FINITE FIELDS AND THEIR APPLICATIONS 1, 83–101 (1995)

A Fast Decoding Method of AG Codes from Miura–Kamiya Curves C_{ab} up to Half the Feng–Rao Bound

S. SAKATA*

Department of Computer Science and Information Mathematics, The University of Electro-Communications, Chofu-shi, Tokyo 182, Japan
E-mail: sakata@shinano.cs.uec.ac.jp

J. JUSTESEN AND Y. MADELUNG

The Institute of Circuit Theory and Telecommunication, Technical University of Denmark, Bldg. 343, DK-2800 Lyngby, Denmark

AND

H. E. JENSEN AND T. HØHOLDT

The Mathematical Institute, Technical University of Denmark, Bldg. 303, DK-2800 Lyngby, Denmark

Communicated by Oscar Moreno

Received January 18, 1994

We present a fast version of the Feng–Rao algorithm for decoding of one-point algebraic-geometric (AG) codes derived from the curves which Miura and Kamiya classified as C_{ab}. Our algorithm performs the Feng–Rao algorithm efficiently by using the Sakata algorithm, i.e., the 2D Berlekamp–Massey algorithm. One can decode the one-point AG codes up to half of the Feng–Rao bound d_{FR} which is greater than or equal to the designed distance d^*. We have proven the validity and the performance of our algorithm in the framework of our own theory, depending little on algebraic geometry.

1. INTRODUCTION

Algorithms for decoding geometric Goppa codes or algebraic-geometric (AG) codes have been given by several authors [1–12]. In particular,

* With Toyohashi University of Technology until March 31, 1994.

1071-5797/95 $6.00

Skorobogatov and Vlăduţ [3] first presented an effective decoding method of general AG codes. Recently Feng and Rao [7], Duursma [8], and Ehrhard [9] have given algorithms for decoding up to $\lfloor (d^* - 1)/2 \rfloor$ or less errors, where d^* is the designed minimum distance of the code. Their algorithms are superior to the Skorobogatov–Vlăduţ algorithm in the sense that the latter can correct only $\lfloor (d^* - g - 1)/2 \rfloor$ or less errors, where g is the genus of the curve from which the code is defined. In fact, as is shown by Kirfel and Pellikaan [13], the Feng–Rao algorithm can correct $\lfloor (d_{\mathrm{FR}} - 1)/2 \rfloor$ or less errors, where d_{FR} is the Feng–Rao bound which is greater than or equal to the Goppa bound d^*. The Feng–Rao algorithm, which can decode one-point codes, is based on Gaussian elimination for solving the system of syndrome equations and on a kind of majority logic, which was generalized by Duursma [8] to general AG codes. From the known syndrome values it finds the unknown syndrome values, which are necessary to decode up to the better bound, by voting for several candidate values. Its computational complexity, which is almost the same as that of Gaussian elimination of a syndrome matrix, is of order $\mathcal{O}(n^3)$, where n is the length of the code.

In this paper, we present a fast version of the Feng–Rao algorithm for the decoding of some one-point codes, i.e., a class of AG codes derived from the curves which Miura and Kamiya treated in [14] and classified as C_{ab} in their paper [15] and which contain Hermitian curves as well. Our algorithm which simulates the Feng–Rao algorithm in an efficient way is an application of the Sakata algorithm, i.e., the 2D Berlekamp–Massey algorithm [16, 17]. We show that one can decode one-point AG codes up to half the Feng–Rao bound d_{FR} by a modification of the Sakata algorithm. We must emphasize that one can obtain the candidate values and the number of votes for them in majority logic decoding by using a minimal polynomial set of a subarray of the 2D syndrome array at each iteration. Furthermore, we can prove the validity of our decoding method in the framework of the Sakata algorithm and the theory of 2D Hankel matrices [18], depending on linear algebra, but not much on algebraic geometry. While a subset or a single element of a minimal polynomial set was used in the previous applications of the Sakata algorithm for decoding of AG codes [4, 10–12], the whole minimal polynomial set is used effectively and without loss in our present method. Our decoding method has complexity of order $\mathcal{O}(n^{7/3})$ for the one-point AG codes derived from the Miura–Kamiya curves C_{ab}.

2. One-Point AG Codes Derived from Miura–Kamiya Curves C_{ab}

In this paper we focus our attention to one-point AG codes $(\mathscr{C}, \mathscr{P}, D)_\Omega$ over $K = GF(q)$ derived from Miura–Kamiya curves $\mathscr{C} = C_{ab}$ in their

terminology [15], where $\mathcal{C}$ is an absolutely irreducible plane curve, $\mathcal{P} = \{P_1, \ldots, P_n\}$ is a set of K-rational points of $\mathcal{C}$, and $D = mQ$ is a divisor of a K-rational point $Q \notin \mathcal{P}$ of the curve $\mathcal{C}$. More exactly, the smooth model of the singular curve $\mathcal{C}$ is used instead. For a pair of coprime integers a and b, $a < b$, $\mathcal{C}$ is defined by

$$Y^a Z^{b-a} + \sum_{(i,j)\in\Sigma(ab-1)} c_{ij} X^i Y^j Z^{b-i-j} + c_{b0} X^b = 0, \tag{1}$$

where $\Sigma(m) := \{(i, j) \in \Sigma \mid ai + bj \le m\}$, $m \in N_0$, is a subset of the 2D integral lattice $\Sigma := N_0 \times N_0$ and $c_{b0} \neq 0$. (We denote the set of positive integers as N, the set of nonnegative integers as N_0, and $\{i \in N \mid i \le m\}$ as N^m, $\{i \in N_0 \mid i \le m\}$ as N_0^m, respectively.) In the following, we often consider subsets of Σ: $\Sigma_{(b)} := \{(i, j) \in \Sigma \mid i < b\}$, and the total order $\le_T$ restricted within each of these subsets, respectively. Except in case of $b = a + 1$, $\mathcal{C}$ is singular and $Q := (0 : 1 : 0)$ is its unique singular point whose multiplicity is $b - a$. By desingularizing $\mathcal{C}$, the genus of its smooth model is shown to be $g = (a - 1)(b - 1)/2$ [15]. Hermitian curves ($a = b - 1 = r$, $q = r^2$), elliptic curves ($a = 2$, $b = 3$), and hyperelliptic curves ($a = 2$, $b = 2g + 1$) are among this class of curves. Q corresponds to exactly one place, which is identified with Q itself for the simplicity of notation. It is shown that the rational functions $\varphi := X/Z$ and $\psi := Y/Z$ have a single pole at Q with orders $O_Q(\varphi) = -a$, $O_Q(\psi) = -b$, respectively, and that a basis of $L(D)$ is given as $B(mQ) := \{\varphi^i \psi^j \mid (i, j) \in \Sigma_{(b)}(m)\}$, where $\Sigma_{(b)}(m) := \Sigma(m) \cap \Sigma_{(b)} = \{(i, j) \in \Sigma \mid ai + bj \le m, i < b\}$. From the defining equation (1) of the curve, we have a linear dependency among $\{\varphi^i \psi^j \mid (i, j) \in \Sigma\}$:

$$\psi^a + \sum_{(i,j)\in\Sigma(ab-1)} c_{ij} \varphi^i \psi^j + c_{b0} \varphi^b = 0. \tag{2}$$

A code of the above type is a linear code of length $n = \#\mathcal{P}$ over K which is the orthogonal complement of the following subspace of K^n: $EV_{\mathcal{P}}(L(D)) := \{(f(P_1), \ldots, f(P_n)) \in K^n \mid f \in L(D)\}$, where $L(D)$ is the subspace of the linear space $K(\mathcal{C})$ of rational functions of the curve $\mathcal{C}$ defined by the divisor D. If $m < n$, the evaluation map $EV_{\mathcal{P}}$ is injective, and hence the code has dimension $k^* := n - m + g - 1$ $(= n - m + ((ab - a - b - 1)/2))$. The designed minimum distance $d^* := m - 2g + 2$ $(= m - ab + a + b + 1)$, provided that $m > 2g - 2$ $(= ab - a - b - 1)$. We want to correct t $(:= \lfloor (d_{\mathrm{FR}} - 1)/2 \rfloor \ge \lfloor (d^* - 1)/2 \rfloor = \lfloor (m + 1)/2 \rfloor - g)$ or less errors, where we determine the Feng–Rao bound d_{FR} of our code in the next section. We know a variety of such codes, among which there are codes treated by [4, 10–12, 19, 20]. For example, Stichtenoth [19], Justesen *et al.* [4], and Shen [10] treated codes from Hermitian curves: $y^r + y -$

$x^{r+1} = 0$ over $K = GF(r^2)$; Kamiya and Miura [11] treated codes from curves $y^a + y - x^b = 0$ for a pair of integers $a < b$, which were discussed by Stichtenoth [20]. The following discussions will be valid for any one-point AG code if $L(mQ)$ is generated by the powers $\{\varphi^i\psi^j \mid (i, j) \in \Sigma_{(b)}(m)\}$ of two rational functions φ and ψ with $O_Q(\varphi) = -a$, $O_Q(\psi) = -b$, where a and b are coprime. Furthermore, it is possible to generalize to the case of $L(mQ)$ generated by powers of k (≥ 2) rational functions, and to apply the kD Sakata algorithm [17] to decoding of such codes.

In our case, the nongaps o_k, $0 \leq k$, for Q are given by

$$\{-O_Q(\varphi^i\psi^j) \mid (i, j) \in \Sigma_{(b)}(m)\} = N_0^m \backslash \Lambda, \tag{3}$$

where $\Lambda := \{bi - aj \mid 1 \leq i \leq a - 1, 1 \leq j \leq [bi/a]\}$ is the gap sequence and $\#\Lambda = \sum_{i=1}^{a-1} [bi/a] = (a - 1)(b - 1)/2$ is the genus g of the curve $\mathcal{C}$. We take the total order $\leq_T$ over the 2D integral lattice Σ defined by

$$\begin{aligned} p = (p_1, p_2) \leq_T q = (q_1, q_2) \\ \Leftrightarrow ap_1 + bp_2 < aq_1 + bq_2 \vee (ap_1 + bp_2 = aq_1 + bq_2 \wedge p_1 \leq q_1) \end{aligned}$$

corresponding to the pair of coprime integers a, b and its restrictions within subsets of Σ such as $\Sigma_{(b)}(m)$, $\Sigma_{(b)}$, etc. We denote the bijection from $\Sigma_{(b)}$ onto N as κ_b and that from $\Sigma_{(b)}(m)$ onto N^{m-g+1} as $\kappa_{m,b}$, respectively, according to each total order $\leq_T$. Thus, we have the natural numbering of the functions composing the following basis of $L(D)$:

$$f_k = \varphi^{k_1}\psi^{k_2}, \quad 1 \leq k \leq m - g + 1. \tag{4}$$

Here $k = \kappa_{m,b}(k_1, k_2) \in N^{m-g+1}$ for $(k_1, k_2) \in \Sigma_{(b)}(m)$. The nongaps $o_{k-1} := -O_Q(f_k)$ are $o_0 = 0 < o_1 < o_2 < \cdots < o_{g-1} < 2g$; $o_k = k + g$, $k \geq g$ (see, e.g., Theorem I.6.7 of [21]), and f_k is denoted as $\phi_{o_{k-1}}$, $1 \leq k \leq m - g + 1$, in the terminology of Feng and Rao [7]. Thus, for $k \geq g$, $k = \kappa_b(k_1, k_2) = ak_1 + bk_2 - g$.

We assume that a codeword $\mathbf{c} = (c_k)$ is sent, and that $\mathbf{u} = (u_k) = \mathbf{c} + \mathbf{e}$ is received, where $\mathbf{e} = (e_k)$ is an error vector of weight $\nu \leq t$, and $e_{k_\mu} \neq 0$, $1 \leq \mu \leq \nu$; $e_k = 0$, $k \notin \{k_1, \ldots, k_\nu\}$. Then, the 2D syndromes

$$s_{i,j} = \sum_{\mu=1}^{\nu} e_{k_\mu}\phi_{o_{i-1}}(P_{k_\mu})\phi_{o_{j-1}}(P_{k_\mu}), \quad 1 \leq i, j \leq m - g + 1, \tag{5}$$

and the syndrome matrix $\mathbf{S} = \|s_{i,j}\|$ are introduced [7], where the elements $s_{i,j}$ of $\mathbf{S}$ are known only for (i, j) with $o_{i-1} + o_{j-1} \leq m$ from the received word $\mathbf{u}$. The elements $s_{i,j}$ and $s_{i',j'}$ are *dependent* on each other if $o_{i-1} +$

$o_{j-1} = o_{i'-1} + o_{j'-1} (= k)$; i.e., one is determined from the other by the linear relation (2) on the basis of the values $s_{i'',j''}$, $o_{i''-1} + o_{j''-1} < k$. In our case, $o_{i-1} = ai_1 + bi_2$, $o_{j-1} = aj_1 + bj_2$, and $s_{i,j} = \sum_{\mu=1}^{\nu} e_{k_\mu} \varphi(P_{k_\mu})^{i_1+j_1} \psi(P_{k_\mu})^{i_2+j_2}$, where $(i_1, i_2), (j_1, j_2) \in \Sigma_{(b)}(m)$ correspond to $i, j \in N^{m-g+1}$, respectively. Thus, $s_{i,j}$ and $s_{i',j'}$ are dependent on each other if and only if $a(i_1 + j_1) + b(i_2 + j_2) = a(i'_1 + j'_1) + b(i'_2 + j'_2)$, where $\kappa_{m,b}(i_1, i_2) = i$, $\kappa_{m,b}(j_1, j_2) = j$ and $\kappa_{m,b}(i'_1, i'_2) = i'$, $\kappa_{m,b}(j'_1, j'_2) = j'$. If $i_1 + j_1 = i'_1 + j'_1$ and $i_2 + j_2 = i'_2 + j'_2$, we have $s_{i,j} = s_{i',j'}$. Thus, we have some distinct elements $s_{i,j}$ and $s_{i',j'}$ which are dependent on each other, where $i, j, i', j' \in N^{m-g+1}$. We call them a dependent pair.

3. Fast Decoding Method up to Half the Feng–Rao Bound

Instead of the matrix terminology, we use the following data structure, i.e., the 2D syndrome array $\bar{s} = (\bar{s}_p)$:

$$\bar{s}_p := \sum_{\mu=1}^{\nu} e_{k_\mu} \varphi(P_{k_\mu})^{p_1} \psi(P_{k_\mu})^{p_2}. \tag{6}$$

Here $p = (p_1, p_2) \in \Sigma$. (*Remark*. The element $\bar{s}_p = \bar{s}_{(p_1,p_2)}$ of our 2D array $\bar{s}$ is different from the element s_{p_1,p_2} of the Feng–Rao matrix **S**. The sequence $(\bar{s}_{(p_1,p_2)} \mid (p_1, p_2) \in \Sigma_{(b)}(m))$ arranged in the total order appears in the first row of the matrix **S**, and its subsequences appear in the remaining rows.) $\bar{s}_p$ and $\bar{s}_q$, $p = (p_1, p_2)$, $q = (q_1, q_2) \in \Sigma_{(2b)}$, are dependent on each other if and only if $ap_1 + bp_2 = aq_1 + bq_2$, where, if $p = (p_1, p_2) \in \Sigma_{(b)}$, $q = (q_1, q_2) = (p_1 + b, p_2 - a) \in \Sigma_{(2b)} \setminus \Sigma_{(b)}$. From now on, we use the notation of the Sakata algorithm [16, 17] freely. Thus, in the following, we have the concepts such as the next point $p \oplus 1$ (with respect to the total order $\leq_T$) of a point $p (\in \Sigma)$, a subarray $\bar{s}^p := (\bar{s}_q \mid q <_T p)$, the excluded point set $\Delta (\subset \Sigma)$ of $\bar{s}^p$, a minimal polynomial set $F (\subset K[x, y])$ of $\bar{s}^p$, etc. In fact, since we know that the curve has a defining polynomial of the form

$$C := y^a + \sum_{(i,j) \in \Sigma(ab-1)} c_{ij} x^i y^j + c_{b0} x^b, \tag{7}$$

a linear relation corresponding to (2) is valid for $\bar{s}_p$, $p \in \Sigma$, and so we have only to execute the algorithm within $\Sigma_{(2b-1)}$ (with respect to the total order $\leq_T$ restricted in $\Sigma_{(2b-1)}$) to determine uniquely the reduced minimal polynomial set of the 2D syndrome array over the whole region Σ. Therefore, we consider the 2D syndrome array $\bar{s}$ over $\Sigma_{(2b-1)}$ mainly. For some additional purpose, we sometimes consider the 2D syndrome array $\bar{s}$ over $\Sigma_{(2b)}$ as well.

For a rational function $f = \sum_{q=(q_1,q_2)\in\Gamma_f} f_q \varphi^{q_1} \psi^{q_2}$ $(\in K(\mathcal{C}))$ or its corresponding polynomial $f = \sum_{q=(q_1,q_2)\in\Gamma_f} f_q x^{q_1} y^{q_2}$ $(\in K[x, y])$, we consider a 2D linear recurrence

$$f[\bar{s}]_p := \sum_{q\in\Gamma_f} f_q \bar{s}_{q+p-d} = 0, \quad \forall p(\in \Sigma_{(2b)}) \geq_P d, \tag{8}$$

which should be *valid* for (or satisfied by) the 2D syndrome array $\bar{s}$, where $\leq_P$ is the natural partial order over Σ and $d(\in \Sigma_{(b)})$ is the degree of f defined as $\deg(f) := \max_T\{q \in \Gamma_f\}$, i.e., the maximum element of the subset $\Gamma_f(\subset \Sigma_{(b)})$ corresponding to the nonzero terms of f. (We denote both function and polynomial by the same symbol f.) The following is a key to decoding by using the Sakata algorithm.

PROPOSITION 1. *The function f has the error locators $I = \{P_{k_1}, \ldots, P_{k_\nu}\}(\subset \mathcal{P})$ among the zeros, i.e.,*

$$f(P) = \sum_{q\in\Gamma_f} f_q \varphi(P)^{q_1} \psi(P)^{q_2} = 0, \quad \forall P \in I, \tag{9}$$

if and only if the 2D linear recurrence (8) *is valid for the subarray $\bar{s}^{(r+d)\oplus 1}$ up to $(r + d) \oplus 1$; i.e.,* (8) *is satisfied at any p such that $d \leq_T p \leq_T r + d$, where $r = (r_1, r_2)$ with $ar_1 + br_2 \geq \nu + 2g - 1$.*

This is derived from the following observation: If we have the identity for $0 \leq_T p \leq_T r$, where $r = (r_1, r_2)$ such that $ar_1 + br_2 \geq \nu + 2g - 1$,

$$\begin{aligned}\sum_{q\in\Gamma_f} f_q \bar{s}_{q+p} &= \sum_{q\in\Gamma_f} f_q \sum_{\mu=1}^{\nu} e_{k_\mu} \varphi(P_{k_\mu})^{q_1+p_1} \psi(P_{k_\mu})^{q_2+p_2} \\ &= \sum_{\mu=1}^{\nu} e_{k_\mu} \varphi(P_{k_\mu})^{p_1} \psi(P_{k_\mu})^{p_2} \left(\sum_{q\in\Gamma_f} f_q \varphi(P_{k_\mu})^{q_1} \psi(P_{k_\mu})^{q_2} \right) = 0,\end{aligned}$$

then (9) holds, because by the Riemann–Roch theorem the evaluation map $EV_I : L(mQ) \to K^\nu$ defined by $f \to (f(P_{k_1}), \ldots, f(P_{k_\nu}))$ is surjective if $m - 2g + 1 \geq \nu$. (If $m - \nu \geq 2g - 1$, then $\dim(L(D)) = m - 2g + 1$ and $\dim(L(mQ - \sum_{i=1}^{\nu} P_{k_i})) = m - 2g + 1 - \nu$. Thus, the image of the evaluation map has dimension ν equal to $\dim(K^\nu)$.) (*Remark.* The above identity holds for $0 \leq_T p \leq_T r$ if and only if (8) holds for $d \leq_T p \leq_T r + d$.) Therefore, our first main concern is how to find an error locator polynomial f or a 2D linear recurrence valid for the given syndrome array $\bar{s}$, where we remark that we do not know the values $\bar{s}_p$ for $p = (p_1, p_2)$ such that $ap_1 + bp_2 > m$.

The above matrix $\mathbf{S} = \|s_{i,j}\|$ is a submatrix of the 2D Hankel matrix $\bar{\mathbf{S}}$ of the 2D array $\bar{s}$ introduced in [18], where $s_{i,j} := \bar{s}_{p+q}$ for $i = \kappa_{m,b}(p)$, $j = \kappa_{m,b}(q)$ $(\in N^{m-g+1})$ which correspond to p, q $(\in \Sigma_{(b)}(m))$, respectively. Thus, the Feng-Rao algorithm fits in well with the framework of the Sakata algorithm. We consider an extension $\bar{\mathbf{S}}$ of $\mathbf{S}$ which contains the rows i and columns j such that $i = \kappa_b(p)$, $j = \kappa_b(q)$, $p, q \in \Sigma_{(b)}$. If

$$\sum_{p' \in \Gamma_f} f_{p'} \bar{s}_{p'+q'} = 0, \quad \forall q' \leq_T q, \tag{10}$$

then $f = \sum_{p'=(p'_1,p'_2)\in\Gamma_f} f_{p'} x^{p'_1} y^{p'_2}$ with $\deg(f) = d(\leq_T p)$ is said to represent a linear row dependency of a submatrix $\bar{\mathbf{S}}_{p,q} := \|s_{i,j}\|_{1\leq i\leq \kappa_b(p), 1\leq j\leq \kappa_b(q)}$ of $\bar{\mathbf{S}}$, which corresponds to a 2D linear recurrence valid for a subarray of $\bar{s}$. In the discussions of 2D Hankel matrix, we have the following lemma (Lemma 2 of [18]).

LEMMA 1. *Let f with $\deg(f) = q$ represent a linear row dependency of the submatrix $\bar{\mathbf{S}}_{q,q}$ of $\bar{\mathbf{S}}$, $q <_T p$. If every column of $\bar{\mathbf{S}}_{q,p}$ is linearly dependent on the columns of $\bar{\mathbf{S}}_{q,q}$, then f represents also a linear row dependency of $\bar{\mathbf{S}}_{q,p}$ (i.e., f corresponds to a 2D linear recurrence).*

From a similar consideration it is easy to see that f is a minimal polynomial of $\bar{s}^{p\oplus 1} := (\bar{s}_q \mid q \leq_T p)$ if and only if the following conditions are satisfied for $d := \deg(f)$:

(1) f represents a linear row dependency of $\bar{\mathbf{S}}_{d,p-d}$;

(2) there exists no linear row dependency of any submatrix $\bar{\mathbf{S}}_{t,p-d}$, $t <_P d$.

Now, we consider how to find the candidate values $\hat{s}_{i,j}$ of $s_{i,j} = \bar{s}_{p+q}$ of $\bar{\mathbf{S}}$ for $i = \kappa_b(p)$, $j = \kappa_b(q)$ such that $a(p_1 + q_1) + b(p_2 + q_2) = m + w$, assuming that the syndrome values $\bar{s}_t$ for $at_1 + bt_2 \leq m + w - 1$ $(1 \leq w \leq g)$ are known. Let $r = (r_1, r_2) = p + q \in \Sigma_{(b)}$ such that $ar_1 + br_2 = m + w$. If $r_2 \geq a$, we have another point $r' = (r'_1, r'_2) = (r_1 + b, r_2 - a) \in \Sigma_{(2b)}\backslash\Sigma_{(b)}$ such that $\bar{s}_r$ and $\bar{s}_{r'}$ are dependent on each other. (*Remark*. There exists a single point $r = (r_1, r_2) \in \Sigma_{(b)}$ such that $ar_1 + br_2 = k$ for each $k \in N_0\backslash\Lambda$. Furthermore, r' is the next point $r \oplus 1$ of r with respect to the total order within $\Sigma_{(2b)}$.) If $r_2 < a$, we do not have the element $\bar{s}_{r'}$ which is dependent on $\bar{s}_r$. We will show that one can use a minimal polynomial set F of the subarray $\bar{s}^r$ for our purpose. In the terminology of Feng and Rao [7], a column containing $\times$ in the row i of the matrix $\bar{\mathbf{S}}$ is not any linear combination of its previous columns with the top $i - 1$ components, where that the row i or the column j of $\bar{\mathbf{S}}$ contains $\times$ means that the row i or the column j of the discrepancy matrix obtained from $\bar{\mathbf{S}}$ by elementary column operations contains $\times$ (We abuse the terminology). This implies that the rows $\bar{\mathbf{S}}$

containing $\times$ are linearly independent, as is seen from linear-algebraic discussions. Therefore, the excluded point set Δ of $\bar{s}^{p+q}$ is composed of the points t ($\in \Sigma_{(b)}$) which correspond to the rows k ($= \kappa_b(t)$) of $\bar{\mathbf{S}}$ containing the mark $\times$, because there exists no valid polynomial f with $\deg(f) \in \Delta$ for $\bar{s}^{p+q}$. On the other hand, it follows from Lemma 1 that we can calculate each candidate value $\hat{s}_{p+q}$ at the position in row $i = \kappa_b(p)$ and in column $j = \kappa_b(q)$ (of the matrix $\bar{\mathbf{S}}$) with $a(p_1 + q_1) + b(p_2 + q_2) = m + w$ and $p, q \notin \Delta$ by using a minimal polynomial $f(\in F)$ with $\deg(f) = d \leq_P p$ of $\bar{s}^{p+q}$ as

$$\hat{s}_{p+q} := -\sum_{u \in \Gamma_f \setminus \{d\}} f_u \bar{s}_{u+p+q-d},$$

where we assume $f_d = 1$ without any loss of generality. (We denote the candidate value $\hat{\bar{s}}_{p+q}$ as $\hat{s}_{p+q}$ for simplicity.) Any such f gives the same candidate value $\hat{s}_{p+q}$ of $\bar{s}_{p+q}$, i.e., that candidate value does not depend on any choice of a minimal polynomial $f \in F$ with $\deg(f) \leq_P p$, as is seen from the following lemma. (Its proof is given in the Appendix.

LEMMA 2. *Let Δ be the excluded point set of $\bar{s}^{p+q}$ with $p, q \notin \Delta$. For a minimal polynomial set F of $\bar{s}^{p+q}$ and $f, f' \in F$ with* $\deg(f), \deg(f') \leq_P p, f$ *is valid at $p + q$ if and only if f' is valid at $p + q$ for any assumed value $\hat{s}_{p+q}$ ($\in K$).*

The above statements imply also that, for $r \in \Sigma_{(b)}$ with $ar_1 + br_2 = m + w$, we can calculate every candidate value $\hat{s}_{i,j} = \hat{s}_r$ at the position in row i and in column j such that $p + q = r$, $p = \kappa_b^{-1}(i)$, $q = \kappa_b^{-1}(j)$, from the known syndrome values $\bar{s}_t$, $t <_T r$, by the same minimal polynomial f with $\deg(f) \leq_P p := \kappa_b^{-1}(i)$. Similarly, for $r' = (r_1', r_2') = r \oplus 1 = (r_1 + b, r_2 - a) \in \Sigma_{(2b)} \setminus \Sigma_{(b)}$ we can calculate every candidate value $\hat{s}_{i',j'} = \hat{s}_{r'}$ at position in row i' and column j' such that $p' + q' = r'$, $p' = \kappa_b^{-1}(i')$, $q' = \kappa_b^{-1}(j')$, which is dependent on $\hat{s}_r$,

$$\hat{s}_r + \sum_{v \in \Sigma(ab-1)} c_v \bar{s}_{r-(0,a)+v} + c_{b0} \hat{s}_{r'} = 0, \tag{11}$$

because of the defining equation (1) of the curve or the linear relation (2). We say that the candidate values $\hat{s}_r$ and $\hat{s}_{r'}$ are consistent to each other if they satisfy (11). We remark that each candidate value $\hat{s}_r$ (or $\hat{s}_{r'}$) denoted by @ in [7] appears just in rows $i = \kappa_b(p)$ such that $p \leq_P r$ (or r'). The number of these @'s are counted in voting in Feng-Rao's majority logic scheme, where the number of @'s either in rows $i = \kappa_b(p)$, $p \in \Delta$, or in columns $j = \kappa_b(q)$, $q \in \Delta$, is excluded in voting. Motivated by these considerations, we introduce the following subsets of $\Sigma_{(2b)}$ for $r = (r_1, r_2) \in \Sigma_{(b)}$ and $r' = (r_1', r_2') = (r_1 + b, r_2 - a) \in \Sigma_{(2b)} \setminus \Sigma_{(b)}$ (if r' exists):

$$\Pi_{r,f} := \Sigma_{\deg(f)} \cap \Gamma_r, \quad \Pi_{r',f} := \Sigma_{\deg(f)} \cap \Gamma_{r'} \cap \Sigma_{(b)},$$
$$\Pi_r := ((\cup_{f\in F}\Pi_{r,f}) \cup (\cup_{f\in F}\Pi_{r',f}))\backslash((r - \Delta) \cup (r' - \Delta)). \tag{12}$$

Here $\Gamma_s := \{q \in \Sigma_{(2b)} \mid q \leq_P s\}$, $\Sigma_s := \{q \in \Sigma_{(2b)} \mid q \geq_P s\}$, and $s - \Delta := \{s - t \mid t \in \Delta \cap \Gamma_s\}$ for $s \in \Sigma_{(2b)}$. (*Remark.* $\Delta + \Sigma\backslash(\cup_{f\in F}\Sigma_{\deg(f)})$.) Here we have the following important observation: We have that the consistency holds for the known values $\bar{s}_p$. That is, for any pair of $p = (p_1, p_2) <_T r$ and $p' = (p_1', p_2') <_T r'$ such that $p_1' = p_1 + b$, $p_2' = p_2 - a$, $\bar{s}_p$ and $\bar{s}_{p'}$ are consistent, from which we have in view of linearity

PROPOSITION 2. *If* $\deg(f) \in \Gamma_r \cap \Gamma_{r'}$, *a single polynomial* $f \in F$ *gives a consistent pair of candidate values* $\hat{s}_r$ *and* $\hat{s}_{r'}$.

On the other hand, Lemma 2 implies that, if $t \in \Pi_r$ and $\deg(f)$, $\deg(f') \leq_P t$, then f and f' give the same and/or consistent candidate values of $\bar{s}_r$ and/or $\bar{s}_{r'}$. In particular, we have

COROLLARY 1. *For* $i = \kappa(p), j = \kappa(q), d = \deg(f) \leq_P p$, *and* $i' = \kappa(p')$, $j' = \kappa(q')$, $d' = \deg(f') \leq_P p'$, *where* $p' + q' = p + q = r$, *if* $\Pi_r \cap \Sigma_d \cap \Sigma_{d'} \neq \emptyset$, *then the candidate values* $\hat{s}_{i,j}$ *and* $\hat{s}_{i',j'}$ *coincide*.

Proposition 2 and Corollary 1 allow us to introduce any maximal subset F' of a minimal polynomial set F of $\bar{s}^r$ as follows:

(*) Any pair $f, f' \in F'$ satisfies one of the following conditions:

(1) If $\deg(f)$, $\deg(f') \leq_P r$, f and f' give the same candidate value of $\bar{s}_r$;

(2) if $\deg(f) \leq_P r$, $\deg(f') \leq_P r'$, f and f' give the candidate values of $\bar{s}_r$ and $\bar{s}_{r'}$, respectively, which are consistent to each other;

(3) if $\deg(f)$, $\deg(f') \leq_P r'$, f and f' give the same candidate value of $\bar{s}_{r'}$.

Then, we stipulate that the number of votes for that subset F' is

$$\#(\Pi_{r,F'} \cup \Pi_{r',F'})\backslash((r - \Delta) \cup (r' - \Delta)), \tag{13}$$

where $\Pi_{r,F'} := \cup_{f\in F'}\Pi_{r,f}$, $\Pi_{r',F'} := \cup_{f\in F'}\Pi_{r',f}$. (*Remark.* The value (13) coincides with the number of votes in Feng-Rao's scheme.) Now, we have the following:

LEMMA 3. *The value of* (13) *coincides with*

$$\#(\tilde{\Gamma}_{r,F'} \cup \tilde{\Gamma}_{r',F'})\backslash\Delta, \tag{14}$$

where $\tilde{\Gamma}_{r,F'} := \cup_{f\in F'}\Gamma_{r-\deg(f)}$, $\tilde{\Gamma}_{r',F'} := (\cup_{f\in F'}\tilde{\Gamma}_{r'-\deg(f)}) \cap \Sigma_{(b)}$.

(For its proof, see the Appendix.)

We call a maximal subset F' of F satisfying the above condition (*) by the name of a consistent subset. (*Remark.* If $f \in F$ such that $\deg(f) \leq_P r$ (or r') but $\Pi_{r,f} \backslash ((r - \Delta) \cup (r' - \Delta))$ (or $\Pi_{r',f} \backslash ((r - \Delta) \cup (r' - \Delta))) = \emptyset$, then the number of votes for that candidate value obtained by f is 0, and it does not influence the decision of the correct candidate values.) Now, we assume that a consistent subset F' of F is preferred; i.e., we take an extended subarray $\hat{s}^{r' \oplus 1}$ which is obtained by appending the candidate values $\hat{s}_r$ and $\hat{s}_{r'}$ given by the polynomials in F' to the known part $\bar{s}^r$. Then, by Lemma 4 of [13], we know that the excluded point set Δ' of the assumed subarray $\hat{s}^{r' \oplus 1}$ for the next point $r' \oplus 1$ of r' has cardinality,

$$\#\Delta' = \#\Delta + \#(\tilde{\Gamma}_{r,F \backslash F'} \cup \tilde{\Gamma}_{r',F \backslash F'}) \backslash \Delta. \tag{15}$$

By Lemma 3, this identity implies the following:

LEMMA 4. *If a consistent subset F' has been preferred, then the increase of the excluded point set, i.e., $\#\Delta' - \#\Delta$ is equal to the number of votes against its preference*: $\#(\Pi_{r,F \backslash F'} \cup \Pi_{r',F \backslash F'}) \backslash ((r - \Delta) \cup (r' - \Delta))$.

Feng and Rao have proven the fact (Theorem 2 of [7]) that the number of candidates in $\mathbf{S}$ is at least $m - 2g$ $(= d^* - 2)$. Let $\bar{\mathbf{S}}'$ be the submatrix of $\bar{\mathbf{S}}$ which is obtained from $\bar{\mathbf{S}}$ by deleting the rows and columns containing $\times$, i.e., corresponding to the points $p \in \Delta$. For the number $\mu (:= \#\Delta)$ of $\times$, the number of these rows and columns is $2(\mu - 1)$, as seen from the discussions of Feng and Rao [7]. Therefore, the number of candidates in $\bar{\mathbf{S}}'$ which is the total number of votes for all the candidate values is at least $m - 2g - 2(\mu - 1) = d^* - 2\mu$. Now, independently from the Feng–Rao theory, we will prove that the total number of votes for all the candidate values is greater than or equal to $d^m_{\mathrm{FR}} - 2\mu$, where d^m_{FR} is the Feng–Rao bound defined by

$$d^m_{\mathrm{FR}} := \min_{r=(r_1,r_2) \in \Sigma_{(b)} : ar_1 + br_2 > m} \#\Gamma_r \cup (\Gamma_{r'} \cap \Sigma_{(b)}). \tag{16}$$

(*Remark.* This just coincides with Kirfel–Pellikaan's definition [13].) First, we have the following:

THEOREM 1. *For any $r \in \Sigma_{(b)}$, we have the total number of votes*

$$\#\Pi_r \geq \#\Gamma_r \cup (\Gamma_{r'} \cap \Sigma_{(b)}) - 2\#\Delta. \tag{17}$$

(For its proof, see the Appendix.)

From Theorem 1, we have the important theorem which is equivalent to Theorem 4 of Feng and Rao [7]. Our reasoning is independent from their arguments.

THEOREM 2. *Let the number t of errors be less than or equal to $\lfloor (d^m_{\mathrm{FR}} - 1)/2 \rfloor$, then, for any $r = (r_1, r_2) \in \Sigma_{(b)}$ and $r' = (r_1 + b, r_2 - a)$ (if r' exists) such that $ar_1 + br_2 > m$, the number of votes for the correct candidate value $\hat{s}_r$ and its consistent correct candidate value $\hat{s}_{r'}$ (if exists) is strictly greater than the number of votes for the incorrect candidate values.*

(For the proof, see the Appendix.)

By the way, we can prove that the Feng–Rao bound d^m_{FR} is not less than the Goppa bound, i.e., the designed distance $d^* = m - 2g + 2$, independently from any other theory such as Kirfel and Pellikaan [13].

LEMMA 5. For $r = (r_1, r_2) \in \Sigma_{(b)}$ such that $m < ar_1 + br_2$, *we have* $\#\Gamma_r \cup (\Gamma_{r'} \cap \Sigma_{(b)}) \geq d^* = m - 2g + 2 = m - ab + a + b + 1$.

(For its proof, see the Appendix.)

Combining Theorem 2 with Lemma 5, we have proven

COROLLARY 2. *For $t = \lfloor (d^* - 1)/2 \rfloor$, t or less errors can be corrected by our method.*

As is shown in the above, we can correct a little more than $\lfloor (d^* - 1)/2 \rfloor$ errors in case $d^m_{\mathrm{FR}} > d^*$. In fact, it occurs rather rarely that $d^m_{\mathrm{FR}} > d^*$ [13].

From the above discussions, we have the following algorithm for finding the unknown syndrome values for decoding one-point AG codes derived from Miura–Kamiya curves C_{ab} up to half the designed minimum distance or rather up to half the Feng–Rao bound. For $r = (r_1, r_2) \in \Sigma_{(b)}$ such that $ar_1 + br_2 = m + 1$, we can assume a subarray $\bar{s}^r$ of the 2D syndrome array $\bar{s}$ and a minimal polynomial set F of $\bar{s}^r$. Then, for each $w = 1, 2, \ldots$, we find the values $\bar{s}_r$ and $\bar{s}_{r'}$ at $r = (r_1, r_2) \in \Sigma_{(b)}$ such that $ar_1 + br_2 = m + w$ and $r' = (r_1 + b, r_2 - a) \in \Sigma_{(2b)} \backslash \Sigma_{(b)}$ from the known part $\bar{s}^r$ by using a minimal polynomial set F of $\bar{s}^r$. From the extended subarray $\bar{s}^{r' \oplus 1}$, we can find a minimal polynomial set F of $\bar{s}^{r' \oplus 1}$ for the next point $r' \oplus 1$ of r' after two iterations of the Sakata algorithm at r and r'. Thus, we can proceed to find the remaining unknown syndrome values of the whole 2D syndrome array. If we have the whole syndrome array, we can find a Gröbner basis of the error locator ideal whose zeros just coincide with the error locations. In fact, we do not need to find the whole syndrome array by Algorithm 1. Later we show a more efficient method for finding the whole syndrome array.

ALGORITHM 1: Finding the Unknown Syndrome Values.

Step 1. (Initialization). $w := 1$.

Step 2. Calculate all the consistent candidate values of the dependent pair $\bar{s}_r$ and $\bar{s}_{r'}$ for $r = (r_1, r_2) \in \Sigma_{(b)}$ such that $ar_1 + br_2 = m + w$ and $r' = r \oplus$

$1 = (r_1 + b, r_2 - a) \in \Sigma_{(2b)} \backslash \Sigma_{(b)}$ and the number of votes for each of them, and decide the correct values of them by majority voting.

Step 3. At r and r', execute the iterative procedures of the Sakata algorithm so that we find a pair of a minimal polynomial set F and the auxiliary polynomial set G of $\bar{s}^{r' \oplus 1}$. [At the same time, we update D, T, and Δ.]

Step 4. If $\#\Delta > t$ then stop. Else $w := w + 1$; $r := r' \oplus 1 \in \Sigma_{(b)}$.

Step 5. If r is a destination point e then stop else go to Step 2.

For the above, we have some remarks as follows.

(1) If $r' \in \Sigma_{(2b)} \backslash \Sigma_{(b)}$ does not exist, we have only to execute the procedure at r.

(2) If $\#\Delta > t$ in Step 4, it means that we have more than t errors, which cannot be corrected.

(3) The new point $r \in \Sigma_{(b)}$ in Step 4 is taken to be the successor of the point r' according to the total order restricted within $\Sigma_{(2b-1)}$.

(4) We have only to iterate the steps up to $e \oplus 1$, where $e = (e_1, e_2)$ is such that $a(e_1 - d_1) + b(e_2 - d_2) \geq t + 2g - 1$ for an error locator polynomial f with $\deg(f) = d = (d_1, d_2)$, in view of Proposition 1. On the other hand, for $D = D(\Delta)(\subset \Sigma)$ with $\#\Delta \leq t$ such that $\Delta = \Sigma \backslash (\cup_{d \in D} \Sigma_d)$, $d(\Delta) := \min_{d \in D(\Delta)} d$ takes its maximum value

$$d^* = \max_{\Delta : \#\Delta \leq t} d(\Delta) \tag{18}$$

if and only if $\Delta = \{p \in \Sigma_{(b)} \mid p <_T d\}$, where max and min mean the maximum and the minimum with respect to $\leq_T$, respectively. Since $\#\Delta \leq t$, $\kappa_b(d) = ad_1 + bd_2 - g \leq t + 1$. Therefore, if $ae_1 + be_2 \geq 2t + 3g$, we have $a(e_1 - d_1) + b(e_2 - d_2) \geq t + 2g - 1$. Furthermore, the uniqueness condition for the valid polynomial f having the minimum degree $\deg(f) = d$ among a reduced minimal polynomial set F of $\bar{s}^{e \oplus 1}$ is satisfies if $2d \leq_T e \oplus 1$, where the inequality holds if $ae_1 + be_2 \geq 2t + 3g$, in view of $2(ad_1 + bd_2 - g) \leq 2(t + 1) \leq 2t + 2g < ae_1 + be_2 - g + 1$. Thus, we can take e such that $ae_1 + be_2 = 2t + 3g$.

(5) If the error locator polynomial f has $\deg(f)$ greater than the degree of the curve defining polynomial C, we hit C firstly. However, we can continue to get an appropriate error locator f different from C up to the above-mentioned point $e \oplus 1$.

Summarizing all the procedures for decoding, we have the following:

ALGORITHM 2: Fast Algorithm for Decoding up to $t = \lfloor (d^m_{\mathrm{FR}} - 1)/2 \rfloor$ Errors.

Step 1. By applying the Sakata algorithm to a subarray $\bar{s}^r$ for $r = (r_1, r_2) \in \Sigma_{(b)}$ such that $ar_1 + br_2 = m + 1$, find a pair of its minimal polynomial set F and the auxiliary polynomial set G such that $D := \{\deg(f) \mid f \in F\}$ and $T := \{\mathrm{span}(g) \mid g \in G\}$ are *dual* in the sense of [17]. (In particular, $\Delta = \Gamma_T = \Sigma \backslash \Sigma_D$.)

Step 2. By using Algorithm 1, find the unknown syndrome values and the unique minimal polynomial f having the minimum (with respect to $\leq_T$) degree $\deg(f) = d = (d_1, d_2)$ among a minimal polynomial set F of $s^{e \oplus 1}$, where $e = (e_1, e_2)$ is such that $ae_1 + be_2 = 2t + 3g$.

Step 3. By using the error locator polynomial f, which just has been obtained in Step 2 and the defining polynomial $C = c_{b0}x^b + y^a + \sum_{(i,j)\in\Sigma(ab-1)} c_{ij}x^i y^j$ (7) of the curve, find the remaining part of the 2D syndrome array $\bar{s}$ over the whole period by a method similar to Justesen *et al.*'s [4]. In fact, if $\deg(f) = d = (d_1, d_2)$, $d_1 < b$, $d_2 \leq a$, we can get the following polynomials valid for $\bar{s}$:

$$f' := y^{d_2}C - c_{b0}x^{b-d_1}f = y^{a+d_2} + \sum_{(i,j)\in\Sigma_{(2b-1)(b(a+d_2))}} f'_{ij}x^i y^j, \tag{19}$$

$$f'' := x^{d_1}C - y^{a-d_2}f = c_{b0}x^{b+d_1} + \sum_{(i,j)\in\Sigma_{(2b-1)(a(b+d_1))}} f''_{ij}x^i y^j. \tag{20}$$

Here, except for the first term y^{a+d_2} (respectively, $c_{b0}x^{b+d_1}$), all the terms of f' (respectively, f'') have degree (i, j) such that $ai + bj < (a + d_2)b$ (respectively, $(b + d_1)a$). To obtain the values $\bar{s}_r$, $ar_1 + br_2 = m + w$, and $\bar{s}_{r'}$, $(r'_1, r'_2) = (r_1 + b, r_2 - a)$, from the values $\bar{s}_r$, $ar_1 + br_2 \leq m + w - 1$, for each w, we can use f' (respectively, f'') if $r_2 \geq a + d_2$ (respectively, $r_1 \geq b + d_1$).

Step 4. From the 2D syndrome array $\bar{s}$ which has been obtained in Step 3, find the error values by the inverse discrete Fourier transform,

$$e_{k_\mu} = \sum_{p_1=0}^{q-2} \sum_{p_2=0}^{q-2} \bar{s}_p(\alpha^{-c_\mu})^{p_1}(\alpha^{-d_\mu})^{p_2}, \tag{21}$$

where $\varphi(P_{k_\mu}) = \alpha^{c_\mu}$, $\psi(P_{k_\mu}) = \alpha^{d_\mu}$ for a primitive element α of $K = GF(q)$. In the above, we assume that the error values e_{k_μ} which correspond to the points P_{k_μ} such that $\varphi(P_{k_\mu}) = 0$ or $\psi(P_{k_\mu}) = 0$ vanish. Even if some of those error values do not vanish, we can find them. For example, the error value at P_{k_μ} such that $\varphi(P_{k_\mu}) = \psi(P_{k_\mu}) = 0$ is obtained as $\bar{s}_{(0,0)} - \bar{s}_{(0,q-1)}$ [22].

The following is taken from an example of Section IV of Feng and Rao [7].

EXAMPLE 1. We consider a code $(\mathscr{C}, \mathscr{P}, D)_\Omega$, where $\mathscr{C}$ is a Hermitian curve: $X^5 + Y^4Z + YZ^4 = 0$ over $K = GF(2^4)$, $\mathscr{P}$ is all the K-rational points of $\mathscr{C}$ except for $Q = (0:1:0)$, and $D = 23Q$. In this case $n = 64$, $a = 4$, $b = 5$, $g = 6$, $m = 23$, and thus, $k^* = n - m + g - 1 = 46$, $d^m_{\mathrm{FR}} = d^* = m - 2g + 2 = 13$, and $t = 6$. $\{\varphi^i\psi^j \mid 4i + 5j \le 23, i \le 4\}$ is a basis of $L(23Q)$, where $\varphi := X/Z$, $\psi := Y/Z$ have Q as a single pole and $a = -O_Q(\varphi) = 4$, $b = -O_Q(\psi) = 5$. The gap sequence is $\{1, 2, 3, 6, 7, 11\}$. $\{\bar{s}_{(5,0)}, \bar{s}_{(0,4)}\}$ is a pair of dependent values; in general, we have

$$\bar{s}_{(i+5,j)} = \bar{s}_{(i,j+1)} + \bar{s}_{(i,j+4)} \tag{22}$$

from the defining equation of the curve. For each $(i, j) \in \Sigma_{(5)}$ (23) the order of each rational function $\varphi^i\psi^j$ is shown in Fig. 1a. For a 6-error pattern $\mathbf{e} = (e_{P_j})$ with $e_{P_j} = 0$, $P_j \in \mathscr{P}$, except for $e_{(1:1:\alpha)} = \alpha^{12}$, $e_{(1:1:\alpha^2)} = \alpha^4$, $e_{(1:1:\alpha^4)} = \alpha^7$, $e_{(1:1:\alpha^8)} = \alpha^8$, $e_{(0:0:1)} = \alpha^9$, and $e_{(1:\alpha:1)} = \alpha^9$, the known part of the 2D syndrome array is shown in Fig. 1b. (α is a primitive element of $GF(2^4)$ whose minimum polynomial is $x^4 + x + 1$.) $\circ$ are the unknown values of the dependent pair $\bar{s}_r$ and $\bar{s}_{r'}$ just in question at the current points $r = (1, 4)$ and $r' = (6, 0)$. The Sakata algorithm is applied with respect to the total order $\le_T$ defined by the pair 4, 5.

Before we begin the iteration of our algorithm 1, we calculate a minimal polynomial set F of $\bar{s}^r$ for $r = (1, 4)$ by the Sakata algorithm. The result is as follows:

$$F = \{f_1 := x^2 + \alpha^{14}x + \alpha^3, f_2 := xy + y + \alpha^3x + \alpha^3,$$
$$f_3 := y^2 + \alpha^4y + \alpha^9x + \alpha^3\}$$

and $G = \{x + \alpha^{13}, y + \alpha x + \alpha^7\}$. (By the way, $D = \{(2, 0), (1, 1), (0, 2)\}$, $T = \{(1, 0), (0, 1)\}$, and $\Delta = \{(0, 0), (1, 0), (0, 1)\}$.) Now, we start by setting $w = 1$ and go to Step 2; i.e., calculate the candidate values of the dependent pair $\bar{s}_{(1,4)}$ and $\bar{s}_{(6,0)}$ by using f_2 (or f_3) and f_1, respectively. Thus, we have the candidate values $\bar{s}_{(1,4)} = \alpha^8$ and $\bar{s}_{(6,0)} = \alpha^{14}$, which are not consistent in the sense that they do not satisfy the identity (22). (*Remark*. f_2 and

(a)

0	5	10	15	20
4	9	14	19	24
8	13	18	23	
12	17	22		
16	21			
20				
24				

(b)

α^1	α^2	α^9	α^5	α^5
α^{14}	α^4	α^{10}	α^2	$\circ$
α^{11}	α^{11}	α^8	α^{10}	
α^4	α^6	α^5		
α^0	α^{12}			
α^1				
$\circ$				

FIG. 1. (a) Orders of $\varphi^i\psi^j$. (b) 2D syndrome array.

$$\begin{array}{ccccccc}
\alpha^1 & \alpha^2 & \alpha^9 & \alpha^5 & \alpha^5 & \circ & * \\
\alpha^{14} & \alpha^4 & \alpha^{10} & \alpha^2 & \alpha^8 & \bullet & \\
\alpha^{11} & \alpha^{11} & \alpha^8 & \alpha^{10} & \bullet & & \\
\alpha^4 & \alpha^6 & \alpha^5 & \bullet & & & \\
\alpha^0 & \alpha^{12} & \bullet & & & & \\
\alpha^1 & \circ & & & & & \\
\alpha^5 & \bullet & & & & & \\
\bullet & & & & & &
\end{array}$$

FIG. 2. The first iteration of the algorithm.

f_3 give the same value of $\bar{s}_{(1,4)}$ by Lemma 2.) The numbers of votes for the former and the latter are $\#(\Sigma_{(0,2)} \cup \Sigma_{(1,1)}) \cap \Gamma_{(1,4)} \backslash ((1, 4) - \Delta) = 4$ and $\#\Sigma_{(2,0)} \cap \Gamma_{(6,0)} \backslash ((6, 0) - \Delta) = 3$, respectively. Thus, the majority logic gives the correct value $\bar{s}_{(1,4)} = \alpha^8$, and so $\bar{s}_{(6,0)}$ should be α^5 by (22), and we update the syndrome array as in Fig. 2. (The symbols $\circ$ are the unknown values of the next dependent pair $\bar{s}_r$ and $\bar{s}_{r'}$ for $r = (0, 5)$ and $r' = (5, 1)$ which we will try to find in Step 2 of the next iteration, and $*$ is a destination $e \oplus 1$.)

Now we proceed to Step 3; i.e., restart the iteration of the Sakata algorithm at (1, 4) and stop at the next point (0, 5). Thus, we have a minimal polynomial set

$$F = \{x^5 + \alpha^{14}x^4 + \alpha^3x^3 + \alpha^{12}x + \alpha^{10},$$
$$xy + y + \alpha^3x + \alpha^3, y^2 + \alpha^4y + \alpha^9x + \alpha^3\}$$

and the auxiliary polynomial set $G = \{x^2 + \alpha^{14}x + \alpha^3, y + \alpha x + \alpha^7\}$, where we remark that the polynomial f_1 has been updated. ($D = \{(5, 0), (1, 1), (0, 2)\}$, $T = \{(4, 0), (0, 1)\}$, and $\Delta = \{(0, 0), (1, 0), (0, 1), (2, 0), (3, 0), (4, 0)\}$. Thus, $\#\Delta = 6$.) Similarly, we proceed to update the syndrome array and a pair of F and G (and D, T, Δ) alternately.

At $e \oplus 1 = (0, 6)$, we have a minimal polynomial set

$$F = \{x^5 + x^4 + \alpha^6x^3 + \alpha^6x^2 + \alpha^{11}y + \alpha^{12},$$
$$xy + x^2 + y + x, y^2 + x^2 + \alpha^4y + \alpha^4x\},$$

among which f_2 and f_3 have the correct error locators $I = \{(1 : 1 : \alpha), (1 : 1 : \alpha^2), (1 : 1 : \alpha^4), (1 : 1 : \alpha^8), (0 : 0 : 1), (1 : \alpha : 1)\}$ as zeros, where $e = (e_1, e_2) = (1, 5)$ satisfies $4(e_1 - d_1) + 5(e_2 - d_2) = 20$, $19 > t + 2g - 1 = 17$ for $d = (d_1, d_2) = \deg(f_2) = (1, 1)$ and for $d = (d_1, d_2) = \deg(f_3) = (0, 2)$, respectively. (*Remark.* f_1 does not have the correct error locators as zeros.) By the way, by restarting the iterations of our algorithm at (0, 6) and continuing them, we obtain a unique minimal polynomial set F of $\bar{s}^{e' \oplus 1}$

for $e' \oplus 1 = (9, 0)$ as

$$F = \{x^5 + x^4 + \alpha^{11}y + \alpha^{12}x, \\ xy + x^2 + y + x, y^2 + x^2 + \alpha^4 y + \alpha^4 x\},$$

which is a (reduced) Gröbner basis and every element of which has the error locators I as zeros. Furthermore, after having the whole period of the array $\bar{s}$, we can find the error values.

(*Remark.* For $m \geq 18$, $d^m_{\mathrm{FR}} = d^*$. However, e.g., for $m = 16$, $d^m_{\mathrm{FR}} = 8 > d^* = 6$ and three errors can be corrected.)

To estimate the computational complexity of our decoding algorithm, we assume that $\mathbb{O}(n) = \mathbb{O}(m) > \mathbb{O}(g)$ and we consider correction of t or less errors with $\mathbb{O}(t) = \mathbb{O}(n)$. For the first stage (Steps 1 and 2 of Algorithm 2) of finding an error locator, we need to update at most b polynomials having size of $\mathbb{O}(t)$ at every point of $\Sigma_{(b)}$ up to $e \oplus 1$, where e depends on $d = \deg(f)$ of the error locator f, but we have only to take $e = (e_1, e_2)$ such that $\mathbb{O}(ae_1 + be_2) = \mathbb{O}(t + g) = \mathbb{O}(n)$. For the first stage, we have complexity of order $\mathbb{O}(btm) = \mathbb{O}(bm^2)$. For good codes, we have that $\mathbb{O}(n) = \mathbb{O}(bq) = \mathbb{O}(g\sqrt{q})$. If $\mathbb{O}(a) = \mathbb{O}(b)$, then $\mathbb{O}(g) = \mathbb{O}(ab) = \mathbb{O}(a^2)$, and so we have $\mathbb{O}(a) = \mathbb{O}(\sqrt{q})$ and $\mathbb{O}(n) = \mathbb{O}(a^3)$, from which it follows that the complexity is $\mathbb{O}(n^{7/3})$ as is the case treated by [4]. If $\mathbb{O}(a) < \mathbb{O}(b)$, then the complexity is much less, because we can take $\Sigma^{(a)} := \{(i, j) \in \Sigma \mid j < a\}$ instead of $\Sigma_{(b)}$ and execute the algorithm over $\Sigma^{(a)}$ with complexity $\mathbb{O}(am^2)$. By $a + d_2 + a + (a/b)d_1 < 4a$, if $(m + w)/b > 4a$, i.e., $m + w > 8g \sim 4ab$, we already have the known syndrome values necessary to use the two polynomials f' and f'' derived from the curve defining polynomial C and the unique minimal polynomial f having the minimum degree $\deg(f) = (d_1, d_2)$. Therefore, provided that the procedure of the first stage is executed until $m + w > 8g$, the second stage (Step 3) of obtaining the error values in our decoding algorithm has computational complexity of order $\mathbb{O}(abq^2) = \mathbb{O}(q^3) = \mathbb{O}(n^2)$, which is less than $\mathbb{O}(n^{7/3})$ for the first stage. (*Remark.* The additional computation in the first stage does not increase the order of complexity.) The final stage (Step 4) also has order $\mathbb{O}(tq^2) = \mathbb{O}(n^{7/3})$.

4. Conclusion

We have presented a fast version of the Feng–Rao decoding algorithm [7] for the class of algebraic-geometric codes derived from Miura–Kamiya curves C_{ab} [15]. It is a kind of simulation by the Sakata algorithm (i.e., the 2D Berlekamp–Massey algorithm) [16, 17]. The Sakata algorithm and the theory of 2D Hankel matrices [18] are applied well for that

purpose. In particular, the majority logic for finding the unknown syndrome values is realized just in the iterative procedures of the Sakata algorithm. We have proven the validity and the performance of our method only in the framework of our theory, depending on linear algebra, but not much on algebraic geometry. (A similar idea is presented in [23], which is carried out in the framework of the previous papers [1, 4] and restricted to codes from nonsingular curves.) The total complexity of computation in finding error locators and error values is $\mathbb{O}(n^{7/3})$ for the class of algebraic–geometric codes. Here, we refer to the result by Feng *et al.* [24, 25]. They gave a fast decoding algorithm by using the method of solving a system of linear equations whose coefficient matrix is of Hankel block-Hankel (or Toeplitz block-Toeplitz) type. The latter has been known in the field of system theory. Their algorithm also has complexity of $\mathbb{O}(n^{7/3})$. Our algorithm, which is distinct from their method, has several merits practically as well as theoretically, owing to its clearcut structure. In particular, we can generalize the present decoding method to a wider class of codes, e.g., to codes defined by $L(mQ)$ having a basis of the form $\{\phi_1^{i_1} \cdot \cdots \cdot \phi_k^{i_k}\}$, which can be decoded by the kD Sakata algorithm [17] with a little more computational complexity.

APPENDIX

Proof of Lemma 2. Let $\deg(f) = d$, $\deg(f') = d'$. Then, from $q \notin \Delta$, there exists $f'' \in F$ such that $\deg(f'') = d'' \leq_P q$. Therefore, from $d, d' \leq_P p$, $d + d'', d' + d'' \leq_P p + q$, which, by Lemma 4 of [16], implies that f is valid at $p + q$ if and only if f'' is valid at $p + q$ and that f'' is valid at $p + q$ if and only if f' is valid at $p + q$. Consequently, f is valid at $p + q$ if and only if f' is valid at $p + q$. Q.E.D.

Proof of Lemma 3. We can show that we have for any subset F' of F the one-to-one correspondence between the subsets $(\Pi_{r,F'} \cup \Pi_{r',F'})\backslash((r - \Delta) \cup (r' - \Delta))$ and $(\tilde{\Gamma}_{r,F'} \cup \tilde{\Gamma}_{r',F'})\backslash\Delta$ of $\Sigma_{(b)}$. We divide both subsets into two parts: $(\Pi_{r,F'} \cup \Pi_{r',F'})\backslash((r - \Delta) \cup (r' - \Delta)) = (\Pi_{r,F'}\backslash(r - \Delta)) \cup ((\Pi_{r',F'}\backslash\Pi_{r,F'})\backslash((r - \Delta) \cup (r' - \Delta)))$ (*Remark.* $\Pi_{r,F'}\backslash((r - \Delta) \cup (r' - \Delta)) = \Pi_{r,F'}\backslash(r - \Delta)$ because $\Gamma_r \cap (r' - \Delta) = \emptyset$ in view of $\Delta \subset \Sigma_{(b)}$ and $r_1' - r_1 = b$.) and $(\tilde{\Gamma}_{r,F'} \cup \tilde{\Gamma}_{r',F'})\backslash\Delta = (\tilde{\Gamma}_{r,F'}\backslash\Delta) \cup (\tilde{\Gamma}_{r',F'}\backslash(\Delta \cup \Gamma_r))$. First, for a point $q \in \Pi_{r,F'}\backslash(r - \Delta)$, we have a point $q' := r - q \in \tilde{\Gamma}_{r,F'}\backslash\Delta$, and vice versa, because $q \leq_P r$, $\deg(f) \leq_P q$ for some $f \in F'$, and $q \notin r - \Delta$ if and only if $q' \leq_P r$, $q' \leq r - \deg(f)$ for some $f \in F'$, and $q' \notin \Delta$. Second, for a point $q \in (\Pi_{r',F'}\backslash\Pi_{r,F'})\backslash((r - \Delta) \cup (r' - \Delta))$ we have a point $q' := r' - q \in \tilde{\Gamma}_{r',F'}\backslash(\Delta \cup \Gamma_r)(\subseteq \Sigma_{(b)})$, and vice versa, because $q \leq_P r'$, $\deg(f) \leq_P q$ for some $f \in F'$, and $q \notin (r' - \Delta) \cup \Gamma_r$, $q \in \Sigma_{(b)}$ if and only if $q' \leq_P r'$, $q' \leq r'$

$-\deg(f)$ for some $f \in F'$, and $q' \notin \Delta \cup \Gamma_r$, $q' \in \Sigma_{(b)}$, in view of $q'_1 \geq r'_1 - b = r_1$ and $q'_1 \leq r'_1 - r_1 = b$. Q.E.D.

Proof of Theorem 1. In view of the one-to-one correspondence between Π_r and $(\tilde{\Gamma}_{r,F} \cup \tilde{\Gamma}_{r',F})\backslash\Delta$ which is shown in the above proof of Lemma 3, we have

$$\begin{aligned}
\#\Pi_r &= \#(\tilde{\Gamma}_{r,F} \cup \tilde{\Gamma}_{r',F})\backslash\Delta \\
&\geq \#((\cup_{f\in F}\Gamma_{r-\deg(f)}) \cup ((\cup_{f\in F}\Gamma_{r'-\deg(f)}) \cap \Sigma_{(b)}) - \#\Delta \\
&= \#((\Gamma_r \cup \Gamma_{r'}) \cap \Sigma_{(b)})\backslash\Delta - \#\Delta \\
&\geq \#\Gamma_r \cup (\Gamma_{r'} \cap \Sigma_{(b)}) - 2\#\Delta.
\end{aligned}$$

Q.E.D.

Proof of Theorem 2. Otherwise, the total number of votes against the correct consistent candidate values is greater than or equal to $\lfloor d^m_{\mathrm{FR}}/2 \rfloor - \#\Delta$ by Theorem 1. Then, when the correct candidate value is assumed (as it should be) and then the next iteration of the Sakata algorithm is executed on the extended correct 2D array, we have by Lemma 4 that $\#\Delta' - \#\Delta \geq \lfloor d^m_{\mathrm{FR}}/2 \rfloor - \#\Delta$, which implies that $\#\Delta' \geq d^m_{\mathrm{FR}}/2$. But, it contradicts $(d^m_{\mathrm{FR}} - 1)/2 \geq \lfloor (d^m_{\mathrm{FR}} - 1)/2 \rfloor = t \geq \#\Delta'$ from the assumption that we have got t or less errors. Q.E.D.

Proof of Lemma 5. We have $\#\Gamma_r = (r_1 + 1)(r_2 + 1)$ and $\#\Gamma_{r'} \cap \Sigma_{(b)}\backslash\Gamma_r = (b - r_1 - 1)(r_2 - a + 1)$. Thus, $\#(\Gamma_r \cup (\Gamma_{r'} \cap \Sigma_{(b)}))\backslash\Gamma_r = r_1 a + r_2 b - ab + a + b > m - ab + a + b$. (*Remark*. Even if $\Gamma_{r'} \cap \Sigma_{(b)}\backslash\Gamma_r = \varnothing$, we have the same inequality, as is seen easily.) Q.E.D.

Acknowledgments

We are indebted to Dr. Shinji Miura and Dr. Norifumi Kamiya for many interesting suggestions and helpful discussions on Miura-Kamiya codes. We also thank anonymous referees for their useful comments.

References

1. J. Justesen, K. J. Larsen, A. Havemose, H. E. Jensen, and T. Høholdt, Construction and decoding of a class of algebraic geometry codes, *IEEE Trans. Inform. Theory* **35** (July 1989), 811–821.
2. R. Pellikaan, On a decoding algorithm for codes on maximal curves, *IEEE Trans. Inform. Theory* (Nov. 1989), 1228–1232.
3. A. N. Skorobogatov and S. G. Vlăduţ, On decoding of algebraic geometric codes, *IEEE Trans. Inform. Theory* **36** (Sept. 1990), 1051–1060.
4. J. Justesen, K. J. Larsen, H. E. Jensen, and T. Høholdt, Fast decoding of codes from algebraic plane curves, *IEEE Trans. Inform. Theory* **38** (Jan. 1992), 111–119.

5. S. C. Porter, B.-Z. Shen, and R. Pellikaan, Decoding geometric Goppa codes using an extra place, *IEEE Trans. Inform. Theory* **38** (Nov. 1992), 1663–1676.
6. B.-Z. Shen, Solving a congruence on a graded algebra by a subresultant sequence and its application, *J. Symbolic Comput.*, to appear.
7. G. L. Feng and T. R. N. Rao, Decoding algebraic–geometric codes up to the designed minimum distance, *IEEE Trans. Inform. Theory* **39** (Jan. 1993), 37–45.
8. I. M. Duursma, Majority coset decoding, *IEEE Trans. Inform. Theory* **39** (May 1993), 1067–1070.
9. D. Ehrhard, Achieving the designed error capacity in decoding algebraic-geometric codes, *IEEE Trans. Inform. Theory* **39** (May 1993), 743–751.
10. B.-Z. Shen, Codes from Hermitian curves and an iterative decoding algorithm, preprint, Eindhoven University of Technology, June 1991.
11. N. Kamiya and S. Miura, On a fast decoding algorithm for geometric Goppa codes defined on certain algebraic curves with at most one higher cusp, presented at the 1993 IEEE International Symposium on Information Theory, San Antonio, TX, Jan. 1993.
12. M. Kurihara and K. Kobayashi, A fast decoding algorithm of algebraic–geometric codes from some plane curves, submitted for publication.
13. C. Kirfel and R. Pellikaan, The minimum distance of codes in an array coming from telescopic semigroups, presented at The Fourth Workshop on Arithmetic Geometry and Coding Theory, Luminy, France, June 1993.
14. S. Miura and N. Kamiya, Geometric-Goppa codes on some maximal curves and their minimum distance, presented at The IEEE Workshop on Information Theory, Susonoshi, Japan, June 1993.
15. S. Miura and N. Kamiya, "On the Minimum Distance of Codes from Some Maximal Curves," Technical Report of IEICE, IT92-147, Mar. 1993. [In Japanese]
16. S. Sakata, Finding a minimal set of linear recurring relations capable of generating a given finite two-dimensional array, *J. Symbolic Comput.* **5** (May 1988), 321–337.
17. S. Sakata, Extension of the Berlekamp–Massey algorithm to N dimensions, *Inform. and Comput.* **84** (Feb. 1990), 207–239.
18. S. Sakata, On the minimal partial realization of 2D discrete linear shift-invariant systems, *IEEE Trans. Automat. Control* **36** (Aug. 1991), 984–988.
19. H. Stichtenoth, A note on Hermitian codes over $GF(q^2)$, *IEEE Trans. Inform. Theory* **34** (Sept. 1988), 1345–1348.
20. H. Stichtenoth, Über die Automorphismengruppe eines algebraischen Funktionenkörpers von Primzahlcharacterisitik, Teil II: Eine spezieller Typ Funktionenkörpern, *Arch. Math.* **24** (1973), 615–631.
21. H. Stichtenoth, "Algebraic Function Fields and Codes," Springer-Verlag, Berlin, 1993.
22. Y. Madelung, "Implementation of a Decoding Algorithm for AG-Codes from the Hermitian Curve," Technical Report of the Institute of Circuit Theory and Telecommunication, ISSN 0105-8541, IT-93-137, Technical University of Denmark, 1993.
23. S. Sakata, J. Justesen, Y. Madelung, H. E. Jensen, and T. Høholdt, Fast decoding of AG-codes up to the designed minimum distance, submitted for publication.
24. G. L. Feng, V. K. Wei, T. R. N. Rao, and K. K. Tzeng, The designed distance decoding of a class of algebraic–geometric codes, Part I: A new theory without Riemann–Roch theorem, presented at the IEEE Workshop on Information Theory, Brazil, 1992.
25. G. L. Feng, V. K. Wei, T. R. N. Rao, and K. K. Tzeng, The designed distance decoding of a class of algebraic–geometric codes, Part II: Fast algorithm and block-Hankel matrices, presented at the IEEE Workshop on Information Theory, Brazil, 1992.