

A Polynomial-Time Attack on the McEliece Cryptosystem on Elliptic Codes with Arbitrary Divisors

Artyom Kuninets¹, Ekaterina Malygina¹, Evgeniy Melnichuk¹

¹QApp, Bolshoy blvd 30b1, Moscow, 121205, Russian Federation.

Contributing authors: artkuninets@yandex.ru; emalygina@qapp.tech; emelnichuk@qapp.tech;

Acknowledgments

The authors of this work, as well as Yu.D. Shkuratov, would like to express their gratitude to the participants of the Summer School–Conference “Cryptography and Information Security”—Ya. P. Vysotsky, R. V. Mashchenkov, P. S. Morozova, and S. A. Orlova—for identifying the limitations in the work [1].

Abstract

The McEliece cryptosystem based on algebraic geometry codes has been proposed as a way to reduce the key size of code-based cryptography, but several structural attacks have demonstrated the vulnerability of particular families of algebraic geometry codes. Despite this, until recently, there remained schemes and parameter sets that were not vulnerable to any known attack. We propose a new structural attack with “hints” that applies to elliptic codes with arbitrary effective divisors. In particular, we prove that, given the elliptic curve, the public generator matrix, and three points from the evaluation divisor, the entire divisor can be recovered in polynomial time, independently of the number of errors used in the cryptosystem. The attack requires $\mathcal{O}(k^2n^2 + |\mathcal{E}(\mathbb{F}_q)| + n)$ operations in $\mathbb{F}_q$ and succeeds with overwhelming probability, after which the second divisor is recovered in $\mathcal{O}(k^2n^2 + (|\mathcal{E}(\mathbb{F}_q)| - n)n^2)$ operations. We further propose an optimized version of the attack that requires no additional information at all. Exploiting the action of the automorphisms of the curve, the three known points are replaced by the enumeration of a single pair of field elements, which yields an equivalent key on the given public curve in $\mathcal{O}(k^2n^2 + q^2 + (|\mathcal{E}(\mathbb{F}_q)| - n)n^2)$ operations on average.

1 Introduction

The McEliece cryptosystem was proposed in 1978 [2] and is one of the most extensively studied post-quantum cryptographic schemes. It possesses several advantages, including high encryption and decryption speeds, however, the large key size required to achieve a given security level hinders the integration of code-based cryptosystems into practical protocols. To address this issue, numerous attempts have been made over the past several decades to incorporate various families of codes into the McEliece cryptosystem. The scheme based on algebraic geometry (AG) codes was first proposed in [3], and subsequent constructions achieved better parameters and significantly reduced key sizes compared to Goppa codes.

Subsequent cryptanalysis demonstrated that schemes based on most AG codes are not secure. In [4], a deterministic polynomial-time algorithm was presented for recovering the structure of Reed–Solomon codes (AG codes associated with curves of genus $g = 0$). The works [5, 6] claimed that the McEliece cryptosystem based on elliptic and hyperelliptic codes ($g \leq 2$) is insecure, however, these attacks require finding minimum weight codewords, which is considered a computationally hard problem [7] unless the code is MDS. In [8], it was shown that an equivalent representation of an AG code can be obtained using only the generator matrix of the code. Finally, in 2017, an attack [9] applicable to codes of arbitrary genus was proposed, resulting in a polynomial-time decoding algorithm without knowledge of the secret parameters.

All of the previously described attacks have limitations, and their applicability can be prevented during key generation by imposing additional conditions on the codes and parameters, as was done, for example, by the authors of the ECC² cryptosystem [10] based on elliptic codes, which until recently was considered secure [1].

Our contribution. In this work, we propose a structural attack with “hints” on the McEliece cryptosystem based on elliptic codes, which, unlike the previously known attacks, applies to arbitrary effective divisors G and to an arbitrary number of inserted errors t . We prove that, given the elliptic curve $\mathcal{E}$, the public generator matrix $\mathbf{G}_{\text{pub}}$ and three points of the evaluation divisor D , the divisor D is recovered in $\mathcal{O}(k^2n^2 + |\mathcal{E}(\mathbb{F}_q)| + n)$ operations in $\mathbb{F}_q$ with probability $1 - \mathcal{O}(|\mathcal{E}(\mathbb{F}_q)|^{-2})$, after which the divisor G is recovered in $\mathcal{O}(k^2n^2 + (|\mathcal{E}(\mathbb{F}_q)| - n)n^2)$ operations. We then propose a modification of the attack that requires no knowledge of the points of D . Since the automorphisms of the curve act on the set of secret keys without changing the code, the three “hints” can be replaced by the enumeration of a single pair of field elements. This modification recovers an equivalent key on the given public curve in $\mathcal{O}(k^2n^2 + q^2 + (|\mathcal{E}(\mathbb{F}_q)| - n)n^2)$ operations on average. The obtained results show that the McEliece cryptosystem based on elliptic codes, including the parameter sets proposed for the ECC² scheme [10], is insecure for arbitrary effective divisors and for any number of inserted errors.

Organization of the paper. Section 2 introduces the preliminary information on divisors, Riemann–Roch spaces and algebraic geometry codes necessary for the rest of the paper. Section 3 is devoted to the cryptanalysis of the McEliece cryptosystem based on elliptic codes. In Section 3.1 we recall the computation of the P -filtration of an algebraic geometry code together with the decoding attack of [9], and in Section 3.2 we revisit the attack of [1] on the ECC² scheme. Next, in Section 3.3 we present our main results, namely the attack recovering the divisor D from three known points and the subsequent recovery of the divisor G for arbitrary effective divisors. Section 3.4 focuses on the optimized version of the attack, which requires no knowledge of the points of D and returns an equivalent key on the given public curve.

2 Preliminaries

2.1 Algebraic geometry codes

Let $\mathcal{X}$ be a smooth, affine, and absolutely irreducible algebraic curve of genus g defined over the finite field $\mathbb{F}_q$. A *divisor* on $\mathcal{X}$ (defined over $\mathbb{F}_q$) is a finite formal sum of places of $\mathcal{X}$, that is,

$$G = \sum_P v_P(G)P,$$

where $v_P(G) \in \mathbb{Z}$ and all but finitely many coefficients are zero. The set of all such divisors forms an abelian group, denoted by $\text{Div}(\mathcal{X})$. We denote by $\mathcal{X}(\mathbb{F}_q)$ the set of $\mathbb{F}_q$ -rational points of $\mathcal{X}$.

For a divisor $G = \sum_P v_P(G)P$, its *support* is defined as

$$\text{Supp}(G) = \{P \mid v_P(G) \neq 0\},$$

and its *degree* is given by

$$\deg(G) = \sum_P v_P(G) \deg(P).$$

A divisor G is said to be *effective*, written $G \geq 0$, if $v_P(G) \geq 0$ for every place P . This induces a partial order on $\text{Div}(\mathcal{X})$ by declaring $G_1 \geq G_2$ whenever $G_1 - G_2$ is effective, that is, whenever $v_P(G_1) \geq v_P(G_2)$ for all P .

The *function field* of the curve $\mathcal{X}$ is denoted by

$$\mathbb{F}_q(\mathcal{X}) = \left\{ \frac{g(x, y)}{h(x, y)} \mid g, h \in \mathbb{F}_q[x, y], h \neq 0 \right\} / \sim,$$

where $\frac{g}{h} \sim \frac{g'}{h'} \Leftrightarrow (gh' - g'h)(P) = 0$ for all $P \in \mathcal{X}$.

For a nonzero rational function $f \in \mathbb{F}_q(\mathcal{X})$, its *principal divisor* is defined by

$$(f) = (f)_0 - (f)_\infty,$$

where $(f)_0$ and $(f)_\infty$ denote the zero and pole divisors of f , respectively.

Given a divisor $G \in \text{Div}(\mathcal{X})$, the associated *Riemann–Roch space* is defined as

$$\mathcal{L}(G) = \{f \in \mathbb{F}_q(\mathcal{X}) \mid (f) + G \geq 0\} \cup \{0\}.$$

This is a finite-dimensional vector space over $\mathbb{F}_q$, and its dimension is denoted by $\ell(G)$.

Let $D = P_1 + \cdots + P_n$ be a divisor consisting of n distinct $\mathbb{F}_q$ -rational points of $\mathcal{X}$, and assume that $\text{Supp}(D) \cap \text{Supp}(G) = \emptyset$.

We define the evaluation map

$$\text{ev}_D : \begin{cases} \mathcal{L}(G) \longrightarrow \mathbb{F}_q^n, \\ f \longmapsto (f(P_1), \dots, f(P_n)). \end{cases}$$

Definition 2.1.1 (Algebraic Geometry Code). *The algebraic geometry code associated with the pair (D, G) is defined as*

$$\mathcal{C}_{\mathcal{L}}(D, G) = \{\text{ev}_D(f) \mid f \in \mathcal{L}(G)\} \subseteq \mathbb{F}_q^n.$$

The code $\mathcal{C}_{\mathcal{L}}(D, G)$ is a linear code with parameters $[n, k, d]$, where:

- $n = \deg(D)$ is the length of the code,
- $k = \dim_{\mathbb{F}_q} \mathcal{L}(G)$ is its dimension,
- d is its minimum Hamming distance.

By the Riemann–Roch theorem (see, e.g., [11, Theorem 2.2.2]), the parameters satisfy:

$$k \geq \deg(G) + 1 - g, \quad d \geq n - \deg(G).$$

Moreover, if $2g - 2 < \deg(G) < n$, then $k = \deg(G) + 1 - g$.

If $\{f_1, \dots, f_k\}$ is a basis of $\mathcal{L}(G)$, then a generator matrix of $\mathcal{C}_{\mathcal{L}}(D, G)$ is given by

$$\begin{pmatrix} f_1(P_1) & f_1(P_2) & \cdots & f_1(P_n) \\ f_2(P_1) & f_2(P_2) & \cdots & f_2(P_n) \\ \vdots & \vdots & \ddots & \vdots \\ f_k(P_1) & f_k(P_2) & \cdots & f_k(P_n) \end{pmatrix}.$$

For more details on function fields and algebraic geometry codes we refer to the book [11].

3 Structural attacks on the McEliece cryptosystem based on elliptic codes

In this work, we consider algebraic geometry codes associated with elliptic curves. An elliptic curve $\mathcal{E}$ over the finite field $\mathbb{F}_q$ is defined in affine coordinates by the smooth equation

$$\mathcal{E} : y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6,$$

where $a_i \in \mathbb{F}_q$. This equation is called the *full Weierstrass form*. The curve has genus $g = 1$, and the set of $\mathbb{F}_q$ -rational points is

$$\mathcal{E}(\mathbb{F}_q) = \{(\alpha, \beta) \in \mathbb{F}_q^2 : \beta^2 + a_1\alpha\beta + a_3\beta = \alpha^3 + a_2\alpha^2 + a_4\alpha + a_6\} \cup \{P_\infty\},$$

where P_∞ is the unique point at infinity. The number of rational points satisfies the Hasse bound

$$q + 1 - \lfloor 2\sqrt{q} \rfloor \leq |\mathcal{E}(\mathbb{F}_q)| \leq q + 1 + \lfloor 2\sqrt{q} \rfloor.$$

Since the genus of $\mathcal{E}$ equals 1, for $0 < k < n$ the Riemann–Roch theorem gives $\dim \mathcal{L}(G) = \deg(G) = k$. Hence, $\mathcal{C}_{\mathcal{L}}(D, G)$ has parameters $[n, k, d]$, where $d \geq n - k$.

Using the results of [12] (see Appendix A), we describe in Algorithm 1 the key generation procedure for the McEliece cryptosystem based on elliptic codes, which will be considered in Section 3.3.

Algorithm 1 Key generation for the McEliece cryptosystem based on elliptic codes

Require: Finite field $\mathbb{F}_q$; elliptic curve $\mathcal{E}/\mathbb{F}_q : y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6$; code parameters $n, k, t \in \mathbb{N}$, where t is the number of errors.

- 1: $D \leftarrow \sum_{i=1}^n P_i$, where $P_i \neq P_j$ for $i \neq j$ and $P_i \in \mathcal{E}(\mathbb{F}_q)$.
- 2: $G \leftarrow \sum_i k_i Q_i$, where $Q_i \notin \text{supp}(D)$ for all i and $k_i \geq 0$. $\triangleright \sum k_i = \deg(G) = k$
- 3: $\mathcal{L}_b(\mathcal{E}) \leftarrow \begin{cases} \{x^i y^j \mid 2i + 3j \leq k, j = \overline{0, 1}\}, & \text{if } G = kP_\infty, \\ \{1, f_{1,s} \mid 2 \leq s \leq k\}, & \text{if } G = kQ, \\ \{1, f_{i,s}, g_j \mid 1 \leq i \leq z, 1 \leq j \leq z-1, 2 \leq s \leq k_i\}, & \text{if } G = \sum_{i=1}^z k_i Q_i. \end{cases}$
- 4: $\mathbf{G}_{\text{gen}} \leftarrow \begin{pmatrix} \widehat{f}_1(P_1) & \widehat{f}_1(P_2) & \cdots & \widehat{f}_1(P_n) \\ \widehat{f}_2(P_1) & \widehat{f}_2(P_2) & \cdots & \widehat{f}_2(P_n) \\ \vdots & \vdots & \ddots & \vdots \\ \widehat{f}_k(P_1) & \widehat{f}_k(P_2) & \cdots & \widehat{f}_k(P_n) \end{pmatrix}$, where $\widehat{f}_i \in \mathcal{L}_b(\mathcal{E})$.
- 5: Transform the matrix $\mathbf{G}_{\text{gen}}$ into systematic form $(\mathbf{I}_k \mid \mathbf{G}')$.
- 6: **if** the transformation is impossible **then**
- 7: Return to Step 1. $\triangleright$ or apply a permutation that transforms the matrix into systematic form
- 8: $\mathbf{G}_{\text{pub}} \leftarrow \mathbf{G}'$.

Ensure: Public key: $(\mathcal{E}, \mathbf{G}_{\text{pub}}, t)$. Private key: (D, G) .

In the considered version of the cryptosystem, the elliptic curve is part of the public key. During the public key generation process, the generator matrix is transformed into systematic form, which is equivalent to left multiplication by an invertible matrix S . The random choice of points in the support of the divisor D is equivalent to right multiplication by a permutation matrix P . Therefore, the public key generation procedure is equivalent to that of the classical McEliece cryptosystem [2].

3.1 Computation of P-filtration for algebraic geometry code from [9]

As before, consider the elliptic code $\mathcal{C}_{\mathcal{L}}(D, G)$ associated with divisors $D, G \in \text{Div}(\mathcal{E})$ such that $\text{supp}(G) \cap \text{supp}(D) = \emptyset$. Let P be one of the points of the divisor D . Denote by D' the divisor obtained by removing the point P from D , i.e. $D' = D - P$, for which $|\text{supp}(D)| = (n - 1)$.

In the work [9] the authors demonstrated the possibility of constructing the following chains of codes using only the generator matrix of the original code $\mathcal{C}_{\mathcal{L}}(D, G)$:

$$\mathcal{V}_i = \mathcal{C}_{\mathcal{L}}(D', G + iP), \quad \mathcal{U}_i = \mathcal{C}_{\mathcal{L}}(D', iP), \quad \forall i \in \mathbb{Z}.$$

The following propositions explain how to compute the elements of the sequences $\mathcal{V}_i$ and $\mathcal{U}_i$ without knowledge of the structure of the AG-code, i.e. without knowledge of the triple $(\mathcal{E}/\mathbb{F}_q, D, G)$.

Proposition 3.1.1 ([9, Proposition 20]). *Let $i \geq 1$, then the elements of the sequence $\{\mathcal{V}_i\}$ can be computed as follows:*

$$\begin{aligned} \mathcal{V}_{-i-1} &= \left\{ \mathbf{z} \in \mathcal{V}_{-i} \mid \mathbf{z} \star \mathcal{V}_{-i+1} \subseteq \mathcal{V}_{-i}^{(2)} \right\}, \text{ where } \deg(G) - \frac{n-4}{2} \leq i \leq \deg(G) - 2g + 1, \\ \mathcal{V}_{i+1} &= \left\{ \mathbf{z} \in \mathbb{F}_q^{n-1} \mid \mathbf{z} \star \mathcal{V}_{i-1} \subseteq \mathcal{V}_i^{(2)} \right\}, \text{ where } 2g + 1 - \deg(G) \leq i \leq \frac{n-4}{2} - \deg(G). \end{aligned}$$

Proposition 3.1.2 ([9, Proposition 23]). *Let i, j, ℓ be such that $i + j = \ell$, $\deg(G) + \ell > 0$ and $\deg(G) + \ell < n - 4$. Then*

$$\begin{aligned} \mathcal{U}_i &= \{ \mathbf{z} \in \mathbb{F}_q^{n-1} \mid \mathbf{z} \star \mathcal{V}_j \subseteq \mathcal{V}_\ell \}, \quad \text{where } \deg(G) + j > 2g, \\ \mathcal{V}_i &= \{ \mathbf{z} \in \mathbb{F}_q^{n-1} \mid \mathbf{z} \star \mathcal{U}_j \subseteq \mathcal{V}_\ell \}, \quad \text{where } j > 2g, \end{aligned}$$

The computation of subsequent terms of the sequences is based on a recursive approach, where each new element is derived directly from the two preceding ones. The initial elements are defined as follows:

- $\mathcal{V}_0 = \mathcal{C}_{\mathcal{L}}(D', G)$ — the *punctured code*, obtained by deleting the component corresponding to the point P_1 , which is equivalent to deleting a column in the generator matrix;

- $\mathcal{V}_{-1} = \mathcal{C}_{\mathcal{L}}(D', G - P)$ — the *shortened code*, obtained by selecting codewords that are zero on the component P_1 , and subsequently deleting this component, which is equivalent to deleting a column and a row in the generator matrix.

Error correcting pairs and arrays.

According to [9], using the above process, we can construct the code $\mathcal{V}_{-t-g} = \mathcal{C}_{\mathcal{L}}(D', G - (t+g)P)$. This, in turn, allows to compute a pair $\mathcal{B} = \mathcal{V}_{-t-g}$ and $\mathcal{A} = (\mathcal{B} \star \mathcal{C}_{\mathcal{L}}(D, G)^{\perp})^{\perp}$ that corrects t errors for the dual code $\mathcal{C}_{\mathcal{L}}(D, G)^{\perp}$. Suppose a vector $\mathbf{y} = \mathbf{c} + \mathbf{e}$ is given, where $\mathbf{c} \in \mathcal{C}_{\mathcal{L}}(D, G)^{\perp}$ and $\mathbf{e}$ is an error vector of weight $\text{wt}(\mathbf{e}) \leq \lfloor \frac{d^* - g - 1}{2} \rfloor$, where d^* is the designed distance of the code. After recovering the pair of codes $(\mathcal{A}, \mathcal{B})$, the adversary can decode the vector $\mathbf{y}' = (y_2, \dots, y_n) = \mathbf{c}' + \mathbf{e}'$ in time $\mathcal{O}(mn^3)$, where $\mathbf{c}'$ and $\mathbf{e}'$ are obtained by shortening the original vectors with respect to the first coordinate. In the final step, it is necessary to recover the first coordinate of the codeword $\mathbf{c}$ by solving a system of linear equations.

Note that using an error-correcting pair allows the attack to be applied when $t \leq \lfloor \frac{d^* - g - 1}{2} \rfloor$. Moreover, in [9] a similar attack was also proposed, but using an error-correcting array. That approach enables the attack for larger values of t , since the array corrects $t \leq \lfloor \frac{d^* - 1}{2} \rfloor$ errors.

In [13], the authors propose an approach to construct power error-locating pairs for AG codes based on a generalized definition of error-locating pairs. This approach makes it possible to correct more errors than the unique decoding radius and, in particular, to construct a power error-locating pair that can be used without any additional knowledge of the underlying curve or divisor. Consequently, the corresponding attack can be applied to a McEliece cryptosystem based on an AG code even when the error weight exceeds half the minimum distance. However, as noted by the authors, this approach cannot be used to construct an effective attack when the error weight is close to the Johnson bound, since it requires a Guruswami–Sudan-like decoder with a decoding radius close to the Johnson radius.

3.2 Shkumatov's attack on ECC² scheme [1]

In [1], the first polynomial-time attack on the McEliece cryptosystem with known divisor $G = kP_{\infty}$ is presented. The attack works for an arbitrary number of errors t under the restriction $5 \leq \deg(G) \leq \frac{n}{2} - 1$. Algorithm 2, described in this work, recovers equivalent keys $\hat{\mathcal{E}}, \hat{D}$ such that $\mathcal{C}_{\mathcal{L}_{\hat{\mathcal{E}}}}(D, G) = \mathcal{C}_{\mathcal{L}_{\hat{\mathcal{E}}}}(\hat{D}, G)$, where $\hat{\mathcal{E}}$ is an elliptic curve isomorphic to $\mathcal{E}$, with complexity $\mathcal{O}((k^2 + q^2)n^2)$.

Algorithm 2 Recovery of $\hat{\mathcal{E}}$ and $\hat{D}$ from [1]

Require: generator matrix $\mathbf{G}_{pub}$ of the code $\mathcal{C}_{\mathcal{L}}(D, kP_{\infty})$.

Ensure: elliptic curve $\hat{\mathcal{E}}$ isomorphic to $\mathcal{E}$; divisor $\hat{D}$ obtained from D via the isomorphism $\mathcal{E} \rightarrow \hat{\mathcal{E}}$.

- 1: Compute $\mathcal{U}_2$ and $\mathcal{U}_3$ from $\mathcal{V}_{-1}$ and $\mathcal{V}_0$ using Propositions 3.1.1 and 3.1.2.
 - 2: Select an arbitrary codeword $\mathbf{v}' \in \mathcal{U}_2$ that does not correspond to a constant, and an arbitrary $\mathbf{v}'' \in \mathcal{U}_3$ that does not belong to the code $\mathcal{U}_2$.
 - 3: Find constants $c_1, c_2 \in \mathbb{F}_q$ such that the vector $\mathbf{v}''' = \left(\frac{v'_i + c_1}{v''_i + c_2} \right)_{i=1}^{n-1}$ belongs to the code $\mathcal{V}_0$.
 - 4: Compute the vector $\mathbf{u} = \mathbf{v}' \star \mathbf{v}''' \star \mathbf{v}'''$.
 - 5: Recover the divisor $\hat{D}' = \sum_{i=1}^{n-1} \hat{P}_i$, where $\hat{P}_i = (v'''_i, u_i)$ ▷ Without loss of generality, $D' = D - P_n$.
 - 6: Recover the equation of the isomorphic curve $\hat{\mathcal{E}}$ by substituting the points $\hat{P}_i$ into the curve equation and solving the resulting system of linear equations.
 - 7: Recover the missing point $\hat{P}_n$ by substituting $x' = 0$ into the curve equation.
-

We note that the correctness proof and the complexity analysis presented in the original work [1] contain several inaccuracies. Nevertheless, the overall idea of the algorithm remains correct. In the considered work, the functions corresponding to vectors $\mathbf{v}'$ and $\mathbf{v}''$ are represented as

$$\begin{aligned} \hat{g}_1 &= a \cdot g_1 + b, & \hat{g}_2 &= c \cdot g_2 + l \cdot g_1 + e, \\ g_1 &= \frac{y + a_0 + a_1(x - x_0)}{(x - x_0)^2}, & g_2 &= \frac{y + a_0 + a_1(x - x_0) + a_2(x - x_0)^2}{(x - x_0)^3}, \end{aligned}$$

where $a, b, c, l, e, a_i \in \mathbb{F}_q$, with $a \neq 0$ and $c \neq 0$. In Step 3 of Algorithm 2, one must find constants $c_1, c_2 \in \mathbb{F}_q$ such that the following equality holds:

$$\hat{g}_2 + c_2 = \frac{\hat{g}_1 + c_1}{c_3(x - x_0)}. \quad (1)$$

Expanding this equation yields

$$g_1 \cdot \left[c + l \cdot (x - x_0) - \frac{a}{c_3} \right] + 1 \cdot \left[c \cdot a_2 + (e + c_2) \cdot (x - x_0) - \frac{b + c_1}{c_3} \right] = 0.$$

The functions g_1 and 1 form a basis of the Riemann–Roch space corresponding to the code $\mathcal{U}_2$. By the linear independence of the basis elements, the coefficients of both basis vectors must vanish. We therefore obtain the following conditions.

Condition 1:

$$c + l \cdot (x - x_0) - \frac{a}{c_3} = 0.$$

Since this condition must hold for all x , it follows that $l = 0$. Furthermore,

$$c = \frac{a}{c_3} \implies c_3 = \frac{a}{c}.$$

Condition 2:

$$c \cdot a_2 + (e + c_2) \cdot (x - x_0) - \frac{b + c_1}{c_3} = 0.$$

Since this condition must hold for all x , we obtain

$$e + c_2 = 0 \implies c_2 = -e \quad \text{and} \quad c_1 = a_2 \cdot a - b.$$

In addition to the constraints arising from equation (1), there is an additional condition related to the curve isomorphism. The isomorphism $\phi(x, y)$ is defined by

$$\begin{aligned} x' &= c_3 \cdot (x - x_0), \\ y' &= a \cdot c_3^2 \cdot (y + y_0 + a_1(x - x_0)) + b \cdot c_3^2 \cdot (x - x_0)^2. \end{aligned}$$

For this mapping to be an isomorphism preserving the standard form of the curve equation, the quadratic term must vanish, which is achieved when $b = 0$. Thus, the resulting constraints $b = 0$ and $l = 0$ make it highly unlikely that relation (1) together with the isomorphism constraint holds for arbitrary vectors $\mathbf{v}'$ and $\mathbf{v}''$. Indeed, under these conditions, the vector $\mathbf{v}'$ corresponds to a function from the space $\langle g_1 \rangle$. The probability of randomly selecting such a vector is, on average, $1/q$. Similarly, the probability of selecting a vector $\mathbf{v}''$ corresponding to a function from the space $\langle g_2 + e \rangle$ is also $1/q$.

Therefore, in order for the vector in Step 3 to satisfy equation (1) together with the condition $l = 0$, it is additionally necessary to search for a vector $\mathbf{v}'' \in \langle \text{ev}_{D'}(g_2), \text{ev}_{D'}(1) \rangle$, which increases the complexity of the attack by a factor of $\mathcal{O}(q)$. Similarly, finding the vector $\mathbf{u}$ corresponding to the function $a \cdot c_3^2 \cdot (y + y_0 + a_1(x - x_0))$ requires an additional search over at most q candidates. Consequently, the final complexity of the attack becomes $\mathcal{O}(k^2 n^2 + q^3 n(n - k) + q)$.

3.3 New structural attack for arbitrary effective divisors G

Here we will present the first polynomial-time structural attack on the McEliece cryptosystem based on elliptic codes (key generation is described in Algorithm 1) that works for arbitrary effective divisors of elliptic codes and for any number of inserted errors t . In the remainder of this section, we assume that the elements of the divisor D are chosen uniformly at random from the set of all rational points of the elliptic curve $\mathcal{E}(\mathbb{F}_q)$. In the proofs, we will focus on the case $\text{char}(\mathbb{F}_q) > 3$, as this case currently presents the greatest practical interest for elliptic codes. In this case, the elliptic curve can be reduced to the short Weierstrass form

$$y^2 = x^3 + a_4 x + a_6, \quad \Delta = -16(4a_4^3 + 27a_6^2), \quad j = 1728 \frac{4a_4^3}{4a_4^3 + 27a_6^2}.$$

Proposition 3.3.1 ([14, Prop. III.3.4, Cor. III.3.5]). Let $\mathcal{E}/\mathbb{F}_q$ be an elliptic curve with neutral element P_∞ , and let $\text{Div}^0(\mathcal{E})$ denote the group of divisors of degree 0. The map

$$\sigma : \text{Div}^0(\mathcal{E}) \longrightarrow \mathcal{E}, \quad \sigma\left(\sum_P n_P P\right) = \bigoplus_P [n_P]P,$$

where $\oplus$ and $[n]\cdot$ denote the group operation and scalar multiplication on $\mathcal{E}$, respectively, is a surjective group homomorphism whose kernel is exactly the group of principal divisors. Equivalently, for $D = \sum_P n_P P \in \text{Div}(\mathcal{E})$,

$$D \sim 0 \iff \deg(D) = 0 \quad \text{and} \quad \bigoplus_P [n_P]P = P_\infty,$$

and σ induces an isomorphism $\text{Pic}^0(\mathcal{E}) \cong \mathcal{E}$.

Lemma 3.3.2. Let $P \in \mathcal{E}(\mathbb{F}_q) \setminus \{P_\infty\}$ and let $f_2^{(P)}$ be the non-constant basis function of $\mathcal{L}(2P)$ from Lemma A.1.1. Then $\text{div}(f_2^{(P)}) = [2]P + P_\infty - 2P$, and for every $R \in \mathcal{E}(\mathbb{F}_q) \setminus \{P\}$ and every $\tilde{f} = af_2^{(P)} + b$, $a \neq 0$,

$$\tilde{f}^{-1}(\tilde{f}(R)) \cap \mathcal{E}(\mathbb{F}_q) = \{R, \tau_P(R)\}, \quad \tau_P(R) = [2]P \ominus R.$$

Consequently, for distinct $P, P' \in \mathcal{E}(\mathbb{F}_q) \setminus \{P_\infty\}$,

$$\exists R : \tau_P(R) = \tau_{P'}(R) \iff \forall R : \tau_P(R) = \tau_{P'}(R) \iff [2]P = [2]P' \iff P \ominus P' \in \mathcal{E}[2].$$

Proof. Since $f_2^{(P)} \in \mathcal{L}(2P)$ is non-constant, its pole divisor is exactly $2P$, so $f_2^{(P)}$ is a morphism of degree 2. The explicit divisor follows because the numerator $y - \beta' - \gamma(x - \alpha)$ is the tangent to $\mathcal{E}$ at $-P$ (resp. the numerator is 1 when $P \in \mathcal{E}[2]$). For $\lambda \in \mathbb{F}_q$ the zero divisor of $f_2^{(P)} - \lambda$ has degree 2, say $A + B$, and $A + B - 2P$ is principal, whence $A \oplus B = [2]P$ by Proposition 3.3.1. The last equivalence is immediate from $\tau_P(R) \ominus \tau_{P'}(R) = [2]P \ominus [2]P'$. $\square$

Theorem 3.3.3. Let $\mathcal{C}_{\mathcal{L}}(D, G)$ be an elliptic code associated with the curve $\mathcal{E}/\mathbb{F}_q$. If $5 \leq \deg(G) \leq \frac{n}{2} - 1$, then, given the curve $\mathcal{E}$, the generator matrix $\mathbf{G}_{\text{pub}}$, and three distinct points $P_1, P_2, P_3 \in D$, the entire divisor D can be recovered in $\mathcal{O}(k^2 n^2 + |\mathcal{E}(\mathbb{F}_q)| + n)$ operations in $\mathbb{F}_q$ with probability $1 - \mathcal{O}(|\mathcal{E}(\mathbb{F}_q)|^{-2})$.

Proof. Without loss of generality, assume that the first three points P_1, P_2, P_3 of the divisor D are known. Since $5 \leq \deg(G) \leq \frac{n}{2} - 1$, using the generator matrix of the code and Propositions 3.1.1 and 3.1.2, we can compute the codes

$$\mathcal{V}_2^{(P_1)} = \mathcal{C}_{\mathcal{L}}(D - P_1, G + 2P_1), \quad \mathcal{U}_2^{(P_1)} = \mathcal{C}_{\mathcal{L}}(D - P_1, 2P_1). \quad (2)$$

Let $\mathcal{E}[2]$ denote the group of 2-torsion points. From Lemma A.1.1 (see Appendix A), one of the bases of the Riemann–Roch space associated with the one-point divisor $2P_1$, where $P_1 \notin \mathcal{E}[2]$, consists of the functions:

$$\mathcal{L}_b(2P_1) = \left\{ 1, f_2^{(P_1)} = \frac{y - \beta' - \gamma(x - \alpha)}{(x - \alpha)^2} \right\}, \quad \text{for } P_1 = (\alpha, \beta) \text{ and } -P_1 = (\alpha, \beta'), \quad (3)$$

where $\beta \neq 0$ and $\gamma = \begin{cases} \frac{\alpha^2 + a_4 + a_1 \beta'}{a_1 \alpha + a_3} & \text{if } \text{char}(\mathbb{F}_q) = 2, \\ \frac{3\alpha^2 + a_4}{2\beta'} & \text{if } \text{char}(\mathbb{F}_q) \geq 3. \end{cases}$

For brevity, we consider only the case $\text{char}(\mathbb{F}_q) \geq 3$, since in characteristic 2 all computations are analogous, except for the formulas for the torsion points $\mathcal{E}[2]$ and the involution $-P$.

Let $\text{char}(\mathbb{F}_q) \geq 3$. If $P_1 \in \mathcal{E}[2]$, i.e. $P_1 = (\alpha, 0)$, then

$$\mathcal{L}_b(2P_1) = \left\{ 1, f_2^{(P_1)} = \frac{1}{(x - \alpha)} \right\}.$$

Since the curve equation $\mathcal{E}$ and the point $P_1 = (\alpha, \beta)$ are known, the function $f_2^{(P_1)}$ can also be evaluated at any point of the curve.

Consider the basis of the code $\mathcal{U}_2^{(P_1)}$ obtained in (2). Using the two known points, we find a linear transformation that maps this basis to the basis corresponding to (3):

$$\mathbf{G}_{\mathcal{U}_2^{(P_1)}} = \begin{pmatrix} g_{1,2} & g_{1,3} & \cdots & g_{1,n} \\ g_{2,2} & g_{2,3} & \cdots & g_{2,n} \end{pmatrix} \longrightarrow \begin{pmatrix} 1 & 1 & \cdots & 1 \\ f_2^{(P_1)}(P_2) & f_2^{(P_1)}(P_3) & \cdots & f_2^{(P_1)}(P_n) \end{pmatrix}.$$

To do this, we need to solve a system of two linear equations for a particular i :

$$\begin{cases} af_2^{(P_1)}(P_2) + b = g_{i,2}, \\ af_2^{(P_1)}(P_3) + b = g_{i,3}, \end{cases} \quad \text{where } (g_{i,2}, \dots, g_{i,n}) \neq (c, \dots, c) \text{ for } c \in \mathbb{F}_q. \quad (4)$$

Clearly, at least one vector $(g_{i,2}, \dots, g_{i,n})$ of the obtained basis satisfies the requirement of the system. The solution of system (4) is unique if and only if the values of f_2 at the two known points do not coincide, i.e. $f_2^{(P_1)}(P_2) \neq f_2^{(P_1)}(P_3)$. We first describe the fibre sets of $f_2^{(P_1)}$, since they determine both this condition and the sets of candidate points obtained below.

1. Let $\beta \neq 0$ ($P_1 \notin \mathcal{E}[2]$). Fix $P_2 = (x_2, y_2)$, the points $P = (x, y)$ with $f_2(P) = f_2(P_2)$ are the solutions of

$$\frac{y_2 - \beta' - \frac{3\alpha^2 + a_4}{2\beta'}(x_2 - \alpha)}{(x_2 - \alpha)^2} = \frac{y - \beta' - \frac{3\alpha^2 + a_4}{2\beta'}(x - \alpha)}{(x - \alpha)^2}. \quad (5)$$

Introduce the notation

$$\gamma = \frac{3\alpha^2 + a_4}{2\beta'}, \quad \lambda = \frac{y_2 - \beta' - \gamma(x_2 - \alpha)}{(x_2 - \alpha)^2},$$

so that (5) becomes

$$\frac{y - \beta' - \gamma(x - \alpha)}{(x - \alpha)^2} = \lambda \implies y = \lambda(x - \alpha)^2 + \gamma(x - \alpha) + \beta'.$$

Substituting y into the curve equation yields

$$[\lambda(x - \alpha)^2 + \gamma(x - \alpha) + \beta']^2 = x^3 + a_4x + a_6.$$

Let $d = x - \alpha$. Then $y = \lambda d^2 + \gamma d + \beta'$, and the right-hand side can be written as

$$\begin{aligned} (\alpha + d)^3 + a_4(\alpha + d) + a_6 &= \alpha^3 + 3\alpha^2d + 3\alpha d^2 + d^3 + a_4\alpha + a_4d + a_6 \\ &= \beta'^2 + (3\alpha^2 + a_4)d + 3\alpha d^2 + d^3 \\ &= \beta'^2 + 2\beta'\gamma d + 3\alpha d^2 + d^3. \end{aligned}$$

The left-hand side is

$$(\lambda d^2 + \gamma d + \beta')^2 = \lambda^2 d^4 + 2\lambda\gamma d^3 + 2\lambda\beta' d^2 + \gamma^2 d^2 + 2\gamma\beta' d + \beta'^2.$$

Equating both sides, we get

$$\lambda^2 d^4 + 2\lambda\gamma d^3 + 2\lambda\beta' d^2 + \gamma^2 d^2 + 2\gamma\beta' d + \beta'^2 = d^3 + 3\alpha d^2 + 2\beta'\gamma d + \beta'^2,$$

$$d^2 [\lambda^2 d^2 + (2\lambda\gamma - 1)d + (2\lambda\beta' + \gamma^2 - 3\alpha)] = 0.$$

The factor d^2 was introduced when clearing the denominator $(x - \alpha)^2$ and is extraneous: it corresponds to $x = \alpha$, that is, to the points $\pm P_1$, and P_1 is a pole of f_2 . The solutions of (5) are therefore given by the roots of

$$\lambda^2 d^2 + (2\lambda\gamma - 1)d + (2\lambda\beta' + \gamma^2 - 3\alpha) = 0, \quad (6)$$

each root d determining exactly one point $P = (\alpha + d, \lambda d^2 + \gamma d + \beta')$, distinct roots giving distinct points. In particular, $d = 0$ is a root of (6) precisely when $2\lambda\beta' + \gamma^2 - 3\alpha = 0$, i.e. when $\lambda = f_2^{(P_1)}(-P_1)$, and the corresponding solution is the point $-P_1$, at which $f_2^{(P_1)}$ is regular. If $\lambda = 0$, equation (6) degenerates to a linear one with the

single root $d = \gamma^2 - 3\alpha$, corresponding to the point $[2]P_1$. The second point of this fibre set is P_∞ , which is a zero of $f_2^{(P_1)}$.

By construction P_2 is a solution, so (6) has a root in $\mathbb{F}_q$, hence both of its roots are rational. Consequently, for every $P \in \mathcal{E}(\mathbb{F}_q)$ the set

$$\left\{ \tilde{P} \in \mathcal{E}(\mathbb{F}_q) : f_2^{(P_1)}(\tilde{P}) = f_2^{(P_1)}(P) \right\}$$

consists of at most two rational points, and of exactly one point precisely when the discriminant of (6) vanishes (see Remark 1).

2. Let $\beta = 0$ ($P_1 \in \mathcal{E}[2]$). Then $f_2^{(P_1)} = \frac{1}{x-\alpha}$, and we have to solve

$$\frac{1}{x_2 - \alpha} = \frac{1}{x - \alpha}. \quad (7)$$

Clearly, the solutions are exactly the points $\pm P_2 = (x_2, \pm y_2)$, so again the fibre set consists of at most two rational points, and of one point exactly when $P_2 \in \mathcal{E}[2]$.

Assume now that $f_2^{(P_1)}(P_2) \neq f_2^{(P_1)}(P_3)$. Then system (4) has a unique solution (a, b) , and, replacing the chosen basis vector by $a^{-1}(\mathbf{g}_i - b \cdot (1, \dots, 1))$, we obtain the generator matrix

$$\mathbf{G}_{\mathcal{U}_2^{(P_1)}} = \begin{pmatrix} 1 & 1 & \dots & 1 \\ f_2^{(P_1)}(P_2) & f_2^{(P_1)}(P_3) & \dots & f_2^{(P_1)}(P_n) \end{pmatrix}. \quad (8)$$

Moreover, as already shown, for each position in the divisor D we have at most two possible points. That is, we obtain

$$\tilde{D}^{(P_1)} = \left\{ \{P_2, \tilde{P}_2^{(P_1)}\}, \dots, \{P_n, \tilde{P}_n^{(P_1)}\} \right\}.$$

Next, to reduce the number of candidates and recover the original divisor D , we perform the same procedure for the codes

$$\mathcal{U}_2^{(P_2)} = \mathcal{C}_{\mathcal{L}}(D - P_2, 2P_2) \quad \text{or} \quad \mathcal{U}_2^{(P_3)} = \mathcal{C}_{\mathcal{L}}(D - P_3, 2P_3)$$

and compute the corresponding sets

$$\tilde{D}^{(P_2)} = \left\{ \{P_1, \tilde{P}_1^{(P_2)}\}, \{P_3, \tilde{P}_3^{(P_2)}\}, \dots, \{P_n, \tilde{P}_n^{(P_2)}\} \right\},$$

$$\tilde{D}^{(P_3)} = \left\{ \{P_1, \tilde{P}_1^{(P_3)}\}, \{P_2, \tilde{P}_2^{(P_3)}\}, \{P_4, \tilde{P}_4^{(P_3)}\}, \dots, \{P_n, \tilde{P}_n^{(P_3)}\} \right\}.$$

By Lemma 3.3.2, the candidate set attached to the i -th position of $\tilde{D}^{(P_1)}$, $\tilde{D}^{(P_2)}$, $\tilde{D}^{(P_3)}$ is $\{P_i, \tau_{P_1}(P_i)\}$, $\{P_i, \tau_{P_2}(P_i)\}$, $\{P_i, \tau_{P_3}(P_i)\}$ respectively. If $[2]P_j \neq [2]P_l$ for some $j \neq l$, then $\tau_{P_j}(P_i) \neq \tau_{P_l}(P_i)$ for every i , and therefore

$$\{P_i, \tau_{P_j}(P_i)\} \cap \{P_i, \tau_{P_l}(P_i)\} = \{P_i\}, \quad i \geq 4.$$

The condition $[2]P_j \neq [2]P_l$ is equivalent to $P_j \ominus P_l \notin \mathcal{E}[2]$ and is verified directly from the known points P_1, P_2, P_3 . It fails for all three pairs only if P_1, P_2, P_3 lie in one coset of $\mathcal{E}[2]$, in which case $\tilde{D}^{(P_1)} = \tilde{D}^{(P_2)} = \tilde{D}^{(P_3)}$ and the three “hints” carry no separating information.

Thus, to recover the original divisor D , we compute the intersection of the corresponding elements in the sets $\tilde{D}^{(P_1)}$, $\tilde{D}^{(P_2)}$ and $\tilde{D}^{(P_3)}$:

$$D = P_1 + P_2 + P_3 + \sum_{i=4}^n \{P_i, \tilde{P}_i^{(P_1)}\} \cap \{P_i, \tilde{P}_i^{(P_2)}\} \cap \{P_i, \tilde{P}_i^{(P_3)}\}.$$

It remains to estimate the probability that the candidate sets $\tilde{D}^{(P_j)}$ can be constructed and that a pair of candidate sets provides the required separation. Let

$$N = |\mathcal{E}(\mathbb{F}_q)|, \quad N_2 = |\mathcal{E}[2](\mathbb{F}_q)|, \quad N_3 = |\mathcal{E}[3](\mathbb{F}_q)|.$$

Since D is chosen uniformly among all sets of n distinct points of $\mathcal{E}(\mathbb{F}_q)$, the triple (P_1, P_2, P_3) is a uniformly distributed ordered triple of pairwise distinct rational points.

For $j, l, m \in \{1, 2, 3\}$, where j, l, m are distinct, define the events

$$\mathcal{A}_j : f_2^{(P_j)}(P_l) \neq f_2^{(P_j)}(P_m), \quad \mathcal{B}_{jl} : [2]P_j \neq [2]P_l.$$

The event $\mathcal{A}_j$ is precisely the condition that the linear system (4) corresponding to the code $\mathcal{U}_2^{(P_j)}$ has a unique solution. Equivalently, the normalized basis (8) can be recovered using the two remaining known points. By Lemma 3.3.2, $\mathcal{A}_j$ fails exactly when

$$P_m = \tau_{P_j}(P_l), \quad \text{equivalently,} \quad P_l \oplus P_m = [2]P_j.$$

Writing $S = P_1 \oplus P_2 \oplus P_3$, this condition can also be expressed as $S = [3]P_j$.

On the other hand, Lemma 3.3.2 shows that $\mathcal{B}_{jl}$ is equivalent to

$$\tau_{P_j}(P) \neq \tau_{P_l}(P) \quad \text{for every } P \in \mathcal{E}(\mathbb{F}_q).$$

Thus, if $\mathcal{B}_{jl}$ holds, the candidate sets obtained from P_j and P_l distinguish the two possible points at every position simultaneously. Consequently, the divisor D is recoverable whenever

$$\exists j \neq l : \quad \mathcal{A}_j \wedge \mathcal{A}_l \wedge \mathcal{B}_{jl}. \quad (9)$$

We next bound the probability that condition (9) fails. We distinguish three cases according to the number of events $\mathcal{A}_j$ that fail.

- *No event $\mathcal{A}_j$ fails.* In this case all three candidate sets can be constructed. Hence, (9) fails only if no pair of the known points separates the corresponding involutions, i.e., $[2]P_1 = [2]P_2 = [2]P_3$. For a fixed P_1 , both P_2 and P_3 must then belong to the coset $P_1 \oplus \mathcal{E}[2]$ and must be distinct from P_1 and from each other. Therefore, there are at most $(N_2 - 1)(N_2 - 2)$ possible ordered pairs (P_2, P_3) . The corresponding probability is at most

$$\frac{(N_2 - 1)(N_2 - 2)}{(N - 1)(N - 2)}.$$

- *Exactly one event $\mathcal{A}_m$ fails.* In this case the only potentially useful pair is (j, l) , where $j, l, m \in \{1, 2, 3\}$. Thus, condition (9) fails precisely when $[2]P_j = [2]P_l$. For a fixed P_j , there are at most $N_2 - 1$ choices for P_l satisfying this equality and $P_l \neq P_j$.

Once P_j and P_l are fixed, the failure of $\mathcal{A}_m$ requires $[2]P_m = P_j \oplus P_l$, which leaves at most N_2 choices for P_m . Summing over the three possible choices of m , we obtain the upper bound

$$\frac{3N_2(N_2 - 1)}{(N - 1)(N - 2)}.$$

- *At least two events $\mathcal{A}_j, \mathcal{A}_l$ fail.* In this case

$$[3]P_j = S = [3]P_l \quad \text{and} \quad P_j \oplus P_l \in \mathcal{E}[3] \setminus \{P_\infty\}.$$

For a fixed P_j , there are therefore at most $N_3 - 1$ possible choices of P_l . Once P_j and P_l are fixed, the failure of $\mathcal{A}_j$ determines $P_m = \tau_{P_j}(P_l)$ uniquely. Summing over the three possible pairs (j, l) gives the upper bound

$$\frac{3(N_3 - 1)}{(N - 1)(N - 2)}.$$

Combining the three bounds yields

$$\Pr[D \text{ is not recovered}] \leq \frac{(N_2 - 1)(N_2 - 2) + 3N_2(N_2 - 1) + 3(N_3 - 1)}{(N - 1)(N - 2)} \leq 1 - \frac{66}{(N - 1)(N - 2)},$$

where the last inequality uses $N_2 \leq 4$ and $N_3 \leq 9$. In particular, the divisor D is recovered with probability $1 - \mathcal{O}(|\mathcal{E}(\mathbb{F}_q)|^{-2})$.

Computing the codes $\mathcal{U}_2^{(P_1)}, \mathcal{U}_2^{(P_2)}, \mathcal{U}_2^{(P_3)}$ requires constructing generator matrices, which involves computing the Schur product of two codes and then reducing to systematic form via Gaussian elimination. Computing the Schur product gives an overall complexity of $\mathcal{O}(k^2 n^2)$ for this step. Solving system (4), which is a system of two linear equations, takes $\mathcal{O}(1)$ operations. To build the candidate sets $\tilde{D}$, we need to enumerate all points of the curve and evaluate the function f_2 at each of them. The obtained values can be stored in a hash table with keys corresponding to the function values. Evaluating f_2 requires $\mathcal{O}(1)$ operations, so building the hash table costs $\mathcal{O}(|\mathcal{E}(\mathbb{F}_q)|)$. Searching for all n values in the hash table takes $\mathcal{O}(n)$ operations. Thus, the overall complexity of this step is $\mathcal{O}(|\mathcal{E}(\mathbb{F}_q)| + n) = \mathcal{O}(|\mathcal{E}(\mathbb{F}_q)|)$. Recovering the original set D reduces to pairwise intersection of elements from at least two sets. Since the size of each element in $\tilde{D}^{(P_1)}, \tilde{D}^{(P_2)}$, and $\tilde{D}^{(P_3)}$ does not exceed 2, the intersection operation takes $\mathcal{O}(1)$ time per position, and for n positions the total complexity is $\mathcal{O}(n)$. Therefore, the overall complexity of the algorithm is $\mathcal{O}(k^2 n^2 + |\mathcal{E}(\mathbb{F}_q)| + n)$. Algorithm 3 implements the procedure described above. $\square$

Remark 1. In the proof of Theorem 3.3.3 the candidate set

$$\tilde{D}_i = \left\{ P \in \mathcal{E}(\mathbb{F}_q) : \tilde{f}_2(P) = g_i \right\} = \{P_i, \tau_{P_1}(P_i)\}$$

is the fibre of the degree-2 map $\tilde{f}_2$ over the value g_i , and in general it consists of two points. It degenerates to a single point exactly at the ramification points of $\tilde{f}_2$, that is, by Lemma 3.3.2,

$$|\tilde{D}_i| = 1 \iff \tau_{P_1}(P_i) = P_i \iff [2]P_i = [2]P_1 \iff P_i \in P_1 \oplus \mathcal{E}[2](\mathbb{F}_q).$$

Consequently, the set of positions at which a single candidate occurs is $\{i \neq 1 : P_i \in P_1 \oplus \mathcal{E}[2](\mathbb{F}_q)\}$, and its cardinality does not exceed $N_2 - 1 \leq 3$, where $N_2 = |\mathcal{E}[2](\mathbb{F}_q)| \in \{1, 2, 4\}$. In particular, for a uniformly random divisor D the expected number of such positions equals $(n-1)(N_2-1)/(N-1)$, and if $\mathcal{E}(\mathbb{F}_q)$ contains no rational 2-torsion points, i.e. $N_2 = 1$, no such situations occurs at all.

Algorithm 3 Recovering the divisor D

Require: Elliptic curve $\mathcal{E}/\mathbb{F}_q$ ($\text{char}(\mathbb{F}_q) > 3$); generator matrix $\mathbf{G}_{\text{pub}}$ of the code $\mathcal{C}_{\mathcal{L}}(D, G)$, where $5 \leq \deg(G) \leq \frac{n}{2} - 1$; points $P_1, P_2, P_3 \in D$.

Ensure: Divisor D of the code $\mathcal{C}_{\mathcal{L}}(D, G)$.

- 1: **if** $S = [3]P_j$ for some $j \in \{1, 2, 3\}$, or $[2]P_1 = [2]P_2 = [2]P_3$, where $S = P_1 \oplus P_2 \oplus P_3$ **then**
- 2: **return** $\perp$ $\triangleright$ Probability $1 - \mathcal{O}(|\mathcal{E}(\mathbb{F}_q)|^{-2})$
- 3: Compute the generator matrix of $\mathcal{U}_2^{(P_1)} = \mathcal{C}_{\mathcal{L}}(\mathcal{E}, D - P_1, 2P_1)$ using $\mathbf{G}_{\text{pub}}$, Propositions 3.1.1 and 3.1.2. Set $P_1 = (\alpha, \beta)$.
- 4: **if** $\beta \neq 0$ **then** $\triangleright P_1 \notin \mathcal{E}[2]$
- 5: $\gamma \leftarrow \frac{3\alpha^2 + a_4}{2\beta}; f_2^{(P_1)} \leftarrow \frac{y + \beta + \gamma(x - \alpha)}{(x - \alpha)^2}.$
- 6: **else**
- 7: $f_2^{(P_1)} \leftarrow \frac{1}{x - \alpha}.$ $\triangleright P_1 \in \mathcal{E}[2]$
- 8: Find a vector $\mathbf{g} = (g_1, \dots, g_{n-1}) \in \mathcal{U}_2^{(P_1)} \setminus \langle \tilde{\mathbf{I}} \rangle.$
- 9: Solve the system for a, b :

$$\begin{cases} a f_2^{(P_1)}(P_2) + b = g_1, \\ a f_2^{(P_1)}(P_3) + b = g_2. \end{cases}$$

- 10: Compute $\tilde{f}_2^{(P_1)} \leftarrow a f_2^{(P_1)} + b.$
 - 11: For each $i \in \{4, \dots, n\}$, find a pair $\{P, \tilde{P}^{(P_1)}\} \subset \mathcal{E}(\mathbb{F}_q)$ such that $\tilde{f}_2^{(P_1)}(P) = \tilde{f}_2^{(P_1)}(\tilde{P}^{(P_1)}) = g_i$. Let $\tilde{D}^{(P_1)} = \{\{P_4, \tilde{P}_4^{(P_1)}\}, \dots, \{P_n, \tilde{P}_n^{(P_1)}\}\}.$
 - 12: Repeat steps 1–11, computing $\mathcal{U}_2^{(P_2)} = \mathcal{C}_{\mathcal{L}}(D - P_2, 2P_2)$ and $\mathcal{U}_2^{(P_3)} = \mathcal{C}_{\mathcal{L}}(\mathcal{E}, D - P_3, 2P_3)$ to obtain $\tilde{D}^{(P_2)}$ and $\tilde{D}^{(P_3)}.$
 - 13: Compute $\{P_i, \tilde{P}_i^{(P_1)}\} \cap \{P_i, \tilde{P}_i^{(P_2)}\} \cap \{P_i, \tilde{P}_i^{(P_3)}\}$ to recover the points $P_i \in D$ for $i \in \{4, \dots, n\}.$
 - 14: **return** $D = \sum_{i=1}^n P_i.$
-

Theorem 3.3.4. Let $\mathcal{C} = \mathcal{C}_{\mathcal{L}}(D, G)$ be an elliptic code associated with an elliptic curve $\mathcal{E}/\mathbb{F}_q$, $n = |\text{supp}(D)|$, and $G = \sum_i k_i Q_i$ with $k_i > 0$. Then, given the elliptic curve $\mathcal{E}$, the generator matrix $\mathbf{G}_{\text{pub}}$, and the divisor D , the second divisor G can be recovered in $\mathcal{O}\left((|\mathcal{E}(\mathbb{F}_q)| - n)^2 \cdot n^2 + \deg(G) \cdot (n + n^2)\right)$ operations in $\mathbb{F}_q$.

Proof. According to Theorem A.1.2, if points $Q_i = (\alpha_i, \beta_i)$, $Q_j = (\alpha_j, \beta_j)$ belong to the support of G , then one of the functions

$$g_{i,j}(x, y) = \frac{y + B_{i,j}(x)}{(x - \alpha_i)(x - \alpha_j)}, \quad g'_{i,j}(x, y) = \frac{1}{x - \alpha_i},$$

where $B_{i,j}(X) = \left(\frac{\beta_j - \beta_i}{\alpha_j - \alpha_i} + a_1\right)X + \left((\beta_i + a_1\alpha_i + a_3) - \left(\frac{\beta_j - \beta_i}{\alpha_j - \alpha_i} + a_1\right)\alpha_i\right)$, belongs to the Riemann–Roch space $\mathcal{L}(G)$. Thus, to recover the support of G , it is necessary to enumerate all possible functions $g_{i,j}$ and $g'_{i,j}$ (i.e., enumerate all possible pairs of curve points except those in the support of D), compute the vectors $\text{ev}_D(g_{i,j})$, $\text{ev}_D(g'_{i,j})$, and test whether they belong to the code $\mathcal{C}$. This requires $\mathcal{O}((|\mathcal{E}(\mathbb{F}_q)| - n)^2 \cdot n^2)$ operations over $\mathbb{F}_q$, since the points in $\text{supp}(D)$ are known by the hypothesis of the theorem. Clearly, $\text{ev}_D(g_{i,j}) \in \mathcal{C}$ if and only if $\{Q_i, Q_j\} \subseteq \text{supp}(G)$, which shows the uniqueness of recovering the support of G .

Recovering the multiplicities of the points of G is possible via Lemma A.1.1. Let $Q \in \text{supp}(G)$ and let the function $f_{k_i} = \frac{y + A_{k_i}(x)}{(x - \alpha_i)^{k_i}}$ be associated with the point Q . Similarly to the previous case, from the definitions of the Riemann–Roch space and the algebraic geometry code, we have:

$$\text{ev}_D(f_{k_i}) \in \mathcal{C} \iff G - k_i Q \geq 0.$$

Thus, by evaluating the functions f_{k_i} at the points of D and iteratively increasing the value of k_i , we can recover the corresponding multiplicities for all $P_i \in \text{supp}(G)$. If $\text{ev}_D(f_{k_i}) \notin \mathcal{C}$, then the divisor $G - k_i Q$ ceases to be effective, contradicting the hypothesis. The complexity of recovering the multiplicities (including the computation of $\text{ev}_D(f_{k_i})$ and the membership test of the vector in the code) is $\mathcal{O}(\deg(G) \cdot (n + n^2))$. Note that the construction of the functions f_{k_i} has complexity $\mathcal{O}(1)$ for any i , since these functions are built iteratively. Algorithm 4 implements the procedure described above. $\square$

Algorithm 4 Recovering the divisor G

Require: Elliptic curve $\mathcal{E}/\mathbb{F}_q$; generator matrix $\mathbf{G}_{\text{pub}}$ of the code $\mathcal{C}_{\mathcal{L}}(D, G)$, where $5 \leq \deg(G) \leq \frac{n}{2} - 1$; divisor D such that $n = |\text{supp}(D)| > 12$.

Ensure: Divisor G of the code $\mathcal{C}_{\mathcal{L}}(D, G)$ such that $|\text{supp}(G)| \geq 2$.

```

1:  $G_{\text{supp}} \leftarrow \{\emptyset\}$ .
2: for  $Q_i = (\alpha_i, \beta_i) \in \mathcal{E}(\mathbb{F}_q) \setminus \text{supp}(D)$  do
3:   for  $Q_j = (\alpha_j, \beta_j) \in \mathcal{E}(\mathbb{F}_q) \setminus \{\text{supp}(D) \cup \{Q_i\}\}$  do
4:     if  $Q_i \notin G_{\text{supp}}$  or  $Q_j \notin G_{\text{supp}}$  then
5:       Compute  $g_{i,j}(x, y) = \frac{y + B_{i,j}(x)}{(x - \alpha_i)(x - \alpha_j)}$  and  $g'_{i,j}(x, y) = \frac{1}{x - \alpha_i}$ , where  $B_{i,j}(X) = \left(\frac{\beta_j - \beta_i}{\alpha_j - \alpha_i} + a_1\right)X + \left((\beta_i + a_1\alpha_i + a_3) - \left(\frac{\beta_j - \beta_i}{\alpha_j - \alpha_i} + a_1\right)\alpha_i\right)$ .
6:       if  $\text{ev}_D(g) \in \mathcal{C}$  or  $\text{ev}_D(g') \in \mathcal{C}$  then
7:          $G_{\text{supp}} \leftarrow G_{\text{supp}} \cup \{Q_i, Q_j\}$ .
8:  $G \leftarrow 0$ .
9: for  $Q_i = (\alpha_i, \beta_i) \in G_{\text{supp}}$  do
10:  if  $\text{ev}_D\left(\frac{y + \beta_i + \frac{3\alpha_i^2 + a_4}{2\beta_i}(x - \alpha_i)}{(x - \alpha_i)^2}\right) \notin \mathcal{C}$  then
11:     $k_i \leftarrow 1$ .
12:  else
13:     $k_i \leftarrow 2$ .
14:    while  $\text{ev}_D(f_{k_i+1}) \in \mathcal{C}$  do
15:       $k_i \leftarrow k_i + 1$ .
16:   $G \leftarrow G + k_i Q_i$ .
17: return  $G$ .
```

$$\triangleright f_{k_i+1} \leftarrow \frac{Y + A_{k_i+1}(x)}{(x - \alpha_i)^{k_i+1}} \text{ (see Algorithm 6)}$$

Remark 2. The algorithms can be straightforwardly adapted to the case of an unknown elliptic curve by recovering an isomorphic representation, similarly to [1]. We also note that although Theorem 3.3.3 considers the case where three points are known, the attack remains polynomial even without knowledge of these points. Indeed, even without exploiting the isomorphism and equivalent keys, the worst-case complexity increases by a factor of $\mathcal{O}(|\mathcal{E}(\mathbb{F}_q)|^3)$ and is equal to $\mathcal{O}(|\mathcal{E}(\mathbb{F}_q)|^3 (n^3 + k^2 n^2 + |\mathcal{E}(\mathbb{F}_q)| + n))$, and is therefore only approximately a factor of q higher than the complexity of the attack from [1].

Remark 3. We note that Theorem 3.3.3 proves the possibility of recovering the parameters under the restriction $5 \leq \deg(G) \leq \frac{n}{2} - 1$. However, the attack can be straightforwardly extended to the case $\deg(G) \geq \frac{n}{2} + 1$ by dualizing and using the techniques from [9], as well as to the case $\deg(G) = \frac{n}{2}$ by considering the punctured code.

3.4 Optimized attack without “hints”

Theorem 3.3.3 assumes that three points of the divisor D are given. We now show that this assumption can be removed at the cost of a single enumeration over pairs of field elements, and that the resulting attack still returns a divisor lying on the given public curve $\mathcal{E}$, in contrast with [1], where an isomorphic curve $\hat{\mathcal{E}}$ is recovered. The key observation is that the group of translations of $\mathcal{E}$ acts on the set of secret keys without changing the code.

Lemma 3.4.1. *Let σ be an automorphism of the curve $\mathcal{E}$ defined over $\mathbb{F}_q$. Then for any divisors $D = P_1 + \dots + P_n$ and $G \geq 0$ with $\text{supp}(D) \cap \text{supp}(G) = \emptyset$, it holds that*

$$\mathcal{C}_{\mathcal{L}}(D, G) = \mathcal{C}_{\mathcal{L}}(\sigma(D), \sigma(G)), \quad \text{where } \sigma\left(\sum_P n_P P\right) = \sum_P n_P \sigma(P).$$

In particular, $(\sigma(D), \sigma(G))$ is an equivalent key for the code $\mathcal{C}_{\mathcal{L}}(D, G)$ on the same curve $\mathcal{E}$.

Proof. The map σ is an isomorphism of $\mathcal{E}$ onto itself defined over $\mathbb{F}_q$, hence $v_P(f \circ \sigma) = v_{\sigma(P)}(f)$ for every $f \in \mathbb{F}_q(\mathcal{E})$ and every point P , so that $\text{div}(f \circ \sigma) = \sigma^{-1}(\text{div}(f))$. Since σ^{-1} preserves effectiveness, for any divisor A we obtain $f \in \mathcal{L}(A) \iff f \circ \sigma \in \mathcal{L}(\sigma^{-1}(A))$, and taking $A = \sigma(G)$ shows that the composition map $f \mapsto f \circ \sigma$ is an $\mathbb{F}_q$ -linear bijection from $\mathcal{L}(\sigma(G))$ onto $\mathcal{L}(G)$, whose inverse is $f \mapsto f \circ \sigma^{-1}$. Moreover, for $f \in \mathcal{L}(G)$,

$$(f \circ \sigma^{-1})(\sigma(P_i)) = f(P_i), \quad i = 1, \dots, n,$$

so the images of the two evaluation maps coincide coordinatewise. Finally, $\sigma(G)$ is effective of degree $\deg(G)$ and its support is disjoint from that of $\sigma(D)$. $\square$

Remark 4. Lemma 3.4.1 applies to the translations $t_R : P \mapsto P \oplus R$ and to the involutions $\bar{t}_S : P \mapsto S \ominus P$, $R, S \in \mathcal{E}(\mathbb{F}_q)$, since $\bar{t}_S = t_S \circ [-1]$ and both t_R and $[-1]$ are automorphisms of $\mathcal{E}$ over $\mathbb{F}_q$.

Proposition 3.4.2. *Let $\mathcal{C} = \mathcal{C}_{\mathcal{L}}(D, G)$ with $5 \leq \deg(G) \leq \frac{n}{2} - 1$, and suppose the divisor D is known. Then for every $Q \in \mathcal{E}(\mathbb{F}_q) \setminus \text{supp}(D)$,*

$$\text{ev}_D(f_2^{(Q)}) \in \mathcal{C}^{(2)} \iff Q \in \text{supp}(G),$$

where $f_2^{(Q)}$ is the non-constant basis element of $\mathcal{L}(2Q)$. Consequently $\text{supp}(G)$ can be recovered in $\mathcal{O}(k^2 n^2 + (|\mathcal{E}(\mathbb{F}_q)| - n)n^2)$ operations in $\mathbb{F}_q$.

Proof. Since $\deg(G) \geq 2g + 1$ and $\deg(2G) = 2\deg(G) \leq n - 2 < n$, one has $\mathcal{C}^{(2)} = \mathcal{C}_{\mathcal{L}}(D, 2G)$, and the evaluation map is injective on $\mathcal{L}(2G)$. The pole divisor of $f_2^{(Q)}$ is exactly $2Q$, whence

$$\text{ev}_D(f_2^{(Q)}) \in \mathcal{C}_{\mathcal{L}}(D, 2G) \iff f_2^{(Q)} \in \mathcal{L}(2G) \iff 2G \geq 2Q \iff v_P(G) \geq 1.$$

The Schur square costs $\mathcal{O}(k^2 n^2)$ and each membership test $\mathcal{O}(n^2)$. $\square$

Theorem 3.4.3. *Let $\mathcal{C} = \mathcal{C}_{\mathcal{L}}(D, G)$ be an elliptic code associated with the curve $\mathcal{E}/\mathbb{F}_q$, $\text{char}(\mathbb{F}_q) > 3$, with $5 \leq \deg(G) \leq \frac{n}{2} - 1$. Then, given only the curve $\mathcal{E}$, automorphism $\sigma \in \text{Aut}(\mathcal{E}/\mathbb{F}_q)$ and the generator matrix $\mathbf{G}_{\text{pub}}$, one can recover a divisor $\hat{D}$ on $\mathcal{E}$ and an effective divisor $\hat{G}$ satisfying*

$$\mathcal{C}_{\mathcal{L}}(\hat{D}, \hat{G}) = \mathcal{C}, \quad \hat{D} = \sigma(D), \quad \hat{G} = \sigma(G),$$

in $\mathcal{O}(k^2 n^2 + nq^2 + (|\mathcal{E}(\mathbb{F}_q)| - n)n^2)$ operations in $\mathbb{F}_q$.

Proof. Choose an arbitrary point $R_0 = (\alpha, \beta) \in \mathcal{E}(\mathbb{F}_q) \setminus \{P_{\infty}\}$ and set $R = R_0 \ominus P_1$. By Lemma 3.4.1 applied to $\sigma = t_R$, the pair $(t_R(D), t_R(G))$ generates the same code $\mathcal{C}$, while the first point of $t_R(D)$ is $t_R(P_1) = R_0$. Thus,

after replacing (D, G) by $(t_R(D), t_R(G))$, we may assume that

$$P_1 = R_0,$$

that is, the first point of the divisor is known. The adversary does not need to know R : the matrix $\mathbf{G}_{\text{pub}}$ is also a generator matrix of the translated key, and any divisor recovered under this normalization gives an equivalent key by Lemma 3.4.1. Since $5 \leq \deg(G) \leq \frac{n}{2} - 1$, the codes

$$\mathcal{U}_2^{(P_1)} = \mathcal{C}_{\mathcal{L}}(D - P_1, 2P_1), \quad \mathcal{U}_2^{(P_j)} = \mathcal{C}_{\mathcal{L}}(D - P_j, 2P_j), \quad j = 2, \dots, 5,$$

can be computed from $\mathbf{G}_{\text{pub}}$ by Propositions 3.1.1 and 3.1.2, exactly as in the proof of Theorem 3.3.3. Let $\{\mathbf{1}, \mathbf{g}\}$, $\mathbf{g} = (g_2, \dots, g_n)$, be a basis of $\mathcal{U}_2^{(P_1)}$ with $\mathbf{g} \notin \langle \mathbf{1} \rangle$, and let $f_2 = f_2^{(R_0)}$ be the function defined in (3), which is explicitly known since R_0 and the equation of $\mathcal{E}$ are known. Because $\{1, f_2\}$ is a basis of $\mathcal{L}(2R_0)$ and the evaluation map is injective on $\mathcal{L}(2R_0)$ for $n - 1 > 2$, there is a unique pair $(a^*, b^*) \in \mathbb{F}_q^* \times \mathbb{F}_q$ such that

$$a^* g_i + b^* = f_2(P_i), \quad i = 2, \dots, n. \quad (10)$$

We find this pair by exhaustive search over $\mathbb{F}_q^* \times \mathbb{F}_q$. To test a candidate (a, b) , consider the value set

$$V = f_2(\mathcal{E}(\mathbb{F}_q) \setminus \{R_0\}) \subseteq \mathbb{F}_q.$$

By Lemma 3.3.2, every fibre of f_2 over an element of V contains either one or two rational points, and by Remark 1 the cardinality of fibre equals 1 exactly for the $N_2 - 1$ points of $(R_0 \oplus \mathcal{E}[2](\mathbb{F}_q)) \setminus \{R_0\}$, where $N_2 = |\mathcal{E}[2](\mathbb{F}_q)|$. Hence

$$|V| = \frac{|\mathcal{E}(\mathbb{F}_q)| - N_2}{2} + (N_2 - 1) = \frac{|\mathcal{E}(\mathbb{F}_q)| + N_2}{2} - 1. \quad (11)$$

The set V together with the corresponding fibres is stored in a hash table T with keys $f_2(P)$ and values P . By (10), the correct pair satisfies $a^* g_i + b^* \in V$ for every $i = 2, \dots, n$. We therefore enumerate all pairs $(a, b) \in \mathbb{F}_q^* \times \mathbb{F}_q$ and test the conditions $ag_i + b \in V$ successively. Let $i_0 \neq i_1$ be two positions with $g_{i_0} \neq g_{i_1}$, which exist because $\mathbf{g} \notin \langle \mathbf{1} \rangle$, and let $i_2, i_3, \dots$ be an arbitrary enumeration of the remaining positions. Denote by $t(a, b) \leq n - 1$ the number of tests performed for (a, b) and by

$$M_m = \#\{(a, b) \in \mathbb{F}_q^* \times \mathbb{F}_q : ag_{i_r} + b \in V, r = 0, \dots, m - 1\}$$

the number of pairs passing the first m tests. Since the map $(a, b) \mapsto (ag_{i_0} + b, ag_{i_1} + b)$ is injective, we have

$$M_1 \leq |V|q, \quad M_m \leq M_2 \leq |V|^2 \quad (\text{for } m \geq 2). \quad (12)$$

Consequently, from (11) and the Hasse bound the total cost of the enumeration is

$$\sum_{(a,b) \in \mathbb{F}_q^* \times \mathbb{F}_q} (1 + t(a, b)) = q(q - 1) + \sum_{m \geq 1} M_m \leq q^2 + |V|q + (n - 2)|V|^2 = \mathcal{O}(nq^2),$$

In particular, the number M of pairs passing all $n - 1$ tests satisfies $M \leq |V|^2 = \mathcal{O}(q^2)$. For each surviving pair (a, b) , put $\lambda_i = ag_i + b$ and $\tilde{D}_i^{(P_1)} = T[\lambda_i]$ for $i = 2, \dots, n$. For the correct pair (a^*, b^*) , Lemma 3.3.2 gives

$$\tilde{D}_i^{(P_1)} = \{P_i, \tau_{R_0}(P_i)\}, \quad i = 2, \dots, n,$$

which are exactly the candidate sets obtained in the proof of Theorem 3.3.3 when $P_1 = R_0$. It remains to resolve this ambiguity. For this purpose, one code $\mathcal{U}_2^{(P_j)}$ is sufficient. Choose $j \geq 2$ such that $[2]W \neq [2]R_0$ for $W \in \tilde{D}_j^{(P_1)}$. This condition is independent of the choice of W , since

$$[2]\tau_{R_0}(P_j) = [4]R_0 \ominus [2]P_j = [2]R_0 \iff [2]P_j = [2]R_0,$$

and it can therefore be checked in $\mathcal{O}(1)$ operations. Such an index exists among $j \in \{2, \dots, N_2 + 1\} \subseteq \{2, \dots, 5\}$, because at most $N_2 - 1 \leq 3$ points of D lie in the coset $R_0 \oplus \mathcal{E}[2](\mathbb{F}_q)$. Fix any $l \neq j$, $l \geq 2$. For each of the at most four pairs $(W_j, W_l) \in \tilde{D}_j^{(P_1)} \times \tilde{D}_l^{(P_1)}$, construct the function $f_2^{(W_j)}$ by (3) and normalize a basis vector $\mathbf{g}^{(P_j)}$ of $\mathcal{U}_2^{(P_j)}$ by solving

$$\begin{cases} a'g_1^{(P_j)} + b' = f_2^{(W_j)}(R_0), \\ a'g_l^{(P_j)} + b' = f_2^{(W_j)}(W_l), \end{cases} \quad (13)$$

which is uniquely solvable whenever $W_l \neq \tau_{W_j}(R_0)$. The first equation is available because position 1 is known to contain R_0 . We do not explicitly construct the candidate sets $\tilde{D}_i^{(P_j)} = \{P_i, \tau_{W_j}(P_i)\}$. Instead, for every $i \notin \{1, j\}$ and every $W \in \tilde{D}_i^{(P_1)}$, we test whether

$$f_2^{(W_j)}(W) = a'g_i^{(P_j)} + b'. \quad (14)$$

By Lemma 3.3.2, the points satisfying this equality are exactly the elements of $\tilde{D}_i^{(P_1)} \cap \tilde{D}_i^{(P_j)}$. Thus, this step requires $\mathcal{O}(n)$ operations for each surviving pair. For the correct choice of $W_j = P_j$, the condition $[2]P_j \neq [2]R_0$ implies

$$\tilde{D}_i^{(P_1)} \cap \tilde{D}_i^{(P_j)} = \{P_i\}, \quad i \notin \{1, j\},$$

so that all the intersections are singletons and the resulting points are pairwise distinct. In this case we obtain the divisor $\hat{D} = R_0 + \sum_{i \neq 1} P_i$. The correct pair (a^*, b^*) passes all membership tests and yields the correct divisor, so the procedure does not fail. We now consider the complexity of recovering the divisor $\hat{D}$. Constructing the hash table T requires $\mathcal{O}(|\mathcal{E}(\mathbb{F}_q)|)$ operations. Computing the five codes $\mathcal{U}_2^{(P_1)}, \mathcal{U}_2^{(P_2)}, \dots, \mathcal{U}_2^{(P_5)}$ requires $\mathcal{O}(k^2 n^2)$ operations and is performed only once. The enumeration of (a, b) costs $\mathcal{O}(nq^2)$, while processing all surviving pairs also costs $\mathcal{O}(nq^2)$, since there are $M = \mathcal{O}(q^2)$ such pairs and only a constant number of combinations (W_j, W_l) is considered for each pair. Since $|\mathcal{E}(\mathbb{F}_q)| = \Theta(q)$ by the Hasse bound, the complexity of this part is

$$\mathcal{O}(k^2 n^2 + nq^2 + |\mathcal{E}(\mathbb{F}_q)|) = \mathcal{O}(k^2 n^2 + nq^2).$$

It remains to recover the second divisor. Since $\hat{D}$ is now known and $\hat{G} = \sigma(G)$ is effective of degree $k = \deg(G)$ with $5 \leq k \leq \frac{n}{2} - 1$, Proposition 3.4.2 applies to the pair $(\hat{D}, \hat{G})$: computing the Schur square $\mathcal{C}^{(2)} = \mathcal{C}_{\mathcal{L}}(\hat{D}, 2\hat{G})$ and testing, for every $Q \in \mathcal{E}(\mathbb{F}_q) \setminus \text{supp}(\hat{D})$, whether

$$\text{ev}_{\hat{D}}(f_2^{(Q)}) \in \mathcal{C}^{(2)},$$

we obtain $\text{supp}(\hat{G})$ in $\mathcal{O}(k^2 n^2 + (|\mathcal{E}(\mathbb{F}_q)| - n)n^2)$ operations. The multiplicities are then determined as in the proof of Theorem 3.3.4: for each $Q \in \text{supp}(\hat{G})$ we increase s as long as $\text{ev}_{\hat{D}}(f_s^{(Q)}) \in \mathcal{C}$, which stops exactly at $s = v_Q(\hat{G})$, because the pole divisor of $f_s^{(Q)}$ equals sQ and hence $\text{ev}_{\hat{D}}(f_s^{(Q)}) \in \mathcal{C}$ if and only if $\hat{G} - sQ \geq 0$. Since the functions $f_s^{(Q)}$ are constructed iteratively (Algorithm 6), this costs $\mathcal{O}(\deg(\hat{G})n^2) = \mathcal{O}(kn^2)$ operations, which is absorbed by $\mathcal{O}(k^2 n^2)$. Note that no assumption on the shape of G is needed here: σ being an automorphism, one has $\hat{G} = \sum_i k_i \sigma(Q_i)$ whenever $G = \sum_i k_i Q_i$, so $\hat{G}$ has the same support size and the same multiplicities as G . By Lemma 3.4.1, the pair $(\hat{D}, \hat{G})$ satisfies $\mathcal{C}_{\mathcal{L}}(\hat{D}, \hat{G}) = \mathcal{C}$ and is therefore an equivalent key. The total complexity of recovering $\hat{D}$ and $\hat{G}$ is

$$\mathcal{O}(k^2 n^2 + nq^2 + |\mathcal{E}(\mathbb{F}_q)| + (|\mathcal{E}(\mathbb{F}_q)| - n)n^2 + kn^2) = \mathcal{O}(k^2 n^2 + nq^2 + (|\mathcal{E}(\mathbb{F}_q)| - n)n^2).$$

□

Remark 5. The bound $M_m \leq |V|^2$ of (12), used in the proof of Theorem 3.4.3, is a worst-case estimate. It does not take into account the reduction in the number of candidates caused by the successive tests. To estimate the average-case complexity, we make the following heuristic independence assumption: for every incorrect pair $(a, b) \neq (a^*, b^*)$, the values $ag_i + b$, $i = 2, \dots, n$, behave as independent uniformly distributed elements of $\mathbb{F}_q$.

Under this assumption an incorrect pair survives each additional test with probability $\rho = |V|/q$, where, by (11), the Hasse bound and $N_2 \leq 4$,

$$\rho = \frac{|V|}{q} \leq \frac{1}{2} + \frac{1}{\sqrt{q}} + \frac{3}{2q}, \quad (15)$$

so that $M_m \approx \rho^m q^2$ and the expected number of tests performed for an incorrect pair equals $(1 - \rho)^{-1} \leq 3$ for $q \geq 64$. Consequently, the expected cost of the enumeration is $\mathcal{O}(q^2)$ instead of $\mathcal{O}(nq^2)$, and the expected number of surviving pairs is $1 + q^2 \rho^{n-1}$. Since $q^2 \rho^{n-1} \leq 1$ as soon as $n \geq 1 + 2 \log_{1/\rho} q$, which holds for all practical parameter sets, only the correct pair survives and has to be processed in the second step. The expected complexity of the attack is therefore

$$\mathcal{O}(k^2 n^2 + q^2 + (|\mathcal{E}(\mathbb{F}_q)| - n)n^2).$$

This assumption is not a consequence of the previous statements: for a fixed pair (a, b) these values are completely determined by (a, b) and by the chosen vector $\mathbf{g}$, and are therefore neither random nor independent. Nevertheless, the resulting probability and the corresponding estimate agree with our practical experiments, in which we obtained a unique pair (a^*, b^*) passing all the tests for almost every code length n .

Algorithm 5 Recovering an equivalent key $(\widehat{D}, \widehat{G})$ without known points

Require: Elliptic curve $\mathcal{E}/\mathbb{F}_q$ ($\text{char}(\mathbb{F}_q) > 3$); generator matrix $\mathbf{G}_{\text{pub}}$ of the code $\mathcal{C}_{\mathcal{L}}(D, G)$, where $5 \leq \deg(G) \leq \frac{n}{2} - 1$.

Ensure: Divisors $\widehat{D} = \sigma(D)$ and $\widehat{G} = \sigma(G)$ for some $\sigma \in \text{Aut}(\mathcal{E}/\mathbb{F}_q)$, or $\perp$.

```

1: Choose an arbitrary point  $R_0 = (\alpha, \beta) \in \mathcal{E}(\mathbb{F}_q) \setminus \{P_\infty\}$  and set  $P_1 \leftarrow R_0$ .
2: if  $\beta \neq 0$  then  $\triangleright R_0 \notin \mathcal{E}[2]$ 
3:    $\gamma \leftarrow \frac{3\alpha^2 + a_4}{2\beta}$ ;  $f_2^{(P_1)} \leftarrow \frac{y + \beta + \gamma(x - \alpha)}{(x - \alpha)^2}$ 
4: else
5:    $f_2^{(P_1)} \leftarrow \frac{1}{x - \alpha}$   $\triangleright R_0 \in \mathcal{E}[2]$ 
6: Compute the value set  $V \leftarrow f_2^{(P_1)}(\mathcal{E}(\mathbb{F}_q) \setminus \{R_0\})$  and, for every  $\lambda \in V$ , the fibre  $T[\lambda] \leftarrow \{P \in \mathcal{E}(\mathbb{F}_q) : f_2^{(P_1)}(P) = \lambda\}$ .
7: Compute the generator matrices of  $\mathcal{U}_2^{(P_j)} = \mathcal{C}_{\mathcal{L}}(D - P_j, 2P_j)$ ,  $j = 1, \dots, 5$ , using  $\mathbf{G}_{\text{pub}}$ , Propositions 3.1.1 and 3.1.2.
8: Find  $\mathbf{g} = (g_2, \dots, g_n) \in \mathcal{U}_2^{(P_1)} \setminus \langle \mathbf{1} \rangle$  and  $\mathbf{g}^{(P_j)} \in \mathcal{U}_2^{(P_j)} \setminus \langle \mathbf{1} \rangle$  for  $j = 2, \dots, 5$ , the coordinates of  $\mathbf{g}^{(P_j)}$  being indexed by  $\{1, \dots, n\} \setminus \{j\}$ .
9: for  $(a, b) \in \mathbb{F}_q^* \times \mathbb{F}_q$  do
10:   if  $ag_i + b \in V$  for all  $i = 2, \dots, n$  then  $\triangleright$  two positions  $i_0, i_1$  with  $g_{i_0} \neq g_{i_1}$  are tested first
11:      $\widetilde{D}_i^{(P_1)} \leftarrow T[ag_i + b]$  for  $i = 2, \dots, n$ .
12:     Find  $j \in \{2, \dots, 5\}$  such that  $[2]W \neq [2]R_0$  for  $W \in \widetilde{D}_j^{(P_1)}$ , and fix any  $l \geq 2$ ,  $l \neq j$ .
13:     for  $W_j \in \widetilde{D}_j^{(P_1)}$  and  $W_l \in \widetilde{D}_l^{(P_1)}$  such that  $W_l \neq \tau_{W_j}(R_0)$  do
14:       Construct  $f_2^{(W_j)}$  as in steps 2–5 with  $W_j$  in place of  $R_0$  and solve (13) for  $(a', b')$ .
15:        $S_i \leftarrow \{W \in \widetilde{D}_i^{(P_1)} \mid f_2^{(W_j)}(W) = a'g_i^{(P_j)} + b'\}$  for  $i \neq 1, j$ .
16:       if all the sets  $S_i$  are singletons and their elements are pairwise distinct then
17:          $\widehat{D} \leftarrow R_0 + W_j + \sum_{i \neq 1, j} S_i$  and recover  $\widehat{G}$  by Theorem 3.3.4 and Proposition 3.4.2.
18:       return  $(\widehat{D}, \widehat{G})$ .
19: return  $\perp$ 

```

Remark 6. The proof of Theorem 3.4.3 and Algorithm 5 use the five codes $\mathcal{U}_2^{(P_j)}$, although in practice a single additional code $\mathcal{U}_2^{(P_2)}$ is almost always sufficient. Indeed, the code $\mathcal{U}_2^{(P_j)}$ is useful only when $[2]P_j \neq [2]R_0$, that is, when $P_j \notin R_0 \oplus \mathcal{E}[2](\mathbb{F}_q)$. Since this coset contains $N_2 \leq 4$ points, one of which is R_0 itself, at most $N_2 - 1 \leq 3$ positions of the divisor D are unusable, and therefore a suitable index always exists among $j \in \{2, \dots, 5\}$. Precomputing these four codes thus guarantees that the divisor is recovered with probability 1.

On the other hand, for a uniformly random divisor D any position is unusable with probability at most

$$\frac{N_2 - 1}{|\mathcal{E}(\mathbb{F}_q)| - 1} \leq \frac{3}{|\mathcal{E}(\mathbb{F}_q)| - 1},$$

so with probability at least $1 - \frac{3}{|\mathcal{E}(\mathbb{F}_q)| - 1}$ the index $j = 2$ is admissible and only the two codes $\mathcal{U}_2^{(P_1)}$ and $\mathcal{U}_2^{(P_2)}$ have to be computed.

References

- [1] Shkuratov, Y.D.: Attack on the ECC2 cryptosystem and the McEliece cryptosystem built on elliptic codes. *Prikl. Diskr. Mat. Suppl.* (18), 188–194 (2025)
- [2] McEliece, R.J.: A public-key cryptosystem based on algebraic coding theory. *DSN Progress Report*, 42–44 (1978)
- [3] Janwa, H., Moreno, O.: McEliece public key cryptosystems using algebraic-geometric codes. *Designs, Codes and Cryptography* **8**(3), 293–307 (1996) <https://doi.org/10.1007/BF00173300>
- [4] Sidelnikov, V.M., Shestakov, S.O.: On insecurity of cryptosystems based on generalized reed-solomon codes. *Discrete Mathematics and Applications* **2**(4), 439–444 (1992) <https://doi.org/10.1515/dma.1992.2.4.439>
- [5] Minder, L.: Cryptography based on error correcting codes. PhD thesis, Verlag nicht ermittelbar (2007)
- [6] Faure, C., Minder, L.: Cryptanalysis of the McEliece cryptosystem over hyperelliptic codes. In: *Eleventh International Workshop on Algebraic and Combinatorial Coding Theory*, Pamporovo, Bulgaria, pp. 99–107 (2008)
- [7] Cheng, Q.: Hard problems of algebraic geometry codes. *IEEE Transactions on Information Theory* **54**, 402–406 (2008) <https://doi.org/10.1109/TIT.2007.911213>
- [8] Márquez-Corbella, I., Martínez-Moro, E., Pellikaan, R.: On the unique representation of very strong algebraic geometry codes. *Designs, Codes and Cryptography* **70**, 215–230 (2014) <https://doi.org/10.1007/s10623-012-9758-3>
- [9] Couvreur, A., Márquez-Corbella, I., Pellikaan, R.: Cryptanalysis of mceliece cryptosystem based on algebraic geometry codes and their subcodes. *IEEE Transactions on Information Theory* **63**(8), 5404–5418 (2017) <https://doi.org/10.1109/TIT.2017.2712636>
- [10] Zhang, F., Zhang, Z., Guan, P.: Ecc2: Error correcting code and elliptic curve based cryptosystem. *Information Sciences* (526), 301–320 (2020) <https://doi.org/10.1016/j.ins.2020.03.069>
- [11] Stichtenoth, H.: *Algebraic Function Fields and Codes* vol. 254, 4th edn. Springer, Switzerland (2009). <https://doi.org/10.1007/978-3-319-16721-3>
- [12] Kuninets, A., Malygina, E.: Explicit bases for riemann-roch spaces on elliptic curves and their application in constructing various elliptic code families. *Cryptography and Communications* (2026) <https://doi.org/10.1007/s12095-026-00923-w>
- [13] Couvreur, A., Panaccione, I.: Power error locating pairs. *Designs, Codes and Cryptography* **88**(8), 1561–1593 (2020) <https://doi.org/10.1007/s10623-020-00774-3>
- [14] Silverman, J.: *The Arithmetic of Elliptic Curves*. Graduate Texts in Mathematics, p. 513. Springer, New York (2009). <https://doi.org/10.1007/978-0-387-09494-6>

Appendix A Auxiliary results for the attack from Section 3.3

A.1 Riemann-Roch bases for arbitrary effective divisors from [12]

Lemma A.1.1 ([12, Lemma 3.1.1]). *Let $\mathcal{E}/\mathbb{F}_q$ be an elliptic curve, and let $P = (\alpha, \beta) \in \mathcal{E}(\mathbb{F}_q) \setminus P_\infty$. The basis $\mathcal{L}_b$ of the Riemann – Roch space associated with the divisor $G = kP$, where $k \in \mathbb{N}_{\geq 2}$, is defined as follows:*

Case 1: If $P \notin \mathcal{E}[2]$ (i.e., $-P \neq P$), then the basis is:

$$\mathcal{L}_b = \{1, f_2, f_3, \dots, f_k\}, \quad f_s(x, y) = \frac{y + A_s(x)}{(x - \alpha)^s},$$

where $A_s(x) \in \mathbb{F}_q(\mathcal{E})$ is a function of degree $\deg(A_s) \leq s-1$, satisfying:

1. $v_{P'}(y + A_s(x)) \geq s$ for $P' = -P$;
2. $v_Q(y + A_s(x)) \geq 0$ for any $Q \neq P'$.

Case 2: If $P \in \mathcal{E}[2]$ (i.e., $-P = P$), then the basis is:

$$\mathcal{L}_b = \{1\} \cup \left\{ \frac{1}{(x-\alpha)^s} \mid 1 \leq s \leq \left\lfloor \frac{k}{2} \right\rfloor \right\} \cup \left\{ \frac{y-\beta}{(x-\alpha)^s} \mid 2 \leq s \leq \left\lfloor \frac{k+1}{2} \right\rfloor \right\}.$$

Theorem A.1.2 ([12, Theorem 3.1.2]). Let $\mathcal{E}/\mathbb{F}_q$ be an elliptic curve in general Weierstrass form:

$$\mathcal{E} : y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6,$$

and let $P_1 = (\alpha_1, \beta_1), \dots, P_z = (\alpha_z, \beta_z)$ be distinct rational points in $\mathcal{E}(\mathbb{F}_q) \setminus \{P_\infty\}$. For a divisor $G = \sum_{i=1}^z k_i P_i$ with $k_i \in \mathbb{N}_{\geq 1}$, the Riemann–Roch space $\mathcal{L}(G)$ admits a basis consisting of the constant function together with functions of the following two forms.

1. Single-point basis functions, with pole divisor sP_i for $1 \leq i \leq z$ and $2 \leq s \leq k_i$:

- If $P_i \notin \mathcal{E}[2]$ (i.e., $-P_i \neq P_i$), then

$$f_{i,s}(x, y) = \frac{y + A_{i,s}(x)}{(x - \alpha_i)^s},$$

where $A_{i,s}(x) \in \mathbb{F}_q[x]$ has degree $\leq s-1$ and satisfies $v_{-P_i}(y + A_{i,s}(x)) \geq s$ (constructed as in Algorithm 6).

- If $P_i \in \mathcal{E}[2]$ (i.e., $-P_i = P_i$), then

$$f_{i,s}(x, y) = \begin{cases} \frac{1}{(x - \alpha_i)^{s/2}}, & \text{if } s \text{ is even,} \\ \frac{y - \beta_i}{(x - \alpha_i)^{(s+1)/2}}, & \text{if } s \text{ is odd.} \end{cases}$$

2. Double-point basis functions, with pole divisor $P_i + P_j$ for $1 \leq i \neq j \leq z$ (simple poles at P_i and P_j):

$$g_{i,j}(x, y) = \begin{cases} \frac{y + B_{i,j}(x)}{(x - \alpha_i)(x - \alpha_j)}, & \text{if } \alpha_i \neq \alpha_j, \\ \frac{1}{x - \alpha_i}, & \text{if } \alpha_i = \alpha_j, \end{cases}$$

where $B_{i,j}(x)$ is the linear polynomial

$$B_{i,j}(x) = \left(\frac{\beta_j - \beta_i}{\alpha_j - \alpha_i} + a_1 \right) x + \left((\beta_i + a_1\alpha_i + a_3) - \left(\frac{\beta_j - \beta_i}{\alpha_j - \alpha_i} + a_1 \right) \alpha_i \right).$$

Algorithm 6 provides one possible method for constructing the functions of the form $f_s(X, Y) = \frac{Y + A_s(X)}{(X - \alpha)^s}$ from Lemma A.1.1.

Algorithm 6 Basis for $\mathcal{L}(kP_{\alpha,\beta})$ from [12]

Require: $\mathcal{E}/\mathbb{F}_q : \begin{cases} y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6, & \text{if } \text{char}(\mathbb{F}_q) = 2, \\ y^2 = x^3 + a_4x + a_6, & \text{if } \text{char}(\mathbb{F}_q) > 3. \end{cases}$

Point $P_{\alpha,\beta} = (\alpha, \beta) \in \mathcal{E}(\mathbb{F}_q) \setminus \mathcal{E}[2]$;

Integer $k \geq 2$.

Ensure: Basis $\{1, f_2, \dots, f_k\}$ for $\mathcal{L}(kP_{\alpha,\beta})$.

1: $\beta' \leftarrow \begin{cases} \beta + a_1\alpha + a_3, & \text{char}(\mathbb{F}_q) = 2, \\ -\beta, & \text{char}(\mathbb{F}_q) > 3. \end{cases}$

2: $\mathcal{L}_b \leftarrow \{1\}$.

3: Set local parameter $t \leftarrow x - \alpha$.

4: **for** $s = 2$ **to** k **do**

▷ Expand $y = \beta' + \sum_{j=1}^{s-1} c_j t^j + O(t^s)$

5: $c_1 \leftarrow \begin{cases} \frac{\alpha^2 + a_4 + a_1\beta'}{a_1\alpha + a_3}, & \text{char}(\mathbb{F}_q) = 2, \\ \frac{3\alpha^2 + a_4}{2\beta'}, & \text{char}(\mathbb{F}_q) > 3. \end{cases}$

6: **for** $j = 2$ **to** s **do**

▷ Substitute $x = t + \alpha$ into the curve equation

7: $S_j \leftarrow \sum_{i=1}^{j-1} c_i c_{j-i}$.

8: $c_j = \begin{cases} \frac{(\alpha + a_2)\delta_{j,2} + \delta_{j,3} + a_1c_{j-1} + S_j}{a_1\alpha + a_3}, & \text{char}(\mathbb{F}_q) = 2, \\ \frac{(3\alpha)\delta_{j,2} + \delta_{j,3} - S_j}{2\beta'}, & \text{char}(\mathbb{F}_q) > 3. \end{cases}$

9: $A_s(x) \leftarrow -\beta' - \sum_{j=1}^{s-1} c_j (x - \alpha)^j$.

10: $f_s \leftarrow \frac{y + A_s(x)}{(x - \alpha)^s}$.

11: $\mathcal{L}_b \leftarrow \mathcal{L}_b \cup \{f_s\}$.

12: **return** $\mathcal{L}_b$.
