are skipped, declare failure
7 **for** *each not skipped* $\sigma_i(x)$, $i = 1, 2$ **do**
8 Find roots $\alpha^{i_1}, \ldots, \alpha^{i_\tau}$ of $\sigma_i(x)$ in . If number of roots of $\sigma_i(x)$ not equal t_i skip $\sigma_i(x)$
9 **for** $l = 1, \ldots, \ell$ **do**
10 compute $c^{(l)}$ by correcting t_i erasures in positions $i_1, \ldots, i_{t_i}$ of $r^{(l)}$, get the $\ell \times n$ matrix $\tilde{C}^{(i)}$ of punctured IRS code
11 compute $C^{(i)}_{n+1} = -C^{(i)}_1 - C^{(i)}_2 - \ldots, -C^{(i)}_n$
Compute the code matrix $C^{(i)} = (\tilde{C}^{(i)}, C^{(i)}_{n+1})$
12 **end**
13 **output:** Code matrix $C^{(i)}$ nearest to R in the burst metric or decoding failure

From the above discussion and from Theorem 3 we obtain

Theorem 4: For the code $\mathcal{IERS}(n+1, k^{(1)}, \ldots, k^{(\ell)})$, Algorithm 3 corrects errors of weight t up to decoding radius $t \leq \min\{t^+_{\max}, n - k_{\max} + 1\}$ with failure probability $P_f(t)$ upper bounded by (31), with

$$t^+_{\max} = \frac{\ell}{\ell+1}\left(n - \overline{k} + 1\right), \tag{32}$$

where $\overline{k}$ and $k_{\max}$ are defined in Theorem 3.

Complexity of Algorithm 3 is also $\mathcal{O}(\ell n^2)$ operations in Q similar to the one for classical IRS codes. We localize errors calling the shift register synthesis algorithm once.

VII. Power Decoding

To describe the idea of "power" decoding [10] of an (extended) RS code consider a code vector $c = (c_1, \dots, c_n \in \mathcal{RS}(n,k))$ and define $c^l = (c_1^l, \dots, c_n^l)$ for $l = 1, 2, \dots$. From Definition 3 immediately follows that if $c \in \mathcal{RS}(n,k)$ then $c^l \in \mathcal{RS}(n, (k-1)l+1)$ for $(k-1)l+1 \le n$. In particular:
if $c \in \mathcal{ERS}(n,k) = \mathcal{C}$ then $c^l \in \mathcal{ERS}(n,(k-1)l+1) = \mathcal{C}^{(l)}$.

The idea of the power decoding is as follows. We transmit a codeword c of $\mathcal{ERS}(n,k)$ and receive a word r with t errors. At the transmitter we can compute ℓ powers of c and get codewords $c, c^2, \dots, c^\ell$, from ERS codes $\mathcal{C}^{(l)}$, $l = 1, \dots, \ell$, where ℓ is the maximum number, such that $(k-1)\ell + 1 \le n$. We can think that we virtually transmitted interleaved words $c, c^2, \dots, c^\ell$, despite in reality we transmit c only.

At the receiver we compute ℓ powers of r and get virtually received words $r, r^2, \dots, r^\ell$. Observe that error free positions in r stay error free in all r^l as well. Hence at the receiver we have a matrix R of interleaved ERS code with t corrupted columns. Since we know how to decode IERS code, we are able to decode the matrix R. This decoding allows to decode a single extended RS code up to the following radius

$$t_{\max}(\ell) = \left\lfloor \frac{2\ell n - \ell(\ell+1)k + \ell(\ell-1)}{2(\ell+1)} \right\rfloor, \qquad (33)$$

which coincides with the Sudan decoding radius [17]. Probability of decoding failure for $\ell = 2$, which corresponds to code rates between $1/3$ and $1/6$ is given by (27) or (28) with slightly different coefficient γ

$$\gamma = \frac{Q}{Q-1}\left(\frac{Q}{Q-1} + \frac{1}{Q}\right)^t \approx 1.$$

For code rates below 1/6 failure probability can be estimated using (27) or (28) as well under the assumption that the virtual error vectors $r^i - c^i$, $i = 1 \dots, \ell$, are statistically independent, correctness of which is supported by simulations.

The power decoding can be applied for interleaved (extended) RS codes [8], if some of them have low rate. In this case every codeword of a low rate code can be virtually extended to two or more codewords. This increases the order of interleaving, the number of equations and the decoding radius or decreases the failure probability.

For interleaving of low rate codes the order of interleaving can be increased even more using "mixed" decoding based on the following observation by Wachter-Zeh et. al. [11]. If $c, \tilde{c}$ are codewords of (n,k) and $(n,\tilde{k})$ RS codes respectively, then the word with component-wise product is a code word of an $(n, k+\tilde{k}-1)$ RS code. Simulations show that we can assume statistical independence of all virtually created error vectors, and one still can use (28) to estimate failure probability, despite this was not proved.

VIII. Decoding Hermitian codes

Decoding of Hermitian code is shown by Algorithm 4.

Algorithm 4: Decoding Hermitian code

input: Received matrix V
begin
  For $j = 1, \dots, Q$ solve the system (12) and get all $f^{(i)}(x_j)$, $i = 1, \dots, q$ (with errors)
  Form matrix R of IERS code (14)
  Decode R by Algorithm 3, get a codeword C of IERS code
  Find information polynomial $f^{(i)}(x)$, $i = 1, \dots, \ell$, for every component ERS codeword $c^{(i)}$
end
output: Information polynomials $f^{(i)}(x)$, $i = 1, \dots, \ell$, or decoding failure

Theorem 5: For a Hermitian (N, K) code $\mathcal{H}_m$, see Definition 1, Algorithm 4 corrects t erroneous columns (bursts) up to decoding radius $t_{\max}$ with failure probability $P_f(t)$ if

$$t \le t_{\max} = \min\left\{ \frac{N-K}{q+1}, N/q - k_{\max} \right\}, \qquad (34)$$

where

$$k_{\max} = \max\{k^{(l)}\} = \lfloor m/q \rfloor + 1$$

see (6), and

$$P_f(t) \le \widehat{P}_f(t) = \gamma Q^{-(q+1)(t_{\max}-t)-1}, \qquad (35)$$

$$\gamma = \left(\frac{Q^l - \frac{1}{Q}}{Q^l - 1}\right)^t \frac{Q}{Q-1} \approx 1.$$

For $t \le (N/q - k_{\max})/2$ holds $P_f(t) = 0$.

Time complexity of Algorithm 4 is $\mathcal{O}(N^{5/3})$ operations in $_Q$.

Proof: After solving the system of linear equations (12) of full rank, every non-zero burst of Hermitian code is mapped in one-to-one manner to a non-zero burst of IERS code. Hence, both codes have the same number of bursts and all non-zero bursts are equiprobable.

Hermitian code will be successfully decoded as soon as we decode correspondent IERS code. Hence, the decoding radius and probability of failure can be obtained from Theorems 3 and 4 by using in Theorem 3 parameters of the Hermitian and IERS codes: order of interleaving $\ell = q$, length of RS codes $n = Q$, dimensions $k^{(l)}$, average dimension of RS codes is $\overline{k} = K/q$. As a result we obtain for Algorithm 4 time complexity is $\mathcal{O}(q^5) = \mathcal{O}(N^{5/3})$ operations in $_Q$, since $N = q^3$. ∎

Observe that in the case, when $\frac{N-K}{q+1} \le N/q - k_{\max}$, which is frequently satisfied, we have an interesting for practice case

$$t_{\max} = \frac{N-K}{q+1},$$

i.e., we can correct $\frac{q}{q+1}(\widehat{d}_B - 1)$ bursts of errors, which is very close to an upper bound $\widehat{d}_B$ (10) on the burst distance

TABLE I

THRESHOLD RATE FOR HERMITIAN CODES TO USE "POWER" DECODING

	$q=4$	$q=8$	$q=16$
$R_{\mathcal{H}_m}$	0.51	0.40	0.36

of the Hermitian code and is almost twice larger than the guaranteed burst correcting radius $(d_B-1)/2$.

Decoding radius and/or failure probability can be improved for low rate codes using power or mixed decoding techniques as described in Section VII. Power decoding can be applied when the minimal rate $k^{(q)}/Q$ of the interleaved RS codes is at most 1/3, i.e., when rate $R_{\mathcal{H}_m}$ of the Hermitian code is bounded by

$$R_{\mathcal{H}_m} = \frac{m-g+1}{q^3} \le \frac{\left\lceil \frac{q^2-3}{3} \right\rceil}{q^2} + \frac{1}{2q} + \frac{1}{2q^2}. \qquad (36)$$

Table I shows dependence of the threshold rate on q. For $q=4$ we can apply power decoding staring from rate 1/2, however, for large q the threshold tends to 1/3. Mixed decoding can be applied when $k^{(q-1)}/Q \le 1/3$.

We made simulations to verify precision of our bounds for failure probability. This can be done when the number of bursts is close to maximum possible, otherwise, the failure probability is usually too small to simulate. Our simulations show that all obtained bounds for the failure probability coincides with simulations results up to a factor < 3. Some simulations results are shown in Table II.

Example 1: To compare performance of our algorithm with the ones by Ren [6] and by Özbudak and Yayla [9], let us consider an example from [6], [9], and [5] of rate $R=1/2$ $(64,32)$ Hermitian code $\mathcal{H}_{37}$ over $_{16}$ with $q=4$ and $Q=16$. Decoding of the code is reduced to decoding of $q=4$ interleaved extended RS $(n+1, k^{(i)}, d+1)$ codes: $(16,10,7)$, $(16,9,8)$, $(16,7,10)$, and $(16,6,11)$.

It follows from Theorems 5 that $t_{\max}=6.4$, hence we can correct up to 6 bursts. Failure probability in case of 6 bursts in the channel is $P_f(6) \le \gamma Q^{-3} = 2.6\times 10^{-4}$, where $\gamma = 1.067$. In case of 5 bursts we have $P_f(5) \le \gamma Q^{-8} = 2.3\times 10^{-10}$.

Decoding algorithm by Özbudak and Yayla [9] also can correct up to 6 busts with failure probability $P_f(t) \le (1-R)Q/(Q+1) = 0.4$. Actually, for these code parameters they give slightly better bound $P_f(t) \le 0.37$. This bound does not depend on the number of bursts t in the channel and is very weak. Their simulations show that for $t=6$ bursts the failure probability is 1.7×10^{-3}.

Ren [6] have shown one example, when his algorithm can correct 6 bursts. Let us estimate probability of decoding failure of his algorithm for $t=6$ bursts. First, he decodes the $(16,6,11)$ ERS code, correcting up to 5 errors, which was the case in his example. However, with probability $(Q-1)/Q)^6 = 0.67$ there will be 6 errors in the fist word and the decoder will fail. Hence, for the Ren's decoder we have $P_f(6) \ge 0.67$.

TABLE II

FAILURE PROBABILITY $P_f(t)$ FOR HERMITIAN CODES OVER $\mathbb{F}_{4^2}$

$N=10^6$	$\mathcal{H}_{26}(64,21,38)$	$\mathcal{H}_{30}(64,25,34)$
$t_{\max}$	8	7
t_{power}	9	8
t	9	8
N_f	31056	34574
$P_f(t)$	$3.1\cdot 10^{-2}$	$3.5\cdot 10^{-2}$
$\widehat{P}_f(t)$	$6\cdot 10^{-2}$	$6\cdot 10^{-2}$

$t_{\max}$: without syndrome extension
t_{power} : with syndrome extension
t : the number of burst errors used in the simulation
N_f : the number of decoding failures
P_f : simulated failure probability
$\widehat{P}_f$: theoretical failure probability upperbound

IX. ACKNOWLEDGMENT

The authors are very thankful to Johan Nielsen and Martin Bossert for very helpful discussions, and to anonymous referees for their valuable comments.

REFERENCES

[1] V. D. Goppa, Codes associated with divisors, Problemy Peredachi Informatsii vol. 13, pp. 33-39, 1977.
[2] G.-L. Feng, T. R. N. Rao, Decoding algebraic-geometric codes up to the designed minimum distance, IEEE Trans. Inform. Theory, vol. 39, pp. 37-45, 1993.
[3] S. Sakata, J. Justesen, Y. Madelung, H. E. Jensen, T. Hoholdt, Fast decoding of algebraic-geometric codes up to the designed minimum distance, IEEE Trans. Inform. Theory, vol. 41, pp. 1672-1677, 1995.
[4] V. Guruswami, M. Sudan, Improved decoding of Reed-Solomon and algebraic-geometry codes, IEEE Trans. Inform. Theory, vol. 45, pp. 1757-1767, 1999.
[5] T. Yaghoobian, I. F. Blake, Hermitian codes as generalized Reed-Solomon codes, Des. Codes Cryptogr., vol. 2, pp. 5-17, 1992.
[6] J. Ren, On the structure of Hermitian codes and decoding for burst errors, IEEE Trans. Inform. Theory, vol. 50 pp. 2850-2854, 2004.
[7] V. Y. Krachkovsky, Reed–Solomon codes for correcting phased error busts, IEEE Trans. Inform. Theory, vol. IT-49, pp. 2975-2984, 2003.
[8] G. Schmidt, V. R. Sidorenko, and M. Bossert, Collaborative decoding of interleaved Reed-Solomon codes and concatenated code designs, IEEE Trans. Inform. Theory, vol. 55, n. 7, pp. 2991-3012, 2009.
[9] F. Özbudak and O. Yayla, Improved probabilistic decoding of interleaved Reed-Solomon codes and folded Hermitian codes, Theoretical Computer Science, vol. 520, pp. 111-123, 2014.
[10] G. Schmidt, V. R. Sidorenko, and M. Bossert, Syndrome decoding of Reed-Solomon codes beyond half the minimum distance based on shift-register synthesis, IEEE Trans. on Inform. Theory, vol. 56, n. 10, pp. 5245-5252, 2010.
[11] A. Wachter-Zeh, A. Zeh, and M. Bossert, Decoding Interleaved Reed-Solomon Codes Beyond Their Joint Error-Correcting Capability, Des. Codes Cryptogr., pp. 1-21, 2012.
[12] H. Stichtenoch, A note on Hermitian codes codes over $GF(q^2)$, IEEE Trans. Inform. Theory, vol. 34, pp. 1345-1348, 1988.
[13] H. J. Tiersma, Remarks on codes from Hermitian curves, IEEE Trans. Inform. Theory, vol. 33, pp. 605-609, 1987.
[14] M. Bossert, V. Sidorenko, Singleton type bounds for Blot Correcting codes, IEEE Trans. on Inf. Theory. vol. 42, pp. 1021-1023, 1996.
[15] R. E. Blahut, Theory and Practice of Error Control Codes, Addison-Wesley Publishing Company, 1983.
[16] V. Sidorenko, and G. Schmidt, A linear algebraic approach to multisequence shift-register synthesis, Problems of Information Transmission, vol. 47, n. 2, pp. 149-165, 2011.
[17] M. Sudan, Decoding of Reed-Solomon codes beyond the error correction bound, J. Complexity, vol. 13, pp. 180-193, 1997.