Design of Guruswami-Sudan List Decoding for Elliptic Codes

Yunqi Wan †, Li Chen ‡, Fangguo Zhang §
† School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou, China
‡ School of Electronics and Communication Engineering, Sun Yat-sen University, Guangzhou, China
§ School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China
Email: wanyq5@mail2.sysu.edu.cn, chenli55@mail.sysu.edu.cn, isszhfg@mail.sysu.edu.cn

***Abstract*—Advancing from Reed-Solomon (RS) codes, the length of algebraic-geometric (AG) codes can exceed the size of finite field, resulting in a greater error-correction capability. However, this is realized with a genus penalty. Usually, they are not maximum distance separable (MDS) codes. One-point elliptic codes are either MDS or almost MDS, yielding a good tradeoff between codeword length and distance property. This paper proposes the Guruswami-Sudan (GS) list decoding algorithm for elliptic codes. To define the interpolated polynomial $Q(x, y, z)$, an explicit construction for the zero basis of each affine point is introduced. Given an interpolation multiplicity m, the error-correction capability τ_m and the maximum decoding output cardinality l_m of the GS algorithm are characterized. An efficient interpolation algorithm is further presented for elliptic codes. Performance of elliptic codes is shown for the first time, demonstrating their advantage over RS codes.**

***Index Terms*—Algebraic-geometric codes, elliptic codes, interpolation, list decoding**

I. Introduction

Algebraic-geometric (AG) codes [1] are constructed based on an algebraic curve. Their codeword length can exceed the size of finite field, yielding a greater error-correction capability than a similar rate Reed-Solomon (RS) codes defined over the same finite field. However, they are usually not maximum distance separable (MDS) codes due to a genus penalty. In the AG family, elliptic codes are either MDS or almost MDS, yielding a good tradeoff between codeword length and distance property.

The early attempt to decode elliptic codes was made by Driencourt [2]. The decoding can correct at most $\lfloor \frac{d^*-1}{4} \rfloor$ errors, where d^* is the designed minimum distance of the code. Justesen *et al.* presented a decoding algorithm for a class of AG codes based on plane curve [3], which can be seen as a generalization of Peterson's algorithm for BCH and RS codes. Skorobogatov and Vladut further generalized the algorithm to decode codes constructed from an arbitrary algebraic curve [4]. For elliptic codes, the algorithm can correct up to $\lfloor \frac{d^*-1}{2} \rfloor$ errors. Feng and Rao introduced majority voting [5] to determine the unknown syndromes and proposed an algorithm that can decode AG codes also up to $\lfloor \frac{d^*-1}{2} \rfloor$ errors. Sakata *et al.* presented an efficient decoding algorithm for Hermitian codes using shift register synthesis [6]. Its decoding performance has been later studied by Johnston and Carrasco [7].

Another branch of decoding for AG codes is the interpolation based list decoding, first proposed by Sudan for low rate RS codes [8]. Shokrollahi and Wasserman generalized Sudan's algorithm for low rate AG codes [9]. Guruswami and Sudan further improved it to decode all rate RS and AG codes, with an error-correction capability beyond $\lfloor \frac{d^*-1}{2} \rfloor$, called the GS algorithm [10]. It constructs a minimum polynomial that has a zero of multiplicity m over a set of points. The intended message can be decoded by finding a root of the polynomial. Since it produces a list of decoded candidates, the GS decoding is also called the list decoding. Høholdt and Nielsen presented a mathematical framework for GS decoding of Hermitian codes [11]. Soft-decision list decoding of Hermitian codes was later proposed by Chen *et al.* [12] and Lee *et al.* [13], independently. Therefore, the interpolation based list decoding has been well developed for Hermitian codes. Their codeword length can be far greater than the size of finite field. But Hermitian codes also suffer a large genus penalty. Trading off the codeword length and distance property, elliptic codes stand out as a promised candidate. Recently, list decoding of elliptic codes has been considered by Zhang and Liu for solving the elliptic curve discrete logarithm problem [14]. However, they only consider affine points with order greater than interpolation multiplicity. To maximize the codeword length, all affine points need to be considered and their zero basis construction become complicated.

This paper proposes the GS list decoding algorithm for one-point elliptic codes. To define the interpolated polynomial $Q(x, y, z)$, we present an explicit construction for the zero basis of each affine point of the curve. Given an interpolation multiplicity m, the error-correction capability τ_m and the maximum decoding output cardinality l_m are further characterized. Based on the above preparations, an efficient interpolation algorithm to determine Q is proposed. Decoding performance of elliptic codes is also shown for the first time, demonstrating their advantage over RS codes.

II. Background Knowledge

A. AG Codes

Let $\mathbb{F}_q$ denote a finite field of size q and χ denote a smooth absolutely irreducible plane curve defined over $\mathbb{F}_q$ with a degree θ. The curve has a genus of $g = \frac{(\theta-1)(\theta-2)}{2}$. Let $\mathbb{F}_q(\chi)$ denote the algebraic function field of χ. Points on the curve

978-1-5386-6900-6/19/$31.00 ©2019 IEEE

χ with all their coordinates over $\mathbb{F}_q$ are called rational points. They are either the affine points (denoted as P_i) or the points of infinity (denoted as P_∞). In this paper, we consider curves with one point of infinity, i.e., $P_\infty = (0, 1, 0)$. Their affine component is chosen for code construction. Hence, the affine points can be denoted as $P_i = (x_i, y_i)$.

Given $h \in \mathbb{F}_q(\chi)$, its order at a rational point P is $v_P(h)$.

***Definition I** [15].* Let $h \in \mathbb{F}_q(\chi)$ and $h \neq 0$, the divisor of h is defined as $\text{div}(h) = \sum_{P \in \mathbf{P}_\chi} v_P(h)[P]$. $\text{div}(h)$ is also called the principle divisor of χ.

***Definition II** [16].* Given χ and one of its divisor D, the finite dimensional vector space over $\mathbb{F}_q(\chi)$ is defined as

$$\mathcal{L}(D) = \{h \in \mathbb{F}_q(\chi) \setminus \{0\} \mid \text{div}(h) + D \succeq 0\} \cup \{0\}, \quad (1)$$

where " $\succeq$ " indicates that the coefficients of $\text{div}(h) + D$ are nonnegative. If $\deg(D) > 2g - 2$, the dimension of $\mathcal{L}(D)$ is $\dim(D) = \deg(D) - g + 1$.

Note that $\mathcal{L}(u[P_\infty])$ with $u \geq 0$ contains the pole basis $Z_{P_\infty} = \{\phi_0(x, y), \phi_1(x, y), \ldots\}$, which is a set of bivariate monomials with an increasing pole order at P_∞, i.e.,

$$-v_{P_\infty}(\phi_a) < -v_{P_\infty}(\phi_{a+1}), a \in \mathbb{N}, \quad (2)$$

where $\mathbb{N}$ is the set of nonnegative integers. Moreover, for each affine point P_i, there exists a zero basis $Z_{P_i} = \{\psi_{P_i,0}(x, y), \psi_{P_i,1}(x, y), \ldots\}$ of $\mathcal{L}(u[P_\infty])$ such that

$$v_{P_i}(\psi_{P_i,\alpha}) < v_{P_i}(\psi_{P_i,\alpha+1}), \ \alpha \in \mathbb{N}, \quad (3)$$

where $\psi_{P_i,\alpha}(x_i, y_i) = 0$ and $v_{P_i}(\psi_{P_i,\alpha}) = \alpha$. Note that

$$\phi_a = \sum_{\alpha \in \mathbb{N}} \xi_{a,P_i,\alpha} \psi_{P_i,\alpha}, \quad (4)$$

where $\xi_{a,P_i,\alpha} \in \mathbb{F}_q$ is the corresponding coefficient between ϕ_a and $\psi_{P_i,\alpha}$ [11] [17].

Given n distinct affine points $P_0, P_1, \ldots, P_{n-1}$ of χ, we have a divisor $G = \sum_{i=0}^{n-1}[P_i]$. Let $f \in \mathcal{L}(u[P_\infty])$ with $u < n$ and

$$f(x, y) = f_0\phi_0 + f_1\phi_1 + \cdots + f_{k-1}\phi_{k-1} \quad (5)$$

is the message polynomial, where $f_0, f_1, \ldots, f_{k-1} \in \mathbb{F}_q$ are message symbols and $k = u - g + 1 < n$ for $u > 2g - 2$. The (n, k) one-point AG code constructed based on χ is

$$\mathcal{C}_\chi(G, u[P_\infty]) = \{(f(P_0), f(P_1), \ldots, f(P_{n-1})), \forall f\}, \quad (6)$$

where codeword $\underline{c} = (c_0, c_1, \ldots, c_{n-1}) = (f(P_0), f(P_1), \ldots, f(P_{n-1}))$. The code has length n and dimension k. It has a minimum distance $d \geq d^*$, where $d^* = n - k - g + 1$ is the designed minimum distance of the code.

B. The GS List Decoding

Let $\mathcal{R} = \bigcup_{u=0}^{\infty} \mathcal{L}(u[P_\infty]) \subset \mathbb{F}_q(\chi)$ and $\mathcal{R}[z]$ denote the trivariate polynomial ring defined over $\mathcal{R}$. The GS decoding consists of interpolation and root-finding. Given $\underline{r} = (r_0, r_1, \ldots, r_{n-1}) \in \mathbb{F}_q^n$ as a received word, it is a variant of $\underline{c}$, as $\underline{r} = \underline{c} + \underline{e}$ where $\underline{e} \in \mathbb{F}_q^n$ is an error vector. The following n interpolation points can be formed

$$(P_0, r_0), (P_1, r_1), \ldots, (P_{n-1}, r_{n-1}). \quad (7)$$

Interpolation constructs a minimum polynomial $Q(x, y, z) \in \mathcal{R}[z]$. It interpolates the n points with a multiplicity of m. Polynomial Q should satisfy $\mathfrak{C} = n\binom{m+1}{2}$ interpolation constraints. Root-finding further decodes the message polynomial f through finding its z-roots, i.e., $Q(x, y, f) = 0$.

Q can be written as $Q = \sum_{a,b \geq 0} Q_{ab}\phi_a z^b$, where $Q_{ab} \in \mathbb{F}_q$. Let $w_z = -v_{P_\infty}(\phi_{k-1})$ denote the weight of z, the $(1, w_z)$-weighted degree of $\phi_a z^b$ is $\deg_{1,w_z}(\phi_a z^b) = -v_{P_\infty}(\phi_a) + w_z b$. Given two distinct monomials $\phi_{a_1} z^{b_1}$ and $\phi_{a_2} z^{b_2}$, we have $\text{ord}(\phi_{a_1} z^{b_1}) < \text{ord}(\phi_{a_2} z^{b_2})$, if $\deg_{1,w_z}(\phi_{a_1} z^{b_1}) < \deg_{1,w_z}(\phi_{a_2} z^{b_2})$, or $\deg_{1,w_z}(\phi_{a_1} z^{b_1}) = \deg_{1,w_z}(\phi_{a_2} z^{b_2})$ and $b_1 < b_2$. Hence, the $(1, w_z)$-weighted degree and leading order of Q can be defined as $\deg_{1,w_z}(Q) = \max\{\deg_{1,w_z}(\phi_a z^b) \mid Q_{ab} \neq 0\}$ and $\text{lod}(Q) = \max\{\text{ord}(\phi_a z^b) \mid Q_{ab} \neq 0\}$. For two distinct polynomials $Q_1, Q_2 \in \mathcal{R}[z]$, $Q_1 < Q_2$, if $\text{lod}(Q_1) < \text{lod}(Q_2)$.

***Theorem 1** [10].* Given polynomial $Q \in \mathcal{R}[z]$ that interpolates the n points of (7) with a multiplicity of m, and a polynomial h in the form of (5), if

$$m(n - |\{i \mid h(P_i) \neq r_i, \ \forall i\}|) > \deg_{1,w_z}(Q), \quad (8)$$

then $(z - h) \mid Q$ or $Q(x, \ y, \ h) = 0$.

Therefore, the message can be decoded by finding z-roots of Q. Since $f(P_i) = r_i$, if $r_i = c_i$, the GS algorithm corrects $|\{i \mid f(P_i) \neq r_i, \ \forall i\}|$ errors and this error-correction capability can be improved by increasing m. Given an $(n, \ k)$ AG code constructed on a curve of genus g, the GS algorithm's error-correction capability is upper bounded by

$$\tau_{\text{GS}} = n - \left\lfloor \sqrt{n(k + g - 1)} \right\rfloor - 1. \quad (9)$$

III. Elliptic Codes

The affine elliptic curve χ_{EL} can be written as $y^2 + \mathfrak{a}_1 xy + \mathfrak{a}_3 y = x^3 + \mathfrak{a}_2 x^2 + \mathfrak{a}_4 x + \mathfrak{a}_6$, where $\mathfrak{a}_1, \mathfrak{a}_2, \mathfrak{a}_3, \mathfrak{a}_4, \mathfrak{a}_6 \in \mathbb{F}_q$ and the discriminant is not null. Hence, elliptic curves have a genus $g = 1$. Note that an (n, k) elliptic code will be an MDS code if and only if for any $\{P_{i_1}, P_{i_2}, \ldots, P_{i_k}\} \subseteq \text{supp}(G)$, $[P_{i_1}] + [P_{i_2}] + \cdots + [P_{i_k}] - k[P_\infty]$ is not a principal divisor. Based on the Hasse-Weil bound [16], the maximum number of rational points on χ_{EL} is $N_q(\chi_{\text{EL}}) = q + 1 + 2\sqrt{q}$. Therefore, curve coefficients $\mathfrak{a}_1, \mathfrak{a}_2, \mathfrak{a}_3, \mathfrak{a}_4, \mathfrak{a}_6$ are chosen such that the Hasse-Weil bound can be reached. As a result, the codeword length can be maximized. Given χ_{EL}, $-v_{P_\infty}(x) = 2$, $-v_{P_\infty}(y) = 3$ and $-v_{P_\infty}(x^\lambda y^\gamma) = 2\lambda + 3\gamma$. The pole basis Z_{P_∞} of $\mathcal{R}$ is

$$Z_{P_\infty} = \{1, x, y, x^2, xy, x^3, x^2y, x^4, x^3y, \ldots\}. \quad (10)$$

In general, $\phi_a = x^\lambda y^\gamma$, where $\lambda \in \mathbb{N}$ and $\gamma \in \{0, 1\}$. It can be seen that for elliptic codes, $w_z = -v_{P_\infty}(\phi_{k-1}) = k$.

To construct an (n, k) elliptic code, let $G = [P_0] + [P_1] + \cdots + [P_{n-1}]$ be the divisor of χ_{EL}, $\mathcal{C}_{\chi_{\text{EL}}}(G, k[P_\infty])$ can be generated following (5) and (6).

Rational points of χ_{EL} form an additive Abelian group with P_∞ as the identity element [15]. That says given two affine

points P_1 and P_2, we can find P_3 such that $P_1+P_2=P_3$. P_1 and P_2 define a line $\mathfrak{b}_1x+\mathfrak{b}_2y+\mathfrak{b}_3=0$, where $\mathfrak{b}_1$, $\mathfrak{b}_2$, $\mathfrak{b}_3\in\mathbb{F}_q$. It also passes through $-P_3$. Therefore, $\text{div}(\mathfrak{b}_1x+\mathfrak{b}_2y+\mathfrak{b}_3)=[P_1]+[P_2]+[-P_3]-3[P_\infty]$. Similarly, $x-x_3=0$ passes through P_3 and $-P_3$, $\text{div}(x-x_3)=[P_3]+[-P_3]-2[P_\infty]$. Therefore,

$$[P_1]+[P_2]=[P_3]+[P_\infty]+\text{div}\left(\frac{\mathfrak{b}_1x+\mathfrak{b}_2y+\mathfrak{b}_3}{x-x_3}\right). \quad (11)$$

***Theorem 2** [15].* Given a divisor D of χ_{EL}, it is a principle divisor if and only if $\deg(D)=0$ and $\text{sum}(D)=P_\infty$.

Therefore, the divisor of $\psi_{P_i,\alpha}$ can be written as

$$\text{div}\left(\psi_{P_i,\alpha}\right)=\alpha[P_i]+[-\alpha P_i]-(\alpha+1)\left[P_\infty\right]. \quad (12)$$

For any P_i of χ_{EL}, there exists a nonnegative integer δ such that $\delta P_i=P_\infty$, where δ is the order of P_i. Consequently, $\text{div}(\psi_{P_i,\delta-1})=(\delta-1)[P_i]+[-(\delta-1)P_i]-\delta[P_\infty]=(\delta-1)[P_i]+[P_i]-\delta[P_\infty]=\delta[P_i]-\delta[P_\infty]$ and $\text{div}(\psi_{P_i,\delta})=\delta[P_i]+[-\delta P_i]-(\delta+1)[P_\infty]=\delta[P_i]-\delta[P_\infty]$. That says $v_{P_i}(\psi_{P_i,\delta-1})=v_{P_i}(\psi_{P_i,\delta})$, which contradicts the definition of Z_{P_i}. Hence, eq. (12) holds when $\alpha<\delta-1$. A practical channel code is often defined in a binary extension field, e.g., $\mathbb{F}_{64}$ and $\mathbb{F}_{256}$, in which $m\geq\delta-1$ holds for some affine points. Since $\alpha\in[0,m]$, this can lead to $\alpha\geq\delta-1$. Consequently, eq. (12) does not hold. The following Theorem redefines the divisors of functions of Z_{P_i}.

Theorem 3. Given an affine point P_i of χ_{EL}, $\delta P_i=P_\infty$. Let $\psi_{P_i,0}=1$. Divisors of $\psi_{P_i,\alpha}(\alpha>0)$ are defined as (i) $\text{div}(\psi_{P_i,\alpha})=\alpha[P_i]+[-\alpha P_i]-(\alpha+1)[P_\infty]$, if $0<\alpha<\delta-1$; (ii) $\text{div}(\psi_{P_i,\alpha})=\alpha[P_i]+[-(\alpha-1)P_i]+[-P_i]-(\alpha+2)[P_\infty]$, if $\alpha=\delta-1$; (iii) $\text{div}(\psi_{P_i,\alpha})=\alpha[P_i]-\alpha[P_\infty]$, if $\alpha=\delta$; (iv) $\text{div}(\psi_{P_i,\alpha})=\text{div}(\psi_{P_i,(\alpha-\delta\lfloor\frac{\alpha}{\delta}\rfloor)})+\lfloor\frac{\alpha}{\delta}\rfloor(\delta[P_i]-\delta[P_\infty])$, if $\alpha>\delta$.

Proof: Based on Theorem 2, we know when $0\leq\alpha\leq\delta$, function $\psi_{P_i,\alpha}$ is a zero basis function. When $\alpha>\delta$ and $0<\alpha-\delta\lfloor\frac{\alpha}{\delta}\rfloor<\delta-1$, $\text{div}(\psi_{P_i,(\alpha-\delta\lfloor\frac{\alpha}{\delta}\rfloor)})=(\alpha-\delta\lfloor\frac{\alpha}{\delta}\rfloor)[P_i]+[-(\alpha-\delta\lfloor\frac{\alpha}{\delta}\rfloor)P_i]-((\alpha-\delta\lfloor\frac{\alpha}{\delta}\rfloor)+1)[P_\infty]=(\alpha-\delta\lfloor\frac{\alpha}{\delta}\rfloor)[P_i]+[-\alpha P_i]-((\alpha-\delta\lfloor\frac{\alpha}{\delta}\rfloor)+1)[P_\infty]$, then $\text{div}(\psi_{P_i,(\alpha-\delta\lfloor\frac{\alpha}{\delta}\rfloor)})+\lfloor\frac{\alpha}{\delta}\rfloor(\delta[P_i]-\delta[P_\infty])=\alpha[P_i]+[-\alpha P_i]-(\alpha+1)[P_\infty]$, i.e., $v_{P_i}(\psi_{P_i,\alpha})=\alpha$. The proof is similar for the rest zero basis function with $\alpha>\delta$. ■

Example 1. Given χ_{EL}: $y^2+y=x^3$, which is defined over $\mathbb{F}_4=\{0,1,\alpha,\alpha^2\}$ and α is a primitive element satisfying $\alpha^2=\alpha+1$. Its affine points are: $P_0=(0,\ 0)$, $P_1=(0,\ 1)$, $P_2=(1,\ \alpha)$, $P_3=(1,\ \alpha^2)$, $P_4=(\alpha,\ \alpha)$, $P_5=(\alpha,\ \alpha^2)$, $P_6=(\alpha^2,\ \alpha)$ and $P_7=(\alpha^2,\ \alpha^2)$. For P_2, $\delta=3$. Based on Theorem 3, the first five functions of Z_{P_2} are $\psi_{P_2,0}=1$, $\psi_{P_2,1}=x+1$, $\psi_{P_2,2}=x^2+1$, $\psi_{P_2,3}=y+x+\alpha^2$ and $\psi_{P_2,4}=xy+x^2+y+\alpha x+\alpha^2$.

IV. GS Decoding of Elliptic Codes

A. Parameterization

Given an interpolation multiplicity m, let $l_m=\deg_z(Q)$ and τ_m denote the error-correction capability, respectively. Since the decoded candidates are z-roots of Q, l_m is also the maximum decoding output cardinality.

Theorem 4. Given an (n,k) elliptic code, then

$$l_m=\left\lfloor\sqrt{\frac{nm(m+1)}{k}+\frac{1}{4}}-\frac{1}{2}\right\rfloor. \quad (13)$$

Proof: Interpolating n points with a multiplicity of m imposes $n\binom{m+1}{2}$ interpolation constraints. Polynomial Q should contain at least $n\binom{m+1}{2}+1$ coefficients so that the linear system will have a nonzero solution. Therefore,

$$l_m=\max\left\{l\mid\text{ord}(z^l)\leq n\binom{m+1}{2}\right\}.$$

Since $w_z=k$ and $\text{ord}(z^l)=\text{ord}(z^{l-1})+|\{\phi_az^b\mid(l-1)k<\deg_{1,k}(\phi_az^b)\leq lk\}|=\text{ord}(z^{l-1})+lk$, $\text{ord}(z^l)=(1+2+\cdots+l)k=\frac{kl(l+1)}{2}$. Substituting it into the above equation will lead to the conclusion. ■

Theorem 5. For an (n,k) elliptic code, if $m(n-\tau_m)-kl_m\neq1$,

$$\tau_m=n-\left\lfloor\frac{1}{m}+\frac{l_mk}{2m}+\frac{(m+1)n}{2(l_m+1)}\right\rfloor-1. \quad (14)$$

Otherwise,

$$\tau_m=n-\frac{1+kl_m}{m}. \quad (15)$$

Proof: Given $\underline{r}$, $\tau_m=|\{i\mid f(P_i)\neq r_i,\forall i\}|$. Based on Theorem 1, we know if $m(n-\tau_m)>\deg_{1,k}(Q)$, $Q(x,y,f)=0$. Hence, the monomial ϕ_az^b of Q should satisfy $\deg_{1,k}(\phi_az^b)<m(n-\tau_m)$, and $\deg_{1,k}(\phi_a)<m\left(n-\tau_m\right)-kb$. Since $b=0,1,\ldots,l_m$, when $m(n-\tau_m)-kl_m\neq1$, $|\{\phi_az^b\mid\deg_{1,k}(\phi_az^b)<m(n-\tau_m)\}|=\sum_{b=0}^{l_m}(m(n-\tau_m)-kb-1)=m(l_m+1)(n-\tau_m)-\frac{kl_m(l_m+1)}{2}-l_m-1$. Therefore, when

$$m(l_m+1)(n-\tau_m)-\frac{kl_m(l_m+1)}{2}-l_m-1>n\binom{m+1}{2},$$

the linear system has a nonzero solution. Solving the above inequality leads to (14). When $m(n-\tau_m)-kl_m=1$, (15) can be straightforwardly reached. ■

Example 2. Applying Theorems 4 and 5, we give the decoding parameters for the $(80,\ 27)$ and the $(288,\ 163)$ elliptic codes, respectively. The listed multiplicities are the minimum values that yield the corresponding τ_m. The two codes are constructed based on: $y^2+y=x^3$ defined over $\mathbb{F}_{64}$, and $y^2+y=x^3+\mathfrak{a}_6$ defined over $\mathbb{F}_{256}$, respectively, where $\mathfrak{a}_6$ satisfies $\mathfrak{a}_6^{2^0}+\mathfrak{a}_6^{2^1}+\cdots+\mathfrak{a}_6^{2^7}=1$.

TABLE I
DECODING PARAMETERS OF THE $(80,\ 27)$ ELLIPTIC CODE

m	1	2	3	4	7	21
l_m	1	3	5	7	12	36
τ_m	25	29	30	31	32	33

B. The Interpolation Constraints

Given an interpolation multiplicity m, we define $\mathcal{R}[z]_{l_m}=\{Q\in\mathcal{R}[z]\mid\deg_z(Q)\leq l_m\}$. An interpolated polynomial

TABLE II
DECODING PARAMETERS OF THE (288, 163) ELLIPTIC CODE

m	1	3	4	5	6	8	12	21	83
l_m	1	4	5	6	8	10	16	28	110
τ_m	61	63	65	66	67	68	69	70	71

$Q \in \mathcal{R}[z]_{l_m}$ can be written as

$$Q = \sum_{a\in\mathbb{N}} \sum_{b\le l_m} Q_{ab}\phi_a z^b, \tag{16}$$

For point (P_i, r_i), (4) holds and $z^b = (z - r_i + r_i)^b = \sum_{\beta\le b} \binom{b}{\beta} r_i^{b-\beta} (z - r_i)^\beta$. Q can be further derived as

$$\begin{aligned} Q &= \sum_{a\in\mathbb{N}} \sum_{b\le l_m} Q_{ab} \left(\sum_{\alpha\in\mathbb{N}} \xi_{a,P_i,\alpha} \psi_{P_i,\alpha} \right) \\ &\quad \cdot \left(\sum_{\beta\le b} \binom{b}{\beta} r_i^{b-\beta} (z - r_i)^\beta \right) \\ &= \sum_{\alpha,\beta\in\mathbb{N}} \left(\mathcal{D}_{\alpha\beta}^{(P_i,r_i)}(Q) \right) \psi_{P_i,\alpha} (z - r_i)^\beta, \end{aligned} \tag{17}$$

where

$$\mathcal{D}_{\alpha\beta}^{(P_i,r_i)}(Q) = \sum_{a\in\mathbb{N}} \sum_{b=\beta}^{l_m} Q_{ab} \binom{b}{\beta} \xi_{a,P_i,\alpha} r_i^{b-\beta} \tag{18}$$

is the evaluation of the (α, β)-Hasse derivative of Q at (P_i, r_i). Note that $Q(P_i, r_i) = 0$, Q interpolates (P_i, r_i). Furthermore, if $\mathcal{D}_{\alpha\beta}^{(P_i,r_i)}(Q) = 0$, $\forall\ \alpha+\beta < m$, Q has a zero of multiplicity m at (P_i, r_i). Eq. (18) implies an interpolation constraint on Q (coefficients Q_{ab} should satisfy (18)). Since there are $\binom{m+1}{2}$ pairs of (α, β) that satisfy $\alpha + \beta < m$, interpolating n points implies $n\binom{m+1}{2}$ interpolation constraints. Eq. (18) also shows the corresponding coefficients $\xi_{a,P_i,\alpha}$ are critical in determining the interpolation property of a polynomial. With Z_{P_∞} and Z_{P_i}, they can be determined prior to the construction of Q using Algorithm 6.1 of [17]. Consequently, the following interpolation can be facilitated.

C. The Interpolation Algorithm

It is now sufficient to introduce the interpolation algorithm for constructing Q. Given χ_{EL}, we know $\mathcal{R} = \mathbb{F}_q[X, Y]/ < Y^2 + \mathfrak{a}_1 XY + \mathfrak{a}_3 Y - X^3 - \mathfrak{a}_2 X^2 - \mathfrak{a}_4 X - \mathfrak{a}_6 >= \mathbb{F}_q[x, y]$, which is a free module over $\mathbb{F}_q[x]$ of rank 2 with a free basis $\{1, y\}$, where x and y denote the residue classes of X and Y, respectively. Therefore, $\mathcal{R}[z]_{l_m}$ can be seen as a free module of rank $2(l_m + 1)$ over $\mathbb{F}_q[x]$. At the beginning, a group of polynomials can be initialized as

$$\begin{aligned} \mathcal{G} &= \{\mathcal{Q}_{\mu+2\nu} = y^\mu z^\nu \mid \mu = 0, 1, \nu = 0, 1, \ldots, l_m\} \\ &= \{1,\ y,\ z,\ yz, \ldots,\ z^{l_m},\ yz^{l_m}\}. \end{aligned} \tag{19}$$

For each polynomial $\mathcal{Q}_t$ $(t = 0, 1, \ldots, 2l_m + 1)$, its (α, β)-Hasse derivative evaluation $\mathcal{D}_{\alpha\beta}^{(P_i,r_i)}(\mathcal{Q}_t)$ will be determined as in (18). If $\mathcal{D}_{\alpha\beta}^{(P_i,r_i)}(\mathcal{Q}_t) = 0$, $\mathcal{Q}_t$ satisfies the current constraint. It does not need to be modified. For those with $\mathcal{D}_{\alpha\beta}^{(P_i,r_i)}(\mathcal{Q}_t) \neq 0$, the minimum one will be picked up as

$$t' = \text{index}\left(\min\{\mathcal{Q}_t \mid \mathcal{D}_{\alpha\beta}^{(P_i,r_i)}(\mathcal{Q}_t) \neq 0\}\right) \tag{20}$$

and let

$$\mathcal{Q}^* = \mathcal{Q}_{t'}. \tag{21}$$

For polynomials $\mathcal{Q}_t$ with $\mathcal{D}_{\alpha\beta}^{(P_i,r_i)}(\mathcal{Q}_t) \neq 0$ and $t \neq t'$, they are updated by

$$\mathcal{Q}'_t = \mathcal{D}_{\alpha\beta}^{(P_i,r_i)}(\mathcal{Q}_t)\mathcal{Q}^* - \mathcal{D}_{\alpha\beta}^{(P_i,r_i)}(\mathcal{Q}^*)\mathcal{Q}_t. \tag{22}$$

Note that $\mathcal{D}_{\alpha\beta}^{(P_i,r_i)}(\mathcal{Q}'_t) = \mathcal{D}_{\alpha\beta}^{(P_i,r_i)}(\mathcal{Q}_t)\mathcal{D}_{\alpha\beta}^{(P_i,r_i)}(\mathcal{Q}^*) - \mathcal{D}_{\alpha\beta}^{(P_i,r_i)}(\mathcal{Q}^*)\mathcal{D}_{\alpha\beta}^{(P_i,r_i)}(\mathcal{Q}_t) = 0$. $\mathcal{Q}'_t$ satisfies the current constraint. For $\mathcal{Q}_{t'}$, it will be updated by

$$\mathcal{Q}'_{t'} = (x - x_i)\mathcal{Q}^*. \tag{23}$$

Again, $\mathcal{D}_{\alpha\beta}^{(P_i,r_i)}(\mathcal{Q}'_{t'}) = \mathcal{D}_{\alpha\beta}^{(P_i,r_i)}(x\mathcal{Q}^*) - x_i\mathcal{D}_{\alpha\beta}^{(P_i,r_i)}(\mathcal{Q}^*) = x_i\mathcal{D}_{\alpha\beta}^{(P_i,r_i)}(\mathcal{Q}^*) - x_i\mathcal{D}_{\alpha\beta}^{(P_i,r_i)}(\mathcal{Q}^*) = 0$.

The above test-and-update process iterates $n\binom{m+1}{2}$ times so that all interpolation constraints are satisfied. Finally, the minimum polynomial of the updated group is chosen as the interpolated polynomial Q as

$$Q = \min\{\mathcal{Q}'_t \mid \mathcal{Q}'_t \in \mathcal{G}\}. \tag{24}$$

Summarizing the above description, the interpolation algorithm for GS decoding of elliptic codes is stated as follows.

Algorithm 1 The Interpolation Algorithm

Input: $\underline{r}$ and m;
Output: Q;

```
Initialize G as in (19);
For each interpolation constraint do
    Compute D_{αβ}^{(P_i,r_i)}(Q_t), ∀Q_t ∈ G as in (18);
    For Q_t with D_{αβ}^{(P_i,r_i)}(Q_t) ≠ 0 do
        Pick up Q* as in (20) and (21);
        Update them as in (22) and (23);
    End for
End for
```

This is Koetter's iterative polynomial construction approach [18]. Based on Theorem 1, the message polynomial f can be further decoded by finding the z-roots of Q, i.e., $Q(x, y, f) = 0$. This can be implemented by the root-finding algorithm of [19], which determines $f_{k-1}, f_{k-2}, \ldots, f_0$ in a recursive manner.

V. DECODING PERFORMANCE

GS decoding of elliptic codes have been simulated over the additive white Gaussian noise (AWGN) channel using BPSK modulation. They are compared with similar rate RS codes that are defined over the same finite field. In our simulations, the decoding is considered to be successful if message polynomial f is included in the decoding output list.

Figs.1 and 2 show the frame error rate (FER) performance of the $(80, 27)$ and the $(288, 163)$ elliptic codes, respectively. They have been introduced in Example 2. Our results show with the same interpolation multiplicity, an elliptic code can outperform a similar rate RS code. This is because the elliptic codes have a greater error-correction capability, also yielding a better asymptotic FER performance. For RS codes, the minimum multiplicities are chosen for the GS decoding. Over the finite field, elliptic codes are longer yielding a greater error-correction capability. It is interesting to point out that this also enables the elliptic codes achieve a similar performance as RS codes but with a smaller decoding complexity. Section IV.C shows there are $\mathfrak{C} = n\binom{m+1}{2}$ iterations for constructing Q. Its complexity can be approximated as $\frac{5}{2}(l_m+1)(\mathfrak{C}^2+\mathfrak{C})$. Hence, $\mathfrak{C}$ defines the interpolation complexity. Fig.1 shows decoding the $(80, 27)$ elliptic code with $m = 4$ performs similarly as decoding the $(63, 21)$ RS code with $m = 5$. However, for the elliptic code $\mathfrak{C} = 800$ and for the RS code $\mathfrak{C} = 945$. A similar phenomenon can also be observed in Fig.2. Note that the optimal GS decoding performances are obtained by assuming the algorithm can decode τ_{GS} errors, where τ_{GS} is defined by (9). For RS codes, $g = 0$.

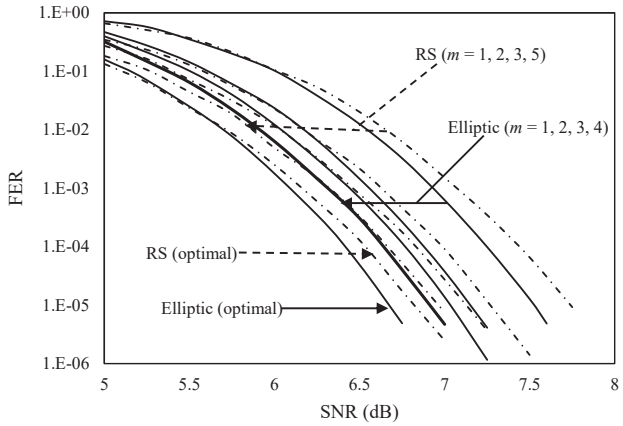

Fig. 1. Performance of the (80, 27) elliptic code and the (63, 21) RS code.

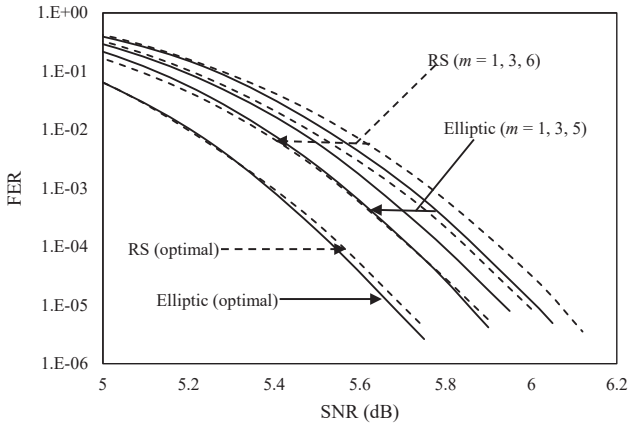

Fig. 2. Performance of the (288, 163) elliptic code and the (255, 144) RS code.

VI. Conclusion

This paper has proposed the GS list decoding algorithm for elliptic codes. An explicit zero basis construction has been proposed for each affine point of the curve, so that the interpolated polynomial can be defined. The error-correction capability and the maximum decoding output cardinality have also been characterized. Finally, an efficient interpolation algorithm has been proposed. Our simulation results have demonstrated elliptic codes can outperform the similar rate RS codes, offering a new option for data communication systems. This work also underpins the development of more advanced decoding for elliptic codes, e.g., the soft-decision list decoding. This will be the authors' future work.

Acknowledgement

This work is sponsored by the National Natural Science Foundation of China (NSFC) with project ID 61671486.

References

[1] V. Goppa, "Codes on algebraic curves," *Soviet Math. Doklady*, vol. 24(1), pp. 170–172, 1981.

[2] Y. Driencourt, "Some properties of elliptic codes over a field of characteristic 2," in *Proc. 3rd Int. Conf. AAECC*. Springer, Berlin, Heidelberg, 1985, pp. 185–193.

[3] J. Justesen, K. Larsen, H. Jensen, A. Havemose, and T. Høholdt, "Construction and decoding of a class of algebraic geometry codes," *IEEE Trans. Inf. Theory*, vol. 35(4), pp. 811–821, Jul. 1989.

[4] A. Skorobogatov and S. Vladut, "On the decoding of algebraic-geometry codes," *IEEE Trans. Inf. Theory*, vol. 36(5), pp. 1051–1060, Sep. 1990.

[5] G. Feng and T. Rao, "Decoding algebraic-geometric codes up to the designed minimum distance," *IEEE Trans. Inf. Theory,*, vol. 39(1), pp. 37–46, Jan. 1993.

[6] S. Sakata, J. Justesen, Y. Madelung, H. Jensen, and T. Høholdt, "Fast decoding of algebraic-geometric codes up to the designed minimum distance," *IEEE Trans. Inf. Theory*, vol. 41(6), pp. 1672–1677, Nov. 1995.

[7] M. Johnston and R. Carrasco, "Construction and performance of algebraic-geometric codes over awgn and fading channels," *IEE Proc. Commun.*, vol. 152(5), pp. 713–722, Oct. 2005.

[8] M. Sudan, "Decoding of Reed-Solomon codes beyond the error-correction bound," *J. Complexity*, vol. 13(1), pp. 180–193, Mar. 1997.

[9] M. Shokrollahi and H. Wasserman, "List decoding of algebraic-geometric codes," *IEEE Trans. Inf. Theory*, vol. 45(2), pp. 432–437, Mar. 1999.

[10] V. Guruswami and M. Sudan, "Improved decoding of Reed-Solomon and algebraic-geometry codes," *IEEE Trans. Inf. Theory*, vol. 45(6), pp. 1757–1767, Sep. 1999.

[11] T. Høholdt and R. Nielsen, "Decoding Hermitian codes with Sudan's algorithm," in *AAECC (Lect. Notes Comput. Sci.)*, vol. 1719. Germany, Berlin:Springer-Verlag, 1999, pp. 260–269.

[12] L. Chen, R. Carrasco, and M. Johnston, "Soft-decision list decoding of Hermitian codes," *IEEE Trans. Commun.*, vol. 57(8), pp. 2169–2176, Aug. 2009.

[13] K. Lee and M. O'Sullivan, "Algebraic soft-decision decoding of Hermitian codes," *IEEE Trans. Inf. Theory*, vol. 56(6), pp. 2587–2600, Jun. 2010.

[14] F. Zhang and S. Liu, "Solving ECDLP via list decoding," *IACR Cryptology ePrint Archive, Report2018/795*, 2018.

[15] L. Washington, *Elliptic curves: number theory and cryptography*. Chapman and Hall/CRC, 2003.

[16] H. Stichtenoth, *Algebraic function fields and codes*. Berlin, Germany: Springer, 1993.

[17] L. Chen, "Design of an efficient list decoding system for Reed-Solomon and algebraic-geometric codes," Ph.D. dissertation, Dept. Electron. Comput. Eng., Newcastle Univ., Newcastle-upon-Tyne, U.K., 2008.

[18] R. Koetter, "On algebraic decoding of algebraic-geometric and cyclic codes," Ph.D. dissertation, Linköping Univ. , Linköping, Sweden, 1996.

[19] X. Wu and P. Siegel, "Efficient root-finding algorithm with application to list decoding of algebraic-geometric codes," *IEEE Trans. Inf. Theory*, vol. 47(6), pp. 2579–2587, Sep. 2001.