**Research Article**

Yasushi Takahashi*, Momonari Kudo, Ryoya Fukasaku, Yasuhiko Ikematsu, Masaya Yasuda, and Kazuhiro Yokoyama

# Algebraic approaches for solving isogeny problems of prime power degrees

https://doi.org/10.1515/jmc-2020-0072
Received Jun 05, 2019; accepted Jul 01, 2019

**Abstract:** Recently, supersingular isogeny cryptosystems have received attention as a candidate of post-quantum cryptography (PQC). Their security relies on the hardness of solving isogeny problems over supersingular elliptic curves. The meet-in-the-middle approach seems the most practical to solve isogeny problems with classical computers. In this paper, we propose two algebraic approaches for isogeny problems of prime power degrees. Our strategy is to reduce isogeny problems to a system of algebraic equations, and to solve it by Gröbner basis computation. The first one uses modular polynomials, and the second one uses kernel polynomials of isogenies. We report running times for solving isogeny problems of 3-power degrees on supersingular elliptic curves over $\mathbb{F}_{p^2}$ with 503-bit prime $p$, extracted from the NIST PQC candidate SIKE. Our experiments show that our first approach is faster than the meet-in-the-middle approach for isogeny degrees up to $3^{10}$.

**Keywords:** Elliptic curves, Isogenies, Vélu's formulae, Gröbner basis computation

**2010 Mathematics Subject Classification:** Primary: 14G50, Secondary: 94A60

## 1 Introduction

Since Koblitz [25] and Miller [27] independently proposed elliptic curve cryptography (ECC) in 1985, elliptic curves have been used in cryptography. Since 2000, pairings over elliptic curves have been an indispensable tool to construct cryptographic protocols and functional encryption schemes. Since 2006, *isogenies* between elliptic curves have began to be used in [9, 29, 34] for constructing several cryptosystems and hash functions. In particular, supersingular isogeny graphs were first proposed in [9] for security, which introduced the supersingular isogeny graph path-finding problem as a hard problem in cryptography. Actually, there exists a subexponential quantum algorithm [10] for computing an isogeny between ordinary elliptic curves since it is reduced to solving the abelian hidden shift problem. (Recently, the key exchange protocol based on group action [8] is regarded as a credible post-quantum system despite of the existence of a sub-exponential attack.)

***Corresponding Author: Yasushi Takahashi:** Fujitsu Laboratories Ltd., 1-1, Kamikodanaka 4-chome, Nakahara-ku, Kawasaki 211-8588, Japan; Email: t-yasushi@fujitsu.com
**Momonari Kudo:** Department of Mathematical Informatics, Graduate School of Information Science and Technology, The University of Tokyo, 7-3-1, Hongo, Bunkyo-ku, Tokyo 113-8656, Japan; Email: kudo@mist.i.u-tokyo.ac.jp
**Ryoya Fukasaku:** Faculty of Mathematics, Kyushu University, 744 Motooka, Nishi-ku, Fukuoka 819-0395, Japan; Email: fukasaku@math.kyushu-u.ac.jp
**Yasuhiko Ikematsu:** Institute of Mathematics for Industry, Kyushu University, 744 Motooka, Nishi-ku, Fukuoka 819-0395, Japan; Email: ikematsu@imi.kyushu-u.ac.jp
**Masaya Yasuda:** Department of Mathematics, Rikkyo University, Nishi-Ikebukuro, Tokyo 171-8501, Japan; Email: myasuda@rikkyo.ac.jp
**Kazuhiro Yokoyama:** Department of Mathematics, Rikkyo University, Nishi-Ikebukuro, Tokyo 171-8501, Japan; Email: kazuhiro@rikkyo.ac.jp

In contrast, the supersingular case is non-abelian and seems to be a promising candidate for PQC. (cf., The security of ECC and pairing-based cryptography relies on the discrete logarithm problem over elliptic curves, which could be solved in polynomial-time by Shor's algorithm [31] using quantum computers.) In 2011, Jao and De Feo [22] introduced the key exchange protocol using using supersingular isogenies as a post-quantum candidate, based on pseudo-random walks in supersingular isogeny graphs. (See [11] for the connection between the hard problems in [22] and the path-finding problem in [9].) Other cryptographic functions were subsequently developed in [14, 23, 35]. In 2017, Jao et al. [21] submitted algorithms of supersingular isogeny key encapsulation, called SIKE, for the NIST PQC standardization process. It remains as a second-round candidate [28].

The template for the security of isogeny cryptosystems is the *general isogeny problem* [20]; Given two elements $j, \tilde{j}$ of a finite field $\mathbb{F}_q$, to find an isogeny $\phi : E \to \tilde{E}$, if exists, such that $j(E) = j$ and $j(\tilde{E}) = \tilde{j}$, where $j(E)$ denotes the $j$-invariant of an elliptic curve $E$. In the supersingular case, it is sufficient to take $\mathbb{F}_{p^2}$ as the base field for a prime $p$ since every supersingular curve over $\overline{\mathbb{F}}_p$ is isomorphic to one defined over $\mathbb{F}_{p^2}$ [33]. A variant of this problem is when the degree of $\phi$ is known, and it arises from the cryptanalysis of the hash function of [9], which requires computing isogenies of degree $\ell_0^e$ for some small $\ell_0$ and large $e$. The most cryptographically-interesting variant is the *supersingular isogeny Diffie-Hellman (SIDH)* problem [14], assuring the security of [21, 22], in which supersingular elliptic curves over $\mathbb{F}_{p^2}$ are chosen for a special prime $p = \ell_1^{e_1}\ell_2^{e_2}f \pm 1$ and a lot of auxiliary information are given (see also [19]). The basic algorithm to solve the general isogeny problem for ordinary curves is due to Galbraith [18]. His procedure is (1) to reduce the problem to the case of elliptic curves $E'$ and $\tilde{E}'$ whose endomorphism ring is maximal, and then (2) to construct an isogeny between $E'$ and $\tilde{E}'$ by building isogeny trees. The time complexity of step (1) for $E$ is negligible when the conductor $[\mathcal{O}_K : \mathrm{End}(E)]$ is smooth, where $\mathcal{O}_K$ denotes the maximal order of the imaginary quadratic field $K = \mathrm{End}(E) \otimes_{\mathbb{Z}} \mathbb{Q}$ with endomorphism ring $\mathrm{End}(E)$. For step (2), a sub-exponential quantum algorithm was discovered in [10]. In contrast, the meet-in-the-middle approach in [18] is applicable to the supersingular isogeny graph by building isogeny trees from $E$ to $\tilde{E}$ directly. The best known quantum algorithm is due to Biasse et al. [3] for the supersingular problem, and its time complexity is exponential. For the SIDH problem, a faster quantum algorithm is given by Tani's claw finding algorithm [36], but its time complexity is still exponential. (See also [24].) In recent, several related computational problems have been discussed in [15] for supersingular elliptic curves, their isogeny graphs, and their endomorphism rings.

Although time complexities of fast quantum algorithms has been discussed actively as described above, we consider *practical* approaches for solving isogeny problems with classical computers. Here we focus on the supersingular isogeny graph path-finding problem in [9] of prime power degree $\ell = \ell_0^e$ for small $\ell_0$. The meet-in-the-middle approach is a practical way to solve the isogeny problem with time complexity equal to the square root of $\ell$. It seems the most practical way at least in the supersingular case. (It is reported in [1] that van Oorschot and Wiener's parallel golden collision search [38] can outperform the meet-in-the-attack for solving isogeny problems with *parallel* computation. See also [12] for improvements of the parallel golden collision search.) While the meet-in-the-middle approach is a generic way for solving graph problems, we propose two algebraic approaches for solving the isogeny problem in this paper. (cf., Several algebraic approaches are considered in [26] to attack the SIDH/SIKE protocol.) Our basic strategy is to reduce the isogeny problem to a system of algebraic equations. (It does not depend on the supersingularity.) Specifically, our first approach uses the modular polynomial of prime level $\ell_0$. We divide a system of equations of modular polynomial into two parts like the meet-in-the-middle approach, and compute their Gröbner bases to efficiently find $j$-invariants of intermediate curves between given two isogenous elliptic curves $E$ and $\widetilde{E}$. In contrast, our second approach takes an intermediate curve $E_0$ between $E$ and $\widetilde{E}$, and consider kernel polynomials $F(x)$ and $\widetilde{F}(\tilde{x})$ of two isogenies $\varphi : E \to E_0$ and $\tilde{\varphi} : \widetilde{E} \to E_0$. Since the curve $E_0$ is unknown, we regard its Weierstrass coefficients as variables, and also represent kernel polynomials $F(x)$ and $\widetilde{F}(\tilde{x})$ as certain multivariate polynomials, based on Schoof's work [30]. Furthermore, by using Vélu's formulae [6, 39], we represent isogenies $\varphi$ and $\tilde{\varphi}$ as rational functions over multivariate polynomial rings, and set up algebraic equations to determine the Weierstrass coefficients of $E_0$. Moreover, we report running times of our algebraic approaches for solving the isogeny problem of 3-power degrees on supersingular elliptic curves over $\mathbb{F}_{p^2}$ with 503-bit prime $p$, extracted from SIKE-p503 parameters [21], to compare with the meet-in-the-middle approach in performance.

# 2 Mathematical background

In this section, we review some basic definitions and properties of elliptic curves and isogenies, which shall be required in Section 3 below.

## 2.1 Elliptic curves over finite fields

An elliptic curve $E$ over a finite field $\mathbb{F}_q$ of characteristic $p \geq 5$ is defined by the (short) Weierstrass form $y^2 = x^3 + ax + b$ with $a, b \in \mathbb{F}_q$ and discriminant $\Delta(E) = -16(4a^3 + 27b^2) \neq 0$. Its $j$-invariant is defined as $j(E) = -1728\frac{(4a)^3}{\Delta(E)}$. Two elliptic curves are isomorphic over $\overline{\mathbb{F}}_q$ if and only if they both have the same $j$-invariant, where $\overline{\mathbb{F}}_q$ denotes the algebraic closure of $\mathbb{F}_q$. Moreover, given an element $j_0$ of $\mathbb{F}_q$, there exists an elliptic curve over $\mathbb{F}_q$ with $j$-invariant equal to $j_0$. The set of $\mathbb{F}_q$-rational points

$$E(\mathbb{F}_q) = \left\{(x, y) \in \mathbb{F}_q^2 : y^2 = x^3 + ax + b\right\} \cup \{\mathcal{O}_E\}$$

forms an abelian group, where $\mathcal{O}_E$ denotes the infinity point on $E$. (See [33, Chap. III] for the group law.) The number of $\mathbb{F}_q$-rational points on $E$, denoted by $\#E(\mathbb{F}_q)$, is finite, and it is represented as $\#E(\mathbb{F}_q) = q + 1 - t$ where $t$ denotes the trace of the $q^{\text{th}}$-power Frobenius map (see [33, Chap. V] for the map). It satisfies $|t| \leq 2\sqrt{q}$ by Hasse's theorem. An elliptic curve $E$ over $\mathbb{F}_q$ is said *supersingular* if the characteristic $p$ divides the trace $t$. Otherwise $E$ is said *ordinary*. Every supersingular elliptic curve over $\overline{\mathbb{F}}_p$ has its $j$-invariant defined over $\mathbb{F}_{p^2}$ [33, Thm. 3.1, Chap. V], and hence it is isomorphic (over $\overline{\mathbb{F}}_q$) to one defined over $\mathbb{F}_{p^2}$. Furthermore, there are about $\frac{p}{12}$ isomorphism classes of supersingular elliptic curves over $\overline{\mathbb{F}}_p$.

## 2.2 Isogenies and Vélu's formulae

Let $E$ and $\widetilde{E}$ be two elliptic curves over a finite field $\mathbb{F}_q$. A morphism $\phi : E \to \widetilde{E}$ satisfying $\phi(\mathcal{O}_E) = \mathcal{O}_{\widetilde{E}}$ is called an *isogeny*. Two elliptic curves $E$ and $\widetilde{E}$ are called *isogenous* if there is a non-zero isogeny between them. Tate's theorem [37] states that $E$ and $\widetilde{E}$ are isogenous over $\mathbb{F}_q$ if and only if $\#E(\mathbb{F}_{q^k}) = \#\widetilde{E}(\mathbb{F}_{q^k})$ for any positive integer $k$. Every non-zero isogeny $\phi : E \to \widetilde{E}$ induces an injection of function fields $\phi^\star : \overline{\mathbb{F}}_q(\widetilde{E}) \to \overline{\mathbb{F}}_q(E)$ (see [33, Chap. III]). The degree of a non-zero isogeny is defined as $\deg \phi = [\overline{\mathbb{F}}_q(E) : \phi^\star\overline{\mathbb{F}}_q(\widetilde{E})]$. We say that $\phi$ is *separable* if the finite extension $\overline{\mathbb{F}}_q(E)/\phi^\star\overline{\mathbb{F}}_q(\widetilde{E})$ is separable. A non-zero isogeny is separable if its degree is not divisible by the characteristic of the base field $\mathbb{F}_q$.

A non-zero isogeny $\phi : E \to \widetilde{E}$ induces a (surjective) group homomorphism from $E(\overline{\mathbb{F}}_q)$ to $\widetilde{E}(\overline{\mathbb{F}}_q)$ [33, Thm. 4.8, Chap. III], and its kernel is a finite subgroup of $E(\overline{\mathbb{F}}_q)$, denoted by $E[\phi]$. It satisfies $\deg \phi = \#E[\phi]$ if $\phi$ is separable. Conversely, given any finite subgroup $S$ of $E(\overline{\mathbb{F}}_q)$, there are a unique elliptic curve $\widetilde{E}$ and a separable isogeny $\phi : E \to \widetilde{E}$ with $E[\phi] = S$ [33, Prop. 4.12, Chap. III]. The curve $\widetilde{E}$ is denoted by the quotient $E/S$. Vélu [39] showed how to explicitly represent the isogeny $\phi : E \to \widetilde{E} = E/S$ and the Weierstrass equation for $\widetilde{E}$. A non-zero isogeny $\phi : E \to \widetilde{E}$ is *normalized* if $\phi^\star(\omega_{\widetilde{E}}) = \omega_E$, where $\omega_E$ and $\omega_{\widetilde{E}}$ denote the invariant differentials of $E$ and $\widetilde{E}$, respectively (see [33, Chap. III] for the invariant differential of an elliptic curve). Vélu's formulae give a normalized separable isogeny, and its modified form is shown in [6] as below (the form was given much earlier by Elkies [16]):

**Modified Vélu's formulae in [6]**
Let $E : y^2 = x^3 + ax + b$ be an elliptic curve over a finite field $\mathbb{F}_q$ of characteristic $p \geq 5$. Let $\ell$ be an odd number, and $S$ a subgroup of $E(\overline{\mathbb{F}}_q)$ of order $\ell$. Set $S^\star = S \setminus \{\mathcal{O}_E\}$. Then a normalized separable isogeny $\phi : E \to \widetilde{E} = E/S$ of degree $\ell$ can be written as

$$\phi(x, y) = \left(\frac{N(x)}{D(x)},\ y\left(\frac{N(x)}{D(x)}\right)'\right), \tag{1}$$

where $D(x)$ is the polynomial defined as

$$D(x) = \prod_{Q \in S^\star} (x - x_Q) = x^{\ell-1} - sx^{\ell-2} + s_2 x^{\ell-3} - s_3 x^{\ell-4} + \cdots$$

and $N(x)$ is related to $D(x)$ through the formula

$$\frac{N(x)}{D(x)} = \ell x - s - (3x^2 + a)\frac{D'(x)}{D(x)} - 2(x^3 + ax + b)\left(\frac{D'(x)}{D(x)}\right)'. \tag{2}$$

Here $T'(x)$ denotes the derivative of a function $T(x)$ and $x_Q$ the $x$-coordinate of a point $Q \in S^\star$. With coefficients $s, s_2, s_3$ of the polynomial $D(x)$, set $v = a(\ell-1) + 3(s^2 - 2s_2)$ and $w = 3as + 2b(\ell-1) + 5(s^3 - 3ss_2 + 3s_3)$. Then the Weierstrass equation for $\widetilde{E}$ is given by $y^2 = x^3 + \tilde{a}x + \tilde{b}$ with $\tilde{a} = a - 5v$ and $\tilde{b} = b - 7w$. (Fast algorithms are shown also in [6] to compute the isogeny $\phi$.)

**Kernel polynomials**

Partition the set $S^\star$ into two parts $S^+$ and $S^-$ such that $S^\star = S^+ \cup S^-$ and $S^- = \{-P : P \in S^+\}$. We consider the kernel polynomial

$$F(x) = \prod_{Q \in S^+} (x - x_Q) = x^d + tx^{d-1} + t_2 x^{d-2} + \cdots + t_d \tag{3}$$

with $d = \frac{\ell-1}{2}$, and it satisfies $D(x) = F(x)^2$. Schoof [30] studied the relation among coefficients $t$ and $t_i$'s of $F(x)$. Instead of working with $\widetilde{E}$, he works with the isomorphic curve $\widehat{E} : y^2 = x^3 + \hat{a}x + \hat{b}$ with $\hat{a} = \ell^4\tilde{a}$ and $\hat{b} = \ell^6\tilde{b}$. To $E$, we associate the reduced Weierstrass $\wp$-function by $\wp(z) = \frac{1}{z^2} + \sum_{k=1}^{\infty} c_k z^{2k}$ with $c_1 = -\frac{a}{5}$, $c_2 = -\frac{b}{7}$ and $c_k = \frac{3}{(k-2)(2k+3)} \sum_{j=1}^{k-2} c_j c_{k-1-j}$ for $k \geq 3$. Similarly, the function $\hat{\wp}$ for $\widehat{E}$ is defined using $\hat{a}$ and $\hat{b}$. Then the polynomial $F(x)$ satisfies

$$z^{\ell-1} F(\wp(z)) = \exp\left(-\frac{1}{2}tz^2 - \sum_{k=1}^{\infty} \frac{\hat{c}_k - \ell c_k}{(2k+1)(2k+2)} z^{2k+2}\right). \tag{4}$$

(This is by reduction of relations over $\mathbb{C}$, and the characteristic $p$ must be large enough to hold over $\mathbb{F}_q$.) From this equation, we can represent every coefficient $t_i$ with $t$, $c_k$'s and $\hat{c}_k$'s, and with $t$, $a$, $b$, $\tilde{a}$ and $\tilde{b}$ since $c_k$'s and $\hat{c}_k$'s are obtained from $a$, $b$, $\tilde{a}$ and $\tilde{b}$. (See [4, Chap. VII] for the first few coefficients of $F(x)$.) In particular, every $t_i$ can be represented as an element of the multivariate polynomial ring $\mathbb{F}_q[t, a, b, \tilde{a}, \tilde{b}]$ when we regard all $t$, $a$, $b$, $\tilde{a}$, $\tilde{b}$ as variables.

## 2.3 Modular polynomials

For every integer $\ell \geq 2$, there exists the modular polynomial $\Phi_\ell(X, Y) \in \mathbb{Z}[X, Y]$ to parameterize pairs of elliptic curves with a cyclic isogeny of degree $\ell$ between them. (See [32, Exercise 2.18, Chap. II].) For two elliptic curves $E$ and $\widetilde{E}$ over a finite field $\mathbb{F}_q$, there is an isogeny of degree $\ell$ from $E$ to $\widetilde{E}$ with cyclic kernel if and only if $\Phi_\ell(j(E), j(\widetilde{E})) = 0$. Every modular polynomial $\Phi_\ell(X, Y)$ is symmetric in each variable, and its integer coefficients become very large as $\ell$ increases. In particular, for a prime $\ell$, the modular polynomial $\Phi_\ell(X, Y)$ is equal to the form

$$X^{\ell+1} - X^\ell Y^\ell + Y^{\ell+1} + \sum_{i,j \leq \ell, i+j < 2\ell} a_{ij} X^i Y^j \tag{5}$$

with $a_{ij} \in \mathbb{Z}$, since there are precisely $\ell + 1$ subgroups of the $\ell$-torsion group of an elliptic curve $E$. (Such each subgroup corresponds to an isogenous curve of degree $\ell$ with a $j$-invariant, which is a zero of the polynomial $\Phi_\ell(X, j(E))$.)

# 3 Algebraic approaches for isogeny problems

In this section, we propose several algebraic approaches. Before presenting our approaches, let us define our setting of isogeny problem as below:

**Problem 1.** Let $\ell = \ell_0^e$ be a power of a small odd prime $\ell_0$. Suppose that there exists an isogeny of degree $\ell$ between elliptic curves $E$ and $\widetilde{E}$ over a finite field $\mathbb{F}_q$. Given $\ell$, $j = j(E)$ and $\tilde{j} = j(\widetilde{E})$, find an isogeny $\phi : E \to \widetilde{E}$ of degree $\ell$.

Compared to the general isogeny problem in [20], the isogeny degree $\ell$ is restricted to a prime power and it is given in our setting. Note that a solution of $\phi$ is not unique in general for the above problem. (In particular, the kernel of $\phi$ is not necessarily cyclic.) There are also many (compact) representations of a solution $\phi$ such as a chain of isogenies of low degrees, a sequence of $j$-invariants of intermediate curves, and a path in an isogeny graph between $E$ and $\widetilde{E}$. The meet-in-the-middle approach is a practical way to solve isogeny problems. For the above isogeny problem, it builds two trees of isogenies of prime degree $\ell_0$ from both the sides of $E$ and $\widetilde{E}$, respectively, and it finds a collision between the two trees to find the shortest path from $E$ to $\widetilde{E}$. While the meet-in-the-middle approach is a generic way for solving graph problems, we shall propose two algebraic approaches for solving the above isogeny problem.

## 3.1 First approach using modular polynomials

In this subsection, we present our first approach. Consider a chain of isogenies $\phi_k$ of prime degree $\ell_0$ from $E$ to $\widetilde{E}$ as

$$E \xrightarrow{\phi_1} E_1 \xrightarrow{\phi_2} E_2 \xrightarrow{\phi_3} \cdots \xrightarrow{\phi_{e-1}} E_{e-1} \xrightarrow{\phi_e} \widetilde{E}.$$

Let $j_k$ denote the $j$-invariant of every elliptic curve $E_k$ for $1 \le k \le e-1$. In this approach, we regard $j$-invariants $j_k$'s as *variables*, and consider a system of equations using modular polynomials (cf., recall that $j$ and $\tilde{j}$ are elements of $\mathbb{F}_q$.)

$$\begin{cases} \Phi_{\ell_0}(j, j_1) = 0, \\ \Phi_{\ell_0}(j_k, j_{k+1}) = 0 \quad (1 \le k \le e-2), \\ \Phi_{\ell_0}(j_{e-1}, \tilde{j}) = 0. \end{cases} \tag{6}$$

A solution of this system gives all $j$-invariants $j_k$'s of intermediate curves $E_k$.

Here we propose a method to solve the system (6) efficiently by Gröbner basis algorithms. (See textbooks [2, 13] for Gröbner basis computation.) We assume that the exponent $e$ of the isogeny degree $\ell$ is even with $e = 2e_0$ for a positive integer $e_0$ for simple discussion. We divide the system (6) into two parts like the meet-in-the-middle approach. In terms of Gröbner basis computation, we consider two ideals in different multivariate polynomial rings

$$I = \left\langle \Phi_{\ell_0}(j, j_1), \Phi_{\ell_0}(j_1, j_2), \ldots, \Phi_{\ell_0}(j_{e_0-1}, j_{e_0}) \right\rangle \subset \mathbb{F}_q[j_1, \ldots, j_{e_0}],$$
$$\widetilde{I} = \left\langle \Phi_{\ell_0}(j_{e_0}, j_{e_0+1}), \Phi_{\ell_0}(j_{e_0+1}, j_{e_0+2}), \ldots, \Phi_{\ell_0}(j_{e-1}, \tilde{j}) \right\rangle \subset \mathbb{F}_q[j_{e_0}, \ldots, j_{e-1}].$$

Both ideals $I$ and $\widetilde{I}$ are zero-dimensional since $j, \tilde{j} \in \mathbb{F}_q$. In particular, the dimensions of $\mathbb{F}_q$-vector spaces $\mathbb{F}_q[j_1, \ldots, j_{e_0}]/I$ and $\mathbb{F}_q[j_{e_0}, \ldots, j_{e-1}]/\widetilde{I}$ are both at most $(\ell_0 + 1)^{e_0}$ due to the form of the modular polynomial (see Equation (5)). Moreover, the above generators give a Gröbner basis for the ideal $I$ (resp., the ideal $\widetilde{I}$) with the lex term order with respect to $j_{e_0} \succ \cdots \succ j_2 \succ j_1$ (resp., $j_{e_0} \succ \cdots \succ j_{e-2} \succ j_{e-1}$). Then we can efficiently compute minimal polynomials $g$ and $\tilde{g}$ of the variable $j_{e_0}$ with respect to ideals $I$ and $\widetilde{I}$, respectively, by using the FGLM algorithm [17]. (In this case, simple linear algebra might be more efficient since degrees of $g$ and $\tilde{g}$ are known. See Remark 3.1 and Appendix A.1 below for details.) By the GCD computation over the univariate polynomial ring $\mathbb{F}_q[j_{e_0}]$, we obtain a common root of two minimal polynomials $g$ and $\tilde{g}$. Such a common root gives a solution of $j_{e_0}$.

Once a solution of $j_{e_0}$ is found, the isogeny problem is divided into two isogeny problems of smaller degree $\ell_0^{e_0} = \sqrt{\ell}$ (i.e., a divide-and-conquer strategy). By repeating this procedure, we can solve the whole isogeny problem.

**Remark 3.1.** Set $g_1 = \Phi_{\ell_0}(j, j_1)$, and let $g_k$ denote the minimal polynomial of the variable $j_k$ with respect to the ideal $\langle \Phi_{\ell_0}(j, j_1), \ldots, \Phi_{\ell_0}(j_{k-1}, j_k) \rangle$ for every $2 \le k \le e_0$. Putting $G_k(X) := \Phi_{\ell_0^k}(j, X)$, we have *generically*

$$g_k(j_k) = G_{k-2}(j_k) G_k(j_k)$$

for $3 \le k \le e_0$. (We verified by experiments that it holds in most cases.) For our target minimal polynomial $g = g_{e_0}$, it seems the best in performance to compute $g_k$ from $g_{k-1}$ and $\Phi_{\ell_0}(j_{k-1}, j_k)$ recursively for $2 \le k \le e_0$, see Appendix A.1. In a similar way, we can compute another minimal polynomial $\tilde{g}$ efficiently.

The time complexity of the first approach shall be analyzed in Appendix A.1. The complexity depends on an algorithm for computing minimal polynomials, such as the FGLM algorithm and the GCD computation. Note that our first approach is not better than the meet-in-the-middle approach in time complexity.

#### Possible improvement (3-section method)

We introduce a possible improvement for the first approach. Our idea is to divide the system (6) into *three parts*. (cf., The original strategy is regarded as the "2-section method".) With two parameters $e_1$ and $e_2$ satisfying $1 < e_1 < e_0 < e_2 < e$ and $e_1 \approx e - e_2$, consider two ideals in different multivariate polynomial rings

$$I_{[1:e_1]} = \left\langle \Phi_{\ell_0}(j, j_1), \Phi_{\ell_0}(j_1, j_2), \ldots, \Phi_{\ell_0}(j_{e_1-1}, j_{e_1}) \right\rangle,$$
$$\widetilde{I}_{[e_2:e-1]} = \left\langle \Phi_{\ell_0}(j_{e_2}, j_{e_2+1}), \Phi_{\ell_0}(j_{e_2+1}, j_{e_2+2}), \ldots, \Phi_{\ell_0}(j_{e-1}, \tilde{j}) \right\rangle.$$

As in the 2-section method, we use the lex term order with $j_{e_1} \succ \cdots \succ j_2 \succ j_1$ (resp., $j_{e_2} \succ j_{e_2+1} \succ \cdots \succ j_{e-1}$) for the zero-dimensional ideal $I_{[1:e_1]}$ (resp., $\widetilde{I}_{[e_2:e-1]}$). Then the ideal $I_{[1:e_1]}$ (resp., $\widetilde{I}_{[e_2:e-1]}$) includes a polynomial $g(j_{e_1})$ (resp., $\tilde{g}(j_{e_2})$) such that the set of its roots contains the roots of $\Phi_{\ell'}(j, j_{e_1})$ of level $\ell' = \ell_0^{e_1}$ (resp., $\Phi_{\tilde{\ell}'}(j_{e_2}, \tilde{j})$ of level $\tilde{\ell}' = \ell_0^{e-e_2}$). Namely, the polynomials $g$ and $\tilde{g}$ are minimal polynomials for zero-dimensional ideals $I_{[1:e_1]}$ and $\widetilde{I}_{[e_2:e-1]}$, respectively. We then consider a new ideal

$$J_{[e_1:e_2]} = \left\langle g(j_{e_1}), \Phi_{\ell_0}(j_{e_1}, j_{e_1+1}), \ldots, \Phi_{\ell_0}(j_{e_2-1}, j_{e_2}), \tilde{g}(j_{e_2}) \right\rangle.$$

For this zero-dimensional ideal, we use the grevlex term order with $j_{e_1} \prec j_{e_2} \prec j_{e_1+1} \prec j_{e_2-1} \prec \cdots \prec j_{e_0}$ to find intermediate $j$-invariants from $j_{e_1}$ to $j_{e_2}$. With these $j$-invariants, we obtain the other $j$-invariants $j_1, \ldots, j_{e_1-1}$ and $j_{e_2+1}, \ldots, j_{e-1}$ as in the 2-section method. The time complexity of this 3-section method might be improved so that it is better than that of the 2-section method.

## 3.2 Second approach using kernel polynomials

In this subsection, we present our second approach for solving the isogeny problem of prime power degree $\ell = \ell_0^e$. As in the first approach, we assume that the exponent $e$ of the isogeny degree is even with $e = 2e_0$ for simple discussion. In this approach, we take an intermediate elliptic curve $E_0$ between $E$ and $\widetilde{E}$ such that there exist two normalized isogenies $\varphi$ and $\tilde{\varphi}$ of same degree $\ell' = \ell_0^{e_0}$ as

$$E \xrightarrow{\varphi} E_0 \xleftarrow{\tilde{\varphi}} \widetilde{E}.$$

Let $a, b$ (resp., $\tilde{a}, \tilde{b}$) denote Weierstrass coefficients in $\mathbb{F}_q$ defining the initial curve $E$ (resp., the final curve $\widetilde{E}$). These coefficients can be chosen with two $j$-invariants $j$ and $\tilde{j}$ of $E$ and $\widetilde{E}$. In contrast, we use two variables $a_0, b_0$ as Weierstrass coefficients defining the intermediate curve $E_0$. (That is, $E_0 : y^2 = x^3 + a_0 x + b_0$.)

Like Equation (3), consider the kernel polynomial

$$F(x) = x^m + tx^{m-1} + t_2x^{m-2} + \cdots + t_m,$$

obtained from the isogeny $\varphi : E \to E_0$ with $m = \frac{\ell'-1}{2}$. Now we regard $t$ as an additional variable. Then we can represent the other coefficients $t_2, \ldots, t_m$ of $F(x)$ as elements of the multivariate polynomial ring $R = \mathbb{F}_q[t, a_0, b_0]$ by Schoof's work [30], described in Subsection 2.2 (recall $a, b \in \mathbb{F}_q$). In other words, we regard the polynomial $F(x)$ as an element of $R[x]$. Furthermore, like Equation (1), we can reconstruct the isogeny $\varphi : E \to E_0$ with rational functions over $R[x]$ as

$$\varphi(x, y) = \left(\frac{N(x)}{D(x)}, y\left(\frac{N(x)}{D(x)}\right)'\right)$$

for any point $(x, y) \in E$, by letting $D(x) = F(x)^2$. Here the rational function $T(x) = \frac{N(x)}{D(x)} \in R(x)$ is given by Equation (2). Since $\varphi(x, y) \in E_0$ for any point $(x, y) \in E$, we clearly have the relation

$$(x^3 + ax + b)T'(x)^2 = T(x)^3 + a_0T(x) + b_0. \tag{7}$$

By expanding this relation with respect to $x$, we obtain a set of equations $S$ defined over $R$. (Every coefficient of $x^i$ in (7) corresponds to an equation in $S$.)

Similarly, we consider the kernel polynomial

$$\widetilde{F}(\tilde{x}) = \tilde{x}^m + \tilde{t}\tilde{x}^{m-1} + \tilde{t}_2\tilde{x}^{m-1} + \cdots + \tilde{t}_m$$

obtained from another isogeny $\tilde{\varphi} : \widetilde{E} \to E_0$. By regarding $\tilde{t}$ as another variable, we can also regard $\widetilde{F}(\tilde{x})$ as an element of the polynomial ring $\widetilde{R}[\tilde{x}]$ with $\widetilde{R} = \mathbb{F}_q[\tilde{t}, a_0, b_0]$. We can also reconstruct the isogeny $\tilde{\varphi}$ as $\tilde{\varphi}(\tilde{x}, \tilde{y}) = (\widetilde{T}(\tilde{x}), \tilde{y}\widetilde{T}'(\tilde{x}))$ for any point $(\tilde{x}, \tilde{y}) \in \widetilde{E}$, for some rational function $\widetilde{T}(\tilde{x}) \in \widetilde{R}(\tilde{x})$. Similarly to the previous paragraph, the relation $(\tilde{x}^3 + \tilde{a}\tilde{x} + \tilde{b})\widetilde{T}'(\tilde{x})^2 = \widetilde{T}(\tilde{x})^3 + a_0\widetilde{T}(\tilde{x}) + b_0$ gives another set of equations $\widetilde{S}$ defined over the ring $\widetilde{R}$.

Finally, we solve the system of equations in the union set $S \cup \widetilde{S}$ by using Gröbner basis algorithms over the multivariate polynomial ring $\mathbb{F}_q[t, \tilde{t}, a_0, b_0]$. (We used the grevlex term order with $t \succ \tilde{t} \succ a_0 \succ b_0$ in our experiments.) A solution of $(t, \tilde{t}, a_0, b_0)$ determines the Weierstrass equation for the intermediate curve $E_0$, and also two kernels of isogenies $\varphi$ and $\tilde{\varphi}$.

**Remark 3.2.** While the first approach requires many variables as the exponent $e$ increases, this approach always requires four variables $t, \tilde{t}, a_0, b_0$. On the other hand, as $e$ increases, total degrees of equations become large in this approach, while total degrees do not change in the first approach due to the use of the modular polynomial of same level $\ell_0$. In contrast, one can consider several variants of this approach. As the simplest variant, we directly consider the kernel polynomial of the isogeny $\phi : E \to \widetilde{E}$ (without taking any intermediate curve between $E$ and $\widetilde{E}$). This variant requires only one variable, but total degrees of equations become very large as $e$ increases. On the other hand, we consider multiple intermediate curves as a variant. This variant requires many variables as the number of intermediate curves increases. As a result, the meet-in-the-middle type described in this subsection seems the best in performance from our preliminary experiments.

As in the first approach, the time complexity of the second approach shall be analyzed in Appendix A.2. In particular, the analysis shows that the second approach can never be asymptotically faster than the first one.

# 4 Experiments

In this section, we report experimental results of our algebraic approaches for solving the isogeny problem of prime power degrees (see Problem 1 for the problem).

## 4.1 Experiment parameters

For Problem 1, we fix parameters $p = 2^{250} \cdot 3^{159} - 1$, $q = p^2$ and $E : y^2 = x^3 + x$, extracted from SIKE-p503 parameters [21]. (The extension field $\mathbb{F}_{p^2}$ is represented as $\mathbb{F}_p[z]/(z^2+1)$.) The initial curve $E$ is a supersingular elliptic curve defined over $\mathbb{F}_{p^2}$ having $\#E(\mathbb{F}_{p^2}) = (p+1)^2 = (2^{250} \cdot 3^{159})^2$ and $j = j(E) = 1728$. We also take 3-powers $\ell = 3^e$ (i.e., $\ell_0 = 3$) as isogeny degrees for even exponents $e = 2e_0$. (cf., SIKE uses a combination of isogenies of degrees 2 and 3.) We follow [21] to generate the final supersingular curve $\widetilde{E}$, isogenous to $E$ of degree $\ell$.

## 4.2 Implementation details

Here we describe details of implementation for our algebraic approaches and the meet-in-the-middle approach with MAGMA [5], a computational algebra system.

For our first approach by 2-section and 3-section methods, we used a combination of the modular polynomials $\Phi_N(X, Y)$ for $N = 3, 3^2, 3^3$, which are pre-computed in MAGMA, in order to obtain the minimal polynomials $g, \tilde{g}$. For example, for $\ell = 3^{10}$, we computed Gröbner bases for $I = \langle \Phi_{3^3}(j, j_3), \Phi_{3^2}(j_3, j_5) \rangle$, $\widetilde{I} = \langle \Phi_{3^2}(j_5, j_7), \Phi_{3^3}(j_7, \tilde{j}) \rangle$ with the MAGMA command `GroebnerBasis` to obtain $g, \tilde{g} \in \mathbb{F}_{p^2}[j_5]$, then computed the GCD of $g, \tilde{g}$ with the MAGMA commands `GCD` for the 2-section method.

For our second approach, we used Equation (4) to represent two kernel polynomials $F(x)$ and $\widetilde{F}(\tilde{x})$ of isogenies $\varphi$ and $\tilde{\varphi}$, described in Subsection 3.2, as polynomials over $\mathbb{F}_q[t, \tilde{t}, a_0, b_0]$. (As an alternative of (4), fast algorithms are introduced in [6] for computing the kernel polynomial of an isogeny, but they require very fast exponentiation.) We also used the grevlex term order with $t \succ \tilde{t} \succ a_0 \succ b_0$ to find a solution $(t, \tilde{t}, a_0, b_0)$ from the union set of equations $S \cup \widetilde{S}$.

For the meet-in-the-middle approach, we construct two sets $J$ and $\widetilde{J}$ of sequences $(j, j_1, \ldots, j_{e_0})$ and $(\tilde{j}, \tilde{j}_1, \ldots, \tilde{j}_{e_0})$ of $j$-invariants of elliptic curves $E_k$ and $\widetilde{E}_k$, respectively. (Recall $j = j(E)$ and $\tilde{j} = j(\widetilde{E})$.) Here $E_{k-1}$ and $E_k$ (resp., $\widetilde{E}_{k-1}$ and $\widetilde{E}_k$) are isogenous of degree 3 for every $1 \le k \le e_0$, where we set $E_0 = E$ (resp., $\widetilde{E}_0 = \widetilde{E}$) for convenience. To construct sequences in $J$ and $\widetilde{J}$, we use the modular polynomial of level 3. For example, we add each solution of $\Phi_3(j_{k-1}, x)$ to a sequence $(j, j_1, \ldots, j_{k-1})$ of length $k$. (We used the MAGMA command `Roots` to find a solution.) In constructing such sequences, we remove a sequence whose ending point already appeared in the other sequences, in order to reduce the sizes of two sets $J$ and $\widetilde{J}$. In our experiments, it terminates when we find a pair of two sequences of $J$ and $\widetilde{J}$ satisfying $j_{e_0} = \tilde{j}_{e_0}$ (i.e., a collision).

## 4.3 Experimental results

In Table 1, we summarize average running times of our two approaches and the meet-in-the-middle approach for solving the isogeny problem of degrees $\ell = 3^e$ with even $e$ from $e = 6$ up to 14. Specifically, we measured the running time of every approach until it finds the $j$-invariant or the Weierstrass coefficients of an intermediate curve between two isogenous curves $E$ and $\widetilde{E}$. We also experimented 5 times for every parameter set. All the experiments were performed using MAGMA 2.24-5 on 4.20 GHz Intel Core i7 CPU with 16 GByte memory. From Table 1, the first approach is the fastest for isogeny degrees up to $\ell = 3^{10}$. In contrast, our second approach is costly and it did not terminate in one day for degrees larger than $\ell = 3^8$. For degrees larger than $\ell = 3^{10}$, the meet-in-the-middle approach is faster than our algebraic approaches. With respect to the memory usage, our first approach by the 2-section method requires about 64, 221 and 526 MByte for $\ell = 3^{10}, 3^{12}$ and $3^{14}$, respectively, while the meet-in-the-middle approach requires about 24, 26 and 32 MByte for the same isogeny degrees.

**Table 1:** Average running times (seconds) of our two approaches and the meet-in-the-middle approach for the isogeny problem of degrees $\ell = 3^e$ on supersingular elliptic curves over $\mathbb{F}_{p^2}$ with 503-bit prime $p = 2^{250} \cdot 3^{159} - 1$ (These parameters are from SIKE-p503 [21], and the initial curve $E$ is given by $y^2 = x^3 + x$ over $\mathbb{F}_{p^2}$)

| Isogeny degrees | Our first approach | | Our second approach | Meet-in-the-middle approach |
|---|---|---|---|---|
| $\ell = 3^e$ | 2-section | 3-section | | |
| $3^6 = 729$ | 0.11 | 0.15 | 336.42 | 2.93 |
| $3^8 = 6561$ | 0.19 | 0.45 | > 1 day | 9.61 |
| $3^{10} = 59049$ | 7.98 | 5.63 | – | 31.09 |
| $3^{12} = 531441$ | 287.77 | 159.39 | – | 96.75 |
| $3^{14} = 4782969$ | 5071.16 | 2725.01 | – | 292.82 |

# 5 Concluding remarks

In this paper, we proposed two algebraic approaches for solving isogeny problems of prime power degrees. The first one is a straightforward way using modular polynomials of small prime levels. The second one is a more complex way, which uses kernel polynomials of isogenies based on [30] and reconstructs the isogenies with Vélu's formulae [6, 39]. From the analysis in Appendix A, our approaches are not asymptotically faster than the meet-in-the-middle approach. However, our experiments showed that the first approach is faster than the meet-in-the-middle approach for isogeny degrees up to $\ell = 3^{10}$ over supersingular elliptic curves of SIKE-p503 [21] with a single classical computer. On the one hand, our first approach is applicable in the collision search step of the meet-in-the-middle approach. Such combination could make it faster in practice and reduce the memory size of the meet-in-the-middle approach. As future work, we would like to use the combination for solving isogeny problems of large degrees.

**Acknowledgement:** This work was supported by JSPS KAKENHI Grant Numbers 18H05836, 19K21026, and 19K22847, Japan

# References

[1] Gora Adj, Daniel Cervantes-Vázquez, Jesús-Javier Chi-Domínguez, Alfred Menezes and Francisco Rodríguez-Henríquez, On the cost of computing isogenies between supersingular elliptic curves, in: *Selected Areas in Cryptography–SAC 2018*, Lecture Notes in Computer Science 11349, Springer, pp. 322–343, 2018.

[2] Thomas Becker and Volker Weispfenning, *Gröbner bases*, Graduate Texts in Mathematics 141, Springer, 1993.

[3] Jean-François Biasse, David Jao and Anirudh Sankar, A quantum algorithm for computing isogenies between supersingular elliptic curves, in: *Progress in Cryptology–INDOCRYPT 2014*, Lecture Notes in Computer Science 8885, Springer, pp. 428–442, 2014.

[4] Ian F Blake, Gadiel Seroussi and Nigel Smart, *Elliptic curves in cryptography*, 265, Cambridge university press, 1999.

[5] Wieb Bosma, John Cannon and Catherine Playoust, The Magma algebra system. I. The user language, *J. Symbolic Comput.* **24** (1997), 235–265, Computational algebra and number theory (London, 1993).

[6] Alin Bostan, François Morain, Bruno Salvy and Éric Schost, Fast algorithms for computing isogenies between elliptic curves, *Mathematics of Computation* **77** (2008), 1755–1778.

[7] Alessio Caminata and Elisa Gorla, Solving multivariate polynomial systems and an invariant from commutative algebra, *arXiv preprint arXiv:1706.06319* (2017).

[8] Wouter Castryck, Tanja Lange, Chloe Martindale, Lorenz Panny and Joost Renes, CSIDH: An efficient post-quantum commutative group action, in: *Advances in Cryptology–ASIACRYPT 2018*, Lecture Notes in Computer Science 11274, Springer, pp. 395–427, 2018.

[9] Denis X Charles, Kristin E Lauter and Eyal Z Goren, Cryptographic hash functions from expander graphs, *Journal of Cryptology* **22** (2009), 93–113.

[10] Andrew Childs, David Jao and Vladimir Soukharev, Constructing elliptic curve isogenies in quantum subexponential time, *Journal of Mathematical Cryptology* **8** (2014), 1–29.

[11] Anamaria Costache, Brooke Feigon, Kristin Lauter, Maike Massierer and Anna Puskás, Ramanujan graphs in cryptography, *IACR ePrint 2018/593* (2018).
[12] Craig Costello, Patrick Longa, Michael Naehrig, Joost Renes and Fernando Virdia, Improved Classical Cryptanalysis of the Computational Supersingular Isogeny Problem, *IACR ePrint 2019/298* (2019).
[13] David Cox, John Little and Donal O'Shea, *Ideals, varieties, and algorithms: An introduction to computational algebraic geometry and commutative algebra*, Springer Science & Business Media, 2013.
[14] Luca De Feo, David Jao and Jérôme Plût, Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies, *Journal of Mathematical Cryptology* **8** (2014), 209–247.
[15] Kirsten Eisenträger, Sean Hallgren, Kristin Lauter, Travis Morrison and Christophe Petit, Supersingular isogeny graphs and endomorphism rings: reductions and solutions, in: *Advances in Cryptology–EUROCRYPT 2018*, Lecture Notes in Computer Science 10822, Springer, pp. 329–368, 2018.
[16] Noam D Elkies et al., Elliptic and modular curves over finite fields and related computational issues, *AMS IP STUDIES IN ADVANCED MATHEMATICS* **7** (1998), 21–76.
[17] Jean-Charles Faugere, Patrizia Gianni, Daniel Lazard and Teo Mora, Efficient computation of zero-dimensional Gröbner bases by change of ordering, *Journal of Symbolic Computation* **16** (1993), 329–344.
[18] Steven D Galbraith, Constructing isogenies between elliptic curves over finite fields, *LMS Journal of Computation and Mathematics* **2** (1999), 118–138.
[19] Steven D Galbraith, Christophe Petit, Barak Shani and Yan Bo Ti, On the security of supersingular isogeny cryptosystems, in: *Advances in Cryptology–ASIACRYPT 2016*, Lecture Notes in Computer Science 10031, Springer, pp. 63–91, 2016.
[20] Steven D Galbraith and Frederik Vercauteren, Computational problems in supersingular elliptic curve isogenies, *Quantum Information Processing* **17** (2018), 265.
[21] D Jao, R Azarderakhsh, M Campagna, C Costello, L DeFeo, B Hess, A Jalali, B Koziel, B LaMacchia, P Longa et al., *SIKE: Supersingular isogeny key encapsulation. Submission to the NIST Standardization Process on Post-Quantum Cryptography*, 2017.
[22] David Jao and Luca De Feo, Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies, in: *Post-Quantum Cryptography–PQCrypto 2011*, Lecture Notes in Computer Science 7071, Springer, pp. 19–34, 2011.
[23] David Jao and Vladimir Soukharev, Isogeny-based quantum-resistant undeniable signatures, in: *Post-Quantum Cryptography–PQCrypt 2014*, Lecture Notes in Computer Science 8772, Springer, pp. 160–179, 2014.
[24] Samuel Jaques and John M Schanck, Quantum cryptanalysis in the RAM model: Claw-finding attacks on SIKE, *IACR Cryptology ePrint Archive 2019/103* (2019).
[25] Neal Koblitz, Elliptic curve cryptosystems, *Mathematics of Computation* **48** (1987), 203–209.
[26] Chloe Martindale and Lorenz Panny, How to not break SIDH, *IACR ePrint 2019/558* (2019).
[27] Victor S Miller, Use of elliptic curves in cryptography, in: *Advances in Cryptology–CRYPTO 1985*, Lecture Notes in Computer Science 218, pp. 417–426, Springer, 1985.
[28] National Institute of Standards and Technology (NIST), *NISTIR 8240: Status Report on the First Round of the NIST Post-Quantum Cryptography Standardization Process*.
[29] Alexander Rostovtsev and Anton Stolbunov, Public-key cryptosystem based on isogenies, *IACR Cryptology ePrint Archive 2006/145* (2006).
[30] René Schoof, Counting points on elliptic curves over finite fields, *Journal de théorie des nombres de Bordeaux* **7** (1995), 219–254.
[31] Peter W. Shor, Algorithms for quantum computation: Discrete logarithms and factoring, in: *Symposium on Foundations of Computer Science (FOCS 1994)*, pp. 124–134, IEEE, 1994.
[32] Joseph H Silverman, *Advanced topics in the arithmetic of elliptic curves*, Graduate Texts in Mathematics 151, Springer-Verlag New York, 1994.
[33] Joseph H Silverman, *The arithmetic of elliptic curves*, second ed, Graduate Texts in Mathematics 106, Springer Science & Business Media, 2009.
[34] Anton Stolbunov, Constructing public-key cryptographic schemes based on class group action on a set of isogenous elliptic curves, *Advances in Mathematics of Communications* **4** (2010), 215–235.
[35] Xi Sun, Haibo Tian and Yumin Wang, Toward quantum-resistant strong designated verifier signature from isogenies, in: *Intelligent Networking and Collaborative Systems–INCoS 2012*, IEEE, pp. 292–296, 2012.
[36] Seiichiro Tani, Claw finding algorithms using quantum walk, *Theoretical Computer Science* **410** (2009), 5285–5297.
[37] John Tate, Endomorphisms of abelian varieties over finite fields, *Inventiones mathematicae* **2** (1966), 134–144.
[38] Paul C Van Oorschot and Michael J Wiener, Parallel collision search with cryptanalytic applications, *Journal of Cryptology* **12** (1999), 1–28.
[39] Jacques Vélu, Isogénies entre courbes elliptiques, *CR Acad. Sci. Paris Sér. AB* **273** (1971), 238–241.

# A Time complexity analysis

In this appendix, we analyze time complexities of our algebraic approaches. Here we use the same notation as in Section 3. In this complexity analysis, we assume that $\ell$ and $\ell_0$ are constants, and we set $e$ and $e_0$ as asymptotic parameters to estimate time complexities.

## A.1 For the first approach

The first approach proceeds with the below two steps:

*Step 1.* Compute minimal polynomials $g$ and $\tilde{g}$ of the variable $j_{e_0}$ with respect to ideals $I$ and $\tilde{I}$, respectively.
*Step 2.* By computing the GCD of $g$ and $\tilde{g}$ over the univariate polynomial ring $\mathbb{F}_q[j_{e_0}]$, we obtain a common root in $\mathbb{F}_q$ of the minimal polynomials.

There are several methods to compute Step 1, such as the FGLM algorithm and the GCD computation. In the below, we estimate costs of the first approach using several different methods for Step 1.

#### Using the FGLM algorithm for Step 1

#### Applying the FGLM algorithm directly to the whole ideal

The cost of Step 1 by using the FGLM algorithm for the ideal $I$ is estimated as $O\left(e_0((\ell_0+1)^{e_0})^3\right) = O\left(e_0(\ell_0+1)^{3e_0}\right)$, since $\dim_{\mathbb{F}_q}\mathbb{F}_q[j_1,\dots,j_{e_0}]/I \le (\ell_0+1)^{e_0}$ and the number of variables is $e_0$. Similarly, the FGLM algorithm for another ideal $\widetilde{I}$ requires the same cost. Hence the complexity of Step 1 is

$$O(2e_0(\ell_0+1)^{3e_0}) = O(e_0(\ell_0+1)^{3e_0}) = O(e(\ell_0+1)^{3e/2}) = O(e\ell_0^{\frac{3}{2}e}).$$

#### Using the normal form computation and simple linear algebra only

Instead of directly using of the FGLM algorithm, we can compute minimal polynomials $g$ and $\tilde{g}$ based on the below well-known fact:

**Lemma A.1.** *Let $K$ be a field, $I$ a zero-dimensional ideal of $S = K[X_1,\dots,X_n]$, and $G$ a Gröbner basis of $I$ with respect to a fixed term order $\succ$. Let $f \in S$ and $g(t) \in K[t]$ with leading coefficient 1. We denote by $m_f(t)$ the minimal polynomial of $f$ with respect to the ideal $I$. Then $g(t) = m_f(t)$ if and only if $g(t)$ satisfies the following two conditions:*

1. $\mathrm{NF}_I(g(f)) = 0$, *where* $\mathrm{NF}_I(g(f))$ *denotes the normal form of* $g(f) \in S$ *with respect to* $G$.
2. *For any* $h(t) \in K[t]$ *with* $\mathrm{NF}_I(h(f)) = 0$, *we have* $\deg(g) \le \deg(h)$.

Recall from Remark 3.1 that we have generically $g = G_{k-2}G_k$ for $3 \le k \le e_0$, where we set $G_k(j_k) := \Phi_{\ell^k}(j, j_k)$ for $1 \le k \le e_0$. (Until the end of this appendix, we assume this generic condition.) Since the degree of $g$ equals to $d := \deg(g) = \deg(G_{e_0-2}) + \deg(G_{e_0}) = (\ell_0+1)\ell_0^{e_0-3} + (\ell_0+1)\ell_0^{e_0-1} = O(\ell_0^{e_0})$, it suffices to compute $\mathrm{NF}_I(X^k)$ for $0 \le k \le d = O(\ell_0^{e_0})$ with $X := j_{e_0}$, and solve the linear equation $\mathrm{NF}_I(X^d) + \sum_{k=0}^{d-1} c_k \mathrm{NF}_I(X^k) = 0$ over $\mathbb{F}_q$, and then we have $g = \sum_{k=0}^{d} c_k X^k$ by Lemma A.1. The cost of computing each normal form $\mathrm{NF}_I(X^k)$ is $O((\ell_0+1)^{2e_0}) = O(\ell_0^{2e_0})$. Hence the cost of computing $\mathrm{NF}_I(X^k)$ for $0 \le k \le d$ is $O((\ell_0^{e_0}\ell_0^{2e_0})) = O(\ell_0^{3e_0})$. The cost of solving the linear equation is not higher than $O(\ell_0^{3e_0})$ since it is $O(d^3) = O(\ell_0^{3e_0})$ if one uses the Gaussian elimination naively. Similarly, computing $\tilde{g}$ requires $O(\ell_0^{3e_0})$. Here we estimate that the cost of Step 1 is $O(2\ell_0^{3e_0}) = O(\ell_0^{3e_0}) = O(\ell_0^{\frac{3}{2}e})$. In contrast, the cost of Step 2 mainly depends on the computation of the GCD of $g$ and $\widetilde{g}$ over $\mathbb{F}_q$. Note that the complexity of computing the GCD of two univariate polynomials of degree at most $d$ over $\mathbb{F}_q$ is $O(d^2)$ arithmetic over $\mathbb{F}_q$ (if one uses classical polynomial arithmetic). Since both degrees of

$g$ and $\tilde{g}$ are equal to $(\ell_0+1)\ell_0^{e_0-3}+(\ell_0+1)\ell_0^{e_0-1}=O(\ell_0^{e_0})$, we estimate that the cost of Step 2 is $O(\ell_0^{2e_0})=O(\ell_0^{e})$. Summing up, we *roughly* estimate that the complexity of the first approach is

$$O(e\ell_0^{\frac{3}{2}e})+O(\ell_0^{e})=O(e\ell_0^{\frac{3}{2}e})$$

if the FGLM algorithm is applied. Moreover, the complexity becomes

$$O(\ell_0^{\frac{3}{2}e})+O(\ell_0^{e})=O(\ell_0^{\frac{3}{2}e})$$

if only the normal form computation and simple linear algebra is used. For the parameter in our experiments (i.e., $\ell_0=3$), the cost is $O(e_0 3^{3e_0})$ or $O(3^{3e_0})$.

### Using sequential GCD for Step 1

As mentioned in Remark 3.1, it is practical to compute minimum polynomials $g_k$ together with $G_k(X):=\Phi_{\ell_0^k}(j,X)$ sequentially for $2\le k\le e_0$. (Note the target minimal polynomial $g$ is given by $g_{e_0}$.) The minimal polynomial $g_k$ is the lowest degree polynomial in $j_k$ belonging to the ideal generated by $G_{k-1}(j_{k-1})=\Phi_{\ell_0^{k-1}}(j,j_{k-1})$ and $\Phi_{\ell_0}(j_{k-1},j_k)$, that is

$$\left\langle G_{k-1}(j_{k-1}),\Phi_{\ell_0}(j_{k-1},j_k)\right\rangle\cap\mathbb{F}_q[j_k]=\left\langle g_k(j_k)\right\rangle.$$

Once $g_k$ is computed, we obtain $G_k$ as the quotient of $g_k$ divided by $G_{k-2}$ since $g_k(X)=G_{k-2}(X)G_k(X)$. Here we estimate the cost of computing $G_k$. Regarding $G_{k-1}(j_{k-1})$ and $\Phi_{\ell_0}(j_{k-1},j_k)$ as polynomials in $j_{k-1}$ over $\mathbb{F}_q[j_k]$, the cost of computing $G_k$ can be upper-bounded by that of computing their greatest common divisor with respect to $j_{k-1}$ by the Euclidean algorithm with pseudo division. This is due to the following reason; In the below steps (1) and (2), the target minimal polynomial can be computed by eliminating $j_{k-1}$ from $G'_{k-1}(j_{k-1},j_k)$ and $\Phi_{\ell_0}(j_{k-1},j_k)$. The Euclidean algorithm with pseudo division corresponds to eliminating $j_{k-1}$ by using leading coefficients as polynomials over $\mathbb{F}_q[j_k]$, whereas the Gröbner basis computation corresponds to eliminating $j_{k-1}$ by using head terms as polynomials in $\mathbb{F}_q[j_{k-1},j_k]$ with respect to the lexicographical term order with $j_{k-1}\succ j_k$. In each elimination step of the Euclidean algorithm with pseudo division, the elimination of a term is just the same as computing an $S$-polynomial (or its multiple by a monomial). To make our estimation simple, we can use so called the subresultant GCD algorithm which is the Euclidean algorithm with pseudo division and *principal subresultant coefficient (PSC)* and computes the resultant, that is, not the minimal polynomial but the characteristic polynomial $(G_{k-2}(j_k))^{\ell_0}G_k(j_k)$ of $j_k$ with respect to $\langle G_{k-1}(j_{k-1}),\Phi_{\ell_0}(j_{k-1},j_k)\rangle$.

Now we estimate the cost of computing the resultant of $G_{k-1}$ and $\Phi_{\ell_0}(j_{k-1},j_k)$ with respect to $j_{k-1}$ by the subresultant GCD, where we may assume that the remainder sequence is normal. Note that $\deg_{j_{k-1}}(G_{k-1}(j_{k-1}))=(\ell_0+1)\ell_0^{k-2}=O(\ell_0^{k-1})$, and $\deg_{j_{k-1}}(\Phi_{\ell_0}(j_{k-1},j_k))=\ell_0+1$. The final target $G_k$ is computed by conducting two procedures below:

1. Using the division algorithm, compute $G'_{k-1}(j_{k-1},j_k):=G_{k-1}(j_{k-1})\bmod\Phi_{\ell_0}(j_{k-1},j_k)$, where $\Phi_{\ell_0}(j_{k-1},j_k)$ is a monic polynomial over $\mathbb{F}_q[j_k]$ of degree $\ell_0+1$ in $j_{k-1}$. Note that $G'_{k-1}$ is a univariate polynomial over $\mathbb{F}_q[j_k]$ in variable $j_{k-1}$ of degree $\le\ell_0$. More specifically, proceed with the following (we use simple notation $A$, $B$, $C$ for polynomials and $N$ for degrees):
   - (1-1) Set $A\leftarrow\Phi_{\ell_0}(j_{k-1},j_k)$, and $B\leftarrow G_{k-1}$.
   - (1-2) Set $N\leftarrow\deg_{j_{k-1}}(B)$, and $C\leftarrow$ (the coefficient of $j_{k-1}^N$ in $B$).
   - (1-3) Compute $B-j_{k-1}^{N-(\ell_0+1)}CA$, and set $B\leftarrow B-j_{k-1}^{N-(\ell_0+1)}CA$.
   - (1-4) If $N\le\ell_0$, stop the loop. Otherwise set $N\leftarrow\deg_{j_{k-1}}(B)$ and go back to (1-2).

   The resulting polynomial $B$ coincides with the target $G'_{k-1}$.
2. Compute the resultant of $G'_{k-1}(j_{k-1},j_k)$ and $\Phi_{\ell_0}(j_{k-1},j_k)$ by the subresultant GCD, and then recover $G_k$ from the resultant.

We estimate the cost of the procedure (1). Let $n$ be the number of times the loop executes, and let $(B_s,N_s,C_s)$ for $1\le s\le n$ be tuples of the successive values $B_s$ of $B$, $N_s$ of $N$ and $C_s$ of $C$ respectively. We denote by $M_s$

the maximum degree of the coefficients of $j_{k-1}^i$ for $0 \le i \le N_s - 1$ in $B_s$. Note that $B_1 = G_{k-1} \in \mathbb{F}_q[j_{k-1}]$, $N_1 = \deg_{j_{k-1}}(G_{k-1}) = O(\ell_0^{k-1})$, $C_1 \in \mathbb{F}_q \setminus \{0\}$ and $M_1 \le 0$. Recall from (2.5) that

$$A = \Phi_{\ell_0}(j_{k-1}, j_k) = j_{k-1}^{\ell_0+1} - (j_k^{\ell_0})j_{k-1}^{\ell_0} + \left( \sum_{i,i' \le \ell_0, i+i' < 2\ell_0} a_{ii'} j_{k-1}^i j_k^{i'} \right) + j_k^{\ell_0+1},$$

and hence all the coefficients of $j_{k-1}^i$ for $0 \le i \le N_s$ of $j_{k-1}^{N_s-(\ell_0+1)} C_s A$ are polynomials over $\mathbb{F}_q[j_k]$ of degrees at most $\deg(C_s)+\ell_0+1$. From this, the degree of the coefficient of each $j_k^i$ with $0 \le i \le N_s$ in $B_{s+1} = B_s - j_{k-1}^{N_s-(\ell_0+1)} C_s A$ is at most $\max\{M_s, \deg(C_s)+\ell_0+1\}$, and thus both $\deg(C_{s+1})$ and $M_{s+1}$ are $\le \max\{M_s, \deg(C_s)+\ell_0+1\}$. Since $M_1 \le 0 \le \deg(C_1)+\ell_0+1$, successively we have $M_s \le (s-1)(\ell_0+1)$ and $\deg(C_s) \le (s-1)(\ell_0+1)$. Since $A$ and $C_s$ have at most $(\ell_0+1)^2+2 = O((\ell_0+1)^2)$ and $\deg(C_s)$ terms respectively, computing $C_s A$ requires $O((s-1)(\ell_0+1)^3)$ arithmetic in $\mathbb{F}_q$, which is clearly dominant in the $n$-th loop. It follows from $n \le \deg(B_1) - \deg_{j_{k-1}}(A) + 1 = O(\ell_0^{k-1})$ and $\sum_{s=1}^{n}(s-1)(\ell_0+1)^3 = O(\ell_0^3 n^2)$ that the total cost of the procedure (1) is

$$O(\ell_0^3(\ell_0^{k-1})^2) = O(\ell_0^{2k}).$$

By the estimation of the cost of computing a resultant of two polynomials, we estimate the complexity of the procedure (2). The procedure (2) requires $O(\ell_0^2)$ arithmetic over $\mathbb{F}_q[j_k]$ since both $G'_{k-1}$ and $\Phi_{\ell_0}(j_{k-1}, j_k)$ have degree $\le \ell_0 + 1 = O(\ell_0)$. Since the degree of the resultant $\hat{g}_k = (G_{k-2})^{\ell_0} G_k$ is $\hat{d}_k := (\ell_0+1)^2 \ell_0^{k-2} = O(\ell_0^k)$, we may assume that all arithmetic over $\mathbb{F}_q[j_k]$ in the procedure (2) are arithmetic of two polynomials of degree $O(\ell_0 \ell_0^k) = O(\ell_0^k)$ by considering pseudo division and PSC. Hence the complexity of the procedure (2) is estimated as $O(\ell_0^2(\ell_0^k)^2) = O(\ell_0^{2k})$ if one uses classical polynomial arithmetic, and $O(\ell_0^2(\ell_0^k)^{1+\epsilon}) = O(\ell_0^{(1+\epsilon)k})$ for some $0 < \epsilon < 1$ if one uses fast polynomial computation (e.g., fast Fourier transform). Dividing the degree $O(\ell_0^k)$-polynomial $\hat{g}_k$ by $(G_{k-2})^{\ell_0}$ provides $G_k$, and it is done in $O\left(\ell_0^{2k}\right)$ (or $O(\ell_0^{(1+\epsilon)k})$), which is equal to the cost of (2). Summing up, we estimate that the total cost is

$$O(\ell_0^4) + O(\ell_0^6) + \cdots + O(\ell_0^{2e_0}) = O(\ell_0^{2e_0}) = O(\ell_0^e) = O(\ell)$$

or $O(\ell_0^{(1+\epsilon)e_0}) = O(\ell^{(1+\epsilon)/2})$. For the parameter in our experiments (i.e., $\ell_0 = 3$), the cost is $O(3^{2e_0})$.

**Remark A.1.** For the meet-in-the-middle approach, we factorize the univariate polynomial $\Phi_3(x, j)$ over $\mathbb{F}_{p^2}$ for the $j$-invariant $j = j(E)$ of an elliptic curve $E$ to find $j$-invariants of all elliptic curves 3-isogeneous to $E$. It costs roughly $O(\log p)$, and hence the total cost of the meet-in-the-middle approach is $O(3^{e_0} \log p)$. On the other hand, the total cost of our first approach is $O(3^{2e_0})$ as discussed above, and hence we expect that our first approach could be faster than the meet-in-the-middle approach when $\log p > 3^{e_0}$. This is a main reason why our first approach is faster than the meet-in-the-middle approach in practice for small exponents $e_0$.

## A.2 For the second approach

We estimate the complexity of our second approach for solving the isogeny problem of prime degree $\ell = \ell_0^e$ with $e = 2e_0$. We use the same notation as in Subsection 3.2. To estimate the complexity, we calculate the number of equations in the system, that is, $\#(S \cup \widetilde{S})$, and the maximum value of the total degrees of the equations with respect to $t$, $\widetilde{t}$, $a_0$ and $b_0$.

First, we bound the total degrees of $t_i$ and $\widetilde{t}_i$ with respect to $t$, $\widetilde{t}$, $a_0$ and $b_0$ for $2 \le i \le m$. The kernel polynomial $F(x)$ for $E$ is written as $F = x^m + tx^{m-1} + \sum_{k=0}^{m-2} t_{m-k} x^k$. For each $2 \le i \le m$, the total degree of $t_i$ with respect to $t$, $a_0$ and $b_0$ is $i$, and thus it is bounded by $m$. Similarly, the total degree of $\widetilde{t}_i$ with respect to $\widetilde{t}$, $a_0$ and $b_0$ is bounded by $m$. Next, we explicitly write down a polynomial $G = \sum_k c_k x^k \in R[x]$ such that $S = \{c_k : 0 \le k \le \deg_x G\}$, where the coefficient ring is $R = \mathbb{F}_q[t, a_0, b_0]$. Recall that we obtain $S$ by expanding (3.2) with respect to $x$. It follows from $T = \frac{N}{D}$ and $D = F^2$ that we have $T' = \frac{N'F - 2NF'}{F^3}$, and thus (3.2) is rearranged as

$$(x^3 + ax + b)\left(\frac{N'F - 2NF'}{F^3}\right)^2 = \frac{N^3}{F^6} + a_0\left(\frac{N}{F^2}\right) + b_0. \tag{8}$$

Note that

$$N = (\ell x + 2t)F^2 - 2(3x^2 + a)F'F - 4(x^3 + ax + b)(F''F - (F')^2)$$

by (2.2), and hence $N$ is of degree $2m + 1$ with respect to $x$. We also note that each coefficient of $N$ is a polynomial in $R = \mathbb{F}_q[t, a_0, b_0]$ of total degree $\le 2m + 1 = O(m)$. Multiplying by $F^6$ the both side of (8), we have

$$(x^3 + ax + b)\left(N'F - 2NF'\right)^2 = N^3 + a_0NF^4 + b_0F^6$$

and set

$$G := (x^3 + ax + b)\left(N'F - 2NF'\right)^2 - N^3 - a_0NF^4 - b_0F^6,$$

which is of degree $6m + 3$ with respect to $x$. Writing $G = \sum_{k=0}^{6m+3} c_k x^k$ with $c_k \in R$ for $0 \le k \le 6m + 3$, we clearly have $S = \{c_k : 0 \le k \le 6m+3\}$, and the total degree of each $c_k$ is equal to or less than $6m+3 = O(m)$. Similarly, we can construct a polynomial $\widetilde{G} = \sum_k \widetilde{c}_k x^k \in \widetilde{R}[x]$ with $\widetilde{R} = \mathbb{F}_q[\widetilde{t}, a_0, b_0]$ such that $\widetilde{S} = \{\widetilde{c}_k : 0 \le k \le \deg_x \widetilde{G}\}$ with $\deg_x \widetilde{G} = 6m + 3$. The total degree of each $\widetilde{c}_k$ is equal to or less than $6m + 3 = O(m)$.

Here we assume that we use the algorithm $F_4$, which is default for computing Gröbner bases in Magma, to solve a system of (non-homogeneous) multivariate equations. Let $I_{S\cup\widetilde{S}}$ denote the ideal defined by $S \cup \widetilde{S}$. Let solv. $\deg(I_{S\cup\widetilde{S}})$ denote the solving degree of the ideal $I_{S\cup\widetilde{S}} \subset \mathbb{F}_q[t, \widetilde{t}, a_0, b_0]$ with respect to the grevlex order. To estimate the complexity of computing a Gröbner basis of $I_{S\cup\widetilde{S}}$, we use the following proposition proved by Caminata and Gorla:

**Proposition A.1** ([7]). *Let $K$ be a field, and $R = K[X_1, \dots, X_n]$ the polynomial ring of $n$ variables. Let $f_1, \dots, f_\ell$ be (not necessarily homogeneous) polynomials in $R$, and $I$ the ideal generated by $f_1, \dots, f_\ell$. We denote by $s =$ solv. $\deg(I)$ the solving degree of $I$, that is, the highest degree of the polynomials involved in the computation of a Gröbner basis of $I$ (we fix the grevlex order for Gröbner bases). Let $d_i$ be the total degree of $f_i$ for $1 \le i \le \ell$, and*

$$m := \sum_{i=1}^{\ell} \binom{n + s - d_i}{s - d_i}.$$

*Then the number of operations in $K$ required to compute a Gröbner basis of $I$ is*

$$O\left(\binom{n+s}{s} m^{\omega-1}\right) \quad \text{if } m \le \binom{n+s}{s},$$
$$O\left(m\binom{n+s}{s}^{\omega-1}\right) \quad \text{otherwise},$$

*where $2 \le \omega \le 3$ is the exponent of matrix multiplication.*

Since the total degree of each $c \in S \cup \widetilde{S}$ is $\approx 6m + 3 = O(m)$, we have

$$\text{solv. } \deg(I_{S\cup\widetilde{S}}) \ge 6m + 3 = O(m),$$

where the solving degree is not less than the maximum degree of input polynomials. It also follows from $\#(S \cup \widetilde{S}) \approx 2(6m + 3) = 12m + 6 = O(m)$ and

$$\binom{4 + \text{solv. } \deg(I_{S\cup\widetilde{S}})}{\text{solv. } \deg(I_{S\cup\widetilde{S}})} = \binom{4 + \text{solv. } \deg(I_{S\cup\widetilde{S}})}{4} \ge O(m^4)$$

that the number of operations in $\mathbb{F}_q$ required to compute a Gröbner basis of $I_{S\cup\widetilde{S}}$ in the second approach is $\ge O\left(m^5\right) = O\left((\ell')^5\right) = O\left(\ell_0^{5e_0}\right)$ by Proposition A.1, and so is the total complexity of the second approach. For the parameter in our experiments (i.e., $\ell_0 = 3$), the complexity is not less than $3^{5e_0}$.