Honam Mathematical J. **35** (2013), No. 4, pp. 793–808
http://dx.doi.org/10.5831/HMJ.2013.35.4.793

FAST UNIQUE DECODING OF PLANE AG CODES

Kwankyu Lee

Abstract. An interpolation-based unique decoding algorithm of Algebraic Geometry codes was recently introduced. The algorithm iteratively computes the sent message through a majority voting procedure using the Gröbner bases of interpolation modules. We now combine the main idea of the Guruswami-Sudan list decoding with the algorithm, and thus obtain a hybrid unique decoding algorithm of plane AG codes, significantly improving the decoding speed.

1. Introduction

A new unique decoding algorithm of Algebraic Geometry codes was proposed in [4], and also in an alternative form in [3]. The algorithm is based on the interpolation modules defined by the received vector, like the list decoding as formulated in [5], rather than the syndromes used in the classical syndrome decoding algorithm. A distinctive feature of the algorithm is that it computes the sent message directly from the received vector by iterating a majority voting procedure, not through the usual two steps, the error location step and the error evaluation step, of the classical decoding algorithm of AG codes. The new algorithm came in two flavors. The algorithm in [4] uses the Gröbner basis over a univariate polynomial ring and has a regular data structure amenable for parallel architecture in hardware implementation. The alternative form proposed in [3] uses the Gröbner basis over the coordinate ring of the curve, which in general consists of fewer generators than in the former approach.

Received November 7, 2013. Accepted November 25, 2013.

2010 Mathematics Subject Classification. Primary 94B35, 94B27, 13P10.

Key words and phrases. algebraic geometry codes, interpolation decoding, Gröbner bases.

The author was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2013R1A1A2009714).

It turns out that the interpolation-based unique decoding algorithm may benefit more from the list decoding, as the central concept of the Guruswami-Sudan list decoding, namely the Q-polynomial can be used along with the majority voting. The idea is that the computationally expensive Gröbner basis computation is iterated only until a Q-polynomial is found, and then the root of the Q-polynomial reveals the rest of the sent message. This simple idea boosts the decoding speed significantly. In this paper, we will present a hybrid unique decoding algorithm that retains the same decoding capacity with the original algorithm but runs faster in a speed inversely proportional to the number of errors in the received vector. Along the way, we add decoding failure detection devices as well, facilitating real applications of AG codes.

In Section 2, we review the original decoding algorithm as presented in [3]. Although the results of the present paper can be equally applied to the algorithm in [4], the mathematical setting of the former paper affords a more convenient exposition of the results. Then in Section 3, we explain the idea of using a Q-polynomial, formulate a hybrid unique decoding algorithm, and analyze the behavior of the new algorithm which can be used to estimate quantitatively the speed improvement. In Section 4, we use the Hermitian codes to demonstrate the new decoding algorithm and present some experimental results on the improved performance. Some proofs in the present paper rely on the standard results in commutative algebra, the main references for which are [1, 2]. For the basic terminology and results on numerical semigroups, see [6] for instance.

2. Unique Decoding by Interpolation

A Miura-Kamiya curve X is an irreducible plane curve defined by an equation

$$y^a + \sum_{ai+bj<ab} c_{i,j}x^iy^j + dx^b = 0$$

over a field $\mathbb{F}$ with $\gcd(a,b) = 1$, $a < b$, and $0 \neq d \in \mathbb{F}$, and has a unique point P_∞ at infinity with a unique valuation v_{P_∞} associated with it. Let $\delta(f) = -v_{P_\infty}(f)$ for f in the coordinate ring R of X. Then $\delta(x) = a$ and $\delta(y) = b$. Note that by the equation of the curve, a function in the coordinate ring $R = \mathbb{F}[x,y]$ can be written as a unique $\mathbb{F}$-linear combination of monomials x^iy^j with $i \geq 0$ and $0 \leq j < a$, called monomials of R. The numerical semigroup of R at P_∞,

$$\begin{aligned}S &= \{s_0, s_1, s_2, \dots\} = \{\delta(f) \mid f \in R\} = \{\delta(x^i y^j) \mid i \geq 0, 0 \leq j < a\} \\ &= \{ai + bj \mid i \geq 0, 0 \leq j < a\}\end{aligned}$$

is a subset of the Weierstrass semigroup at P_∞. We easily see that the monomials of R are in one-to-one correspondence with elements in S, called nongaps. For a nongap s, let φ_s be the unique monomial with $\delta(\varphi_s) = s$.

Let $P_1, P_2, \dots, P_n$ be nonsingular rational points of X. The evaluation ev from R to the Hamming space $\mathbb{F}^n$ defined by

$$\varphi \mapsto (\varphi(P_1), \varphi(P_2), \dots, \varphi(P_n))$$

is a linear map over $\mathbb{F}$. Let u be a fixed positive integer and define

$$L_u = \{f \in R \mid \delta(f) \leq u\} = \langle \varphi_s \mid s \in S, s \leq u\rangle,$$

where brackets denote the linear span over $\mathbb{F}$. Then the AG code C_u is defined as the image of L_u under ev. We assume $u < n$ such that the evaluation is one-to-one on L_u. Therefore $\dim C_u = \dim_{\mathbb{F}} L_u = |\{s \in S \mid s \leq u\}|$. Let $k = \dim C_u$. Recall that the genus g of S is the number of nonnegative integers not in S called gaps. Then it is clear that $k = u - g + 1$ for $k \geq 2g$ since $2g + \mathbb{N} \subset S$, where $\mathbb{N}$ is the set of nonnegative integers.

We assume a codeword c in C_u was sent through a noisy communication channel. Let $v \in \mathbb{F}^n$ be the received vector such that $v = c + e$ with the error vector e. Then $c = \mathrm{ev}(\mu)$ for a unique

$$\mu = \sum_{s \in S, s \leq u} \omega_s \varphi_s \in L_u, \qquad \omega_s \in \mathbb{F}$$

Under evaluation encoding, the vector $(\omega_s \mid s \in S, s \leq u) \in \mathbb{F}^k$ is the message encoded into the codeword c. The decoding problem is essentially to find ω_s for all nongap $s \leq u$ from the given v.

For $s \geq u$, let $v^{(s)} = v$, $c^{(s)} = c$, and $\mu^{(s)} = \mu$. For nongap $s \leq u$, let

$$\begin{aligned}\mu^{(s-1)} &= \mu^{(s)} - \omega_s \varphi_s, \\ c^{(s-1)} &= c^{(s)} - \mathrm{ev}(\omega_s \varphi_s), \\ v^{(s-1)} &= v^{(s)} - \mathrm{ev}(\omega_s \varphi_s),\end{aligned}$$

and for gap $s \leq u$, let $v^{(s-1)} = v^{(s)}$, $c^{(s-1)} = c^{(s)}$, and $\mu^{(s-1)} = \mu^{(s)}$. Note that

$$\mu^{(s)} \in L_s, \quad c^{(s)} = \mathrm{ev}(\mu^{(s)}) \in C_s, \quad v^{(s)} = c^{(s)} + e$$

for all s. Hence the decoding problem can be solved iteratively if we can figure out ω_s for each nongap s decreasing from u to 0.

A polynomial in $R[z]$ defines a function on the product surface of X and the affine line $\mathbb{A}^1_{\mathbb{F}}$, and can be evaluated at a point (P, α) with $P \in X, \alpha \in \mathbb{F}$. Hence we can define the *interpolation module*

$$I_v = \{f \in Rz \oplus R \mid f(P_i, v_i) = 0, 1 \leq i \leq n\}$$

for v and similarly $I_{v^{(s)}}$ for $v^{(s)}$. These interpolation modules are indeed modules over R, and finite-dimensional vector space over $\mathbb{F}$. Note that

$$I_v = R(z - h_v) + J$$

where

$$J = \bigcap_{1 \leq i \leq n} \mathfrak{m}_i, \qquad \mathrm{ev}(h_v) = v,$$

and $\mathfrak{m}_i = \langle x - \alpha_i, y - \beta_i \rangle$ is the maximal ideal of R associated with $P_i = (\alpha_i, \beta_i)$. Recall that by Lagrange interpolation, h_v can be computed fast from v.

Our decoding algorithm works with the Gröbner basis of $I_{v^{(s)}}$ with respect to a monomial order $>_s$ on $Rz \oplus R$, which is defined as follows. Let s be an integer. The monomial $x^i y^j z^k$ of $R[z]$ is given the weight $\delta(x^i y^j) + sk$. In particular, the weighted degrees of the monomials $x^i y^j z$ and $x^i y^j$ of $Rz \oplus R$ are $ai + bj + s$ and $ai + bj$, respectively. The monomial order $>_s$ orders the monomials of $Rz \oplus R$ by their weighted degrees, and breaks the tie with higher z-degree. For f in $Rz \oplus R$, the notations $\mathrm{lt}_s(f)$, $\mathrm{lm}_s(f)$, and $\mathrm{lc}_s(f)$ denote the leading term, the leading monomial, and the leading coefficient of f, with respect to $>_s$, and $\deg_s(f)$ the weighted degree of $\mathrm{lt}_s(f)$. Let M be a submodule of $Rz \oplus R$. A subset B of M is called a *Gröbner basis* with respect to $>_s$ if the leading term of every element of M is divided by the leading term of some element of B. We will write

$$B = \{G_i, F_j\}$$

with i and j in some index sets, where we assume $\mathrm{lt}_s(G_i) \in R$ and $\mathrm{lt}_s(F_j) \in Rz$. The *sigma set* $\Sigma_s(M)$ is the set of all leading monomials of polynomials in M with respect to $>_s$. The *delta set* $\Delta_s(M)$ is the complement of $\Sigma_s(M)$ in the set of all monomials of $Rz \oplus R$. For the case that M is an ideal of R, we may omit the superfluous s from the above notations, and denote $>_s$ simply by $>$. Note that if $\{\eta_i\}$ is a Gröbner basis of J with respect to $>$, then the set $\{\eta_i\} \cup \{z - h_v\}$ is a Gröbner basis of I_v with respect to $>_{\delta(h_v)}$.

Suppose $B^{(s)}$ is a Gröbner basis of $I_{v^{(s)}}$ with respect to $>_s$. The majority voting procedure described in [3] makes a guess $w^{(s)}$ of ω_s from $B^{(s)}$, and we have

Theorem 2.1. *The guess by majority voting is correct, that is,* $w^{(s)} = \omega_s$ *if*

$$\nu(s) = |\Delta(J) \cup \Delta(R\varphi_s)| - s > 2\mathrm{wt}(e).$$

This result leads to an obvious decoding algorithm, in which $B^{(s)} = \{G_i, F_j\}$ denotes a Gröbner basis of $I_{v^{(s)}}$ with respect to $>_s$.

```
Unique Decoding Algorithm. Let $v$ be the received vector.
Initialize: Compute $h_v$. Let $B^{(N)} = \{\eta_i\} \cup \{z - h_v\}$ where $N = \delta(h_v)$.
Main: Repeat the following for $s$ from $N$ to $0$.
    M1: If $s$ is a nongap $\leq u$, then make a guess $w^{(s)}$ for $\omega_s$, and let $\tilde{B} = \{G_i(z+w^{(s)}\varphi_s), F_j(z+w^{(s)}\varphi_s)\}$. Otherwise, let $\tilde{B} = B^{(s)}$.
    M2: Compute $B^{(s-1)}$ from $\tilde{B}$.
Finalize: Output $(w^{(s)} \mid \text{nongap } s \leq u)$.
```

The details of how to *make a guess* $w^{(s)}$ *for* ω_s by majority voting in the step M1 and *compute* $B^{(s-1)}$ *from* $\tilde{B}$ in the step M2 are found in [3]. By Theorem 2.1, the decoding algorithm outputs $w^{(s)} = \omega_s$ for all $s \in S, s \leq u$ if

$$d_u = \min_{s \in S, s \leq u} \nu(s) > 2\mathrm{wt}(e).$$

Note that $d_u \geq n - u$ as $\nu(s) \geq n - s$. Let $\tau = \tau_u = \lfloor (d_u - 1)/2 \rfloor$. This is the number of errors that can be corrected by the Unique Decoding Algorithm for C_u.

3. Fast Unique Decoding

We will prove a special form of the fundamental theorem of the Guruswami-Sudan list decoding, which is found in [5], adapted for our definition of the interpolation module I_v as a submodule of $Rz \oplus R$.

Lemma 3.1. *For* $v \in \mathbb{F}^n$,

$$\dim_{\mathbb{F}} Rz \oplus R / I_v = n.$$

For $\mu \in R$ *and* $\mathrm{wt}(v - \mathrm{ev}(\mu)) = t$,

$$\dim_{\mathbb{F}} Rz \oplus R / \langle I_v, z - \mu \rangle = n - t.$$

Proof. It is easy to see

$$I_v = \langle z - h_v \rangle + \bigcap_{i=1}^{n} \mathfrak{m}_i.$$

Recall that $\mathfrak{m}_i$ are associated with nonsingular rational points over $\mathbb{F}$. Hence

$$\dim_{\mathbb{F}} Rz \oplus R/I_v = \dim_{\mathbb{F}} R/\bigcap_{i=1}^{n} \mathfrak{m}_i = \sum_{i=1}^{n} \dim_{\mathbb{F}} R/\mathfrak{m}_i = n.$$

Let us suppose $\mathrm{ev}(\mu) = (c_1, c_2, \ldots, c_n)$. As $\dim_{\mathbb{F}} Rz \oplus R/\langle I_v, z - \mu\rangle \leq n$, the R-module $Rz \oplus R/\langle I_v, z - \mu\rangle$ is of finite length, and its support is contained in $\{\mathfrak{m}_1, \mathfrak{m}_2, \ldots, \mathfrak{m}_n\}$, and hence via the Chinese remainder theorem for modules of finite length, we have

$$\dim_{\mathbb{F}} Rz \oplus R/\langle I_v, z - \mu\rangle = \sum_{i=1}^{n} \dim_{\mathbb{F}} (Rz \oplus R)_{\mathfrak{m}_i}/\langle I_v, z - \mu\rangle,$$

where we take on the right the localizations at $\mathfrak{m}_i$ of the module on the left. Moreover we have for each $1 \leq i \leq n$,

$$\dim_{\mathbb{F}} (Rz \oplus R)_{\mathfrak{m}_i}/\langle I_v, z - \mu\rangle = \dim_{\mathbb{F}} \widehat{(Rz \oplus R)}_{\mathfrak{m}_i}/\langle I_v, z - \mu\rangle,$$

where we take on the right the completion of the module on the left. Finally observe that $\widehat{(Rz \oplus R)}_{\mathfrak{m}_i}/\langle I_v, z - \mu\rangle$ is isomorphic to the $\mathbb{F}[[x]]$-module

$$\mathbb{F}[[X]]Z \oplus \mathbb{F}[[X]]/\langle \{f(X)Z + g(X) \mid f(0)v_i + g(0) = 0\}, Z - c_i\rangle,$$

whose $\mathbb{F}$-dimension is 1 or 0 depending on whether $v_i = c_i$ or not. □

Theorem 3.2. *Suppose* $\mu \in L_s$ *and* $\mathrm{wt}(v - \mathrm{ev}(\mu)) = t$. *If* $\deg_s(f) < n - t$ *with a nonzero* $f \in I_v \subset Rz \oplus R$, *then* $f(\mu) = 0$.

Proof. If $f(\mu)$ is not zero in R, then

$$\begin{aligned}\deg_s(f) \geq \delta(f(\mu)) &= \dim_{\mathbb{F}}(R/f(\mu)) \\ &= \dim_{\mathbb{F}} Rz \oplus R/\langle f, z - \mu\rangle \\ &\geq \dim_{\mathbb{F}} Rz \oplus R/\langle I_v, z - \mu\rangle = n - t.\end{aligned}$$

This proves the contraposition of our assertion. □

We will call any polynomial f in $I_{v^{(s)}}$ satisfying the condition $\deg_s(f) < n - \mathrm{wt}(e)$ a *Q-polynomial* for $v^{(s)}$. The above theorem says that a root of the Q-polynomial f for $v^{(s)}$ is $\mu^{(s)}$. Therefore once we find a Q-polynomial f in $B^{(s)}$, then we may stop the main iteration and compute the root $\mu^{(s)}$ of f in R. In fact, as f is in the form $\varphi z + \psi$ with $\varphi, \psi \in R$, the root is simply $-\psi/\varphi$. Thus $\mu^{(s)}$ can be computed by a single long division of $-\psi$ by φ, which can be done very fast using, for instance, the root-finding algorithm in [7]. A caveat is that we do not know $\mathrm{wt}(e)$ in advance. Hence the actual condition that we use is $\deg_s(f) + \tau < n$ where τ is half of the order bound d_u. If we assume $\mathrm{wt}(e) \leq \tau$, then a polynomial f in $B^{(s)}$ satisfying the condition $\deg_s(f) + \tau < n$ is a Q-polynomial for $v^{(s)}$. See the step M2 and the step Division in the Fast Unique Decoding Algorithm given below.

On the other hand, if $\mathrm{wt}(e) > \tau$, then it is advantageous to detect this fact at the earliest possible point during decoding so that we can avoid useless iterations of the Main step and the Division step. The following lemma is useful for this purpose.

Lemma 3.3. $|\Delta_s(I_{v^{(s)}}) \cap Rz| \leq \mathrm{wt}(e)$

Proof. Let $J_e = \bigcap_{e_i \neq 0} \mathfrak{m}_i$. Then clearly $J_e(z - \mu^{(s)}) \subset I_{v^{(s)}}$, which implies

$$\Delta_s(I_{v^{(s)}}) \cap Rz \subset \Delta(J_e)z.$$

Since $|\Delta(J_e)| = \dim_{\mathbb{F}}(R/J_e) = |\{e_i \neq 0\}| = \mathrm{wt}(e)$, the assertion follows. □

If we have $|\Delta_s(I_{v^{(s)}}) \cap Rz| > \tau$ during decoding, then by the lemma, we must have $\mathrm{wt}(e) > \tau$, that is, there occurred more errors to the sent codeword than the decoding algorithm can cope with. Hence *decoding failure* may be declared once the condition $|\Delta_s(I_{v^{(s)}}) \cap Rz| > \tau$ is satisfied. Note that by the definition of a Gröbner basis, a monomial φ in Rz belongs to $\Delta_s(I_{v^{(s)}})$ if and only if φ is divisible by none of the leading terms of F_j in $B^{(s)}$. See the step M1 below.

We now present the Fast Unique Decoding Algorithm combining the original algorithm with the two enhancements described above. Note that the step Initialize is also slightly changed to detect no error case, that is, $h_v \in L_u$ so that $\mu = h_v$.

Fast Unique Decoding Algorithm. *Let v be the received vector.*

Initialize: *Compute h_v. Let $B^{(N)} = \{\eta_i\} \cup \{z - h_v\}$ where $N = \delta(h_v)$. If $N \leq u$, then set $w^{(s)}$ to the coefficient of the monomial φ_s in h_v for every nongap $s \leq u$, and go to the step Finalize.*

Main: *Repeat the following for s from N to 0. Then go to the step Finalize.*

M1: *Let Δ be the set of monomials in Rz divisible by none of the leading terms of F_j in $B^{(s)}$. If $|\Delta| > \tau$, then declare* Decoding Failure.

M2: *If there is an* $f \in B^{(s)}$ *such that* $\deg_s(f) + \tau < n$, *then go to the step Division.*

M3: *If* s *is a nongap* $\leq u$, *then make a guess* $w^{(s)}$ *for* ω_s, *and let* $\tilde{B} = \{G_i(z+w^{(s)}\varphi_s), F_j(z+w^{(s)}\varphi_s)\}$. *Otherwise, let* $\tilde{B} = B^{(s)}$.

M4: *Compute* $B^{(s-1)}$ *from* $\tilde{B}$.

Division: *Let* $f = \varphi z + \psi$. *Compute* $\mu^{(s)} = -\psi/\varphi$, *and proceed to the step Finalize. If* ψ *is not divisible by* φ, *then declare* Decoding Failure.

Finalize: *Output* $(w^{(s)} \mid$ *nongap* $s \leq u)$.

To estimate the improvement of the new algorithm over the original one, we now consider the question when the condition in the step M2 is satisfied so that the algorithm branch to the step Division.

Lemma 3.4. *Let* $B^{(s)}$ *be a Gröbner basis of* $I_{v^{(s)}}$ *with respect to* $>_s$. *There exists an* $f \in B^{(s)}$ *such that* $\deg_s(f) \leq s_t + s$, *where* $t = \mathrm{wt}(e)$.

Proof. Recall that $J_e(z - \mu^{(s)}) \subset I_{v^{(s)}}$, and hence $\Sigma(J_e)z \subset \Sigma_s(I_{v^{(s)}})$. Recall that $s_0, s_1, \ldots, s_t$ are the first $t + 1$ nongaps. Since $|\Delta(J_e)| = \mathrm{wt}(e) = t$, not all of $s_0, s_1, \ldots, s_t$ corresponds to monomials in $\Delta(J_e)$. Therefore there exists a monomial φ in $\Sigma(J_e)$ for which $\delta(\varphi) \leq s_t$. Then as $\varphi z \in \Sigma_s(I_{v^{(s)}})$, there exists an $f \in B^{(s)}$ such that $\mathrm{lt}_s(f)$ divides φz by the definition of Gröbner basis, and hence $\deg_s(f) \leq s_t + s$. □

The lemma immediately implies

Theorem 3.5. *If* $s_t + s + t < n$, *then there exists an* $f \in B^{(s)}$ *such that* f *is a* Q*-polynomial for* $v^{(s)}$.

Let us define two values

$$\tau_Q(s) = \max\{t \mid s_t + s + t < n\},$$
$$\tau_M(s) = \max\{t \mid 2t + s < |\Delta(J) \cup \Delta(R\varphi_s)|\}$$

as functions of nongap $s < n$. In view of Theorem 2.1, the value $\tau_M(s)$ is the largest number of errors for which the majority voting succeeds for s. Also by Theorem 3.5, the value $\tau_Q(s)$ is the largest number of errors for which a Q-polynomial exists in $B^{(s)}$. We will now show that $\tau_M(s)$ is larger than $\tau_Q(s)$, and their difference is bounded by a small constant.

First we prove several lemmas regarding ideals of a numerical semigroup S with genus g. An ideal A of a numerical semigroup S is a nonempty subset A of S satisfying $S + A \subset A$. Note that the set $\Delta(A) = S \backslash A$ is finite as $2g + \mathbb{N} \subset S$.

Lemma 3.6. *For s in S, it holds that $|\Delta(s+S)| = s$.*

Proof. We prove by induction on the genus of S. If S has no gap, then $S = \mathbb{N}$, and our assertion is clear. If the genus of S is positive, then let f be the Frobenius element of S, that is, the largest integer in $\mathbb{N}\backslash S$, and let $S' = S \cup \{f\}$. Then S' is clearly a numerical semigroup with genus one less than that of S, and $|S'\backslash(s+S')| = s$ by induction hypothesis. As $|S\backslash(s+S)| = |S'\backslash(s+S')|$, it follows that $|S\backslash(s+S)| = s$. □

Lemma 3.7. *Let A and B be ideals of a numerical semigroup S. Let $|\Delta(A)| = a$, $|\Delta(B)| = b$. Suppose $a \geq b$. Then*

$$a \leq |\Delta(A) \cup \Delta(B)| \leq a + g.$$

Moreover, $|\Delta(A) \cup \Delta(s+S)| = a$ for $s \leq a - 2g$.

Proof. Let m_A be the least integer in the ideal A. Then $m_A + S \subset A$. Therefore at most g integers greater than m_A are in $\Delta(A)$. Hence at least $a-g$ first elements of S are in $\Delta(A)$. Similarly at least $b-g$ first elements of S are in $\Delta(B)$. As $b - g \leq a - g$, we see that $|\Delta(A) \cap \Delta(B)| \geq b - g$. Therefore

$$|\Delta(A) \cup \Delta(B)| \leq a + b - (b-g) = a + g.$$

For the second assertion, assume $s \leq a - 2g$. Recall that $s + 2g + \mathbb{N} \subset s + S$. Hence for every $x \in \Delta(s+S)$,

$$x < s + 2g \leq a = |\Delta(A)| \leq |\Delta(m_A + S)| = m_A,$$

which implies $x \in \Delta(A)$. Hence $\Delta(s+S) \subset \Delta(A)$ □

By considering the ideals of the numerical semigroup S corresponding to $\Sigma(J)$ and $\Sigma(R\varphi_s)$, we have by the lemma,

$$n \leq |\Delta(J) \cup \Delta(R\varphi_s)| \leq \begin{cases} n+g & n - 2g < s < n, \\ n & s \leq n - 2g. \end{cases}$$

Theorem 3.8. *For nongap $s < n$,*

$$\lfloor (n-g-s-1)/2 \rfloor \leq \tau_Q(s) \leq \lfloor (n-s-1)/2 \rfloor \leq \tau_M(s) \leq \lfloor (n+g-s-1)/2 \rfloor.$$

Moreover for $s \leq n - 2g$,

$$\tau_M(s) = \lfloor (n-s-1)/2 \rfloor,$$

and for $s < n - 3g$,

$$\tau_Q(s) = \lfloor (n-g-s-1)/2 \rfloor.$$

In particular, $\tau_Q(s) \leq \tau_M(s)$ for nongap $s < n$, and for $s < n - 3g$,

$$\tau_M(s) - \tau_Q(s) = \lceil g/2 \rceil.$$

Proof. Let $s < n$. We have $t \le s_t \le t + g$, and hence

$$\max\{t \mid 2t + s + g < n\} \le \tau_Q(s) \le \max\{t \mid 2t + s < n\},$$

that is

$$\lfloor (n - g - s - 1)/2 \rfloor \le \tau_Q(s) \le \lfloor (n - s - 1)/2 \rfloor.$$

Recall that $s_t = t + g$ for $t \ge g$. Hence if $s < n - 3g$, then $s_g + s + g = 3g + s < n$. Therefore

$$\tau_Q(s) = \max\{t \mid t + g + s + t < n\} = \lfloor (n - g - s - 1)/2 \rfloor.$$

On the other hand, by the bounds on $|\Delta(J) \cup \Delta(R\varphi_s)|$ given above,

$$\max\{t \mid 2t + s < n\} \le \tau_M(s) \le \begin{cases} \max\{t \mid 2t + s < n + g\} & n - 2g < s < n, \\ \max\{t \mid 2t + s < n\} & s \le n - 2g, \end{cases}$$

that is

$$\lfloor (n - s - 1)/2 \rfloor \le \tau_M(s) \le \begin{cases} \lfloor (n + g - s - 1)/2 \rfloor & n - 2g < s < n, \\ \lfloor (n - s - 1)/2 \rfloor & s \le n - 2g, \end{cases}$$

□

Recall that the actual condition used in the step M2 to find a Q-polynomial in $B^{(s)}$ is $\deg_s(f) + \tau < n$. Lemma 3.7 also implies

Theorem 3.9. *Suppose* $t = \mathrm{wt}(e) \le \tau$. *If* $s_t + s + \tau < n$, *then there exists an* $f \in B^{(s)}$ *satisfying* $\deg_s(f) + \tau < n$, *and* f *is a* Q*-polynomial for* $v^{(s)}$.

By the theorem, the jump at the step M2 to the step Division occurs for some $s \ge s_Q(t) = n - \tau - s_t - 1$ depending on $t = \mathrm{wt}(e)$, and at the latest for some $s \ge s_Q(\tau) = n - \tau - s_\tau - 1$ if we assume $t \le \tau$. We have the relations

$$\tau = \min_{s \le u} \tau_M(s), \quad s_Q(\tau) = \max\{s \mid \tau_Q(s) \ge \tau\}, \tag{1}$$

which can be verified by unwinding the definitions.

4. Hermitian codes

We demonstrate the Fast Unique Decoding Algorithm with the Hermitian codes defined on the curves with equation

$$y^q + y - x^{q+1} = 0$$

over $\mathbb{F}_{q^2}$. There are $n = q^3$ nonsingular rational points on the affine part of the Hermitian curve, and $J = \langle x^{q^2} - x \rangle$. In [3] and [4], it was shown that

$$\nu(s) = s_2 \max\{s_1 - s_2 + q + 1 - q^2, 0\} + q^3 - s$$

for $s = qs_1 + s_2 < n$, and hence for the Hermitian code C_u with nongap $u < q^3$,

$$d_u = \begin{cases} q^3 - aq & b \leq a + q - q^2, \\ q^3 - u & b > a + q - q^2, \end{cases}$$

if $u = aq + b$, $0 \leq b < q$.

Let $q = 3$, and let us consider the Hermitian codes C_u of length 27 over $\mathbb{F}_9$. In Figure 1, the graphs of $\tau_M(s)$ and $\tau_Q(s)$ are displayed as asterisks and black dots respectively. Note that the upper and lower bounds in Theorem 3.8 are depicted by circles and the middle bound by squares. For example, for $s = 16$, the majority voting succeeds up to $\tau_M(16) = 5$ errors and a Q-polynomial is found in $B^{(16)}$ for up to $\tau_Q(16) = 3$ errors. On the other hand, Figure 2 shows τ_u along with $s_Q(\tau_u)$ for u on the horizontal axis. For example, $\tau_{16} = 5$ marked by an asterisk and $s_Q(\tau_{16}) = 13$ (the black dot that lies on the left horizontally to the asterisk), indicating that for the code C_{16}, the decoding algorithm can correct up to 5 errors and branching to the Division step occurs at the latest before $s = 13$. Notice that these two figures are closed related, and indeed one can easily derive Figure 2 from Figure 1 using the relations (1).

Now let us choose $u = 16$, and work with the Hermitian code C_{16} of length 27 and dimension 14 on the curve $y^3 + y - x^4 = 0$ over $\mathbb{F}_9$, where $\mathbb{F}_9 = \mathbb{F}_3(\alpha)$ with $\alpha^2 - \alpha - 1 = 0$. As noted above, the decoding algorithm can correct up to $\tau = \tau_{16} = 5$ errors. As $\delta(x) = 3$ and $\delta(y) = 4$, the numerical semigroup S is

$$\{s_0, s_1, s_2, s_3, s_4, s_5, s_6, \dots\} = \{0, 3, 4, 6, 7, 8, 9, \dots\}.$$

Since S has three gaps 1, 2, and 5, the genus of S is $g = 3$.

The monomials of R correspond to nongaps in S and are displayed in the right array below. The monomials of Rz are similarly displayed on the left array below with z being omitted.

y^2	xy^2	x^2y^2	x^3y^2	x^4y^2	x^5y^2	x^6y^2	x^7y^2	x^8y^2	x^9y^2	$\cdots$
y	xy	x^2y	x^3y	x^4y	x^5y	x^6y	x^7y	x^8y	x^9y	$\cdots$
1	x	x^2	x^3	x^4	x^5	x^6	x^7	x^8	x^9	$\cdots$

y^2	xy^2	x^2y^2	x^3y^2	x^4y^2	x^5y^2	x^6y^2	x^7y^2	x^8y^2	x^9y^2	$\cdots$
y	xy	x^2y	x^3y	x^4y	x^5y	x^6y	x^7y	x^8y	x^9y	$\cdots$
1	x	x^2	x^3	x^4	x^5	x^6	x^7	x^8	x^9	$\cdots$

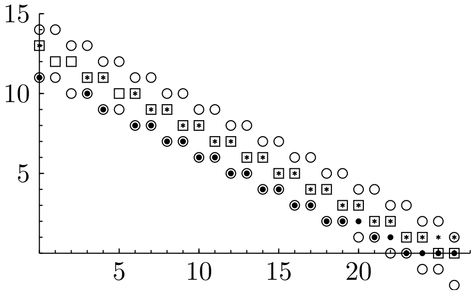


FIGURE 1. $\tau_M(s)$ and $\tau_Q(s)$

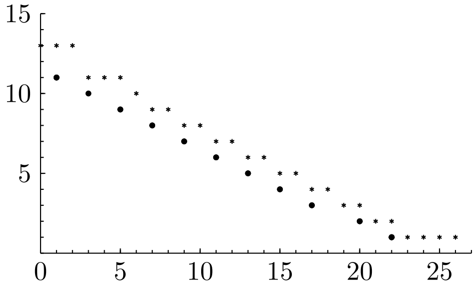


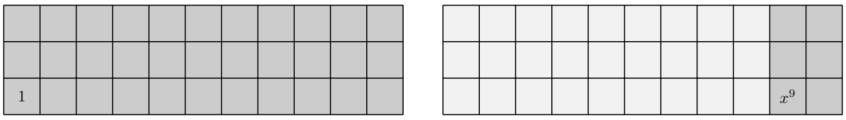
FIGURE 2. τ_u and $s_Q(\tau_u)$

Suppose that we received the vector

$$v = (\alpha^2, \alpha^2, \alpha, \alpha, \alpha^2, \alpha^2, \alpha^2, \alpha, \alpha, \alpha, \alpha, \alpha, \alpha, \alpha, \alpha, \alpha, \alpha, \alpha, \alpha, \alpha, \alpha, \alpha, \alpha, \alpha, \alpha, \alpha, \alpha)$$

from a noisy channel. We now follow the steps of the decoding algorithm. The algorithm starts with the Gröbner basis of $I_{v^{(32)}}$,

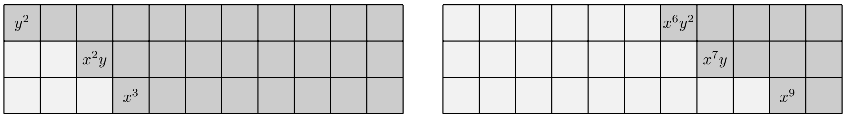
$$B^{(32)} = \left\{ \begin{array}{lclcl} G_1 & = & & & x^9 + \cdots \\ F_1 & = & z & + & 2x^8y^2 + \cdots \end{array} \right\}$$



In the above figure as well as in the following, the monomials in $\Sigma_s(I_{v^{(s)}}) \cap Rz$ lie in the dark gray area of the left array, with z being omitted, while the monomials in $\Sigma_s(I_{v^{(s)}}) \cap R$ lie in the dark gray area of the right array.

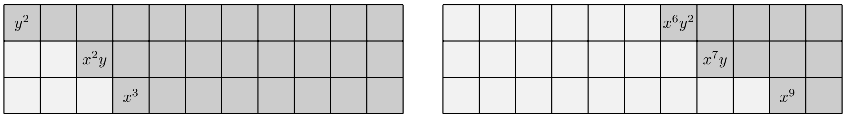
For s decreasing from 32 to 17, the Gröbner basis $B^{(32)}$ of $I_{v^{(32)}}$ is iteratively updated toward the Gröbner basis of $I_{v^{(16)}}$,

$$B^{(16)} = \left\{ \begin{array}{lcrcl} G_1 & = & & & x^9 + \cdots \\ G_2 & = & (2x^2 + \cdots)\, z & + & \alpha x^6 y^2 + \cdots \\ G_3 & = & (\alpha xy + \cdots)\, z & + & \alpha^5 x^7 y + \cdots \\ F_1 & = & (\alpha^7 x^3 + \cdots)\, z & + & \alpha x^7 y + \cdots \\ F_2 & = & (\alpha^2 x^2 y + \cdots)\, z & + & 2x^6 y^2 + \cdots \\ F_3 & = & (\alpha^3 y^2 + \cdots)\, z & + & y^2 + \cdots \end{array} \right\}$$



From this point on, when $s = 16$, the original decoding algorithm would compute $w^{(16)}, w^{(15)}, \dots, w^{(0)}$ by majority voting iteratively until $s = -1$. In the same way, the new decoding algorithm computes $w^{(16)} = 0, w^{(15)} = 0, w^{(14)} = 0$ by majority voting for $s = 16$, 15, and 14. However, at $s = 13$ when it starts the next iteration with the Gröbner basis of $I_{v^{(13)}}$,

$$B^{(13)} = \left\{ \begin{array}{lcrcl} G_1 & = & & & x^9 + \cdots \\ G_2 & = & (2x^2 + \cdots)\, z & + & \alpha x^6 y^2 + \cdots \\ G_3 & = & (\alpha xy + \cdots)\, z & + & \alpha^5 x^7 y + \cdots \\ F_1 & = & (\alpha^6 x^3 + \cdots)\, z & + & \alpha^3 x^3 + \cdots \\ F_2 & = & (x^2 y + \cdots)\, z & + & \alpha^5 x^2 y + \cdots \\ F_3 & = & (\alpha^3 y^2 + \cdots)\, z & + & y^2 + \cdots \end{array} \right\}$$



in the step M2, it is found that the generator F_3 satisfies the condition $\deg_{13}(F_3) + \tau < n$ as $\deg_{13}(F_3) + \tau = \delta(y^2) + s + \tau = 8 + 13 + 5 = 26 < n = 27$. So the step Division is triggered with a Q-polynomial

$$f = F_3 = (\alpha^3 y^2 + 2xy + x^2 + \alpha y + \alpha^2 x)z + y^2 + \alpha xy + \alpha^5 x^2 + \alpha^6 y + \alpha^7 x$$

By a long division, the algorithm quickly computes

$$\mu^{(13)} = 0x^3y + 0x^4 + 0xy^2 + 0x^2y + 0x^3 + 0y^2 + 0xy + 0x^2 + 0y + 0x + \alpha,$$

which set $w^{(s)} = 0$ for $s = 13, 12, \dots, 4, 3$, and $w^{(0)} = \alpha$. The algorithm then finishes with the output $w^{(s)}$ for nongap $s \leq 16$. As $\mu = \alpha$, the decoding is correct.

In the decoding example above, the number of errors was $t = 5$, and the jump to the Division step occurred at $s = 13$. Using Theorem 3.9, we may tabulate when the jump occurs dependant on the number of errors $t = \mathrm{wt}(e)$.

t	$s_Q(t)$	by voting	by division
0	21		$\omega_{16}, \omega_{15}, \dots, \omega_0$
1	18		$\omega_{16}, \omega_{15}, \dots, \omega_0$
2	17		$\omega_{16}, \omega_{15}, \dots, \omega_0$
3	15	ω_{16}	$\omega_{15}, \omega_{14}, \dots, \omega_0$
4	14	ω_{16}, ω_{15}	$\omega_{14}, \omega_{13}, \dots, \omega_0$
$\tau = 5$	$s_Q(\tau) = 13$	$\omega_{16}, \omega_{15}, \omega_{14}$	$\omega_{13}, \omega_{12}, \dots, \omega_0$

We experimented with the Fast Unique Decoding Algorithm implemented in software. We report the results on the algorithm decoding the $[64, 53, 8]$ Hermitian code C_{58} over $\mathbb{F}_{16}$, to demonstrate the speed improvement of the new algorithm over the original algorithm. Figure 3 and Figure 4 again provide the basic data that defines the speed improvement, for the Hermitian codes of length 64 over $\mathbb{F}_{16}$.

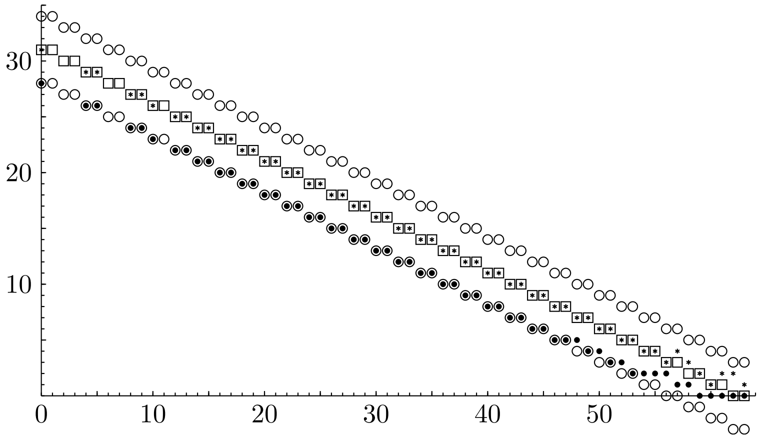


FIGURE 3. $\tau_M(s)$ and $\tau_Q(s)$

The table below reports the average timings (in milliseconds) in correcting random errors of weight t. The absolute time is meaningless as no attempt was made to optimize the software implementations, but the table clearly shows the speed improvements of the new algorithm over the original. Observe that the added devices to detect the no error case and out-of-the-capacity errors are in action.

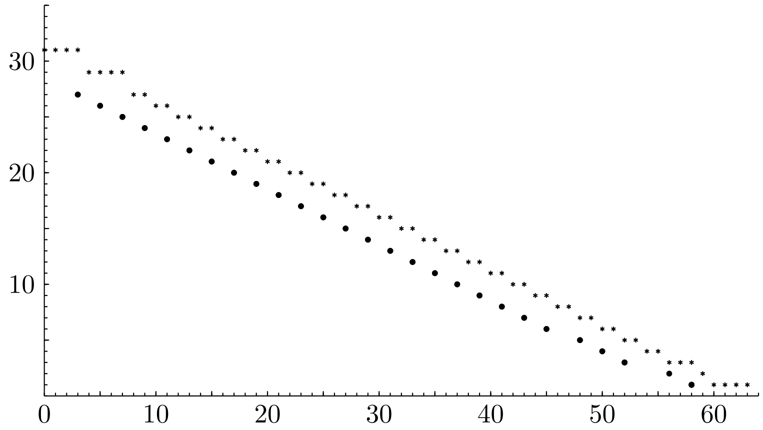


FIGURE 4. τ_u and $s_Q(\tau_u)$

t	*n*ew	*o*riginal	ratio of n to o
0	40	435	9%
1	121	456	27%
2	128	448	29%
3	149	462	32%
4	45	471	10%
5	45	481	9%

Recall that the errors out of the capacity of the code are detected either at the step M1 or at the step Division. As an indication of the effectiveness of the detectors, we present an experimental result of $100,000$ decoding instances with random errors of weight 4, 5, 6, and 7.

t	I	II	III	IV
4	99849		151	
5	99972	24	1	2(7–8)
6	99976	21		3(3–5)
7	99977	17		6(3–8)

I: failure detected at M1
II: failure detected at Division
III: correct decoding
IV: false decoding (range of distances)

Observe that most of these out-of-the-capacity errors are detected successfully, and even in the worst case when the decoder outputs a wrong codeword (IV), the output codeword is usually within a small

Hamming distance (the numbers in the parentheses) from the received vector.

5. Remarks

We improved the original interpolation-based unique decoding algorithm by incorporating the central idea of the Guruswami-Sudan list decoding and moreover the no error and decoding failure detection devices. We analyzed the behavior of the new algorithm and presented experimental results that show the expected improvement in the decoding speed. We conclude that the improved unique decoding algorithm is faster and more suitable for real applications of AG codes than the original.

Finally the author wants to express sincere thanks to the developers and contributors of the free mathematics software Sage [8], with which he spent many hours implementing the decoding algorithms.

References

[1] M. F. Atiyah and I. G. MacDonald, *Introduction to Commutative Algebra*, Perseus Books, 1969.

[2] D. Eisenbud, *Commutative Algebra with a View toward Algebraic Geometry*, Number 150 in GTM, Springer-Verlag, 1995.

[3] K. Lee, *Unique decoding of plane AG codes revisited*, arXiv, Apr. 2012. arXiv:1204.0052v2.

[4] K. Lee, M. Bras-Amorós, and M. E. O'Sullivan, *Unique decoding of plane AG codes via interpolation*, IEEE Trans. Inf. Theory, **58(6)** (2012), 3941-3950.

[5] K. Lee and M. E. O'Sullivan, *List decoding of Hermitian codes using Gröbner bases*, Journal of Symbolic Computation, **44** (2009), 1662-1675.

[6] J. C. Rosales and P. A. García-Sánchez, *Numerical semigroups*, volume 20 of Developments in Mathematics, Springer, New York, 2009.

[7] R. M. Roth and G. Ruckenstein, *Efficient decoding of Reed-Solomon codes beyond half the minimum distance*, IEEE Trans. Inf. Theory, **46(1)** (2000), 246-257.

[8] W. A. Stein et al, *Sage Mathematics Software (Version 4.8)*, The Sage Development Team, 2012. http://www.sagemath.org

Kwankyu Lee
Department of Mathematics and Education, Chosun University,
Gwangju 501-759, Korea.
E-mail: kwankyu@chosun.ac.kr