MINIMUM DISTANCE DECODING OF GENERAL ALGEBRAIC GEOMETRY CODES VIA LISTS

NATHAN DRAKE AND GRETCHEN L. MATTHEWS
DEPARTMENT OF MATHEMATICAL SCIENCES
CLEMSON UNIVERSITY
CLEMSON, SC 29634-0975
U.S.A.
E-MAIL: GMATTHE@CLEMSON.EDU

Abstract. Algebraic geometry codes are defined by divisors D and G on a curve over a finite field $\mathbb{F}$. Often, G is supported by a single $\mathbb{F}$-rational point and the resulting code is called a one-point code. Recently, there has been interest in allowing the divisor G to be more general as this can result in superior codes. In particular, one may obtain a code with better parameters by allowing G to be supported by m distinct $\mathbb{F}$-rational points, where $m > 1$ In this paper, we demonstrate that a multipoint algebraic geometry code C may be embedded in a one-point code C'. Exploiting this fact, we obtain a minimum distance decoding algorithm for the multipoint code C. This is accomplished via list decoding in the one-point code C'.

1. Introduction

Algebraic geometry codes (AG codes) are defined by divisors D and G on a curve over a finite field $\mathbb{F}$. Often, G is supported by a single $\mathbb{F}$-rational point and the resulting code is called a one-point code. Recently, there has been interest in allowing the divisor G to be more general as this can result in superior codes [2, 4, 5, 26, 35, 36]. In particular, one may obtain a code with better parameters by allowing G to be supported by m distinct $\mathbb{F}$-rational points, where $m > 1$ [6, 15, 16, 17, 23, 24, 25]. We refer to such a code as a multipoint code. While multipoint codes may have better parameters than comparable one-point codes on the same curve, most decoding algorithms have been designed specifically for one-point codes; those that apply to general algebraic geometry codes tend to decode up to a bound on the minimum distance (such as the Goppa bound or the order bound) rather than the minimum distance itself. For example, Beelen's adaptation of majority voting (see [3] and references therein) and the modification of the Berlekamp-Massey-Sakata Algorithm [29] due to Sakata and and Fujisawa [28], decode up to the generalized order bound, a lower bound on the minimum distance; however, the generalized order bound does not agree with the actual minimum distance in general (though it does in certain cases, for example, for two-point Hermitian codes). The informative survey on decoding algebraic geometry codes [3] includes a decoding scheme which allows the possibility of correcting errors beyond the order bound. In this paper, we provide a simple algorithm for decoding

Key words and phrases. algebraic geometry code, list decoding, minimum distance decoding, multipoint code.

This project was supported by NSF DMS-0201286 and NSA H-98230-06-1-0008.

a multipoint code up to its minimum distance by embedding the multipoint code in a one-point code and list decoding in a supercode. This algorithm also applies to general AG codes defined by a rational divisors G and D, provided not all rational points are in the support of D.

While list decoding dates back to the late 1950's [8, 9, 34], its power was not fully exploited for over 40 years. This changed with Sudan's observation that list decoding could be applied to Reed-Solomon codes to give a polynomial time algorithm which decodes beyond the minimum distance of the code (meaning more than $\lfloor \frac{d-1}{2} \rfloor$ errors) [33]. A major breakthrough in decoding AG codes came with the generalizations of Sudan's algorithm to arbitrary AG codes of low rate by Shokrollahi and Wasserman [30] and to one-point AG codes by Guruswami and Sudan [11]. While the results in [30] apply to general AG codes of restricted rate, nearly all subsequent improvements (including, for example, [10, 19, 20, 21, 27]) are restricted to the one-point case. By embedding a multipoint code into a one-point code, we can capitalize on these and provide a minimum distance decoder for a multipoint code.

This paper is organized as follows. This section concludes with a summary of notation to be used throughout the paper. Section 2 demonstrates how a general AG code (in particular, a multipoint code) may be embedded in a one-point code. Section 3 presents a decoding algorithm for such codes based on a list decoding algorithm for one-point codes. In Section 4, this algorithm is modified to handle multipoint codes embedding in multiple one-point supercodes.

Notation. Let X be a projective curve of genus g over a finite field $\mathbb{F}$. Let $\mathbb{F}(X)$ denote the field of rational functions on X defined over $\mathbb{F}$. The divisor of a nonzero rational function f is denoted by (f). The coefficient of a point P on X in a divisor A on X is written as $v_P(A)$, or $v_P(f)$ if $A = (f)$ for some rational function f. Given a divisor A on X defined over $\mathbb{F}$, let $\mathcal{L}(A)$ denote the set of rational functions f on X defined over $\mathbb{F}$ with divisor $(f) \geq -A$ together with the zero function. We often use the fact that given divisors A and B on X with $A \leq B$, $\mathcal{L}(A) \subseteq \mathcal{L}(B)$. Let $\ell(A)$ denote the dimension of $\mathcal{L}(A)$ as an $\mathbb{F}$-vector space. As is standard, given a plane curve X with defining equation $f(x, y) = 0$, P_{ab} denotes the affine point which is the common zero of $x - a$ and $y - b$.

Let G be an $\mathbb{F}$-rational divisor on X and $D = \sum_{i=1}^{n} P_i$ where $P_1, \ldots, P_n$ are pairwise distinct $\mathbb{F}$-rational points on X, none of which are in the support of G. An AG code defined by D and G is

$$C_{\mathcal{L}}(D, G) := \{ev(f) : f \in \mathcal{L}(G)\}$$

where

$$ev(f) := (f(P_1), \ldots, f(P_n)).$$

We refer to such a code as an m-point code if and only if the support of the divisor G consists of m distinct $\mathbb{F}$-rational points. We do not assume that the divisor D is supported by all rational points other than those in the support of G as is sometimes taken to be the case. If the support of G is precisely $Q_1, \ldots, Q_m$, then we say that the code $C_{\mathcal{L}}(D, G)$ is supported by $Q_1, \ldots, Q_m$. If $\deg G < n$, then $C_{\mathcal{L}}(D, G)$ has length n, dimension $\ell(G)$, and designed distance $n - \deg G$. The minimum distance of the code $C_{\mathcal{L}}(D, G)$ is at least its designed distance. We use $d(C)$ (resp., $d^*(C)$) to denote the minimum distance (resp., designed distance) of a code C. A code of length n, dimension k, and minimum distance d (resp. at least d) is called an

$[n,k,d]$ (resp. $[n,k,\geq d]$) code. The Hamming distance between words $w,w'\in\mathbb{F}^n$ is $d(w,w'):=|\{i:w_i\neq w'_i\}|$. A minimum distance decoder for an $[n,k,d]$ code C over $\mathbb{F}$ is a decoder that given a received word $w\in\mathbb{F}^n$ returns the unique codeword $c\in C$ with $d(w,c)<\frac{d(C)}{2}$ if such a codeword exists and declares failure otherwise. General references for algebraic geometry codes are [14, 32].

The set of positive integers is denoted $\mathbb{Z}^+$. As usual, given $v\in\mathbb{F}^n$ where $n\in\mathbb{Z}^+$, the i^{th} coordinate of v is denoted by v_i.

2. Embedding a general AG code in a one-point code

In this section, we demonstrate that a general AG code $C_{\mathcal{L}}(D,G)$ on a curve X over a finite field $\mathbb{F}$ may be embedded in a one-point code provided that the divisor D is not supported by all $\mathbb{F}$-rational points on X. As a consequence, we see that each multipoint code $C_{\mathcal{L}}(D,G)$ embeds in a one-point code, as the divisor G contains rational points in its support. Examples are provided at the end of the section.

Lemma 2.1. *Let X be a nonsingular projective curve over a finite field $\mathbb{F}$. Suppose G is an $\mathbb{F}$-rational divisor and $D:=P_1+\cdots+P_n$ is supported by n distinct $\mathbb{F}$-rational points, none of which are in the support of G. If there exists an $\mathbb{F}$-rational point P on X not in the support of D, then $C_{\mathcal{L}}(D,G)$ is isometric to a subcode of a one-point code $C_{\mathcal{L}}(D,\alpha P)$ on X for some $\alpha\in\mathbb{Z}$.*

Proof. Write $G=G_+-G_-$ where $G_+,G_-\geq 0$. Then

$$\deg\left(G_+-(\deg G_+)P\right)=0.$$

Since the field $\mathbb{F}$ is finite, the group of divisor classes of degree zero has finite order. Hence, some multiple of the divisor $G_+-(\deg G_+)P$ is a principal divisor. Consequently, there exists a rational function f with divisor

$$(f)=\lambda\left(G_+-(\deg G_+)P\right)$$

for some $\lambda\in\mathbb{Z}$. Multiplication by f induces an isomorphism of Riemann-Roch spaces

$$\begin{array}{ccc}\mathcal{L}(G) & \to & \mathcal{L}\left((\lambda\deg G_+)P-(\lambda-1)G_+-G_-\right)\\ h & \mapsto & fh.\end{array}$$

Since

$$\mathcal{L}\left((\lambda\deg G_+)P-(\lambda-1)G_+-G_-\right)\subseteq\mathcal{L}\left((\lambda\deg G_+)P\right),$$

$\mathcal{L}(G)$ is isomorphic to a subspace of $\mathcal{L}\left((\lambda\deg G_+)P\right)$.

Moreover, multiplication by the function f induces a vector space isomorphism

$$\begin{array}{rccc}\phi: & \mathbb{F}_q^n & \to & \mathbb{F}_q^n\\ & v & \mapsto & ev(f)*v\end{array}$$

where

$$ev(f)*v:=(f(P_1)\cdot v_1,\ldots,f(P_n)\cdot v_n).$$

Since f has no zeros among $P_1,\ldots,P_n$, the map ϕ is weight-preserving, and hence, distance-preserving. Thus, restriction of ϕ to $C_{\mathcal{L}}(D,G)$ induces an isometry ϕ of codes

$$C_{\mathcal{L}}(D,G)\overset{\phi}{\cong}C_{\mathcal{L}}\left(D,(\lambda\deg G_+)P-(\lambda-1)G_+-G_-\right) \tag{1}$$

Therefore, $C_{\mathcal{L}}\left(D,(\lambda\deg G_+)P\right)$ contains (an isometric copy of) the code $C_{\mathcal{L}}(D,G)$. □

Lemma 2.1 is crucial for the decoding algorithm presented in Section 3. The following examples provide isometries for commonly studied multipoint codes.

Example 2.2. In this example, we consider Hermitian codes. Recall that the Hermitian curve X is defined by $y^q + y = x^{q+1}$ over $\mathbb{F}_{q^2}$. Since the automorphism group of this curve is doubly-transitive, to study two-point codes, we may restrict our attention to a code of the form $C := C_{\mathcal{L}}(D, aP_\infty + bP_{00})$. Suppose $a, b \in \mathbb{Z}^+$. Then multiplication by $f := y^{\lceil \frac{b}{q+1} \rceil}$ induces a vector space isomorophism

$$\mathcal{L}(D, aP_\infty + bP_{00}) \cong \mathcal{L}\left(\left(a + \left\lceil \frac{b}{q+1} \right\rceil (q+1)\right) P_\infty + \left(b - \left\lceil \frac{b}{q+1} \right\rceil (q+1)\right) P_{00}\right),$$

because $(y) = (q+1)(P_{00} - P_\infty)$. Hence,

$$\begin{aligned} C &\cong C_{\mathcal{L}}\left(D, \left(a + \left\lceil \tfrac{b}{q+1} \right\rceil (q+1)\right) P_\infty - \left(\left\lceil \tfrac{b}{q+1} \right\rceil (q+1) - b\right) P_{00}\right) \\ &\subseteq C_{\mathcal{L}}\left(D, \left(a + \left\lceil \tfrac{b}{q+1} \right\rceil (q+1)\right) P_\infty\right). \end{aligned}$$

Now consider an m-point Hermitian code

$$C := C_{\mathcal{L}}\left(D, a_1 P_\infty + \sum_{i=2}^{m} a_i P_{\alpha\beta_i}\right)$$

suppported by collinear points $P_\infty, P_{\alpha\beta_2}, \ldots, P_{\alpha\beta_m}$ where $a_i \in \mathbb{Z}^+$ for all i, $1 \leq i \leq m$ and $2 \leq m \leq q+1$. Such codes were studied in [22] where the authors show that if $\tau_{\alpha\beta_i} := y - \beta_i - \alpha^q(x - \alpha)$ then $(\tau_{\alpha\beta_i}) = (q+1)(P_{\alpha\beta_i} - P_\infty)$. Thus we can take

$$f = \prod_{i=2}^{m} \tau_{\alpha\beta_i}^{\lceil \frac{a_i}{q+1} \rceil}.$$

The multiplication by f induces a vector space isomorphism

$$\mathcal{L}\left(a_1 P_\infty + \textstyle\sum_{i=2}^{m} a_i P_{\alpha\beta_i}\right) \cong$$

$$\mathcal{L}\left(\left(a_1 + (q+1)\textstyle\sum_{i=2}^{m} \left\lceil \frac{a_i}{q+1} \right\rceil\right) P_\infty + \textstyle\sum_{i=2}^{m} \left(a_i - \left\lceil \frac{a_i}{q+1} \right\rceil (q+1)\right) P_{\alpha\beta_i}\right)$$

and an isometry of codes

$$\begin{aligned} C &\cong C_{\mathcal{L}}\left(D, \left(a_1 + (q+1)\textstyle\sum_{i=2}^{m} \left\lceil \frac{a_i}{q+1} \right\rceil\right) P_\infty - \textstyle\sum_{i=2}^{m} \left(\left\lceil \frac{a_i}{q+1} \right\rceil (q+1) - a_i\right) P_{\alpha\beta_i}\right) \\ &\subseteq C_{\mathcal{L}}\left(D, \left(a_1 + (q+1)\textstyle\sum_{i=2}^{m} \left\lceil \frac{a_i}{q+1} \right\rceil\right) P_\infty\right) \end{aligned}$$

since

$$\begin{aligned} &\mathcal{L}\left(\left(a_1 + (q+1)\textstyle\sum_{i=2}^{m} \left\lceil \frac{a_i}{q+1} \right\rceil\right) P_\infty + \textstyle\sum_{i=2}^{m} \left(a_i - \left\lceil \frac{a_i}{q+1} \right\rceil (q+1)\right) P_{\alpha\beta_i}\right) \\ &\subseteq \mathcal{L}\left(\left(a_1 + (q+1)\textstyle\sum_{i=2}^{m} \left\lceil \frac{a_i}{q+1} \right\rceil\right) P_\infty\right). \end{aligned}$$

Example 2.3. In this example, let $C := C_{\mathcal{L}}(D, aP_\infty + bP_{00})$ be a two-point Suzuki code where $a, b \in \mathbb{Z}^+$. Recall that the Suzuki curve is defined over $\mathbb{F}_q$ by the

equation $y^q - y = x^{q_0}(x^q - x)$ where $q_0 = 2^n$, $q = 2^{2n+1}$, and $n \in \mathbb{Z}^+$. Let $w := y^{\frac{q}{q_0}} x^{\frac{q}{q_0}+1} + \left(y^{\frac{q}{q_0}} - x^{\frac{q}{q_0}+1}\right)^{\frac{q}{q_0}}$. Since

$$(w) = \left(q + \frac{q}{q_0} + 1\right)(P_{00} - P_\infty)$$

as shown in [12], multiplication by

$$f := w^{\left\lceil \frac{b}{q+\frac{q}{q_0}+1} \right\rceil}$$

gives rise to an isomorphism of Riemann-Roch spaces and consequently an isometry of codes

$$C \cong C_{\mathcal{L}}(D, \alpha P_\infty - \beta P_{00}) \subseteq C_{\mathcal{L}}(D, \alpha P_\infty)$$

where

$$\alpha = a + \left\lceil \frac{b}{q+\frac{q}{q_0}+1} \right\rceil \left(q + \frac{q}{q_0} + 1\right) \text{ and } \beta = \left\lceil \frac{b}{q+\frac{q}{q_0}+1} \right\rceil \left(q + \frac{q}{q_0} + 1\right) - b.$$

Remark 2.4. *Determination of a suitable function $f \in \mathbb{F}(X)$ as in the proof of Lemma 2.1 may be found as follows. Consider $\ell(\lambda(G_+ - \deg G_+ P))$ for increasing $\lambda \in \mathbb{Z}^+$ until one is found with $\ell(\lambda(G_+ - \deg G_+ P)) \neq 0$. Then compute a basis of $\mathcal{L}(\lambda(G_+ - \deg G_+ P))$ to find such an f. Hess* [13] *gives an algorithm for effectively computing bases of Riemann-Roch spaces. For certain divisors on maximal or optimal function fields, bases of such spaces are known* [22].

3. A minimum distance decoder for general AG codes

In this section, we outline the decoding algorithm for general AG codes $C_{\mathcal{L}}(D, G)$, where not all rational points are in the support of D. Note that this algorithm applies to multipoint codes and that the point of view taken here can be utilized with any list decoding algorithm for one-point codes. For clarity of exposition, we focus on the Guruswami-Sudan list decoding algorithm as found in [11, Section IV. B.].

Consider a nonzero AG code $C := C_{\mathcal{L}}(D, G)$ on a nonsingular projective curve X of genus g over a field $\mathbb{F}$ where $D := P_1 + \cdots + P_n$, and assume that there is an $\mathbb{F}$-rational point P on X not in the support of D. For the purpose of decoding it is sufficient to consider the code $\phi(C)$ where ϕ is as in (1). To see this, suppose that $w \in \mathbb{F}_q^n$ is a received word using the code C and that E errors have occurred, where $E \leq \left\lfloor \frac{d(C)-1}{2} \right\rfloor$. We may identify w and $ev(f) * w$ via the map ϕ. Then $ev(f) * w$ is treated as a received word using $\phi(C)$. Since ϕ is distance preserving, $E \leq \left\lfloor \frac{d(\phi(C))-1}{2} \right\rfloor$. Hence, there is a unique nearest codeword $ev(h) \in \phi(C)$ to $ev(f) * w$. It follows that $ev\left(f^{-1}h\right)$ is the unique codeword in C nearest w. Consequently, throughout the remainder of this section, we assume that $G = \alpha P - G'$ where $\alpha \in \mathbb{Z}^+$, P is an $\mathbb{F}$-rational point not in the support of D, and $G' > 0$.

Algorithm 3.1. Let $C := C_{\mathcal{L}}(P_1 + \cdots + P_n, \alpha P - G')$ be a nonzero AG code over the field $\mathbb{F}_q$. Suppose that $w \in \mathbb{F}_q^n$ is a received word in which $\left\lfloor \frac{d(C)-1}{2} \right\rfloor$ or fewer errors have occurred.

Input: $P_1, \ldots, P_n$, P, α, G', received word $w \in \mathbb{F}_q^n$, agreement parameter $t :=$

$n - \left\lfloor \frac{d(C)-1}{2} \right\rfloor$.
Assumptions: $t^2 > \alpha n$.

(1) Initialization: Set

$$r := \left\lfloor \frac{2gt + \alpha n + \sqrt{(2gt + \alpha n)^2 - 4(g^2 - 1)(t^2 - \alpha n)}}{2(t^2 - \alpha n)} \right\rfloor + 1,$$

$$s := \left\lfloor \frac{rt - 1 - g}{\alpha} \right\rfloor,$$

and $\Omega' := \emptyset$.
(2) Interpolation: Find a nonzero polynomial $Q(T) \in \mathbb{F}_q(X)[T]$ satisfying:
  (a) $Q(f) \in \mathcal{L}((rt-1)P)$ for all $f \in \mathcal{L}(\alpha P)$, and
  (b) $v_{P_i}(Q(f)) \geq r$ for each $i \in \{1, \ldots, n\}$ with $f(P_i) = w_i$.
(3) Factorization: Find all roots $h \in \mathcal{L}(\alpha P)$ of the polynomial Q. For each such h, if $h(P_i) = w_i$ for at least t values of i, then add h to Ω'. In this way, we find all functions $h \in \mathcal{L}(\alpha P)$ that possibly give rise to the codewords in $C' := C_{\mathcal{L}}(D, \alpha P)$ at distance at most $\left\lfloor \frac{d(C)-1}{2} \right\rfloor$ from w; that is, $\Omega := \{h \in \mathcal{L}(\alpha P) : d(ev(h), w) \leq n - t\}$ is determined.
(4) Check for zeros: Compute the order of h at Q_i for each $h \in \Omega$ until the one is found with $v_{Q_i}(h) \geq v_{Q_i}(G')$ for all Q_i in the support of G'.

Output: $ev(h)$, the unique word in C with $d(ev(h), w) \leq \left\lfloor \frac{d(C)-1}{2} \right\rfloor$.

Remark 3.2. (1) *Step (2) of Algorithm 3.1 produces a polynomial Q which fits the points (P_i, w_i) in the sense that $Q(w_i)(P_i) = 0$ for all i, $1 \leq i \leq n$. Hence, the elements of Ω are among the roots of Q. See* [11] *for further discussion on this as well as the correctness of Steps (1)-(3).*

(2) *Steps (1)-(3) of Algorithm 3.1 may be replaced with those of any list decoding algorithm for $C_{\mathcal{L}}(D, \alpha P)$ which yields Ω as its output.*

(3) *According to* [7]*, the parameters r and s may be replaced with the following values in Step (1):*

$$r := \left\lfloor \frac{\alpha(n-t) + 2tg + \sqrt{\Delta}}{2(t^2 - \alpha n)} \right\rfloor + 1 \text{ and } s := \left\lfloor \frac{rt - \frac{\alpha}{2} - g - \sqrt{\Delta'}}{\alpha} \right\rfloor + 1$$

where

$$\Delta := \alpha^2 \left((n-t)^2 - 4gn \right) + 4\alpha gn(t+g)$$

and

$$\Delta' := \left(t^2 - \alpha n\right) r^2 + (\alpha t - \alpha n - 2tg) r + \frac{\alpha^2}{4} + g^2 - \alpha g.$$

This may result in a lower degree interpolating polynomial Q, as $\deg Q \leq s$*. In addition, it provides a better bound on $|\Omega|$ and the number of functions that must be checked in Step (4).*

(4) *The goal of Step (4) may be achieved via parity checks. Specifically, given a parity check matrix H for C, compute $Hev(h)^T$ for each $h \in \Omega$ until one is found with $Hev(h)^T = 0$. Alternatively, one could determine the additional parity checks $v_1, \ldots, v_r$ that words in C' must satisfy to be in the subcode*

C. Then, for each h found in Step (3), compute $ev(h) * v_i$, $1 \leq i \leq r$, until an h is found satisfying all r checks.

(5) *It is natural to compare Algorithm 3.1 with majority voting as in* [3]*. However, majority voting only corrects up to $\left\lfloor \frac{d_o(C)-1}{2} \right\rfloor$ errors, where $d_o(C)$ denotes the order bound on C. Because the only step in Algorithm 3.1 beyond list decoding is the check for zeros (or parity checks as mentioned in (4) above), this algorithm compares as favorably to majority voting as the underlying algorithm for list decoding in the one-point code (the choices for which seem to be ever-improving, for instance* [21]*) and has the advantage of correcting up to $\left\lfloor \frac{d(C)-1}{2} \right\rfloor$ errors. Due to the nature of Step (4) and Remark 3.2 (4) above, it is advantageous to minimize the difference in the dimension of C' and C.*

Example 3.3. Let X denote the Hermitian curve defined by $y^8 + y = x^9$ over $\mathbb{F}_{64}$. Then X has genus $g = 28$. Consider the two-point code $C := C_{\mathcal{L}}(D, 344P_\infty - 8P_{00})$ where $D := P_1 + \cdots + P_{511}$ is the sum of all $\mathbb{F}_{64}$-rational points other than P_∞ and P_{00}. According to [17], C is a $[511, 309, 175]$ code, and, thus can correct any 87 errors. Suppose $w \in \mathbb{F}_{64}^{511}$ is a received word in which 87 errors have occured. To decode w, embed C in $C' := C_{\mathcal{L}}(D, 344P_\infty)$, a $[511, 317, 167]$ code [15]. Applying Steps (1)-(3) in Algorithm 3.1 with the parameter choices given in Remark 3.2(3) produces a list of (at most) 21 functions $h_1, \ldots, h_{21} \in \mathcal{L}(344P_\infty)$ with

$$d(w, ev(h_i)) \leq 87. \tag{2}$$

There is a unique $h_i \in \mathcal{L}(344P_\infty - 8P_{00})$ satisfying Equation 2. To determine this function, evaluate $v_{P_{00}}(h_i)$ for $1 \leq i \leq 21$ until $h \in \Omega$ is found so that $v_{P_{00}}(h) \geq 8$. Then decode w as $ev(h) = (h(P_1), h(P_2), \ldots, h(P_{511}))$.

4. A minimum distance decoder for multipoint codes using lists, multiple embeddings, and GCD

In this section, we discuss a modification of Algorithm 3.1 in which a multipoint code is embedded in multiple one-point codes and the interpolating polynomial is obtained as a greatest common divisor. This idea was inspired by [1].

Consider a multipoint code $C := C_{\mathcal{L}}(D, \sum_{i=1}^m a_i Q_i)$ where $a_i \in \mathbb{Z}$. Given any function f whose divisor is supported only by points among $Q_1, \ldots, Q_m$, multiplication by f induces a vector space isomorphsim

$$\mathcal{L}\left(\sum_{i=1}^m a_i Q_i\right) \cong \mathcal{L}\left(\sum_{i=1}^m (a_i - v_{Q_i}(f)) Q_i\right)$$

and an isometry of codes

$$C \overset{\phi}{\cong} C_{\mathcal{L}}\left(D, \sum_{i=1}^m (a_i - v_{Q_i}(f)) Q_i\right).$$

Hence, for each such function f with $v_{Q_j}(f) < a_j$ for exactly one j, $1 \leq j \leq m$, C is isometric to a subcode of the one-point code $C_{\mathcal{L}}\left(D, \left(a_j - v_{Q_j}(f)\right) Q_j\right)$; that is,

$$\phi(C) \subseteq C_{\mathcal{L}}\left(D, \left(a_j - v_{Q_j}(f)\right) Q_j\right).$$

To emphasize that the embedding is induced by

$$f \in \mathcal{L}\left(\left(a_j - v_{Q_j}(f)\right) Q_j\right),$$

we sometimes write ϕ_j instead of ϕ. The following algorithm exploits these multiple embeddings.

Algorithm 4.1. Let $C := C_{\mathcal{L}}\left(D, \sum_{i=1}^{m} a_i Q_i\right)$ be an m-point code over the finite field $\mathbb{F}_q$ where $D := P_1 + \cdots + P_n$. Suppose that $w \in \mathbb{F}_q^n$ is a received word in which $\left\lfloor \frac{d(C)-1}{2} \right\rfloor$ or fewer errors have occurred.

Input: n, $a_1, \ldots, a_m$, received word $w \in \mathbb{F}_q^n$, agreement parameter $t := n - \left\lfloor \frac{d(C)-1}{2} \right\rfloor$

(1) Embedding: Choose a nonempty subset $J \subseteq \{1, \ldots, m\}$. For each $j \in J$, find a one-point code $C_j := C_{\mathcal{L}}\left(D, (a_j - b_{jj})\, Q_j\right)$ such that

$$C \overset{\phi_j}{\cong} C_{\mathcal{L}}\left(D, (a_j - b_{jj})\, Q_j - \sum_{\substack{1 \le i \le m \\ i \ne j}} (b_{ij} - a_i) Q_i\right) \subseteq C_j$$

is the embedding induced by a rational function f_j whose divisor is supported by no points other than $Q_1, \ldots, Q_m$ with $v_{Q_i}(f_j) = b_{ij}$ for all $1 \le i \le m$, $b_{ij} \ge a_i$ for all $i \ne j$, $b_{jj} < a_j$ and $t^2 > (a_j - b_{jj})\, n$. Set $\Omega := \emptyset$.

(2) Interpolation: For each $j \in J$, apply Steps (1) and (2) of Algorithm 3.1 to C_j with received word $\phi_j(w)$ to yield nonzero polynomials $H_j(T) \in \mathbb{F}_q(X)[T]$ satisfying Conditions (a) and (b) of Step (2). Set

$$Q(T) := gcd\left\{H_j(f_j T) : j \in J\right\}.$$

(3) Factorization: Find the roots of $Q(T)$ as in the standard factorization step. If a root h of $Q(T)$ satisfies $h(P_i) = w_i$ for at least t values of i, $1 \le i \le m$, then add h to Ω. In this way, we obtain

$$\Omega = \left\{ h \in \mathcal{L}\left(\sum_{j \in J} a_j Q_j + \sum_{\substack{1 \le i \le m \\ i \notin J}} b_i Q_i\right) : d\left(ev(h), w\right) \le \left\lfloor \frac{d(C) - 1}{2} \right\rfloor \right\},$$

that is, we find all functions that possibly give rise to the codewords in C at distance $\left\lfloor \frac{d-1}{2} \right\rfloor$ from w, where $b_i = \min\{b_{ij} : j \notin J\}$.

(4) Check for zeros: Compute the order of h at Q_i for each h found in Step (4) until the one is found with $v_{Q_i}(h) \ge -a_i$ for all $i \notin J$.

Output: $ev(h)$, the unique word in C with $d\left(ev(h), w\right) \le \left\lfloor \frac{d(C)-1}{2} \right\rfloor$.

Theorem 4.2. *Given a multipoint code $C := C_{\mathcal{L}}\left(D, \sum_{i=1}^{m} a_i Q_i\right)$ as above, Algorithm 4.1 provides a mininum distance decoder for C.*

Proof. Suppose that w is a received word in which at most $\left\lfloor \frac{d(C)-1}{2} \right\rfloor$ errors have occured. Then there exists a unique codeword $ev(h)$ that is the transmitted word (resulting in the received word w). We must show that the output of Algorithm 4.1 is $ev(h)$.

Assume $h \in \mathcal{L}\left(\sum_{i=1}^{m} a_i Q_i\right)$ and $d\left(ev(h), w\right) \le \left\lfloor \frac{d(C)-1}{2} \right\rfloor$. We claim that h is a root of $Q(T)$. To see this, we show that h is a root of $H_j(f_j T)$ for all $j \in J$. Note

that among the roots of $H_j(T)$ are elements of

$$\Omega'_j := \left\{ f \in L\left(\left(a_j - b_{jj}\right) Q_j\right) : d\left(ev\left(f\right), \phi_j\left(w\right)\right) \leq \left\lfloor \frac{d\left(C\right) - 1}{2} \right\rfloor \right\}.$$

We prove that $f_j h \in \Omega'_j$ for all $j \in J$. Let $j \in J$. It is immediate that $f_j \in L\left(\left(a_j - b_{jj}\right) P_j\right)$ by definition of f_j. Since $ev\left(f_j h\right) = \phi_j\left(ev\left(h\right)\right)$ and ϕ_j is distance preserving,

$$d\left(ev\left(f_j h\right), \phi_j\left(w\right)\right) = d\left(ev\left(h\right), w\right) \leq \left\lfloor \frac{d\left(C\right) - 1}{2} \right\rfloor.$$

Hence, h is a root of $Q(T)$ and so will be found in Step (4) of Algorithm 4.1. □

As seen in Algorithm 4.1 and Theorem 4.2, multiple embeddings and greatest common divisor may be combined to yield an interpolating polynomial Q of smaller degree. While these multiple embeddings may be advantageous in the root-finding step of list decoding, the cost of calcuating multiple interpolating polynomials to produce Q may outweigh the benefits.

5. Conclusion

In this paper, we present a minimum distance decoding algorithm for a general AG code $C_{\mathcal{L}}(D, G)$ on a curve X over a finite field $\mathbb{F}$, provided there is an $\mathbb{F}$-rational point on X not in the support of D. This decoding algorithm applies to all multipoint codes. It relies on embedding the code $C_{\mathcal{L}}(D, G)$ into a one-point code $C_{\mathcal{L}}(D, \alpha P)$ and applying list decoding to the one-point code. A method for utilizing multiple embeddings and greatest common divisor is also presented.

Acknowledgements. The authors thank the referees for suggestions that improved the content and presentation of this work.

References

[1] A. Barg, E. Krouk, H. C. van Tilborg, On the complexity of minimum distance decoding of long linear codes. IEEE Trans. Inform. Theory **45** (1999), no. 5, 1392–1405.
[2] P. Beelen, The order bound for general AG codes, Finite Fields Appl. **13** (2007), no. 3, 665–680.
[3] P. Beelen and T. Hoholdt, The decoding of algebraic geometry codes, in Advances in Algebraic Geometry Codes, Series on Coding Theory and Cryptology (World Scientific) **5**, E. Martinez-Moro, C. Munuera, and D. Ruano, eds., 49–98.
[4] C. Carvalho, C. Munuera, E. da Silva, and F. Torres, Near orders and codes, IEEE Trans. Inform. Theory **53** (2007), no. 5, 1919–1924.
[5] C. Carvalho and F. Torres, On Goppa codes and Weierstrass gaps at several points. Des. Codes Cryptogr. **35** (2005), no. 2, 211–225.
[6] I. Duursma and R. Kirov, An extension of the order bound for AG codes, preprint.
[7] N. Drake and G. L. Matthews, Parameter choices and a better bound on the list size in the Guruswami-Sudan algorithm for algebraic geometry codes, Des. Codes Cryptogr., to appear.
[8] P. Elias, List decoding for noisy channels, Tech Rep. 335, Res. Lab. Electron., MIT, Cambridge, MA, 1957.
[9] P. Elias, Error-correcting codes for list decoding, IEEE Trans. Inform. Theory **37** (2001), 5–12.
[10] V. Guruswami and M. Sudan, On representations of algebraic-geometry codes, IEEE Trans. Inform. Theory **47** (2001), no. 4, 1610–1613.
[11] V. Guruswami and M. Sudan, Improved decoding of Reed-Solomon and algebraic-geometric codes, IEEE Trans. Inform. Theory **45** (1999), 1757–1767.
[12] J. Hansen and H. Stichtenoth, Group codes on certain algebraic curves with many rational points, Appl. Algebra Engrg. Comm. Comput. **1** (1990), no. 1, 67–77.

[13] F. Hess, Computing Riemann-Roch spaces in algebraic function fields and related topics, J. Symbolic Comput. **33** (2002), no. 4, 425–445.
[14] T. Høholdt, J. H. van Lint, and R. Pellikaan, Algebraic geometry codes, in Handbook of Coding Theory, V. Pless, W. C. Huffman, and R. A. Brualdi, Eds., **1**, Elsevier, Amsterdam (1998), 871–961.
[15] M. Homma and S. J. Kim, Toward the determination of the minimum distance of two-point codes on a Hermitian curve, Des. Codes Cryptogr. **37** (2005), no. 1, 111–132.
[16] M. Homma and S. J. Kim, The two-point codes on a Hermitian curve with designed minimum distance, Des. Codes Cryptogr. **38** (2006), no. 1, 55–81.
[17] M. Homma and S. J. Kim, The two-point codes with the designed minimum distance on a Hermitian curve in even characteristic, Des. Codes Cryptogr. **39** (2006), no. 3, 375–386.
[18] M. Homma and S. J. Kim, The complete determination of the minimum distance of two-point codes on a Hermitian curve, Des. Codes Cryptogr. **40** (2006), no. 1, 5–24.
[19] H. O'Keeffe and P. Fitzpatrick, Gröbner basis solutions of constrained interpolation problems, Linear Algebra Appl. **351/352** (2002), 533–551.
[20] R. Koetter and A. Vardy, Algebraic soft-decision decoding of Reed-Solomon codes. IEEE Trans. Inform. Theory **49** (2003), no. 11, 2809–2825.
[21] K. Lee and M. E. O'Sullivan, List decoding of Hermitian codes using Gröbner bases, J. Symbolic Comput., to appear.
[22] H. Maharaj, G. Matthews, and G. Pirsic, Riemann-Roch spaces of the Hermitian function field with applications to algebraic geometry codes and low-discrepancy sequences, J. Pure Appl. Algebra, **195** (2005), no. 3, 261–280.
[23] G. L. Matthews, Codes from the Suzuki function field, IEEE Trans. Inform. Theory, **50** (2004), no. 12, 3298–3302.
[24] G. L. Matthews, Weierstrass pairs and minimum distance of Goppa codes, Des. Codes Cryptogr. **22** (2001), 107–121.
[25] G. L. Matthews, Weierstrass semigroups and codes from a quotient of the Hermitian curve, Des. Codes Cryptogr. **37** (2005), no. 3, 473–492.
[26] G. L. Matthews and T. W. Michel, One-point codes using places of higher degree, IEEE Trans. Inform. Theory **51** (2005), no. 4, 1590-1593.
[27] S. Sakata, On fast interpolation method for Guruswami-Sudan list decoding of one-point algebraic-geometry codes. Applied algebra, algebraic algorithms and error-correcting codes (Melbourne, 2001), 172–181, Lecture Notes in Comput. Sci. **2227**, Springer, Berlin, 2001.
[28] S. Sakata and M. Fujisawa, Fast decoding of two-point AG codes, preprint.
[29] S. Sakata, J. Justesen, Y. Madelung, H. E. Jensen, and T. Høholdt, Fast decoding of AG-codes up to the designed minimum distance, IEEE Trans. on Inform. Theory, IT-41 (1995), 1672–1677.
[30] M. A. Shokrollahi and H. Wasserman, List decoding of algebraic-geometric codes, IEEE Trans. Inform. Theory **45** (1999), 432–437.
[31] J. H. Silverman, The arithmetic of elliptic curves. Graduate Texts in Mathematics, 106. Springer-Verlag, New York, 1986.
[32] H. Stichtenoth, Algebraic Function Fields and Codes, Springer-Verlag, 1993.
[33] M. Sudan, Decoding of Reed-Solomon codes beyond the error correction bound, J. Compl. **13**, 180–193, 1997.
[34] J. M. Wozencraft, List decoding, Quarterly progress report, Research Laboratory of Electronics, MIT, 48:90-95, 1958.
[35] C. P. Xing and H. Chen, Improvements on parameters of one-point AG codes from Hermitian codes, IEEE Trans. Inform. Theory **48** no. 2 (2002), 535–537.
[36] L. Xu, Improvement on parameters of Goppa geometry codes from maximal curves using the Vlădut-Xing method, IEEE Trans. Inform. Theory **51** no. 6 (2005), 2207–2210.