Faster Algorithms for Isogeny Problems Using Torsion Point Images

Christophe Petit(✉)

School of Computer Science, University of Birmingham, Birmingham, UK
christophe.f.petit@gmail.com

Abstract. There is a recent trend in cryptography to construct protocols based on the hardness of computing isogenies between supersingular elliptic curves. Two prominent examples are Jao-De Feo's key exchange protocol and the resulting encryption scheme by De Feo-Jao-Plût. One particularity of the isogeny problems underlying these protocols is that some additional information is given as input, namely the image of some torsion points with order coprime to the isogeny. This additional information was used in several active attacks against the protocols but the current best passive attacks make no use of it at all.

In this paper, we provide new algorithms that exploit the additional information provided in isogeny protocols to speed up the resolution of the underlying problems. Our techniques lead to heuristic polynomial-time key recovery on two non-standard variants of De Feo-Jao-Plût's protocols in plausible attack models. This shows that at least some isogeny problems are easier to solve when additional information is leaked.

1 Introduction

Following calls from major national security and standardization agencies, the next cryptographic standards will have to be "post-quantum secure", namely they will have to rely on computational problems that will (at least to the best of our knowledge) remain hard for quantum computers. Several directions are currently explored for post-quantum cryptography, including lattice-based cryptography, code-based cryptography, multivariate cryptography, hash-based cryptography and most recently cryptography based on isogeny problems. The latter are appealing for their mathematical elegance but also for the relatively small key sizes compared to other post-quantum candidates.

The interest in isogeny problems as potential cryptographic building blocks is relatively new, and there has therefore not been much cryptanalytic work on them. The most established isogeny problem is the endomorphism ring computation problem, which was already considered by Kohel in his PhD thesis [12]. In the supersingular case this problem is (heuristically at least) equivalent to the problem of computing an isogeny between two randomly chosen curves [15], and it remains exponential time even for quantum algorithms today.

The supersingular key exchange protocol of Jao-De Feo [11] and the encryption scheme and signature schemes that are derived from it [7,9,24] rely on

T. Takagi and T. Peyrin (Eds.): ASIACRYPT 2017, Part II, LNCS 10625, pp. 330–353, 2017.
https://doi.org/10.1007/978-3-319-70697-9_12

variants of these problems, where special primes and relatively small degree isogenies are used. More importantly for this paper, the attacker is also provided with the image by the isogeny of a large torsion group, in addition to the origin and image curves. Although it was observed that this additional information could a priori make the problems easier, all security evaluations against passive attacks were based on a meet-in-the-middle strategy that makes no use at all of it.

1.1 Contributions

In this paper, we study the impact of revealing the images of torsion points on the hardness of isogeny problems. We provide new techniques to successively exploit this additional information and improve on the best previous attacks, namely meet-in-the-middle attacks (see Sect. 2). Among other results, these techniques lead to polynomial-time algorithms to compute isogenies between two curves E_0 and E_1 assuming

1. Some non scalar endomorphisms of E_0 are known and/or are of small degree.
2. The images of N_2 torsion points are revealed, where N_2 is significantly larger than the degree of the isogeny N_1.

So far our techniques do not invalidate the parameters proposed in the original protocol (where $N_1 \approx N_2$). However, we describe two natural variants, which we call unbalanced variant and optimal degree variant, which can be attacked by our methods in plausible attack scenarios. We believe these generalizations are of independent interest, as they have some advantages over the original protocol when appropriate parameters are chosen.

Our main contribution in this paper is our new attack techniques. We illustrate their potential with the following results:

1. (Sect. 3.) A nearly square root speedup on the problem of computing an endomorphism of a supersingular elliptic curve of a certain degree, when provided with some torsion point images through this endomorphism.
2. (Sect. 4.4.) A polynomial time key recovery attack on our optimal degree variant, provided $N_2 > N_1^4$ and E_0 is "special" (such special curves were suggested in previous implementations [4,7] for efficiency reasons).
3. (Sect. 4.5.) A polynomial time key recovery attack on both variants, provided $\log N_2 = O(\log^2 N_1)$ and E_0 has a small degree non scalar endomorphism.

These attacks show that (at least some) isogeny problems are easier to solve when the images of torsion points through the isogeny are revealed. Some of these attacks require further assumptions on N_2; we refer to the next sections for details. We provide a heuristic analysis for all these attacks. The heuristics used involve factorization patterns and other properties of integers of particular forms appearing in our algorithms, which we treat as random numbers of the same size. For the first two attacks these heuristics are very plausible, and we believe that they can either be proved or made unnecessary (though any of

those options would require significant work). For the third attack they are still a priori plausible, but they may be very hard to prove or remove. Indeed the attack involves a recursive step, and a rigorous result would have to take into account correlations between successive steps. For this reason we additionally provide some experimental support for our third attack.

We believe the three attacks we develop here are only some examples of what our new techniques can achieve, and we leave further developments to further work.

1.2 Background Reading

We refer to the books of Silverman [18] and Vignéras [21] for background results on elliptic curves and quaternion algebras. Recent cryptographic constructions based on isogeny problems include [2,7,9,11,17,23]. Computational aspects related to isogenies are covered in David Kohel's PhD thesis [12] and more recently in [8,9].

1.3 Complexity Model

Unless otherwise stated all complexity estimations in this paper use elementary bitwise operations as units. We use the standard "big O notation" to describe asymptotic complexities of algorithms. Recall that for any two functions $f, g : \mathbb{N} \to \mathbb{Z}_+$ we have $f = O(g)$ if and only if there exists $N \in \mathbb{N}$ and $c \in \mathbb{Z}_+$ such that $g(n) \geq cf(n)$ for any $n \geq N$. We also use the "big O tilde notation" to hide any polylogarithmic factors in our complexity statements: namely for any two functions $f, g : \mathbb{N} \to \mathbb{Z}_+$ we have $f = \tilde{O}(g)$ when there exists $d \in \mathbb{Z}_+$ such that $f = O(g \log^d g)$. The security levels of the protocols studied in this paper are functions of one or several security parameters. When we refer to "polynomial time" complexity we mean complexity $O(f)$, where f is a polynomial function of these security parameters.

1.4 Outline

In Sect. 2 we first describe the supersingular key exchange protocol of Jao-De Feo [11] and our two variants of this protocol, then we recall the most relevant cryptanalysis results on it. In Sect. 3 we describe faster algorithms to compute an endomorphism of a given supersingular elliptic curve, given the image of torsion points by this endomorphism. In Sect. 4 we turn to the problem of computing an isogeny between two supersingular elliptic curves given the images of torsion points by this isogeny, and we describe two attacks faster than the state-of-the-art meet-in-the-middle algorithm in this context. Finally, we summarize the impact of our techniques and results in Sect. 5, and we give perspectives for further work.

2 Supersingular Isogeny Key Exchange

2.1 Jao-De Feo's Key Exchange

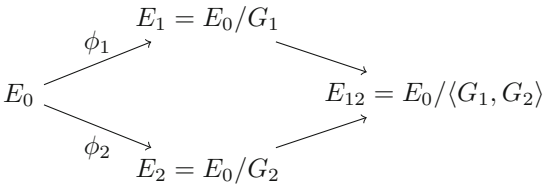
We recall the supersingular key exchange protocol of Jao-De Feo [11].

Setup. Let ℓ_1, ℓ_2 be two small primes. Given a security parameter λ, let e_1, e_2 be the smallest integers such that $\ell_1^{e_1}, \ell_2^{e_2} \geq 2^{2\lambda}$ (or $2^{3\lambda}$ for post-quantum security). Let f be the smallest integer such that $p = \ell_1^{e_1}\ell_2^{e_2} f - 1$ is prime. Let E_0 be a supersingular elliptic curve over $\mathbb{F}_{p^2}$. Let P_1, Q_1 and P_2, Q_2 be respectively bases of the $\ell_1^{e_1}$ and $\ell_2^{e_2}$ torsions on E_0.

First round. Alice chooses a random cyclic subgroup of order $\ell_1^{e_1}$, say $G_1 = \langle \alpha_1 P_1 + \beta_1 Q_1 \rangle$ with at least one of α_1, β_1 coprime to ℓ_1. She computes the corresponding isogeny ϕ_1 and image curve E_1, as well as $\phi_1(P_2)$ and $\phi_1(Q_2)$. She sends E_1, $\phi_1(P_2)$ and $\phi_1(Q_2)$ to Bob. Bob proceeds similarly, permuting the roles of ℓ_1 and ℓ_2.

Second round. Upon receiving E_2, $\phi_2(P_1)$ and $\phi_2(Q_1)$, Alice computes $G_1' = \langle \alpha_1\phi_2(P_1) + \beta_1\phi_2(Q_1) \rangle$, the corresponding isogeny ϕ_1', the image curve $E_{12} = E/\langle G_1, G_2 \rangle$ and its j-invariant j_{12}. Bob computes $j_{21} = j_{12}$ similarly with the information sent by Alice. The shared secret is the value $j_{12} = j_{21}$, or the result of applying some key derivation function to this value.

The protocol is summarized in the following commutative diagram:



This protocol can be broken if one can compute isogenies between two given curves. However we stress that the curves appearing in this protocol are closer to each other in the isogeny graphs than random curves would be: indeed for any fixed E_0 there are only $(\ell_i + 1)\ell_i^{e_i - 1} \approx \sqrt{p}$ possible curves for E_1, while there are roughly $p/12$ supersingular j-invariants over $\mathbb{F}_{p^2}$. This allows more efficient meet-in-the-middle attacks in complexity $\tilde{O}(\sqrt[4]{p})$ instead of $\tilde{O}(\sqrt{p})$ for a generic curve pair. More importantly for this paper, some information on the isogenies is leaked by the protocol, as the image of a full torsion coprime with the isogeny degree is revealed. Finally, special primes are used to ensure that the $\ell_i^{e_i}$ torsions are defined over $\mathbb{F}_{p^2}$. For arbitrary p these torsions subgroups would be defined over large field extensions, resulting in an inefficient protocol.

Remark. Let $N_1 = \ell_1^{e_1}$. If the image of the N_1 torsion by a degree N_1 isogeny was revealed it would be straightforward to recompute the isogeny, as this image would be the kernel of the dual isogeny. More generally if N_1 is not coprime with the degree then part of the isogeny can be recovered efficiently.

2.2 Unbalanced and Optimal Degree Variants

We now present two variants of the protocol, which we call unbalanced and optimal degree variants.

Unbalanced variant. In their paper Jao and De Feo suggested parameters such that $\ell_1^{e_1} \approx \ell_2^{e_2}$. We suggest to generalize the setup to allow for unbalanced parameters $\ell_2^{e_2} \gg \ell_1^{e_1}$ in some contexts. The size of $\ell_i^{e_i}$ determines the security of the corresponding secret key G_i with respect to all previous attacks (see next subsection), while the size of p would influence efficiency. Jao and De Feo therefore chose $\ell_1^{e_1} \approx \ell_2^{e_2}$ to provide the same security level on both Alice and Bob's ephemeral keys. However in some contexts as in the public key encryption scheme [7] one secret key is static and it may therefore make sense to protect it more strongly. This is achieved by our unbalanced variant.

In the unbalanced variant, the setup procedure takes two security parameters λ_1 and λ_2 as input. For $i = 1, 2$ it computes the smallest integer e_i such that $\ell_i^{e_i}, \geq 2^{2\lambda_i}$ (or $2^{3\lambda_i}$ for post-quantum security), and then the smallest integer f such that $p = \ell_1^{e_1}\ell_2^{e_2}f - 1$ is prime. The rest of the protocol is as in Jao and De Feo.

Optimal degree variant. We now generalize the parameters such that the isogeny degrees are large enough to ensure uniform distribution of E_i among all curves on the isogeny graphs: we call the resultant protocol "optimal degree variant" for this reason. In addition, this variant allows for arbitrary primes p rather than the very special primes used by Jao and De Feo.

We recall that a number $N = \prod p_i^{e_i}$ is B-powersmooth if for all i we have $p_i^{e_i} < B$. In this paper we say that a number is powersmooth if it is B-powersmooth for some bound B that is polynomial in the security parameter.

For an arbitrary prime p, we replace $\ell_1^{e_1}$ and $\ell_2^{e_2}$ in the protocol by any powersmooth numbers N_1 and N_2 that are coprime to each other and of size about p^2. Note that the N_1 and N_2 torsions are a priori not defined over $\mathbb{F}_{p^2}$; however the powersmooth requirement ensures that they can be efficiently represented in a Chinese remainder manner (see [9]). On the other hand, the coprimality requirement ensures that the isogeny diagram commutes as in the original protocol. Finally, the condition $N_i \approx p^2$ on the isogeny degrees guarantees that E_1 and E_2 are close to uniformly distributed [9], while for the original parameters and the unbalanced variant above we have $N_1N_2 \approx p$.

In the optimal degree variant, the setup procedure takes a security parameter λ. It chooses a random prime p with 2λ bits (or 3λ bits for post-quantum security). Then N_1 and N_2 are chosen coprime to each other, such that both of them are powersmooth and have at least $2\log p$ bits. Then for each maximal prime power $\ell_j^{e_j}$ dividing either N_1 or N_2 we fix a basis for the $\ell_j^{e_j}$ torsion. Note that this is defined over an extension field of degree at most $2\ell_j^{e_j}$, which has a size polynomial in λ.

If $N_1 = \prod p_j^{e_j}$ then in the first round Alice chooses for each j one cyclic subgroup $G_{1j} = \langle \alpha_j P_j + \beta_j Q_j \rangle$ with at least one of α_j, β_j coprime to p_j. This

implicitly defines a cyclic subgroup G_1 of order N_1 such that $G_1 = G_{1j} \bmod E_0[\ell_j^{e_j}]$. She computes the corresponding isogeny ϕ_1 as a composition of isogenies of prime degrees, the image curve $E_1 = E_0/G_1$, and the image by ϕ_1 of the $\ell_j^{e_j}$ torsion basis points, for each $\ell_j^{e_j}$ dividing N_2. Alice sends E_1 and all torsion point images to Bob. Note that although the torsion points and their images are defined over some field extensions, all isogenies computed are defined over $\mathbb{F}_{p^2}$. Moreover the degree of any extension field involved is bounded by $2\ell_j^{e_j}$ which is polynomial in the security parameter, so all elements can be efficiently represented and the computation runs in polynomial time. Bob proceeds similarly.

In the second round, Alice computes $\phi_2(G_{1j})$ using the information sent by the other party (as in the original protocol), then she computes $E_2/\phi_2(G_1)$ as above, and finally the j-invariant of this curve. Bob proceeds similarly.

A first implementation of this variant is given in [3]. Because it allows both for arbitrary primes and for "large enough" degree isogenies, the optimal degree variant can a priori be more secure than the original protocol. On the other hand, working over field extensions, even of moderate degrees, has a significant efficiency cost in practice. We leave a precise complexity estimation and a thorough comparison of this variant with the original protocol to further work.

Remark. Of course, one could also allow intermediate parameters where $\gcd(N_1N_2, (p+1)^2)$ is a medium size factor of $(p+1)^2$ to ensure that the primes are not too special and at the same limit the size of the extension fields needed.

2.3 State-of-the Art on Cryptanalysis

We refer to [9] for a thorough discussion of existing cryptanalysis results, and only describe the most relevant work for this paper. With the exception of active attacks in [8,10,20], previous cryptanalysis results have ignored the additional information revealed in De Feo-Jao-Plût's protocols. They therefore considered the following problem:

Problem 1. *Let N be a positive integer, let p be a prime and let E_1, E_2 be two supersingular elliptic curves defined over $\mathbb{F}_{p^2}$, such that there exists an isogeny ϕ of degree N such that $E_2 = E_1/\ker\phi$. Compute ϕ.*

Remark. The most natural representation of ϕ is some canonical representation as two elements of the function field $E_1(x, y)$. In cryptographic contexts the degree of ϕ is of exponential size so this representation is not efficient. However in these contexts the degree is often a smooth number so that the isogeny can be efficiently returned as a composition of rational maps.

When N is large enough any pair of elliptic curves are connected by an isogeny of degree N, and this problem is heuristically equivalent to the endomorphism ring computation problem [15]. In De Feo-Jao-Plût's protocols, however, $N = O(\sqrt{p})$ is too small to ensure this, and as N is moreover smooth one can do a meet-in-the-middle attack with complexity $\tilde{O}(\sqrt[4]{p})$ (respectively $\tilde{O}(\sqrt[6]{p})$ with a quantum computer) even if the endomorphism ring computation problem

remains of complexity $\tilde{O}(\sqrt{p})$ (respectively $\tilde{O}(\sqrt[3]{p})$ with a quantum computer). We stress that the optimal degree variant we introduced above does not suffer from this problem, as the isogeny degrees are chosen large enough to ensure a uniform distribution of E_2.

The following lemma generalizes the meet-in-the-middle strategy when the smoothness bound on N is not polynomial in $\log p$.

Lemma 1. *Assume $N = N_1 \cdot N_2$ where both N_1 and N_2 are B-smooth. Then the meet-in-the-middle strategy has a time and memory complexity $\tilde{O}(B \max(N_1, N_2))$, neglecting log factors.*

Proof: The factorization of N can be obtained in subexponential time, which is negligible with respect to $\max(N_1, N_2)$. Isogenies of prime degree can be computed in quasilinear time in the degree. The meet-in-the-middle strategy computes $O(N_1)$ isogenies of degree N_1 and $O(N_2)$ isogenies of degree N_2, each of them as a composition of isogenies of degrees at most B. □

The active attack presented in [8] runs $O(\log p)$ executions of the key exchange protocol with the same party. Assuming this party uses a static secret key G_1, the attacker provides them with incorrect values for $\phi_2(P_1), \phi_2(Q_1)$, observes variations in the resulting shared key $j(E_{12})$, and progressively deduces the key G_1. The loop-abort fault attack developed in [10] is similar to this attack. A fault attack is also used in [20] to replace $\phi_2(P_1)$ and $\phi_2(Q_1)$ by points whose order is not coprime with the isogeny degree. Our goal in this paper is to show how to exploit the "torsion image" information revealed in De Feo-Jao-Plût's protocols but using only passive attacks, namely for normal executions of the protocols.

3 Computing an Endomorphism from Additional Information

From a computational number theory point of view, computing endomorphisms of a curve is a somewhat more natural task than computing isogenies between two curves. At the same time, there are strong relations between the two problems (see [9,15]). In this section we define an "endomorphism computation" counterpart to De Feo-Jao-Plût's isogeny problem, and we show how leaking the image of torsion points helps in solving this problem.

3.1 Endomorphism Computation Problem with Additional Information

We consider the following problem:

Problem 2. *Let p be a prime and let E be a supersingular elliptic curve defined over $\mathbb{F}_{p^2}$. Let ϕ be a non scalar endomorphism of E with smooth degree N_1. Let N_2 be a smooth integer with $\gcd(N_1, N_2) = 1$, and let P, Q be a basis of $E[N_2]$. Let R be a subring of $\mathrm{End}(E)$ that is either easy to compute, or given. Given E, P, Q, $\phi(P)$, $\phi(Q)$, $\deg \phi$, R, compute ϕ.*

Remark. This problem is similar to the problem appearing in De Feo-Jao-Plût protocols, with the additional requirement $E_1 = E_2$.

Remark. When no endomorphism subring is explicitly given one can take for R the subring of scalar multiplications, which we will denote $R = \mathbb{Z}$.

Remark. If we do not use the additional information the best algorithm for this problem will be a meet-in-the-middle approach: compute all isogenies of degree approximately $\sqrt{N_1}$ from E and search for a collision. As N_1 is smooth the cost for each isogeny is polynomial, resulting in an algorithm with roughly $\tilde{O}(\sqrt{N_1})$ complexity.

3.2 General Strategy

Our general strategy is summarized in Algorithm 1.

Algorithm 1. Computing an Endomorphism from Additional Information

Require: As in Problem 2, plus parameter B.
Ensure: A description of ϕ as a composition of low degree maps.
1: Find $N_1' \in \mathbb{N}$ and $\theta_1, \theta_2 \in R$ such that $\deg(\theta_1\phi + \theta_2) = N_1'N_2$ and $\gcd(\deg\theta_1, N_1) = 1$, and such that N_1' is B-smooth and as small as possible.
2: Compute $\ker\psi_{N_2}$ using the additional information, where $\theta_1\phi + \theta_2 = \psi_{N_1'}\psi_{N_2}$ and $\psi_{N_1'}, \psi_{N_2}$ are respectively of degrees N_1' and N_2.
3: Compute $\psi_{N_1'}$ using a meet-in-the-middle approach.
4: Compute $\ker\phi = \ker(\theta_1^{-1}(\psi_{N_1'}\psi_{N_2} - \theta_2))$ by evaluating all maps on the N_1 torsion.
5: Compute ϕ from $\ker\phi$.

From what is given in the problem we can compute the image of ϕ on any point in $E[N_2]$. Let $\theta_1, \theta_2 \in R$ be known endomorphisms of E, to which we associate another endomorphism

$$\psi := \theta_1\phi + \theta_2.$$

Of course we do not know ϕ so far, but since we know θ_1, θ_2, and the action of ϕ on $E[N_2]$ we can nevertheless evaluate ψ on any point of $E[N_2]$.

Let us now assume that the maps θ_1, θ_2 are chosen such that $\deg\psi = N_1'N_2$ for some $N_1' \in \mathbb{Z}$. An algorithm to achieve that together with an additional smoothness condition on N_1' will be described in the next subsection for the case $R = \mathbb{Z}$. The endomorphism ψ can then be written as a composition of two isogenies

$$\psi = \psi_{N_1'}\psi_{N_2}$$

with $\psi_{N_1'}$ and ψ_{N_2} respectively of degrees N_1' and N_2.

By computing ψ on a basis of $E[N_2]$ and solving some discrete logarithm problems in $E[N_2]$ we deduce the kernel of ψ_{N_2} and then deduce ψ_{N_2} itself. This is efficient since N_2 is smooth by assumption.

At this point, the map $\psi_{N_1'}$ is an isogeny of degree N_1' between two known j-invariants, namely the curve image of ψ_{N_2} and the original curve E. We recover this isogeny using the meet-in-the-middle approach analyzed in Lemma 1. The efficiency of this step depends on the factorization of N_1'.

At this point, we have computed the map ψ as a composition $\psi = \psi_{N_1'}\psi_{N_2}$. We deduce an expression for ϕ, namely $\theta_1^{-1}(\psi_{N_1'}\psi_{N_2} - \theta_2)$, and assuming $\gcd(\deg\theta_1, N_1) = 1$ we evaluate this map on the N_1 torsion to identify $\ker\phi$, from which we recompute a more canonical description of ϕ. This is efficient as N_1 is smooth.

Remark. We do not use the additional information to compute $\psi_{N_1'}$. Note that part of the N_2 torsion is annihilated by ψ_{N_2}, so we only know $\psi_{N_1'}$ and its dual on one cyclic subgroup of the respective N_2 torsions.

Remark. There is no gain in generality in considering maps of the form $\psi := \theta_1\phi\theta_3 + \theta_2$ for $\theta_1, \theta_2, \theta_3 \in R$. Indeed we have $\theta_1\phi\theta_3 + \theta_2 = \hat{\phi}\hat{\theta}_1\theta_3 + \theta_2 + \mathrm{Tr}(\theta_1\phi)\theta_3$. Taking conjugates we obtain an element $\hat{\psi} = \hat{\theta}_3\theta_1\phi + \hat{\theta}_2 + \mathrm{Tr}(\theta_1\phi)\hat{\theta}_3 \in R\phi + R$ with the same norm. Similarly, there is no gain in generality in using powers of ϕ since $\phi^2 = -(\mathrm{Tr}\,\phi)\phi - \deg\phi$.

3.3 Attack When $R = \mathbb{Z}$

We first consider the most generic case where the only known endomorphisms of E are scalar multiplications. We define

$$\psi = \psi_{a,b} = a\phi + b$$

for $a, b \in \mathbb{Z}$, which has degree

$$\deg\psi_{a,b} = a^2\deg\phi + b^2 + ab\,\mathrm{Tr}\,\phi = \left(b + a\frac{\mathrm{Tr}\,\phi}{2}\right)^2 + a^2\left(\deg\phi - \left(\frac{\mathrm{Tr}\,\phi}{2}\right)^2\right).$$

Our goal is to find a, b such that $\deg\psi_{a,b} = N_1'N_2$, where N_1' is as small and as smooth as possible.

Parameter Restriction. The attack we describe below requires two assumptions on the parameters.

1. We require $N_2 > 2\sqrt{N_1}$.
2. We also require that $-D$ is a square modulo N_2, where $D = \deg\phi - \left(\frac{\mathrm{Tr}(\phi)}{2}\right)^2$.

Note that in the Jao-De Feo key exchange protocol we have $N_2 \approx N_1$ so the first assumption does not look too strong. By Hensel's lifting lemma the second condition is equivalent to $-D$ being a square modulo every odd prime factor of N_2 and congruent to 1 modulo 8 when 2 divides N_2. If we consider the endomorphism ϕ as fixed, this condition restricts N_2 values as follows:

- If $N_2 = \ell^e$ is a prime power (as for Jao-De Feo's parameters and the unbalanced variant), the second condition is satisfied if and only if $-D$ is a quadratic residue modulo ℓ, and heuristically we expect this to occur for half of the primes ℓ.
- In our optimal degree variant, N_2 is a powersmooth number whose prime factors will be as small as possible. We can heuristically expect that about one half of these prime factors ℓ_i will be such that $-D$ is a quadratic residue modulo ℓ_i. There will therefore exist a factor N_2' of N_2 such that $N_2' \approx \sqrt{N_2}$ and $-D$ is a quadratic residue modulo N_2'. Moreover if $N_2 \approx N_1$ or bigger, we can use N_2' in the attack instead of N_2, and still satisfy the first condition.

This suggests that the conditions above are relatively mild, in the sense that they are satisfied for a large set of parameters with the expected forms. In the remaining of this section we assume that both conditions above are satisfied.

Algorithm. Remember that from the additional information given in the problem we can compute the image of ϕ on any point in $E[N_2]$. Note that since N_1 and N_2 are coprime, ϕ is a one-to-one map on $E[N_2]$. From the relation $\phi\hat{\phi} = [\deg \phi]$ we can also compute the image of any point in $E[N_2]$ by the dual map $\hat{\phi}$. We can therefore also evaluate $\mathrm{Tr}(\phi)$ on $E[N_2]$. By solving a discrete logarithm problem in $E[N_2]$ we deduce $\mathrm{Tr}(\phi) \bmod N_2$. By the Cauchy-Schwarz inequality we also have $\mathrm{Tr}(\phi) \leq 2\sqrt{\deg \phi}$ so under our first parameter restriction that $N_2 > 2\sqrt{N_1}$ we actually recover $\mathrm{Tr}(\phi)$ exactly.

Let $D = \deg \phi - \frac{1}{4}(\mathrm{Tr}(\phi))^2$ and let τ such that $\tau^2 = -D \bmod N_2$. Such a τ exists under our second parameter restriction, and can be efficiently computed using Tonelli-Shanks algorithm and Hensel's lifting lemma. Points (x, y) in the lattice generated by the two vectors $(N_2, 0)$ and $(\tau, 1)$ correspond to solutions of the equation $x^2 + Dy^2 = 0 \bmod N_2$. We compute a reduced basis for the lattice, with respect to a weighted inner product norm where the second component is weighted by $\sqrt{D}$. This can be done in polynomial time. Finally we let $a := y_0$, $b = x_0 - \frac{\mathrm{Tr}(\phi)}{2} y_0$ and $N_1' = \frac{x_0^2 + Dy_0^2}{N_2}$, where (x_0, y_0) is a well-chosen short vector in the lattice.

To choose (x_0, y_0) we proceed as follows. Using the short basis computed above we generate short vectors and compute the corresponding N_1' values, until we obtain N_1' such that the meet-in-the-middle strategy is efficient enough (see Lemma 1).

Complexity analysis. We first analyze the expected norms of minimal lattice vectors.

Lemma 2. *Under plausible heuristic assumptions, the shortest vectors in the lattice have norm $N_1' N_2$ where $N_1' \approx \sqrt{N_1}$.*

Proof: Heuristically, a proportion about $1/N_2$ pairs (x, y) will satisfy the congruence $x^2 + Dy^2 = 0 \bmod N_2$ so we expect $xy \approx N_2$. We can also expect that minimal vectors (x, y) in the lattice have their coefficients balanced such that $x^2 \approx Dy^2 \approx N_1' N_2$. (If $N_2^2 < D$ then the smallest element will of course be

$(N_2, 0)$, however by our parameter restriction we have $D \approx N_1 < N_2^2$.) Combining all these approximations gives $(N_1' N_2)^2 \approx x^2 D y^2 = D(xy)^2 \approx D N_2^2$, hence $N_1' \approx \sqrt{D} \approx \sqrt{N_1}$. □

By construction any lattice vector will have a norm divisible by N_2. In our algorithm we generate random short vectors until the cofactor N_1' is smooth enough. To estimate the number of random trials needed, we (heuristically) approximate the smoothness probability of N_1' by the smoothness probability of a random number of the same size.

For any positive integers X, Y, let $\pi(X, Y)$ be the proportion of integers smaller than X that are Y-smooth. For any positive integer X and any $0 \leq \alpha \leq 1$, let $L_X(\alpha) = \exp^{(\log X)^\alpha (\log \log X)^{1-\alpha}}$ be the subexponential function. We recall the following well-known fact [16]:

Lemma 3. *For any $0 \leq \alpha \leq 1$ and any large enough X, we have $\pi(X, L_X(\alpha)) \approx (L_X(1-\alpha))^{-1}$.*

We deduce the following result:

Proposition 1. *Subject to the above parameter restrictions and under plausible heuristic assumptions, Problem 2 can be solved in time $O(N_1^{1/4+\epsilon})$ for any $\epsilon > 0$.*

Proof: The algorithm and heuristic assumptions required have been described earlier in this section. The cost of the algorithm depends mostly on the smoothness bound required on lattice vectors, which decides both the cost of finding a suitable vector and the meet-in-the-middle cost needed to compute $\psi_{N_1'}$.

Using a smoothness bound $L_{N_1}(\alpha)$ for any $0 < \alpha < 1$, the cost of finding a suitable vector in the lattice is bounded by $(L_{N_1}(1-\alpha))^{-1} << O(N_1^{1/4+\epsilon})$, and the meet-in-the-middle computation has a cost $\tilde{O}(N_1^{1/4} L_{N_1}(\alpha)) = O(N_1^{1/4+\epsilon})$ by Lemma 1. □

By exploiting the images of torsion points, our algorithm provides a near-square root speedup for Problem 2 over the previous state-of-the-art algorithm.

Improvement when $\gcd(D, N_2) \neq 1$. For any $r | \gcd(D, N_2)$ there exist $a, b \in \mathbb{Z}$ with $(a\phi + b)/r \in \mathrm{End}(E)$ and $r \nmid \gcd(a, b)$. Moreover we can normalize pairs of this form such that $a = 1$. We can identify the corresponding correct b by trying every possibility until $\phi + b$ annihilates the r torsion. This has a cost $O(r)$. Alternatively we can solve some discrete logarithm problems to find b in at most $O(\sqrt{r})$ operations. In any case since N_2 is smooth we can process small factors one at the time, and efficiently deduce $\phi' \in \frac{1}{r}\mathbb{Z}[1, \phi] \cap \mathrm{End}(E)$, with a new D value $D' = D/\gcd(D, N_2)$. Moreover we can evaluate ϕ' on the $N_2/\gcd(D, N_2)$ torsion. Following the analysis above, we expect that this will reduce the complexity by a factor $\sqrt{\gcd(D, N_2)}$.

3.4 Variants and Extensions

We could consider variants of Problem 2 where information is given on several endomorphisms of a single curve, and develop similar attacks.

A particular case of interest is the case of subfield curves, namely curves defined over $\mathbb{F}_p$, when we can use $R = \mathbb{Z} + \pi_p\mathbb{Z}$ where $\pi_p : (x, y) \to (x^p, y^p)$.

Remark: When E is defined over $\mathbb{F}_p$ one can compute the full endomorphism ring in expected time $\tilde{O}(p^{1/4})$ using the techniques of Delfs and Galbraith [6].

Under the (reasonable) parameter restriction that $N_2 > 2\sqrt{N_1}$ we can compute $\operatorname{Tr}\phi$ as above, and substitute ϕ by $\phi' = \phi - \frac{\operatorname{Tr}\phi}{2}$ in the problem so that $\operatorname{Tr}\phi' = 0$. Let $\Delta := \deg\phi' = N_1 - \frac{1}{4}(\operatorname{Tr}\phi)^2$. We can consider an endomorphism of the form

$$\psi = (a\phi' + b)\pi_p + c\phi' + d,$$

with degree

$$\begin{aligned}\deg\psi &= (a^2\Delta + b^2)p + (c^2\Delta + d^2) + \operatorname{Tr}\left((a\phi' + b)\pi_p(-c\phi' + d)\right)\\ &= (a^2\Delta + b^2)p + (c^2\Delta + d^2) + (ad - bc)\operatorname{Tr}(\phi'\pi_p).\end{aligned}$$

If $N_2 > 2\sqrt{N_1 p}$ we can evaluate $\operatorname{Tr}(\pi_p\phi')$. We are then left with finding $a, b, c, d, N_1' \in \mathbb{Z}$ such that $\deg\psi = N_1'N_2$ and moreover N_1' is both small and smooth such that the meet-in-the-middle strategy (Lemma 1) is efficient.

Note that for the minimal solution we expect $a^2pN_1 \approx b^2N_1 \approx c^2p \approx d^2 \approx N_1'N_2$ and $abcd \approx N_2$, hence $d^4 \approx N_2N_1p$ and $N_1' \approx N_1^{1/2}p^{1/2}N_2^{-1/2}$. This means that if $N_2 \approx N_1p$ we can expect a solution with $N_1' = O(1)$.

Remark: The discussion in this section provides a reduction from an isogeny problem to a Diophantine equation problem, arguably a step forward in the cryptanalysis. We leave the construction of an efficient (classical or quantum) algorithm to solve this Diophantine equation to further work.

Remark: Efficient solutions for quaternary quadratic form equations exist over the rationals [5,19]; however we are not aware of any efficient algorithm that would return integer solutions.

4 Attacks on (Variants of) the Key Exchange Protocol

We now turn to isogeny problems with additional information, as in De Feo-Jao-Plût's protocols.

4.1 Problem Statement

In this section we consider the following problem.

Problem 3. *Let p be a prime. Let $N_1, N_2 \in \mathbb{Z}$ be coprime. Let E_0 be a supersingular elliptic curve over $\mathbb{F}_{p^2}$. Let $\phi_1 : E_0 \to E_1$ be an isogeny of degree N_1. Let R_0, R_1 be subrings of* $\mathrm{End}(E_0)$, $\mathrm{End}(E_1)$ *respectively. Given N_1, E_1, R_0, R_1 and the image of ϕ_1 on the whole N_2 torsion, compute ϕ_1.*

Remark. The most generic case for this problem is $R_0 = R_1 = \mathbb{Z}$, namely only the scalar multiplications are known (and do not need to be explicitly given). If E_0 is defined over $\mathbb{F}_p$ we can take $R_0 = \mathbb{Z}[\pi_p]$ where π_p is the Frobenius. In some previous implementation works [4,7] it was suggested for efficiency reasons to use special curves in the key exchange protocol, such as a curve with j-invariant $j = 1728$. In this case we have $R_0 = \mathrm{End}(E_0)$, and moreover R_0 contains some non scalar elements of small degrees.

4.2 Attack Model and General Strategy

We provide algorithms that use the additional information provided by the image of torsion points to solve Problem 3 with dramatic speedups compared to the basic meet-in-the-middle strategy.

All our attacks assume that the subring of endomorphisms R_0 contains more than the scalar multiplications. They are particularly efficient when special curves E_0 are used, such as in [4,7].

Another current limitation of our attacks is that they require N_2 significantly larger than N_1. This condition could plausibly be met in practice (should this paper not have warned against them!) in the following scenarios:

- In the unbalanced variant of the original protocol. We recall that this variant could a priori have been used when one party uses a static key and the other party uses an ephemeral key, as is the case for example in the public key encryption scheme.
- In the optimal degree variant of the protocol, a server might have used a static key and published the images of a very large torsion group $E_0[N_2]$, for example to allow connections with a wide range of clients using different sets of parameters.

Our basic strategy is as follows. For any known endomorphism $\theta \in \mathrm{End}(E_0)$ we can consider the endomorphism $\phi = \phi_1 \theta \hat{\phi}_1 \in \mathrm{End}(E_1)$. Moreover if θ is non scalar then ϕ is also non scalar. Using our knowledge of how ϕ_1 acts on the N_2 torsion we can also evaluate ϕ on the N_2 torsion, and hence apply the techniques from the previous section. Once we have an expression for ϕ we can use it to evaluate $\phi_1 \theta \hat{\phi}_1$ on the N_1 torsion. Since N_1 is smooth an easy discrete logarithm computation gives generators for $\ker(\phi_1 \theta \hat{\phi}_1) \cap E_1[N_1]$. The latest group contains $\ker \hat{\phi}_1$ as a cyclic subgroup of order N_1. When it is cyclic we directly recover $\ker \hat{\phi}_1$ and deduce ϕ_1; in Sect. 4.3 below we show how to do it in the general case.

Remark. Our resolution strategy requires that R_0 contains more than the scalar multiplications, as otherwise ϕ is just a scalar multiplication.

In Sects. 4.4 and 4.5 below we give two examples of attacks that can be developed using our techniques.

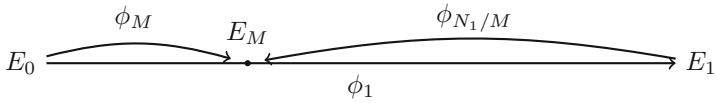
- The first attack assumes that E_0 is defined over $\mathbb{F}_p$, and moreover that E_0 has a small degree endomorphism ι such that $\mathrm{Tr}(\iota) = \mathrm{Tr}(\iota\pi_p) = 0$. This is the case for example if $j(E_0) = 1728$. Currently the attack applies only to our optimal degree variant. For well-chosen values of N_2 larger than N_1^4 the attack recovers the secret key G_1 in polynomial time.
- The second attack only requires that E_0 has a small degree endomorphism, but on the other hand it needs $\log N_2 = O(\log^2 N_1)$ to recover the secret key G_1 in polynomial time. This attack deviates from the basic strategy explained above and it instead uses some recursive step. We provide a heuristic analysis and some experimental support for this attack for both the unbalanced and the optimal degree variants.

Both attacks are heuristic, as their analysis makes unproven assumptions on factorization properties of certain numbers. We leave a better analysis, further variants and improvements to further work.

4.3 Recovering ϕ_1 from $\ker(\phi_1\theta\hat{\phi}_1)$

In the strategy outlined above we need to recover ϕ_1 from $\ker(\phi_1\theta\hat{\phi}_1)$. Here we give a method to do this and we show that the method is efficient. For simplicity we assume without loss of generality that $\deg\theta$ is coprime with N_1.

Let $G := \ker(\phi_1\theta\hat{\phi}_1) \cap E_1[N_1]$. Clearly $\ker\hat{\phi}_1$ is a cyclic subgroup of order N_1 in G. When G is cyclic this immediately gives $\ker\hat{\phi}_1$. When G is not cyclic, let $M|N_1$ be the largest integer such that $E_1[M] \subset G$. The isogeny $\phi_1 : E_0 \to E_1$ can be decomposed as an isogeny ϕ_M of degree M from E_0 to a curve E_M, and a second isogeny of degree N_1/M from E_M to E_1. We denote by $\phi_{N_1/M}$ the *dual* of this second isogeny, namely $\phi_{N_1/M} : E_1 \to E_M$ and $\phi_1 = \hat{\phi}_{N_1/M}\phi_M$. This is represented in the picture below:



Clearly, recovering ϕ_M and $\phi_{N_1/M}$, or equivalently their kernels, is sufficient to recover ϕ_1. The second isogeny $\phi_{N_1/M}$ is the easiest one to recover:

Lemma 4. *We have* $\ker\phi_{N_1/M} = M\left(\ker(\phi_1\theta\hat{\phi}_1) \cap E_1[N_1]\right)$.

Proof: Clearly $\ker\phi_{N_1/M} = M\ker\hat{\phi}_1$. The later is a cyclic subgroup of $M\left(\ker(\phi_1\theta\hat{\phi}_1) \cap E_1[N_1]\right)$ of order N_1/M. By our definition of M, the group $M\left(\ker(\phi_1\theta\hat{\phi}_1) \cap E_1[N_1]\right)$ is cyclic, hence equal to $M\ker\hat{\phi}_1$ as well. □

We now focus on ϕ_M, and we first identify a property that its kernel must satisfy:

Lemma 5. *We have* $\theta(\ker \phi_M) = \ker \phi_M$.

Proof: Equivalently, we want to prove $\theta^{-1}(\ker \phi_M) = \ker \phi_M$. We have $\ker \phi_M = \ker \phi_1 \cap E_0[M] = \hat{\phi}_1(E_1[M])$ and similarly $\theta^{-1}(\ker \phi_M) = \theta^{-1}(\ker \phi_1) \cap E_0[M] = \ker(\phi_1\theta) \cap E_0[M]$, so we can rephrase the lemma as $\hat{\phi}_1(E_1[M]) = \ker(\phi_1\theta) \cap E_0[M]$.

Since $\hat{\phi}_1(E_1[N_1])$ is cyclic, so is $\hat{\phi}_1(E_1[M])$. Therefore $E_1[M] \subset \ker(\phi_1\theta\hat{\phi}_1) \cap E_1[M]$ if and only if $\hat{\phi}_1(E_1[M]) \subset \ker \phi_1\theta$.

By the definition of M we have $E_1[M] \subset \ker(\phi_1\theta\hat{\phi}_1) \cap E_1[M]$ so $\hat{\phi}_1(E_1[M]) \subset \ker \phi_1\theta$. Moreover M is the largest such integer and $\hat{\phi}_1(E_1[M])$ is cyclic, so the equality holds. □

As we know the endomorphism θ, we can evaluate its action on the M torsion and identify potential candidates for $\ker \phi_M$.

Lemma 6. *Let* k *be the number of distinct prime factors of* M. *Then there are at most* 2^k *cyclic subgroups* H *of order* M *in* $E_0[M]$ *such that* $\theta(H) = H$.

Proof: Let $\{P, Q\}$ be a basis for $E_0[M]$, and let α, β be integers such that $\ker \phi_M = \langle \alpha P + \beta Q \rangle$. We have $\gcd(\alpha, \beta, M) = 1$.

The action of θ on $E_0[M]$ can be described by a matrix $m = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \in GL(2, \mathbb{Z}_M)$ such that $\theta(P) = aP + bQ$ and $\theta(Q) = cP + dQ$. Moreover we have $\det(m) = ad - bc = \deg \theta \bmod M$ and $\mathrm{Tr}(m) = a + d = \mathrm{Tr}(\theta) \bmod M$.

The condition $\theta(\ker \phi_M) = \ker \phi_M$ from Lemma 5 now becomes

$$\langle \alpha P + \beta Q \rangle = \langle (a\alpha + c\beta)P + (b\alpha + d\beta)Q \rangle$$

or equivalently

$$(a\alpha + c\beta)\beta = (b\alpha + d\beta)\alpha \bmod M,$$

or

$$c\beta^2 + (a - d)\alpha\beta - b\alpha^2 = 0 \bmod M.$$

The latest has solutions if and only the discriminant

$$(a - d)^2 - 4bc = (\mathrm{Tr}(\theta))^2 - 4 \deg \theta \bmod M$$

is a quadratic residue, and this is the case by assumption. Clearly there are at most two solutions modulo any prime $\ell | M$, and by Hensel's lifting lemma a solution modulo a prime $\ell | M$ determines a unique solution modulo any power of ℓ dividing M. □

We remark that when N_1 is smooth, our proof implicitly provides an efficient algorithm to identify all the candidate kernels. When N_1 is a prime power then k is at most one, and we are done. Our last lemma shows that for powersmooth numbers, the expected value of k is small enough to allow a polynomial time exhaustive search of all candidate kernels.

Lemma 7. *Let N_1 be a powersmooth number. Assume ϕ_1 be chosen uniformly at random among all isogenies of degree N_1 from E_0. Then the expected value of k is bounded by $2 \log \log N_1$.*

Proof: Clearly the number of distinct prime factors of N_1 is smaller than $\log_2 N_1$. In the proof of the previous lemma we showed that for every prime ℓ dividing $M|N_1$, there are at most two candidate cyclic subgroups H_ℓ such that $\theta(H_\ell) = H_\ell$. We can therefore bound the expected value of k by

$$E[k] \leq \sum_{\substack{\ell | N_1, \ell \text{ prime} \\ \ell \leq \log N_1}} \frac{2}{\ell + 1} < \sum_{\ell \leq \log N_1} \frac{2}{\ell} \approx 2 \int_1^{\log N_1} \frac{1}{\ell} \approx 2 \log \log N_1.$$

□

4.4 Attack When E_0 Is Special

In this section we focus on the optimal degree variant of the protocol. We assume E_0 is defined over $\mathbb{F}_p$, so that $\mathrm{End}(E_0)$ contains the Frobenius endomorphism $\pi_p : (x, y) \to (x^p, y^p)$. Moreover we assume $\mathrm{End}(E_0)$ contains some non scalar element ι with small norm q such that $\mathrm{Tr}(\iota) = \mathrm{Tr}(\iota\pi_p) = 0$. (Maximal orders with minimal such ι were called special in [13].) Then clearly the attacker knows π_p and they can efficiently compute ι by testing all isogenies of small degree. We consider an endomorphism of E_1 defined by

$$\phi = \phi_1(a\iota\pi_p + b\pi_p + c\iota)\hat{\phi}_1 + d,$$

with degree

$$\deg \phi = N_1^2 pqa^2 + N_1^2 pb^2 + N_1^2 qc^2 + d^2.$$

Remark: There is no gain of generality in allowing scalar components in R_0: indeed $\phi_1 \mathbb{Z} \hat{\phi}_1 = N_1 \mathbb{Z} \subset R_1$.

Similarly as before, our goal is now to find tuples of integers (a, b, c, d) such that $\deg \phi = N_1' N_2$ and N_1' is small. We first discuss some elementary properties of the solutions.

Lemma 8. *Let (a, b, c, d) defining ϕ as above, with $\deg \phi = N_1' N_2$ for some N_1'. Then*

- *$N_1' N_2$ is a square modulo N_1^2;*
- *except for "exceptional" parameters, $N_1' N_2$ is not much smaller than N_1^4;*
- *except for "exceptional" parameters, N_1 is not much smaller than p.*

Proof: We have $d^2 = N_1' N_2 \bmod N_1^2$. For any N_1' this defines d modulo N_1^2 up to sign, hence except for exceptional parameters d^2 will not be much smaller than N_1^4. We then have

$$pqa^2 + pb^2 + qc^2 = \frac{N_1' N_2 - d^2}{N_1^2} \approx N_1^2,$$

and the value of c is defined modulo p up to sign (assuming such a value exists). Except for exceptional parameters c^2 will not be much smaller than p^2, hence N_1 will not be much smaller than p. □

Parameter restriction. Recall that in this section we focus on the optimal degree variant, hence N_1 and N_2 are powersmooth numbers. From now on, we assume that $N_1 > p$, that $N_2 \approx N_1^4$ and that N_2 is a square modulo N_1^2. This ensures that all the conditions identified in Lemma 8 are satisfied provided N_1' is a square modulo N_1^2.

Note that we can always ensure that a powersmooth number N_2 is also a square modulo N_1^2 by dividing and/or multiplying it by a well-chosen small prime. In the first case we will have to work with a slightly smaller N_2 value in our attack, and in the second case we will have to perform some small guess on the images of the full N_2 torsion.

Algorithm. We now describe an algorithm that computes a tuple (a, b, c, d) that can be used in our attack. We first attempt to find a solution with $N_1' = 1$, and when this fails we successively increase N_1' to the next square. For a given N_1', the value of d is determined modulo N_1^2 up to the sign. We try possible values of d until we find one such that $q \cdot \frac{N_1' N_2 - d^2}{N_1^2}$ is a square modulo p. At this point we try random values for c satisfying the congruence condition, until the equation

$$a^2 q + b^2 = \frac{N_1' N_2 - d^2 - pc^2}{pN_1^2}$$

has a solution, which we compute with Cornacchia's algorithm. This algorithm is detailed below in Algorithm 2.

The complexity of this algorithm is analyzed in the following lemma.

Lemma 9. *Let all parameters be restricted as above. Under plausible heuristic assumptions Algorithm 2 terminates in polynomial time.*

Proof: Computing quadratic residues (Step 5), modular square roots (Steps 3 and 12) and Cornacchia's algorithm (Step 17) all run in polynomial time.

Heuristically, the quadratic residuosity condition in Step 5 will be satisfied for every other value of d, so the algorithm will reach Step 12 with a very small value of N_1'. Consequently in Step 4 we expect $m = \frac{N_1' N_2 - d^2}{N_1^2} \approx N_1^2$ and in Step 13 we expect $m/p \approx N_1^2/p > p$. In Step 15 we expect $n = \frac{N_1' N_2 - d^2 - c^2 N_1^2 q}{N_1^2 p} \approx \frac{N_1' N_2}{N_1^2 p} \approx \frac{N_1' N_1^2}{p} > N_1' p \approx p$.

As long as $\log N_1 = O(\log p)$, the expected number of random trials on c until n is prime is therefore $\log n \approx O(\log p)$. Moreover by Dirichlet's density theorem the density of primes represented by the norm form $a^2 q + b^2$ is $1/2H(q) > \sqrt{q}/2$ where $H(q) < \sqrt{q}$ is the class number of $\mathbb{Q}[\sqrt{-q}]$. Finally under the Generalized Riemann Hypothesis we have $q = O((\log p)^2)$ [1]. This shows that a polynomial number (in the security parameter λ) of values r must be tested in Step 13 until

Algorithm 2. Finding attack parameters when E_0 is special

Require: N_1, N_2, q as above.
Ensure: Parameters (a, b, c, d) and N_1' for an attack.
1: $i \leftarrow 1$.
2: $N_1' \leftarrow i^2$.
3: Let d such that $0 \leq d \leq N_1^2$ and $d^2 = N_1' N_2 \bmod N_1^2$.
4: $m \leftarrow \frac{N_1' N_2 - d^2}{N_1^2}$.
5: **if** mq is not a square modulo p **then**
6: **if** $d < N_1' N_2 - N_1^2$ **then**
7: $d \leftarrow d + N_1^2$.
8: **go to** Step 4.
9: **else**
10: $i \leftarrow i + 1$.
11: **go to** Step 2.
12: Let $\hat{c}$ such that $0 \leq \hat{c} < p$ and $q\hat{c}^2 = m \bmod p$.
13: Let r be a random integer in $[0, m/p]$.
14: $c \leftarrow \hat{c} + rp$.
15: $n \leftarrow \frac{N_1' N_2 - d^2 - c^2 N_1^2 q}{N_1^2 p}$.
16: **if** n has an easy factorization (for example if n is prime) **then**
17: Solve equation $a^2 q + b^2 = n$ with Cornacchia's algorithm
18: **if** there is no solution **then**
19: **go to** Step 13.
20: **return** (a, b, c, d, N_1').

a suitable one is found. Since the expected size of m/p is bigger than p, the algorithm is expected to terminate in polynomial time with a solution. □

We deduce the following:

Proposition 2. *Let N_1 and N_2 be powersmooth numbers as in our optimal degree variant of Jao-De Feo's protocol. Assume moreover that $N_1 > p$, that $N_2 \approx N_1^4$ and that N_2 is a square modulo N_1^2. Then under plausible heuristic assumptions Problem 3 can be solved in polynomial time when the initial curve E_0 has j invariant $j = 1728$, or more generally when the curve is "special" in the sense of [13].*

Remark. In the original and unbalanced variants of the protocol we have $N_1 N_2 < p$ so $N_1' > N_1^2 p / N_2 > N_1$, unless $a = b = 0$. In the next section we provide an attack that works in this setting.

4.5 Attack When $R_0 = \mathbb{Z} + \theta\mathbb{Z}$ (with $\deg \theta$ Small) and $R_1 = \mathbb{Z}$

An algorithm to recover ϕ using only the scalar multiplications of E_1 and the image of ϕ on the N_2 torsion was described in Sect. 3.3. However this in combination with our basic strategy above does not a priori provide any speedup on the straighforward meet-in-the-middle approach. Indeed we have

$\deg\phi = N_1^2 \deg\theta \approx N_1^2$ in the most favorable case (when $\deg\theta = 1$) so by the analysis of Sect. 3.3 we expect to have at best $N_1' \approx \sqrt{D} \approx \sqrt{\deg\phi} \approx N_1$. We therefore modify the basic strategy.

Modified Strategy. We adapt the techniques of Sect. 3.3 to reduce Problem 3 to another instance of itself with smaller parameters $N_1' < N_1/2$ and N_2' some factor of N_2. After repeating this reduction step $O(\log N_1)$ times we end up with an instance of Problem 3 where N_1 is sufficiently small that it can be solved in polynomial time with a meet-in-the-middle approach.

Parameter Restriction. We will require that $\mathrm{End}(E_0)$ has some non scalar element θ of small degree (which does not need to be explicitly given, as it can then be computed efficiently by trying all isogenies of this degree). This is for example the case in Costello et al.'s implementation [4] where $j = 1728$. In our reduction we will also require $N_2/N_2' > 2N_1\Delta_\theta$ where $\Delta_\theta = \deg\theta - \frac{1}{4}\mathrm{Tr}^2\,\theta$. This implies that we will need to start with parameters such that $\log N_2$ is at least $O(\log^2 N_1)$. Note that in the original De Feo-Jao-Plût protocols we had $N_1 \approx N_2$.

Reduction Step. We fix some $\theta \in \mathrm{End}(E_0)$ with small norm q, and let $D_\theta := \deg\theta - \frac{1}{4}\mathrm{Tr}^2\,\theta$. Then we choose some factor $\tilde{N}_2$ of N_2 such that $\tilde{N}_2 > KN_1q$ for some $K > 1$, and $-D_\theta$ is a square modulo $\tilde{N}_2$. We proceed as in Sect. 3.3 to compute a, b and N_1' such that $\deg(a\phi_1\theta\hat{\phi}_1 + b) = N_1'\tilde{N}_2$ and N_1' is as small as possible. Namely, we choose τ such that $\tau^2 = -D_\theta \bmod \tilde{N}_2$, then we compute a short vector in a two-dimensional lattice generated by two vectors $(\tilde{N}_2, 0)$ and $(\tau, 1)$ with a weighted norm $||(x, y)|| = (x^2 + D_\theta y^2)^{1/2}$, and we deduce a, b and N_1'. If $N_1' > N_1/2$ we start again with a new square root of $-D_\theta$ modulo $\tilde{N}_2$, or with a new $\tilde{N}_2$ value.

If $N_1' < N_1/2$ we define $\phi_{N_1'}, \phi_{\tilde{N}_2}$ two (still unknown) isogenies of degrees N_1' and $\tilde{N}_2$ such that $a\phi_1\theta\hat{\phi}_1 + b = \phi_{N_1'}\phi_{\tilde{N}_2}$. We evaluate $a\phi_1\theta\hat{\phi}_1 + b$ on the $\tilde{N}_2$ torsion to identify the $\tilde{N}_2$ part of the kernel of $a\phi_1\theta\hat{\phi}_1 + b$, then the corresponding isogeny. We evaluate this isogeny on the $N_2' = N_2/\tilde{N}_2$ torsion, and deduce the action of $\phi_{N_1'}$ on the N_2' torsion. We then apply the reduction step recursively to compute some representation of $\phi_{N_1'}$. Finally, we evaluate $(\phi_{N_1'}\phi_{\tilde{N}_2} - b)/a$ on the N_1 torsion to identify $\ker\hat{\phi}_1$, and from there we compute a more canonical expression for ϕ.

Complexity analysis. Our reduction procedure implicitly relies on the following informal assumption:

Assumption 1. *Let $K > 1$ be a "small" constant, and suppose that D_θ is "small". The probability that a "random" powersmooth value $\tilde{N}_2 > KN_1q$ leads to $N_1' < N_1/2$ is "large".*

Note that following the analysis of Lemma 2 we expect to find N_1' of size at most $N_1\sqrt{D_\theta}$. Assumption 1 tells that with some probability on the choice of $\tilde{N}_2$,

we can find a value N_1' smaller than this bound by at least a (small) factor $2\sqrt{D_\theta}$. This assumption seems very plausible. Using lattice terminology, the expectation on N_1' comes from the well-known Gaussian heuristic, and Assumption 1 tells that the proportion of lattices with small deviations from this heuristic is significant. In continued fraction terminology, Assumption 1 considers the proportion of values $\tilde{N}_2$ such that some rational fraction approximation of τ/N_1 is a little bit better than what is guaranteed by the bounds, and tells that this proportion is significant. Finally, Assumption 1 receives further support from our experiments described below.

We deduce the following result:

Proposition 3. *Let N_1 and N_2 be coprime smooth numbers, with $\log N_2 = O(\log^2 N_1)$. Then under plausible heuristic assumptions Problem 3 can be solved in polynomial time when the initial curve E_0 has a small degree endomorphism.*

Proof: All subroutines in our reduction procedure require at most polynomial time, and under Assumption 1 these steps will only be executed a polynomial number of times.

Remark: Suppose $p = 3 \bmod 4$ and suppose $j_0 = 1728$ as in Costello et al.'s implementation [4]. In this case there exists a non scalar endomorphism $\iota \in \mathrm{End}(E_0)$ with norm 1 and trace 0. Any $\theta \in \mathrm{End}(E_0)$ must either have a large norm or be of the form $\theta = a\iota + b$ for two small $a, b \in \mathbb{Z}$. In the last case we then have $\Delta_\theta = a^2$, so $-\Delta_\theta$ is a square modulo some prime r if and only if -1 is a square modulo r. This implies that no prime factor r of N_2 with $r = 3 \bmod 4$ can be used in our attack. On the other hand, any prime factors with $r = 1 \bmod 4$ can be used in the attack.

Experiments for the optimal degree variant. We wrote a small Magma program [22] to compute the successive pairs of parameters (a, b) to use in our attack, and test the heuristic assumptions involved in our analysis (the code is available in the eprint version of this paper [14]). In our experiments we generate random p values, choose N_1 powersmooth and then search for a coprime $\tilde{N}_2 > 2qN_1$ leading to $N_1' < N_1/2q$. We repeat this recursively until N_1' is small enough (smaller than some polylog bound in p). We used $K = 2$ in these experiments. For 80-bit security parameters our program gives the parameters of an attack in a few seconds. The full attack requires isogenies of degree at most about 36000.

Experiments for the unbalanced variant. We also ran attack experiments for the unbalanced protocol variant. In all experiments we took $\ell_1 = 2$ and $\ell_2 = 5$. We considered values of e_1 between 20 and 100, and we searched for the minimal value of e_2 such that the attack could reduce N_1 to a value smaller than 100. Table 1 provides some successful attack parameters. In addition to e_1 and e_2 it shows the value $\left\lceil \frac{e_2 \log_2 5}{e_1^2} \right\rfloor$ (which seems close to a constant 1/2, as expected), the value K used for these parameters, and the number of reduction steps used. Our Magma code is provided in the eprint version of this paper [14].

Table 1. Some successful attack parameters against the unbalanced variant ($\ell_1 = 2$ and $\ell_2 = 5$)

e_1	e_2	$\left\lceil \frac{e_2 \log_2 5}{e_1^2} \right\rceil$	K	# steps
20	102	0.59	50	11
30	194	0.50	50	17
40	330	0.48	50	22
50	405	0.38	10	30
60	610	0.39	10	38
70	1047	0.50	2	61
80	1473	0.53	2	72
90	1775	0.51	2	80
100	2180	0.51	2	90

Remark: The parameter K must be larger than 1 in our attack as $N_1' > N_1^2 q/\tilde{N}_2$ for any $a \neq 0$. We experimentally observed that $K = 2$ was sufficient in the optimal degree variant to make N_1 decrease by a factor 2 at each reduction step. The unbalanced variant leaves less flexibility in the parameter choice, so we did not impose a factor 2 decrease on N_1 (and in fact we even allowed it to increase in some reduction steps). We observed that lower values of K were then sufficient. We have also observed experimentally that the value of K has some moderate impact on the overall performances of the attack (required size for N_2, number of reduction steps). We leave a thorough investigation of optimal parameter choices for our attack to further work.

Remark: When N_2 is too small to execute $O(\log N_1)$ reduction steps, then we may replace the missing last reduction steps by a final meet-in-the-middle strategy. Depending on the final size of N_1' and on its largest prime factor this may still provide some exponential speedup over the basic meet-in-the-middle strategy. We note, however, that for the original parameters proposed by Jao and De Feo, at most one recursive step can be performed. In this case it might be possible to find some (exceptional) set of parameters that would improve the best attack by a few bits, but for most parameters we do not expect any savings.

Possible Extensions. One can vary R_0 and R_1 depending on the attack model, or consider variants of Problem 3 involving several isogenies, and derive similar attacks. We leave details to the reader and further work.

5 Impact and Perspectives

The techniques developed in this paper solve some isogeny problems using the images of certain torsion points by the isogenies. Such images are revealed in De Feo-Jao supersingular key exchange protocol as well as the public key encryption

and signature scheme that derive from it (see [7,24] and the first signature scheme of [9]). Until now all existing attacks against these protocols made no use at all of this auxiliary information.

At the moment our techniques do not apply to the parameters originally proposed in these protocols. However they apply on some natural variants of them, and they issue a warning that the auxiliary information might weaken isogeny problems. One could also fear that further developments of our techniques and particular attack models will be able to threaten the original protocol itself.

In anticipation of potential future improvements of our attacks, we recommend to avoid the use of special E_0 in the protocols, as any (partial) knowledge of the endomorphism ring of E_0 may a priori be useful to the attacker with our techniques. We stress, however, that the only known algorithm to avoid special curves for E_0 consists in generating a special curve and then performing a random walk from there to obtain a truly random curve; depending on the context this procedure might still allow some form of backdoor attack. An algorithm that could generate a random supersingular j-invariant without performing a random walk from a curve with known endomorphism ring would be a handy tool for designing cryptosystems based on supersingular isogeny problems. Of course, the algorithm may come with additional insight on the underlying Mathematics, which might also help further cryptanalysis. We would like to encourage research in this direction.

We note that the hash function proposed by Charles-Goren-Lauter [2] can also be attacked when starting from a curve with known endomorphism ring. There is also a corresponding "backdoor collision attack"; however the attack is less powerful than above as it can be detected and any use of the backdoor will leak it. We refer to [15] for details of this attack.

The second signature scheme of [9] relies on the endomorphism ring computation problem for random curves, with no extra information leaked, and is not affected by our techniques. In contrast to the isogeny problem variants considered in this paper, we are not aware of any cryptanalysis result that affects the endomorphism ring computation problem, and we believe that cryptosystems based on this problem offer the strongest security guarantees in the area of isogeny-based cryptography. Of course, cryptanalysis research in this direction is also fairly scarce despite some early work by Kohel [12], and more cryptanalysis will be needed to gain confidence on their security.

Acknowledgments. We thank Bryan Birch, Jonathan Bootle, Luca De Feo, Steven Galbraith, Chloe Martindale, Lorenz Panny and Yan Bo Ti, as well as the anonymous reviewers of the Asiacrypt 2017 conference for their useful comments on preliminary versions of this paper. This work was developed while the author was at the Mathematical Institute of the University of Oxford, funded by a research grant from the UK government.

References

1. Ankeny, N.C.: The least quadratic non residue. Ann. Math. **55**(1), 65–72 (1952)
2. Charles, D.X., Lauter, K.E., Goren, E.Z.: Cryptographic hash functions from expander graphs. J. Cryptol. **22**(1), 93–113 (2009)
3. Coggia, D.: Implémentation d'une variante du protocole de key-exchange SIDH (2017). https://github.com/dnlcog/sidh_variant
4. Costello, C., Longa, P., Naehrig, M.: Efficient algorithms for supersingular isogeny Diffie-Hellman. In: Robshaw, M., Katz, J. (eds.) CRYPTO 2016. LNCS, vol. 9814, pp. 572–601. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-53018-4_21
5. Cremona, J.E., Rusin, D.: Efficient solution of rational conics. Math. Comput. **72**(243), 1417–1441 (2003)
6. Delfs, C., Galbraith, S.D.: Computing isogenies between supersingular elliptic curves over $\mathbb{F}_p$. Des. Codes Crypt. **78**(2), 425–440 (2016)
7. Feo, L.D., Jao, D., Plût, J.: Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. J. Math. Cryptol. **8**(3), 209–247 (2014)
8. Galbraith, S.D., Petit, C., Shani, B., Ti, Y.B.: On the security of supersingular isogeny cryptosystems. In: Cheon, J.H., Takagi, T. (eds.) ASIACRYPT 2016. LNCS, vol. 10031, pp. 63–91. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-53887-6_3
9. Galbraith, S.D., Petit, C., Silva, J.: Identification protocols and signature schemes based on supersingular isogeny problems. In: Takagi, T., Peyrin, T. (eds.) ASIACRYPT 2017, Part I. LNCS, vol. 10624, pp. 3–33. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-70694-8_1
10. Gélin, A., Wesolowski, B.: Loop-abort faults on supersingular isogeny cryptosystems. In: Lange, T., Takagi, T. (eds.) PQCrypto 2017. LNCS, vol. 10346, pp. 93–106. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-59879-6_6
11. Jao, D., De Feo, L.: Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. In: Yang, B.-Y. (ed.) PQCrypto 2011. LNCS, vol. 7071, pp. 19–34. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-25405-5_2
12. Kohel, D.: Endomorphism rings of elliptic curves over finite fields. PhD thesis, University of California, Berkeley (1996)
13. Kohel, D., Lauter, K., Petit, C., Tignol, J.-P.: On the quaternion ℓ-isogeny path problem. LMS J. Comput. Math. **17A**, 418–432 (2014)
14. Petit, C.: Faster algorithms for isogeny problems using torsion point images. IACR Cryptology ePrint Archive, 2017:571 (2017)
15. Petit, C., Lauter, K.: Hard and easy problems in supersingular isogeny graphs (2017)
16. Canfield, R., Erdös, P., Pomerance, C.: On a problem of Oppenheim concerning "factorisatio numerorum". J. Number Theory **17**, 1–28 (1983)
17. Rostovtsev, A., Stolbunov, A.: Public-key cryptosystem based on isogenies. Cryptology ePrint Archive, Report 2006/145 (2006). http://eprint.iacr.org/
18. Silverman, J.: The Arithmetic of Elliptic Curves. Springer Verlag, New York (1986)
19. Simon, D.: Quadratic equations in dimensions 4, 5 and more. Preprint (2005). http://www.math.unicaen.fr/~simon/
20. Ti, Y.B.: Fault attack on supersingular isogeny cryptosystems. In: Lange, T., Takagi, T. (eds.) PQCrypto 2017. LNCS, vol. 10346, pp. 107–122. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-59879-6_7

21. Vignéras, M.-F.: Arithmétique des Algèbres de Quaternions. Springer, Heidelberg (2006). https://doi.org/10.1007/BFb0091027
22. Fieker, C., Steel, A., Bosma, W., Cannon, J.J. (eds.): Handbook of Magma functions, edition 2.20 (2013). http://magma.maths.usyd.edu.au/magma/
23. Xi, S., Tian, H., Wang, Y.: Toward quantum-resistant strong designated verifier signature from isogenies. Int. J. Grid Util. Comput. **5**(2), 292–296 (2012)
24. Yoo, Y., Azarderakhsh, R., Jalali, A., Jao, D., Soukharev, V.: A post-quantum digital signature scheme based on supersingular isogenies. Financial Crypto (2017)