Computational aspects of class group actions and applications to post-quantum cryptography

Houben, M.R.

Chapter 2

Isogeny-based cryptography

This chapter is a concise, mostly informal, and highly incomplete introduction to the field of isogeny-based cryptography. Our main focus in this introduction will be on key-exchange protocols; in particular those based on class group actions. Such schemes will be our main concern for the results presented in Chapter 3.

2.1 Diffie–Hellman key exchange

Elliptic curves are widely used in modern-day cryptography. For example, to securely send information in many end-to-end encrypted messaging apps (such as WhatsApp, Signal, and LINE), and to sign transactions in blockchain (such as Bitcoin and Ethereum). The way that messaging apps employ elliptic curves, is through a cryptographic protocol known as a *Diffie–Hellman key exchange*. The idea of such a protocol was first published in 1976 [6]. It is a method for two parties, typically known as *Alice* and *Bob*, to establish a common secret over an insecure channel of communication. Such a common secret is typically subsequently used to encrypt and decrypt information that Alice and Bob would like to send to each other securely (like a WhatsApp message). There are various ways to perform a Diffie–Hellman key exchange, but the main idea is always the same. It can be roughly described as follows. First, Alice and Bob agree publicly on a set X and an element $x_0 \in X$. They also both compute a secret function $X \to X$. Let us say Alice computes (and keeps to herself) the function $f_A : X \to X$, and that Bob does the same for the function $f_B : X \to X$. They each evaluate their secret functions on x_0, and send the result to each other, so that Bob receives $f_A(x_0)$ and Alice receives $f_B(x_0)$. Then, they compute $f_A(f_B(x_0))$ and $f_B(f_A(x_0))$ respectively. If the protocol is designed in such a way that their functions commute, they end up at the same element of X, which is then their common secret.

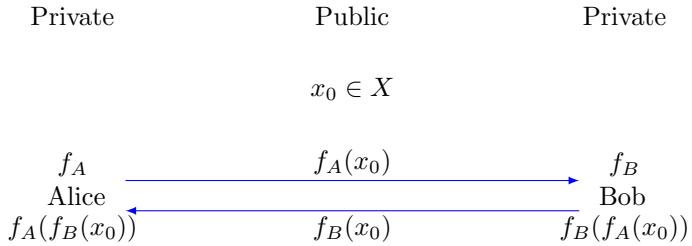

Figure 2.1: A blueprint for a Diffie–Hellman key-exchange.

As an example, in Elliptic Curve Diffie–Hellman (ECDH), the protocol used by WhatsApp, Alice and Bob publicly agree on an elliptic curve E over a finite field, and a point $P \in E$ of large order n. They hold secret elements $a, b \in \{1, \ldots, n\}$, and they compute and send each other $f_A(P) = aP$ and $f_B(P) = bP$ respectively. Their eventual shared secret is then the point $abP = baP$ on E. For example, in WhatsApp, the elliptic curve used is $E : y^2 = x^3 + 486662x^2 + x$ over the finite field $\mathbf{F}_p$, where $p = 2^{255} - 19$. The public point $P \in E$ is one of the points whose x-coordinate is 9, and has order $n = 2^{252} + 27742317777372353535851937790883648493$.

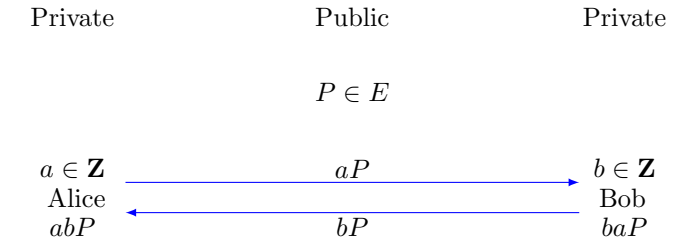

Figure 2.2: An Elliptic Curve Diffie–Hellman key-exchange.

The security of such a protocol is based on the assumption that it is computationally infeasible to obtain any secret information by only using the publicly available information. More concretely, consider the following set of problems that can be associated to the cryptographic scheme described above.

Problems 2.1.1 (i) *Discrete Logarithm Problem (DLP).* Suppose we are given P and aP. Compute $a \in \{1, \ldots, n\}$.

(ii) *Computational Diffie–Hellman Problem (CDH).* Suppose we are given P, aP, and bP. Compute $abP \in E$.

(iii) *Decisional Diffie–Hellman Problem (DDH).* Suppose we are given either a quadruple $\{P, aP, bP, abP\}$ or a quadruple $\{P, aP, bP, Q\}$, where $Q \in E$ is random. Distinguish, with non-negligible probabilistic advantage, which of the two options it is.[1]

[1] A more formal version of this problem can be found in, e.g. [8, Def. 8.63].

An algorithm to solve problem (i) can be used to solve problem (ii), and an algorithm to solve problem (ii) can be used to solve problem (iii). In a sense, this makes (i) the "hardest" of the problems and (iii) the "easiest". Many security proofs for cryptographic schemes rely on the hypothesis that an analogue of one, or all, of these problems is computationally infeasible. Such a hypothesis is also called a *computational hardness assumption*. In the case of ECDH, it is known that there exists a quantum algorithm that solves (i) in polynomial time. This is known as *Shor's algorithm* [16], and applies to a wide variety of cryptographic schemes that are deployed in practice today, including more classical schemes that are not based on elliptic curves, such as RSA [13]. This means that ECDH and many other protocols used today are insecure in the era of large-scale quantum computing. While it is unknown whether a sufficiently large quantum computer exists, or will exist in the near future, that can be practically used to break schemes that are employed in the real world, this has sparked an area of research known as *post-quantum cryptography*, which looks for ways to encrypt information using algorithms on classical computers that are safe against quantum attacks. One such proposal is *isogeny-based cryptography*.

2.2 Class group action based key exchanges

The primary computational hardness assumption central to essentially all of isogeny-based cryptography is the *isogeny path problem*, and can be described as follows.

Problem 2.2.1 (Isogeny path problem.) *Given a pair of elliptic curves E_0, E_1 over a field k, find an isogeny $\varphi : E_0 \to E_1$.*

Most isogeny-based protocols are not based solely on the pure isogeny path problem as described above, but rather on a version of it that includes some form of extra structure or information. Sometimes, as we will see later, this extra structure turns out to make the protocol insecure. The historically first, and still unbroken, isogeny-based scheme is a Diffie–Hellman key exchange protocol known as CRS, named after Couveignes, Rostovtsev, and Stolbunov [4, 15]. CRS, and later variants such as CSIDH [2] and OSIDH [3], are known as *class group action based key exchange protocols*.

2.2.1 Obtaining a key exchange from a class group action

Let k be a perfect field, and let $\mathcal{O}$ be an imaginary quadratic order. Then there is a free action

$$\mathrm{Cl}(\mathcal{O}) \circlearrowright \{(E, \iota) \mid E/\overline{k} \text{ ell. curve}, \iota : \mathcal{O} \hookrightarrow \mathrm{End}(E) \text{ primitive } \mathcal{O}\text{-orientation}\}/\cong \quad (2.1)$$

of the ideal class group $\mathrm{Cl}(\mathcal{O})$ of $\mathcal{O}$ on the set of primitively $\mathcal{O}$-oriented elliptic curves over $\overline{k}$ up to isomorphism. We will explain in a bit where this comes from, how this action is defined, and how to evaluate it in practice, but let us first see how this gives rise to an idea for a Diffie–Hellman key exchange protocol.

Alice and Bob agree publicly on an elliptic curve E_0/k that is primitively oriented by $\mathcal{O}$. They select secret elements $[\mathfrak{a}]$ and $[\mathfrak{b}]$ of $\mathrm{Cl}(\mathcal{O})$ respectively, and compute $E_A := [\mathfrak{a}]E_0$ and $E_B := [\mathfrak{b}]E_0$. After exchanging E_A and E_B, they both compute $[\mathfrak{b}][\mathfrak{a}]E_0 = E_{AB} = [\mathfrak{a}][\mathfrak{b}]E_0$.

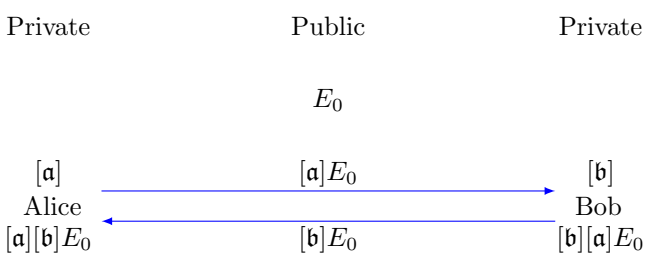

Figure 2.3: A class group action based key exchange.

This gives rise to the following analogues of Problems 2.1.1.

Problems 2.2.2 (i) *Vectorization Problem.* Suppose we are given E_0 and $[\mathfrak{a}]E_0$. Compute $[\mathfrak{a}]$.

(ii) *Computational Diffie–Hellman Problem (CDH).* Suppose we are given E_0, $[\mathfrak{a}]E_0$, and $[\mathfrak{b}]E_0$. Compute $[\mathfrak{a}][\mathfrak{b}]E_0$.

(iii) *Decisional Diffie–Hellman Problem (DDH).* Suppose we are given either a quadruple $\{E_0, [\mathfrak{a}]E_0, [\mathfrak{b}]E_0, [\mathfrak{a}][\mathfrak{b}]E_0\}$ or a quadruple $\{E_0, [\mathfrak{a}]E_0, [\mathfrak{b}]E_0, [\mathfrak{c}]E_0\}$, where $[\mathfrak{c}] \in \mathrm{Cl}(\mathcal{O})$ is a random ideal class. Distinguish, with non-negligible probabilistic advantage, which of the two options it is.

2.2.2 Defining the class group action

We now describe the group action from (2.1) explicitly. Let k be a perfect field of characteristic $p \geq 0$, let $\mathcal{O}$ be an imaginary quadratic order and let E/k be a primitively $\mathcal{O}$-oriented elliptic curve. Note that $\iota(\mathcal{O})$ is then a subring of $\mathrm{End}(E)$. Let $0 \neq \mathfrak{a}$ be an $\mathcal{O}$-ideal such that p does not divide the norm of $\mathfrak{a}$. We define the *kernel of* $\mathfrak{a}$, denoted $E[\mathfrak{a}]$, as

$$E[\mathfrak{a}] := \bigcap_{\alpha \in \iota(\mathfrak{a})} \ker(\alpha) \tag{2.2}$$

$$= \{P \in E(\overline{k}) \mid \alpha(P) = 0 \quad \forall\, \alpha \in \iota(\mathfrak{a})\}. \tag{2.3}$$

The number of elements of $E[\mathfrak{a}]$ equals the norm of the ideal $\mathfrak{a}$. We denote $\mathfrak{a} \cdot E := E/E[\mathfrak{a}]$, and write $\varphi_{\mathfrak{a}} : E \to \mathfrak{a} \cdot E$ for the separable isogeny with kernel $E[\mathfrak{a}]$ (which is unique up to post-composition with an isomorphism by Lemma 1.1.1). Moreover, the isomorphism class of the curve $\mathfrak{a} \cdot E$ together with its orientation induced by (1.4)

through $\varphi_{\mathfrak{a}}$ only depends on the ideal class of $\mathfrak{a}$. Since any ideal class contains a representative whose norm is not divisible by p, this defines a group action as in (2.1).

2.2.3 Computing the class group action

We now describe how (part of) the class group action can be explicitly computed in the setting of CRS and CSIDH. Suppose E is defined over a finite field $k = \mathbf{F}_q$ of characteristic p. Let π denote the q-Frobenius endomorphism. Suppose that π is imaginary quadratic (or, equivalently, not an element of $\mathbf{Z}$), and that E is primitively oriented by the imaginary quadratic order $\mathcal{O} := \mathbf{Z}[\pi]$. Denote by $f := X^2 - tX + q$ the characteristic polynomial of Frobenius. If $\ell \neq p$ is a prime number that splits in $\mathcal{O}$, then f splits modulo ℓ, that is

$$f = X^2 - tX + q \equiv (X - \lambda)(X - \mu) \pmod{\ell}.$$

for $\lambda \neq \mu \in (\mathbf{Z}/\ell\mathbf{Z})^\times$. This corresponds to a splitting of the principal $\mathcal{O}$-ideal

$$(\ell) = (\ell, \pi - \lambda)(\ell, \pi - \mu) = \mathfrak{l}\bar{\mathfrak{l}} \tag{2.4}$$

into two ideals $\mathfrak{l} = (\ell, \pi - \lambda)$ and $\bar{\mathfrak{l}} = (\ell, \pi - \mu)$ of norm ℓ. For both of these ideals, the kernel as defined by (2.3) is a subgroup of E of order ℓ, hence corresponds to an ℓ-isogeny with domain E. The orbit of E under the action by the subgroup of $\mathrm{Cl}(\mathcal{O})$ generated by $[\mathfrak{l}]$ is a cycle whose length equals the order of $[\mathfrak{l}] \in \mathrm{Cl}(\mathcal{O})$. We can associate to this cycle a directed graph, whose set of nodes is the orbit of E and whose edges are the ℓ-isogenies corresponding to the ideals $\mathfrak{l}$ and $\bar{\mathfrak{l}}$, i.e. corresponding to the two eigenvalues of Frobenius λ and μ. Since $\mathfrak{l}\bar{\mathfrak{l}} = (\ell)$ is principal, i.e. $[\mathfrak{l}]$ and $[\bar{\mathfrak{l}}]$ are each other's inverses in $\mathrm{Cl}(\mathcal{O})$, the edges belonging to different eigenvalues point in opposite directions along the cycle.

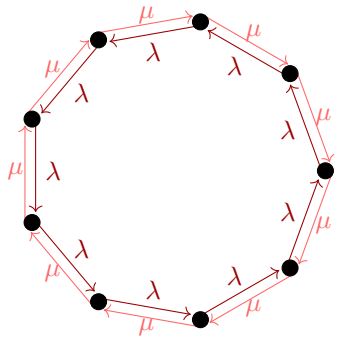

Figure 2.4: Directed cycles associated to ℓ-isogenies corresponding to the eigenvalues of Frobenius λ and μ.

The idea of CRS and CSIDH is now that we compose random walks along these graphs for different small primes ℓ_i. That is, we restrict the class group action to

ideals of the form $[\mathfrak{a}] = \prod_i [\mathfrak{l}_i]^{a_i}$, where $\mathfrak{l}_i$ is a prime ideal above ℓ_i and the $a_i \in \mathbf{Z}$ are sampled randomly from predetermined bounded intervals.

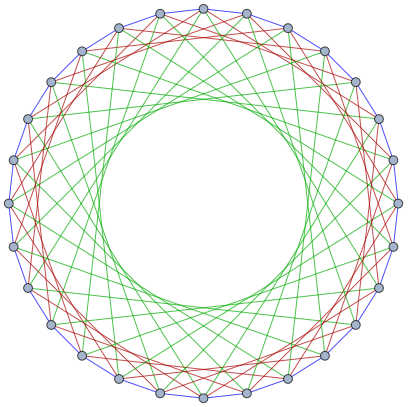

Figure 2.5: A union of three ℓ_i-isogeny graphs.

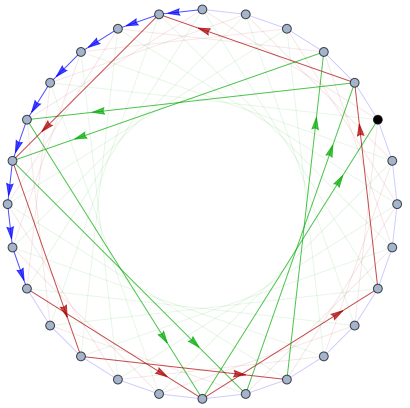

Figure 2.6: An isogeny walk.

If the class group is sufficiently large, these random walks should still give us many different options for, e.g., Alice's public curve $[\mathfrak{a}]E_0$, so that finding an isogeny between the starting and ending curve of the walk exhaustively will remain infeasible.

Now, let us say that we would like to compute the action of the ideal $\mathfrak{l}$ given by (2.4) on E, i.e. one step of the walk in Figure 2.6. According to (2.3), the kernel of $\mathfrak{l}$ is given by

$$E[\mathfrak{l}] = \{P \in E(\overline{k}) \mid P \in E[\ell],\ (\pi - \lambda)(P) = 0\}.$$

Denoting by r the order of $\lambda \in (\mathbf{Z}/\ell\mathbf{Z})^\times$, we see that

$$(\pi - \lambda)(P) = 0 \iff \pi(P) = \lambda P \implies \pi^r(P) = \lambda^r P = P \implies P \in E(\mathbf{F}_{q^r}).$$

It follows that $E[\mathfrak{l}] \subseteq E[\ell](\mathbf{F}_{q^r})$. One way to compute $[\mathfrak{l}]E$ is thus to sample an ℓ-torsion point $P \in E(\mathbf{F}_{q^r})$ whose eigenvalue under the action of Frobenius is λ, and then to compute the codomain of the ℓ-isogeny with kernel $\langle P \rangle$ using Vélu's formulae (1.3). All of this can be done in time polynomial in ℓ and $\log(q)$. In practice, the computational complexity depends heavily (although polynomially) on r, as field arithmetic in larger fields is more expensive. The optimal situation for efficient evaluation of ℓ-isogeny walks is thus the case where the multiplicative orders of the eigenvalues of Frobenius modulo ℓ are as small as possible, i.e. $\lambda = 1, \mu = -1$. This is equivalent to

$$t \equiv 0 \pmod{\ell}, \qquad \text{and} \qquad q + 1 \equiv 0 \pmod{\ell}.$$

Demanding this for many primes ℓ automatically forces $t = 0$ by the Chinese remainder theorem, i.e. the curve E to be supersingular. This gives rise to CSIDH (Commutative Supersingular Isogeny Diffie–Hellman).

Example 2.2.3 (CSIDH-512) Let p be the prime number

$$p := 4 \cdot \underbrace{(3 \cdot 5 \cdot \ldots \cdot 373)}_{73 \text{ consecutive primes}} \cdot 587 - 1 \approx 2^{511}. \tag{2.5}$$

Let $E_0/\mathbf{F}_p$ be the supersingular elliptic curve given by $E_0 : y^2 = x^3 + x$. Then $\mathcal{O} := \mathrm{End}_{\mathbf{F}_p}(E_0) = \mathbf{Z}[\pi]$, where $\pi : E_0 \to E_0$ denotes the p-Frobenius, and $\mathcal{O} \hookrightarrow \mathrm{End}(E_0)$ is a primitive orientation. We denote by $\ell_1, \ldots, \ell_{74}$ the odd prime factors of $p+1$, and by $\mathfrak{l}_i := (\ell_i, \pi - 1)$ the $\mathcal{O}$-ideal above ℓ_i corresponding to Frobenius eigenvalue $+1$. This gives rise to the following Diffie–Hellman key exchange procedure.

(i) Alice samples a random element $(a_1, \cdots, a_{74}) \in \{-5, \ldots, 5\}^{74}$, computes $E_A := \prod_i [\mathfrak{l}_i]^{a_i} E_0$, and sends E_A to Bob.

(ii) Bob samples a random element $(b_1, \cdots, b_{74}) \in \{-5, \ldots, 5\}^{74}$, computes $E_B := \prod_i [\mathfrak{l}_i]^{b_i} E_0$, and sends E_B to Alice.

(iii) Alice computes $\prod_i [\mathfrak{l}_i]^{a_i} E_B = \prod_i [\mathfrak{l}_i]^{a_i + b_i} E_0$.

(iv) Bob computes $\prod_i [\mathfrak{l}_i]^{b_i} E_A = \prod_i [\mathfrak{l}_i]^{a_i + b_i} E_0$. ☆

CRS follows the same protocol as CSIDH, but with an ordinary starting curve $E_0/\mathbf{F}_q$. The computational performance of CRS depends heavily on the trace of E_0; one for which the eigenvalues of Frobenius have small multiplicative order modulo many primes ℓ_i is typically better. Other than essentially by exhaustive search, currently no method is known for computing an ordinary elliptic curve over a finite field with both a favorable trace and a large class group (the latter requirement rules out the use of the CM method 1.2.1, since (Hilbert) class polynomials become impractically large; more on this in Chapter 6). Moreover, as explained above, if sufficiently many eigenvalue pairs are "optimal", i.e. ± 1, this forces the curve to be supersingular. As such, all known instantiations of CRS that offer cryptographic levels of security are several orders of magnitude slower than CSIDH, and the protocol is widely considered impractical. However, the CRS scheme is still interesting from a theoretical standpoint, since it is conceivable that the structure of supersingular curves (particularly their additional endomorphisms) might some day be used in an attack against CSIDH.

2.2.4 OSIDH

Oriented Supersingular-Isogeny Diffie–Hellman [3] (OSIDH) is another class group action based protocol, which can be roughly described as follows. Let E_0 be an elliptic curve over a (large) finite prime field $\mathbf{F}_p$ oriented by an imaginary quadratic order $\mathcal{O}$ of class number one. Let $n \in \mathbf{Z}_{>0}$ and let ℓ be a (small) prime number. The idea is for Alice and Bob to act by (secret) elements of the class group of the order $\mathcal{O}_n := \mathbf{Z} + \ell^n \mathcal{O}$ on length-n chains of descending ℓ-isogenies, starting from a given one $E_0 \to E_1 \to \ldots \to E_n$. More specifically, Alice acts by an ideal class $[\mathfrak{a}] = \prod_i [\mathfrak{q}_i]^{a_i}$, for some prime ideals $\mathfrak{q}_i$ of small norm coprime to ℓ and exponents $-r \leq a_i \leq r$ (note the similarity to the ideal class in CSIDH and CRS), to obtain $[\mathfrak{a}] \cdot (E_0 \to E_1 \to \ldots \to$

$E_n) = F_0 \to F_1 \to \ldots \to F_n$. Bob does the same with an ideal class $[\mathfrak{b}] = \prod_i [\mathfrak{l}_i]^{b_i}$ of the same form to obtain $[\mathfrak{b}] \cdot (E_0 \to E_1 \to \ldots \to E_n) = G_0 \to G_1 \to \ldots \to G_n$. Then, instead of exchanging the full descending chains $(F_k)_{0 \le k \le n}$ and $(G_k)_{0 \le k \le n}$ (which would be insecure; see e.g. [3, Section 5.1]), Alice and Bob publish F_n and G_n together with the action of $[\mathfrak{l}_i]^j$ for all i and all $-r \le j \le r$ on F_n and G_n respectively. This is sufficient for Alice and Bob to both be able to compute $[\mathfrak{b}][\mathfrak{a}]E_n = [\mathfrak{a}][\mathfrak{b}]E_n$ (see [3, Section 5.2] for more details). There exist exponential-time attacks against OSIDH [5] that are practical for a large set of parameter choices; in particular for the original proposal of [3] that claimed a security level equivalent to CSIDH-512.

2.3 SIDH

Supersingular Isogeny Diffie–Hellman [7] (SIDH) is an isogeny-based key exchange protocol that does not rely on class group actions. For a long time, SIDH was considered the most promising post-quantum candidate for isogeny-based cryptography. Its major advantage compared to class group action based key exchanges was in its efficiency, and in the fact that the best-known quantum attacks had exponential complexity, whereas CRS and CSIDH are known to admit subexponential quantum attacks [9, 10, 12]. In 2022, classical polynomial time attacks against SIDH were found [1, 11, 14].

Example 2.3.1 (SIKEp503) Let p be the prime number

$$p := 2^{250} \cdot 3^{159} - 1 = 2^a \cdot 3^b - 1. \tag{2.6}$$

Let $E_0/\mathbf{F}_p$ be the supersingular elliptic curve given by $E_0 : y^2 = x^3 + x$. Let $E_0[2^a] = \langle P_A, Q_A \rangle$ and $E_0[3^b] = \langle P_B, Q_B \rangle$.

(i) Alice samples a random integer $m_A \in \{1, \ldots, 2^a\}$, computes $\varphi_A := E_0 \to E_A := E_0/\langle P_A + m_A Q_A \rangle$ and sends $E_A, \varphi_A(P_B), \varphi_A(Q_B)$ to Bob.

(ii) Bob samples a random integer $m_B \in \{1, \ldots, 3^b\}$, computes $\varphi_B := E_0 \to E_B := E_0/\langle P_B + m_B Q_B \rangle$ and sends $E_B, \varphi_B(P_A), \varphi_B(Q_A)$ to Alice.

(iii) Alice computes $E_B/\langle \varphi_B(P_A) + m_A \varphi_B(Q_A) \rangle = E_0/\langle P_A + m_A Q_A, P_B + m_B Q_B \rangle$.

(iv) Bob computes $E_A/\langle \varphi_A(P_B) + m_B \varphi_A(Q_B) \rangle = E_0/\langle P_A + m_A Q_A, P_B + m_B Q_B \rangle$. ☆

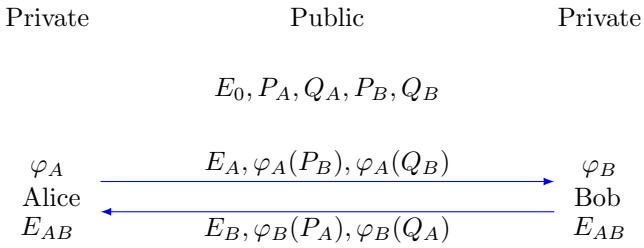

Figure 2.7: Supersingular Isogeny Diffie–Hellman

2.4 Bibliography

[1] Wouter Castryck and Thomas Decru. An efficient key recovery attack on SIDH. In *Eurocrypt 2023 Pt. 5*, volume 14008 of *Lecture Notes in Computer Science*, pages 423–447. Springer, 2023.

[2] Wouter Castryck, Tanja Lange, Chloe Martindale, Lorenz Panny, and Joost Renes. CSIDH: An efficient post-quantum commutative group action. In *Asiacrypt 2018 Pt. 3*, volume 11274 of *Lecture Notes in Computer Science*, pages 395–427. Springer, 2018.

[3] Leonardo Colò and David Kohel. Orienting supersingular isogeny graphs. *Journal of Mathematical Cryptolology*, 14(1):414–437, 2020.

[4] Jean-Marc Couveignes. Hard homogeneous spaces, 2006. Unpublished article, available at `https://eprint.iacr.org/2006/291`.

[5] Pierrick Dartois and Luca De Feo. On the security of OSIDH. In *PKC (1)*, volume 13177 of *Lecture Notes in Computer Science*, pages 52–81. Springer, 2022. `https://ia.cr/2021/1681`.

[6] W. Diffie and M. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, 1976.

[7] David Jao and Luca De Feo. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. In *International Workshop on Post-Quantum Cryptography*, pages 19–34. Springer, 2011.

[8] J. Katz and Y. Lindell. *Introduction to Modern Cryptography: Principles and Protocols*. Chapman & Hall/CRC, second edition, 2014.

[9] Greg Kuperberg. A subexponential-time quantum algorithm for the dihedral hidden subgroup problem. *SIAM Journal on Computing*, 35(1):170–188, 2005.

[10] Greg Kuperberg. Another subexponential-time quantum algorithm for the dihedral hidden subgroup problem. In *8th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2013)*, volume 22 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 20–34. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2013.

[11] Luciano Maino, Chloe Martindale, Lorenz Panny, Giacomo Pope, and Benjamin Wesolowski. A direct key recovery attack on SIDH. In *Eurocrypt 2023 Pt. 5*, volume 14008 of *Lecture Notes in Computer Science*, pages 448–471. Springer, 2023.

[12] Oded Regev. A subexponential time algorithm for the dihedral hidden subgroup problem with polynomial space, 2004.

[13] Ronald L. Rivest, Adi Shamir, and Leonard M. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Commun. ACM*, 21:120–126, 1978.

[14] Damien Robert. Breaking SIDH in polynomial time. In *Eurocrypt 2023 Pt. 5*, volume 14008 of *Lecture Notes in Computer Science*, pages 472–503. Springer, 2023.

[15] Alexander Rostovtsev and Anton Stolbunov. Public-key cryptosystem based on isogenies, 2006. Unpublished article, available at `https://eprint.iacr.org/2006/145`.

[16] Peter W. Shor. Algorithms for quantum computation: discrete logarithms and factoring. *Proceedings 35th Annual Symposium on Foundations of Computer Science*, pages 124–134, 1994.