Trade-off between Performance and Security for Supersingular Isogeny-Based Cryptosystems

Claudio Téllez[1], **Fábio Borges**[1]

[1]National Laboratory for Scientific Computing (LNCC)
25.651-075 – Petrópolis – RJ – Brazil

{ctellez,borges}@lncc.br

Abstract. *Cryptosystems based on isogenies between supersingular elliptic curves are considered promising candidates for a post-quantum era. Their security is based on the intractability of the Computational Supersingular Isogeny Problem (CSSIP) and of the Decisional Supersingular Product Problem (DSSPP). For this reason, there have been many important breakthroughs in supersingular isogeny cryptography in recent years. The purpose of our work is to provide a complexity analysis of the trade-off between performance and security for supersingular isogeny-based cryptosystems (SSI) in comparison with Discrete Logarithm Problem (DLP) and Integer Factorization Problem (IFP). We show how the complexities increase for the attack algorithms when the key lengths become longer. As SSI achieves small key sizes at practical security levels, it is a strong potential quantum-resistant cryptosystem.*

1. Introduction

Most cryptosystems nowadays considered safe are based on hard mathematical problems as the Integer Factorization Problem (IFP) and the Discrete Logarithm Problem (DLP). Such problems are significantly hard to be solved by computers of the kind that we have today. However, a hypothetical strong quantum computer could easily solve them and break their related cryptosystems using Shor's algorithm [Shor 1995]. For problems that Shor's algorithm cannot handle, Grover's algorithm [Grover 1996] can speed up the searching for cryptographic keys. Hence, the eventual construction of a cryptographically relevant quantum computer poses an important threat to privacy and information security.

Although sufficiently strong quantum computers are not a reality yet, there is a considerable amount of research effort towards progress in quantum computing hardware. Besides, the current race to build quantum computers includes large high-tech companies like IBM, Google, Intel, and Microsoft. A recent NIST's report [Chen et al. 2016] estimates that a quantum computer capable of breaking 2000-bit RSA could appear around 2030. This strongly indicates that we need further efforts in post-quantum cryptography that take into account both security and performance requirements.

Among the several alternatives that come on the scene as quantum-safe, cryptosystems based on isogenies between supersingular elliptic curves are considered promising candidates. Their security is based on the supposed intractability of the Computational Supersingular Isogeny Problem (CSSIP) and of the Decisional Supersingular Product Problem (DSSPP) for which we still do not know any quantum algorithms able to solve them. Although the original proposal [Rostovtsev and Stolbunov 2006] of the problem of finding isogenies between ordinary elliptic curves was proven to be vulnerable to quantum

computers [Childs et al. 2010], this weakness was overcome when supersingular elliptic curves are used instead of ordinary ones.

The purpose of this paper is to discuss the trade-off between performance and security for supersingular isogeny-based cryptosystems (SSI), to provide a better assessment regarding whether the protocols meet performance objectives for desired levels of security. In the next section, we present a theoretical overview of SSI. After that, in the third section, we discuss the performance and security of SSI in comparison to other relevant cryptosystems. We present our conclusions in the closing section.

2. Supersingular isogeny-based cryptosystems

The first proposal to the use of elliptic curves in cryptography (ECC) was made independently by [Koblitz 1987] and [Miller 1985] in the mid-80's. The security of ECC is based on the DLP in the group of points formed by an elliptic curve defined over a finite field. As for their attractiveness, it is due to their capacity to provide equivalent security of other existing public key schemes with smaller keys (about an order of magnitude smaller than RSA). An extensive presentation on the theory of elliptic curves can be found in [Silverman 1986] and for a review of ECC, we refer the reader to [Washington 2008].

In a quantum computing era, however, Shor's algorithm could find discrete logarithms in polynomial time, rendering elliptic curve cryptosystems inappropriate for post-quantum cryptography. In fact, a recent resource estimate shows that Shor's algorithm could break a curve with 128-bit security level (256-bit module) using 2330 qubits and 1.26×10^{11} Toffoli gates, much less than the 6146 qubits and 1.86×10^{13} Toffoli gates necessary for factoring an RSA 3072 [Roetteler et al. 2017]. That result shows that it would be easier for Shor's algorithm to break ECC than RSA.

In 2006, Rostovtsev and Stolbunov proposed a new mathematical problem related to elliptic curves, which would be hypothetically strong against quantum attacks: the problem of computing isogenies between ordinary elliptic curves E_1 and E_2 defined over a finite field $\mathbb{F}_q$. Isogeny-based cryptography using ordinary elliptic curves was dismissed as impractical since 2010 [Childs et al. 2010]. However, we present some of its basic ideas to clarify the discussion of the supersingular case in the next subsection.

2.1. Foundations

In the general case, given an elliptic curve over a field $\mathbb{F}$ in the Weierstraß form

$$E(\mathbb{F}) : y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6 \tag{1}$$

where the coefficients are elements of $\mathbb{F}$. Let

$$\begin{aligned}
b_2 &= a_1^2 + 4a_2 \\
b_4 &= 2a_4 + a_1a_3 \\
b_6 &= a_3^2 + 4a_6 \\
b_8 &= a_1^2a_6 + 4a_2a_6 - a_1a_3a_4 + a_2a_3^2 - a_4^2 \\
c_4 &= b_2^2 - 24b_4 \\
c_6 &= -b_2^3 + 36b_2b_4 - 216b_6 \\
\Delta &= -b_2^2b_8 - 8b_4^3 - 27b_6^2 + 9b_2b_4b_6.
\end{aligned}$$

We define Δ as the *discriminant* of $E(\mathbb{F})$ and the j-invariant of $E(\mathbb{F})$ as

$$j(E) = c_4^3/\Delta.$$

When $char(\mathbb{F}) \neq 2,\ 3$, we can reduce the curve equation (1) to

$$E(k) : y^2 = x^3 + Ax + B. \tag{2}$$

Where $A = -c_4/12$, $B = -c/216$ and the j-invariant becomes

$$j(E) = 1728\frac{4A^3}{4A^3 + 27B^2}.$$

Given a finite field $\mathbb{F}_p$ and an elliptic curve E over $\mathbb{F}_p$, an endomorphism is a homomorphism from E to itself. The endomorphisms of E form a ring $End(E)$ with the group addition of E as the group operation and composition as the multiplication.

An isogeny $\phi : E_1 \to E_2$ is a rational morphism that maps the point at infinity $\mathcal{O}_1$ of E_1 to the point at infinity $\mathcal{O}_2$ of E_2 (that is, that preserves identity). As such, isogenies preserve both the geometry of elliptic curves and their group structures (they are group homomorphisms). If $\phi : E_1 \to E_2$ is an isogeny defined over a field $\mathbb{F}_p$, and $\mathbb{F}_p(E_1)$ and $\mathbb{F}_p(E_2)$ are the function fields of E_1, E_2 respectively, the composition of ϕ with the functions of $\mathbb{F}_p(E_2)$ gives a subfield $\phi^*(\mathbb{F}_p(E_2))$ of $\mathbb{F}_p(E_1)$. The degree of an isogeny is the degree of the extension $[\mathbb{F}_p(E_1) : \phi^*(\mathbb{F}_p(E_2))]$ [Feo 2017]. We call isogenies of degree l as l-isogenies. When $l = 1$, an isogeny is an isomorphism, and the j-invariant is an element of the algebraic closure $\overline{\mathbb{F}}_p$ that determines an isomorphism class of elliptic curves.

Given an l-isogeny $\phi : E_1 \to E_2$, there is a unique l-isogeny $\hat{\phi} : E_2 \to E_1$, called the dual of ϕ, such that $\hat{\phi} \circ \phi = [l]$ on E_1 and $\phi \circ \hat{\phi} = [l]$ on E_2 (where $[l]$ is the application of scalar multiplication). This fact guarantees that isogeny is an equivalence relation.

If k is a finite field $k = \mathbb{F}_p$ and we have an elliptic curve E of the form (2) with coefficients from $\mathbb{F}_p$, the map $\pi : (x, y) \to (x^p, y^p)$ is called the Frobenius endomorphism with the Frobenius trace $T = p - \#E(\mathbb{F}_p)$. The Frobenius endomorphism satisfies the characteristic equation given by

$$\pi^2 - T\pi + p = 0. \tag{3}$$

The discriminant D_π of (3) plays a significant role in our discussion because of the following result.

Theorem 1. *Consider E an elliptic curve over a finite field $\mathbb{F}_p$ with Frobenius discriminant D_π, and $\left(\frac{D_\pi}{l}\right)$ a Kronecker symbol for some l-degree isogeny. If $\left(\frac{D_\pi}{l}\right) = -1$, there are no l-isogenies; if $\left(\frac{D_\pi}{l}\right) = 1$, there are two l-isogenies; if $\left(\frac{D_\pi}{l}\right) = 0$, then there are 1 or $l + 1$ l-isogenies.*

For a proof of the theorem above, see [Kohel 1996].

In the case $\left(\frac{D_\pi}{l}\right) = 1$, l is called an *Elkies isogeny degree* [Elkies 1998]. Elkies isogenies are useful for counting points on elliptic curves over finite fields.

Isogeny-based cryptosystems are based on isogeny graphs whose vertices are equivalence classes of elliptic curves, defined by the j-invariant, and whose edges are isogenies between them. For constructing those graphs, Rostovtsev and Stolbunov consider the set U of elliptic curves with equal number of points (which are isogenous, according to a theorem of [Tate 1966]) as a category whose set of morphisms is given by the isogenies between elements of U. To properly define routes on isogeny graphs, they notice that when $\#U$ is prime, the elements of U form a single cycle, and that switching to dual isogenies changes direction in a cycle. The roots of (3) are the *Frobenius eigenvalues* π_1 and π_2, each one corresponding to one cycle direction.

The resultant graphs, which encompasses prime numbers of elliptic curves connected by isogenies, are called *isogeny stars*. Rostovtsev and Stolbunov use wide enough isogeny stars for constructing cryptographic algorithms. The technical details concerning walks along rational cycles of isogenous curves are given by [Couveignes et al. 1996]. In short, given an isogeny star S, a set $L = \{l_i\}$ of Elkies isogeny degrees and a set $F = \{\pi_i\}$ of Frobenius eigenvalues, which specify positive direction for every l_i, we define a set $R = \{r_i\}$ as a *route* on S. Each r_i corresponds to a number of steps by the l_i-isogeny in the direction specified by π_i.

In their original paper, Rostotsev and Stolbunov proposed a version of the ElGamal public-key encryption technique using isogeny stars. The common parameters of their cryptosystem are:

- A finite field $\mathbb{F}_p$.
- An initial elliptic curve E_{init} specified by coefficients (A_{init}, B_{init}) from equation (2).
- A certain number d of isogeny degrees.
- A set $L = \{l_i\}$, with $1 \leq i \leq d$ of Elkies isogeny degrees.
- A set $F = \{\pi_i\}$, with $1 \leq i \leq d$ of Frobenius eigenvalues, to specify the positive direction for each $l_i \in L$.
- A limit k for number of steps by one isogeny degree in a route. For any route $\{r_i\}$, numbers of steps are selected in $-k \leq r_i \leq k$.

In this system, the private key is a route R_{priv} and the public key is an elliptic curve calculated as $E_{pub} = R_{priv}(E_{init})$, specified by (A_{pub}, B_{pub}).

For an isogeny star of order n, the required complexity of attacks is estimated at $O(n)$ isogeny computations. Using the *meet-in-the-middle* technique, the complexity of the attack is estimated at $O(\sqrt{n})$ isogeny computations [Rostovtsev and Stolbunov 2006]. [Galbraith 1999] gives another estimation at $O(p^{1/4})$. To compute a chain of q isogenies, q equations must be solved consecutively, because the j-invariant (parameter) changes at every step. This indicates that computations cannot be parallelized. The strength of the cryptosystem, then, is estimated at $O(\sqrt{n}) \sim O(p^{1/4})$ and it is exponential from $log\, p$.

Rostotsev and Stolbunov believed that their cryptosystem would be quantum-resistant because the parallelization problem also related to a quantum computer and because [Hashimoto 2018] showed that to break a cryptosystem based on multivariate polynomials would be hard for a quantum computer as well.

Soon after Rostotsev and Stolbunov's proposal of a cryptosystem based on isogenies between ordinary elliptic curves, however, [Childs et al. 2010] showed how to con-

struct elliptic curve isogenies in quantum subexponential time assuming only the Generalized Riemann Hypothesis (GRH) and using expansion properties of a Cayley graph [Jao et al. 2009]. In this way, the authors found a subexponential algorithm for computing endomorphism rings of ordinary elliptic curves exploring a connection between isogenies and hidden shifts [Stolbunov 2010] and showing that the hidden shift problem in any finite abelian group can be solved by a quantum computer using only polynomial space.

The result by [Childs et al. 2010] seemed to imply that isogeny-based cryptosystems are unfeasible for a post-quantum era. Besides presenting a poor performance in comparison to other post-quantum proposals, especially lattice-based NTRU [Hermans et al. 2010], they could be vulnerable to quantum attacks in subexponential time.

2.2. Isogeny cryptosystems based on supersingular elliptic curves

It is important to remark that the subexponential quantum attack by [Childs et al. 2010] against isogeny-based cryptosystems relies on the endomorphism ring being commutative. However, this is not the case for a supersingular elliptic curve whose endomorphism ring is isomorphic to an order in a quaternion algebra. This inspired further research on the *supersingular* case.

[Jao and Feo 2011] presented a proposal for a version of Diffie-Hellman based on isogenies between supersingular elliptic curves (SIDH). To this moment, SIDH stays immune to quantum attacks.

In comparison with Elliptic Curve Diffie-Hellman (ECDH), which uses points on one fixed elliptic curve and the group of rational points of this curve, SIDH uses supersingular isogeny classes and replaces exponentiations by quotients. Its security relies on the non-abelian structure of the set of isogenies of a supersingular elliptic curve together with the operation of composition. Table 1 shows a comparison between the original group-based Diffie-Hellman protocol, ECDH, and SIDH.

Table 1. Comparison between the algorithms.

	DH	ECDH	SIDH
Elements	Integers g	Points P in E	Curves E in isogeny classes
Secrets	exponents x	scalars k	isogenies ϕ
Computations	$g, x \mapsto g^x$	$k, P \mapsto [k]P$	$\phi, E \mapsto \phi(E)$
Hard Problem	Given g, g^x, find x	Given P, $[k]P$, find k	Given E, $\phi(E)$, find ϕ

There are several ways of defining supersingular elliptic curves in the literature. For our purposes, we can consider a given elliptic curve over a field k of characteristic $p > 0$ as *supersingular* if and only if its endomorphism ring over $\overline{k}$ has rank 4 (an order in a quaternion algebra). In comparison, the endomorphism ring of other elliptic curves has only rank 1 or 2. For a development of the basic theory of supersingular elliptic curves, we refer the reader to [Deuring 1941].

Just as Rostovtsev and Stolbunov's isogeny-based cryptosystem uses wide enough isogeny stars, the supersingular case also employs graphs for key generation and cryptographic protocols. Now, *expander graphs* are attractive because they are simultaneously

sparse and highly connected. Their expanding property makes them useful to construct communication networks. A survey of the theory of expander graphs can be found at [Hoory et al. 2006]. Here, we present just some results that will be important for our subject.

Given an expander graph G, its *expander constant* is defined as

$$h(G) = min\left\{ \frac{|\partial F|}{|F|} \mid F \subseteq V \text{ is such that } 0 < |F| < \frac{|V|}{2} \right\},$$

where F is a subset of the set of vertices V of G.

Here, we are interested in the spectral properties of expander graphs. It is a known result that if a graph G with n vertices is finite, connected, and k-regular (that is, there are exactly k edges incident with any vertex of the graph), its eigenvalues can be ordered as

$$\mu_0 = k > \mu_1 \geq \ldots \geq \mu_{n-1} \geq -k.$$

From this result, [Alon and Milman 1985] and [Dodziuk 1984] give a way to estimate the expanding constant of an expander k-regular graph G, i.e.,

$$\frac{k - \mu_1}{2} \leq h(G) \leq \sqrt{2k(k - \mu_1)},$$

where μ_1 is the first non-trivial eigenvalue of the adjacency matrix of G. In other words, the spectral gap $k - \mu_1$ provides an estimate on the expansion of a graph. If we want good expanders, we must answer the question of how big the spectral gap can be. Theorem 2 by N. Alon and R. Boppana (see [Nilli 1991] for the proof) gives a bound.

Theorem 2. *For every k-regular graph G with n vertices, we denote $\mu = \mu(G) = max\{|\mu_1|, |\mu_n|\}$ (the largest eigenvalue other than $\mu_0 = k$. Then, we have*

$$\mu \geq 2\sqrt{k-1} - o_n(1),$$

where the $o_n(1)$ term tends to zero for fixed k as $n \to \infty$.

In other words, the bound $2\sqrt{k-1}$, also known as *Ramanujan bound*, provides an optimal expanding property, which is an important feature to guarantee that a random walk on an expander graph mixes very quickly.

From the results presented above, we can define a *Ramanujan graph* as a k-regular graph G satisfying

$$\mu(G) \leq 2\sqrt{k-1}.$$

As discussed in the previous subsection, isogenies create a graph structure on the set of j-invariants defined over a finite field. In fact, the vertices of an isogeny graph are equivalence classes of elliptic curves defined by the j-invariant, and its edges represent isogenies between them. As isogeny-based cryptography requires large isogeny graphs, expander graphs in general and Ramanujan graphs in particular appear as obvious candidates. Moreover, a theorem presented in [Feo 2017] establishes that supersingular graphs (i.e., isogeny graphs from supersingular elliptic curves) are Ramanujan (for a proof, see Theorem 4.1 in [Silverman 1986]).

Supersingular curves are defined over finite fields of the form $\mathbb{F}_{p^2}$. For every prime $l \nmid p$, there are $l+1$ l-isogenies that originate from any given such supersingular curve.

[Feo et al. 2011] propose a public-key cryptosystem based on isogenies between supersingular elliptic curves. They fix the field $\mathbb{F}_q = \mathbb{F}_{p^2}$, where p is a prime of the form $l_A^{e_A} l_B^{e_B} \cdot f \pm 1$ (l_A and l_B are small primes and f is a cofactor such that p is prime).

Alice and Bob take random walks on isogeny graphs (preferably Ramanujan, for the reasons exposed above). Alice uses the graph of l_A-isogenies, and Bob uses the graph of l_B-isogenies. Those random walks on (Ramanujan) isogeny graphs are used by Alice and Bob to choose random elements that will be used to produce the proper kernels and compute the isogenies (using, for example, Vélu's formulae [Vélu 1971], from which isogenies can be constructed explicitly given their kernels). The process of key exchange is described in the algorithm 1.

Algorithm 1: SIDH

Input: A supersingular elliptic curve $E_0/\mathbb{F}_{p^2}$ with $E_0(\mathbb{F}_{p^2}) = E[l_A^{e_A} l_B^{e_B}]$, where $p = l_A^{e_A} l_B^{e_B} \pm 1$ is a prime. Public bases $\{P_A, Q_A\}$ for $E[l_A^{e_A}]$ and $\{P_B, Q_B\}$ for $E[l_B^{e_B}]$.

Output: A secret key k.

1 ALICE chooses random m_A, $n_A \in \mathbb{Z}/l_A^{e_A}\mathbb{Z}$.
2 ALICE uses $\phi_A : E \to E_A$, where $E_A = E_0/\langle m_A P_A + n_A Q_A \rangle$, $\mathfrak{P}_A = \phi_A(P_B)$, e $\mathfrak{Q}_A = \phi_A(Q_B)$.
3 ALICE sends $\mathfrak{P}_A$, $\mathfrak{Q}_A$, and E_A to BOB.
4 BOB chooses random m_B, $n_B \in \mathbb{Z}/l_B^{e_B}\mathbb{Z}$.
5 BOB uses $\phi_B : E \to E_B$, where $E_B = E_0/\langle m_B P_B + n_B Q_B \rangle$, $\mathfrak{P}_B = \phi_B(P_A)$, e $\mathfrak{Q}_B = \phi_B(Q_A)$.
6 BOB sends $\mathfrak{P}_B$, $\mathfrak{Q}_B$, and E_B to ALICE.
7 ALICE computes $E_{AB} := E_B/\langle m_A \phi_B(P_A) + n_A \phi_B(Q_A) \rangle$.
8 BOB computes $E_{BA} := E_A/\langle m_B \phi_A(P_B) + n_B \phi_A(Q_B) \rangle$.
9 ALICE computes $k = j(E_{AB})$
10 BOB computes $k = j(E_{BA})$

At the end of the process, we have

$$ker\, \phi_{AB} = \langle m_A P_A + n_A Q_A, m_B P_B + n_B Q_B \rangle = ker\, \phi_{BA}.$$

Hence, $E_{AB} \simeq E_{BA}$. Alice and Bob share the secret $j(E_{AB}) = j(E_{BA})$, the same j-invariant.

Figure 1 depicts the SIDH. The variables in green are parameters, in blue are public, and in red are private.

3. Performance and security

3.1. Security

In the case of isogeny-based cryptosystems that use supersingular elliptic curves, the DLP is not important. The security assumptions come from the following problems, which are believed to be intractable for quantum computers:

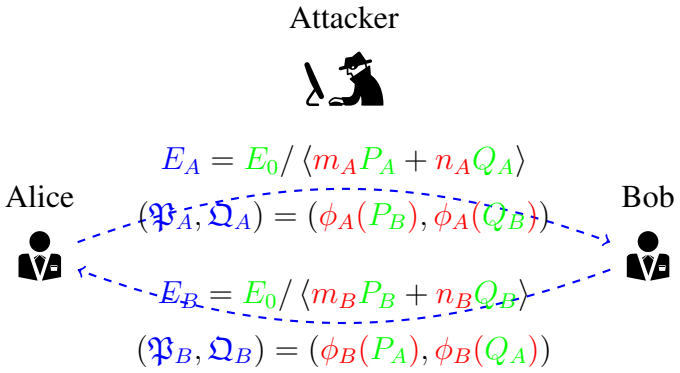

Figure 1. Picture illustrating the SIDH.

- **Computational Supersingular Isogeny Problem (CSSIP):** Let $\phi_A : E_0 \to E_A$ be an isogeny with kernel $\langle R_A \rangle$, where R_A is a random point with order $l_A^{e_A}$. Given E_A, $\phi_A(P_B)$, and $\phi_A(Q_B)$, find a generator of $\langle R_A \rangle$.
- **Decisional Supersingular Product Problem (DSSPP):** Let $\phi : E_0 \to E_3$ be an isogeny of degree $l_A^{e_A}$. Given (E_1, E_2, ϕ') sampled with probability $1/2$ from one of the distributions below, determine which distribution it comes from:
  - A random point R of order $l_B^{e_B}$ is chosen and $E_1 = E_0/\langle R \rangle$, $E_2 = E_3/\langle \phi(R) \rangle$, and $\phi' : E_1 \to E_2$ is a $l_A^{e_A}-$isogeny.
  - E_1 is chosen randomly among curves of the same cardinality of E_0. $\phi' : E_1 \to E_2$ is a random $l_A^{e_A}-$isogeny.

According to [Jao and Feo 2011], the DSSPP is at least easier than the CSSIP. Hence, the security of the SIDH protocol depends on the problem of computing an isogeny between isogenous supersingular curves. In the general case, the fastest known algorithm to find isogenies between supersingular elliptic curves has complexity of $O(\sqrt{p}\, log^2 p)$ [Charles et al. 2009]. However, for the cryptosystem presented in the previous section, we use a more specific case, for which most recent known complexities for solving the CSSIP are $O(p^{1/4})$ against classical attacks [Feo et al. 2011] and $O(p^{1/6})$ against quantum attacks [Tani 2009].

To solve the IFP, we use the general number field sieve (GNFS), which has subexponential complexity. We can compare a brute force attack in a key of x bits with the GNFS. Matching the complexity for brute force with the complexity of GNFS, we have

$$2^x = \exp\left(\left(\left(\frac{64}{9}\right)^{1/3} + O(1)\right)(\ln n)^{1/3}(\ln \ln n)^{2/3}\right), \tag{4}$$

where n is the number for factorization. Since we know parameters for brute force, we can find the length of n.

To solve the DLP, we use Pollard's Rho algorithm for logarithms, because it is the best known algorithm to solve the discrete logarithm problem. Similarly, matching the complexities, we have

$$2^x = \sqrt{\frac{\pi o}{2}}, \tag{5}$$

where o is the order of the group.

Let us name classic isogeny (CI) for the algorithm of Galbraith and Stolbunov [Galbraith and Stolbunov 2013], considered the best algorithm for classic computers to solve the isogeny problem [Feo et al. 2011]. Thus, the matching of CI complexity is given by

$$2^x = p^{1/4}, \tag{6}$$

where p is the characteristic of the field, where the classes of supersingular elliptic curves are defined. Similarly, let us name quantum isogeny (QI) for Tani's algorithm [Tani 2009], the best quantum algorithm to solve the isogeny problem [Adj et al. 2018]. Thus, the matching of QI complexity is given by

$$2^x = p^{1/6}. \tag{7}$$

Table 2 resumes the values found with the equations. We have one more column with the values recommended by the National Institute of Standards and Technology (NIST) [Barker 2016].

Table 2. Comparison between brute force and minimum key length.

Brute Force	DLP - Eq. (5)	GNFS - Eq. (4)	NIST	CI - Eq. (6)	QI - Eq. (7)
80	160	851	1 024	320	480
112	224	1 853	2 048	448	672
128	256	2 538	3 072	512	768
192	384	6 707	7 680	768	1152
256	512	13 547	15 360	1024	1536

Using the Grover's algorithm for brute force attack, a n-bits key can be found with complexity $O(\sqrt{n})$. Therefore, every cryptographic algorithm should at least double the key length to keep the same level of security in the face of quantum computing. The performance is directly proportional to the key length. Figure 2 depicts a trade-off between security and key bit length, with the interpolation polynomials from the data in Table 2.

3.2. Performance

Jao and De Feo provide two algorithms for the task of computing isogenies of a given kernel in a key exchange protocol [Jao and Feo 2011]. The main point is to compute

$$\phi_A : E_0 \to E_A, \text{ where } E_A = E_0/\langle [m_A]P_A + [n_A]Q_A \rangle.$$

Defining $R_0 := [m_A]P_A + [n_A]Q_A$, the order of R_0 is $l_A^{e_A}$. Then, for $0 \leq i < e_A$, we have

$$E_{i+1} = E_i/\langle l_A^{e_A - i - 1} R_i \rangle, \quad \phi_i : E_i \to E_{i+1}, \quad R_{i+1} = \phi_i(R_i),$$

where ϕ_i is a l_A-isogeny. Hence, $E_A = E_{e_A}$ and ϕ_A is found by composition

$$\phi_{e_A - 1} \circ \ldots \circ \phi_0.$$

It is important to remark that walks on expander (Ramanujan) graphs can be used to compute the isogenies that compose ϕ_A. This is not explicit in the algorithm

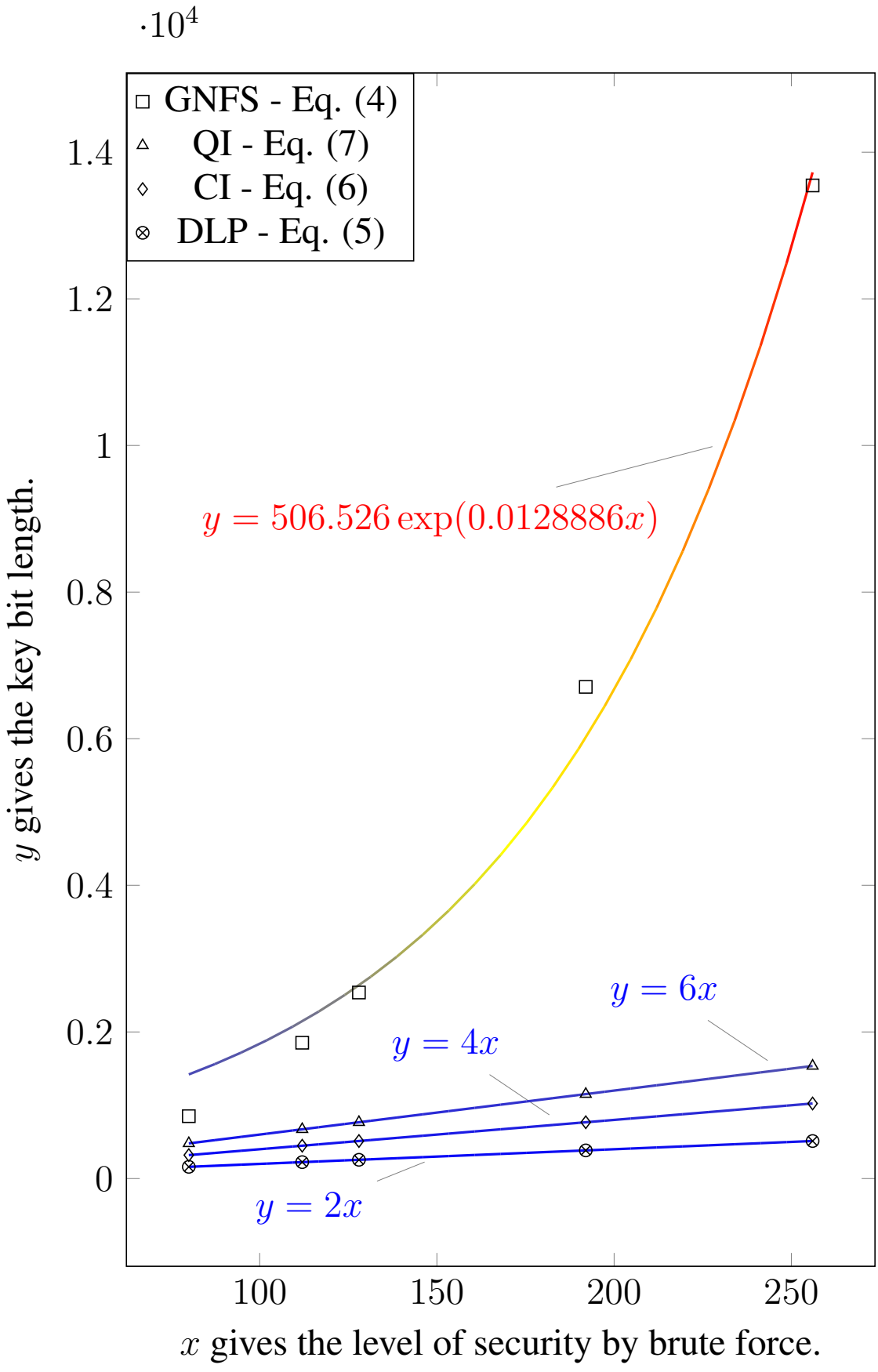

Figure 2. Comparison between brute force and minimum key length.

Algorithm 2: Multiplication-based

Input: E_0, R_0
Output: E_{e_A}
1 **for** $0 \leq i < e_A$ **do**
2 $\quad P_i \leftarrow l_A^{e_A-i-1} R_0$
3 $\quad$ Compute $\phi_i : E_i \rightarrow E_i/\langle P_i \rangle$
4 $\quad E_{i+1} \leftarrow E_i/\langle P_i \rangle$
5 $\quad R_{i+1} \leftarrow \phi_i(R_i)$

Algorithm 3: Isogeny-based

Input: E_0, R_0
Output: E_{e_A}
1 $Q_0 \leftarrow R_0$
2 **for** $0 \leq i < e_A - 1$ **do**
3 $\quad Q_{j+1} \leftarrow l_A Q_j$
4 **for** $0 \leq i < e_A$ **do**
5 $\quad$ Compute $\phi_i : E_i \rightarrow E_i/\langle Q_{e_A-1} \rangle$
6 $\quad E_{i+1} \leftarrow E_i/\langle Q_{e_A-1} \rangle$
7 $\quad$ **for** $i \leq j < e_A - 1$ **do**
8 $\quad\quad Q_{j+1} \leftarrow \phi_i(Q_j)$

by [Jao and Feo 2011]. They give two cost-equivalent strategies, *multiplication-oriented* (Algorithm 2) and *isogeny-oriented* (Algorithm 3) for the iterative computation of the required isogenies.

For key exchanges, Alice and Bob can choose between two algorithms (multiplication-oriented or isogeny-oriented). Both have a cost of $O(log^2 p)$ in the chosen finite field. The major cost corresponds to the isogeny evaluation at step 7 of algorithm 3. Hence, as l_A grows, the multiplication-oriented algorithm becomes preferable over the isogeny-based algorithm. Thus, the performance cost is 2 times the key length, which grows by a factor of 4 for classic computers and of 6 by quantum computers. Therefore, its performance cost increases by a factor of 8 and 18, respectively.

The complexity for modular exponentiations is $O(\log e)$, where e is the exponent [Borges et al. 2017]. Therefore, its performance cost increases by a factor of 2. If e is chosen randomly it has the size of the module. Therefore, its performance cost increases exponentially. See Figure 2.

4. Conclusions

This paper has discussed the trade-off between performance and security for cryptosystems based on isogenies between supersingular elliptic curves. We based our analysis on the original proposal of [Jao and Feo 2011]. Since that proposal, several refinements have been proposed to enhance the performance of SSI-based cryptosystems. For example, [Azarderakhsh et al. 2016] present a method to compress the public transcript for SSI-based key exchange, without affecting the security of the cryptosystem and at the

expense of a modest additional computational cost for the compression. In another work, [Costello et al. 2016] give even more efficient algorithms and show an implementation that runs up to 2.9 times faster than the proposal by [Azarderakhsh et al. 2016]. Following a similar approach, [Costello et al. 2017] propose new algorithms capable to accelerate SIDH public-key compression while further reducing the size of the compressed public keys and also reducing the computational effort. Therefore, this work is presenting the worst-case scenario for the performance of SIDH.

There are several research efforts to provide more efficient SSI-based cryptosystems. Our results show that even using the original proposal by [Jao and Feo 2011], the trade-off between performance and security indicates that SSI achieves small key sizes with good performance at the practical security levels recommended by NIST. Moreover, when the security level increases, the cost for SIDH increases exponentially slower than for classical cryptographic algorithms. Therefore, we conclude that SSI cryptosystems are strong potential post-quantum candidates.

Acknowledgements

We would like to thank LNCC for the institutional support and the anonymous reviewers for their helpful comments and suggestions.

References

Adj, G., Cervantes-Vázquez, D., Chi-Domínguez, J.-J., Menezes, A., and Rodríguez-Henríquez, F. (2018). On the cost of computing isogenies between supersingular elliptic curves. *IACR Cryptology ePrint Archive*, 2018:313.

Alon, N. and Milman, V. D. (1985). λ_1, isoperimetric inequalities for graphs, and superconcentrators. *J. Combin. Theory Ser. B*, 38(1):73–88.

Azarderakhsh, R., Jao, D., Kalach, K., Koziel, B., and Leonardi, C. (2016). Key compression for isogeny-based cryptosystems. In *Proceedings of the 3rd ACM International Workshop on ASIA Public-Key Cryptography*, AsiaPKC '16, pages 1–10, New York, NY, USA. ACM.

Barker, E. (2016). Recommendation for Key Management. Part 1: General. NIST Special Publication 800-57 Part 1 Revision 4 [Online]. Available: `http://dx.doi.org/10.6028/NIST.SP.800-57pt1r4` [January 2016]. National Institute of Standards and Technology, Gaithersburg, MD.

Borges, F., Lara, P., and Portugal, R. (2017). Parallel algorithms for modular multiexponentiation. *Appl. Math. Comput.*, 292(C):406–416.

Charles, D. X., Lauter, K. E., and Goren, E. Z. (2009). Cryptographic hash functions from expander graphs. *Journal of Cryptology*, 22(1):93–113.

Chen, L., Jordan, S., Liu, Y.-K., Moody, D., Peralta, R., and an Daniel Smith-Tone, R. P. (2016). NIST Report on Post-Quantum Cryptography NISTIR 8105 [Online]. Available: `http://dx.doi.org/10.6028/NIST.IR.8105` [April 2016]. National Institute of Standards and Technology, Gaithersburg, MD.

Childs, A. M., Jao, D., and Soukharev, V. (2010). Constructing elliptic curve isogenies in quantum subexponential time. *CoRR*, abs/1012.4019.

Costello, C., Jao, D., Longa, P., Naehrig, M., Renes, J., and Urbanik, D. (2017). Efficient compression of sidh public keys. In *EUROCRYPT (1)*, pages 679–706. Springer.

Costello, C., Longa, P., and Naehrig, M. (2016). Efficient algorithms for supersingular isogeny diffie-hellman. In *Proceedings, Part I, of the 36th Annual International Cryptology Conference on Advances in Cryptology — CRYPTO 2016 - Volume 9814*, pages 572–601, Berlin, Heidelberg. Springer-Verlag.

Couveignes, J.-M., m. Couveignes, J., Dewaghe, L., Dewaghe, L., Morain, F., and Morain, F. (1996). Isogeny cycles and the schoof-elkies-atkin algorithm. In *Research Report LIX/RR/96/03, LIX*, page 96.

Deuring, M. (1941). Die Typen der Multiplikatorenringe elliptischer Funktionenkörper. *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg*, 14(1):197–272.

Dodziuk, J. (1984). Difference equations, isoperimetric inequality and transience of certain random walks. *Trans. Amer. Math. Soc.*, 284(2):787–794.

Elkies, N. D. (1998). Elliptic and modular curves over finite fields and related computational issues. In *Computational perspectives on number theory (Chicago, IL, 1995)*, volume 7 of *Studies in Advanced Mathematics*, pages 21–76. AMS International Press.

Feo, L. D. (2017). Mathematics of isogeny based cryptography. *CoRR*, abs/1711.04062.

Feo, L. D., Jao, D., and Plût, J. (2011). Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. Cryptology ePrint Archive, Report 2011/506. `https://eprint.iacr.org/2011/506`.

Galbraith, S. and Stolbunov, A. (2013). Improved algorithm for the isogeny problem for ordinary elliptic curves. *Applicable Algebra in Engineering, Communication and Computing*, 24(2):107–131.

Galbraith, S. D. (1999). Constructing isogenies between elliptic curves over finite fields. *LMS J. Comput. Math*, 2:118–138.

Grover, L. K. (1996). A fast quantum mechanical algorithm for database search. In *ANNUAL ACM SYMPOSIUM ON THEORY OF COMPUTING*, pages 212–219. ACM.

Hashimoto, Y. (2018). Multivariate public key cryptosystems. In *Mathematical Modelling for Next-Generation Cryptography*, pages 17–42. Springer.

Hermans, J., Vercauteren, F., and Preneel, B. (2010). Speed records for ntru. In Pieprzyk, J., editor, *Topics in Cryptology - CT-RSA 2010*, pages 73–88, Berlin, Heidelberg. Springer Berlin Heidelberg.

Hoory, S., Lilian, N., and Wigderson, A. (2006). Expander graphs and their applications. *Bulletin (New Series) of the American Mathematical Society*, 43(4):349–561.

Jao, D. and Feo, L. D. (2011). Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. In Yang, B.-Y., editor, *Post-Quantum Cryptography*, pages 19–34. Springer-Verlag.

Jao, D., Miller, S. D., and Venkatesan, R. (2009). Expander graphs based on grh with an application to elliptic curve cryptography. *Journal of Number Theory*, 129(5):1491–1504.

Koblitz, N. (1987). Elliptic curve cryptosystems. *Mathematics of Computation*, 48(177):203–209.

Kohel, D. (1996). *Endomorphism rings of elliptic curves over finite fields*. PhD thesis, University of California at Berkley.

Miller, V. S. (1985). Use of elliptic curves in cryptography. In *Advances in Cryptology - CRYPTO '85, Santa Barbara, California, USA, August 18-22, 1985, Proceedings*, pages 417–426.

Nilli, A. (1991). On the second eigenvalue of a graph. *Discrete Math.*, 91(2):207–210.

Roetteler, M., Naehrig, M., Svore, K. M., and Lauter, K. (2017). Quantum resource estimates for computing elliptic curve discrete logarithms. In *Proc. ASIACRYPT 2017*, volume 10625, pages 241–270. Springer.

Rostovtsev, A. and Stolbunov, A. (2006). Public-key cryptosystem based on isogenies. Cryptology ePrint Archive, Report 2006/145. `https://eprint.iacr.org/2006/145`.

Shor, P. W. (1995). Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5):1484–1509.

Silverman, J. (1986). *The Arithmetic of Elliptic Curves*. Applications of Mathematics. Springer.

Stolbunov, A. (2010). Constructing public-key cryptographic schemes based on class group action on a set of isogenous elliptic curves. *Advances in Mathematics of Communications*, 4(1930-5346):215.

Tani, S. (2009). Claw finding algorithms using quantum walk. *Theoretical Computer Science*, 410(50):5285 – 5297.

Tate, J. (1966). Endomorphisms of abelian varieties over finite fields. *Invent. math.*, 2:134–144.

Vélu, J. (1971). Isogénies entre courbes elliptiques. *C. R. Acad. Sci. Pars Sér. A-B*, 273:A238–A241.

Washington, L. C. (2008). *Elliptic Curves: Number Theory and Cryptography*. Chapman & Hall/CRC.