Post-Quantum Constant-Round Group Key Exchange from Static Assumptions

Katsuyuki Takashima

Abstract We revisit a generic compiler from a two-party key exchange (KE) protocol to a group KE (GKE) one by Just and Vaudenay. We then give two families of GKE protocols from *static* assumptions, which are obtained from the general compiler. The first family of the GKE protocols is a constant-round GKE by using secure key derivation functions (KDFs). As special cases, we have such GKE from *static* Ring-LWE (R-LWE), where "static" means that the parameter size in the R-LWE does not depend on the number of group members, n, and also from the standard SI-DDH and CSI-DDH assumptions. The second family consists of two-round GKE protocols from isogenies, which are proven secure from *new* isogeny assumptions, the first (resp. second) of which is based on the SIDH (resp. CSIDH) two-party KE. The underlying new *static* assumptions are based on indistinguishability between *a product value of supersingular invariants* and a random value.

Keywords Post-quantum cryptography · Constant-round group key exchange · Static assumptions · Lattice-based cryptography · Isogeny-based cryptography

1 Introduction

1.1 Background

It is well known that widely deployed cryptographic schemes (e.g., RSA and ECC) can be broken by using a large-scale quantum computer (Shor 1997). Hence, we should develop new cryptosystems based on quantum-resistant mathematical problems (called post-quantum cryptography (PQC)).

Group key exchange (GKE) is an important cryptographic primitive, and has been studied for a long time (since the seminal two-party Diffie–Hellman key exchange). In GKE, the number of rounds is a crucial measure for evaluating the efficiency and to obtain a constant-round GKE protocol is considered as a minimum desirable require-

K. Takashima (✉)
Mitsubishi Electric, 5-1-1 Ofuna, Kamakura, Kanagawa 247-8501, Japan
e-mail: Takashima.Katsuyuki@aj.MitsubishiElectric.co.jp

T. Takagi et al. (eds.), *International Symposium on Mathematics, Quantum Theory, and Cryptography*, Mathematics for Industry 33,
https://doi.org/10.1007/978-981-15-5191-8_18

ment. Traditionally, the Burmester and Desmedt (BD) KE protocol (Burmester and Desmedt 1994) has been widely known from its simplicity and small round complexity, just two rounds. Subsequently, Just and Vaudenay (JV) (1996) generalized the BD construction in which *any* two-party KE can be used for obtaining GKE. However, their description was sketchy and a *rigorous* security proof was not presented before (see Boyd and Mathuria 2003 also).

In the post-quantum setting, there exist two variants BD-type GKE protocols from lattices (Apon et al. 2019) and isogenies (Furukawa et al. 2018).[1] Apon et al. (2019) proposed a lattice-based BD-type GKE from the Ring-LWE (R-LWE) assumption (in the random oracle model), in which the authors elaborately adjusted the original security proof to their new post-quantum setting. However, since the underlying R-LWE assumption depends on the number of group members, n, the size of data also gets large depending on n. Furukawa et al. (2018) proposed an isogeny-based BD-type GKE protocol called SIBD. However, the security proof of SIBD (Theorem 4 in Furukawa et al. 2018) is imperfect, and several points remain unclear, for example, on how to simulate some public variables. Applying the JV-type compiler to a post-quantum two-party KE is also considered as a reasonable approach, however, we should give a rigorous treatment on its (post-quantum) security proof.

As a result, we lack a post-quantum constant-round GKE protocol with a rigorous and reasonable security proof. We next consider what are reasonable underlying assumptions. The size of a problem instance in the above R-LWE setting is linear in the number of group members, n. Traditionally, in pairing-based cryptography, such linear-sized assumptions are called "non-static", "dynamic", or "q-type", which are not desirable from efficiency and security viewpoints. And, in a line of researches, we succeeded to replace q-type ones to static ones (e.g., Kowalczyk and Wee 2019; Okamoto and Takashima 2010; Takashima 2014) in paring cryptography. Hence, we have the following problem as our target:

> *Can we obtain (provably secure) post-quantum constant-round group key exchange from static assumptions* ?

Recent cryptography research also considers *tight* security reduction (from a static assumption). In fact, the original BD GKE is proven tightly secure from the standard DDH assumption (Theorem 6). For obtaining tight security proof, it is not enough to employ a general form of the JV-type transformation which includes a *general KDF* function to a cyclic group $\mathbb{G}$ (denoted $\mathrm{KDF}_{\mathbb{G}}$). We need a construction without using (general) $\mathrm{KDF}_{\mathbb{G}}$ functions *for tight security* since $\mathrm{KDF}_{\mathbb{G}}$ breaks mathematical structures in the underlying two-party KE.

[1] Boneh et al. (2018) recently proposed a one-round GKE from isogenies. However, it has a crucial mathematical difficulty so that it cannot be realized yet.

1.2 Our Contributions

We revisit previous post-quantum BD-type GKE schemes (Apon et al. 2019; Furukawa et al. 2018 and the JV compiler for GKE Boyd and Mathuria 2003; Just and Vaudenay 1996, and reformulate them under a provably secure generic compiler. We have two families of GKE protocols from *static* assumptions.

The first family of GKE protocols obtained from the general compiler is a constant-round GKE (from a two-party KE protocol) by using a secure $\mathrm{KDF}_{\mathbb{G}}$ (Theorem 3). As special cases, we have such GKE from *static* Ring-LWE (R-LWE), where "static" means that the parameter size in the R-LWE does not depend on the number of group members, n (Corollary 1) and the standard SI-DDH and CSI-DDH assumptions (Corollary 2). The first family has a limitation that they cannot have a tight security proof since a general $\mathrm{KDF}_{\mathbb{G}}$ is used.

The second family consists of two-round GKE protocols, which are proven secure from *new* isogeny assumptions, the first (resp. second) of which is based on the SIDH (resp. CSIDH) KE (Theorem 4 (resp. Theorem 5)). They are called SI-PBD and CSI-PBD GKEs, respectively. The underlying new *static* assumptions are obtained from indistinguishability between a random *product value* of supersingular invariants and a random value (in some appropriate finite field), which seem to have independent interests. They are called DSJP (Decisional Supersingular j-invariants Product) and DSMP (Decisional Supersingular Montgomery coefficients Product) assumptions, respectively. As the second family needs no $\mathrm{KDF}_{\mathbb{G}}$'s, it may have some merits for approaching to tightly secure GKE. (However, we do not yet succeed it.)

Note that we have the Katz–Yung (KY) generic compiler from KE to authenticated KE (AKE) (Katz and Yung 2007), in which a signature scheme is required. Very interestingly, the first *practical* isogeny-based signature scheme, CSI-FiSh, was recently proposed (Beullens et al. 2019). Therefore, we have a practical authenticated GKE (AGKE) by applying the KY compiler to our isogeny-based GKE and CSI-FiSh, both of which are post-quantum from isogenies. (Refer to Bernstein et al. 2019; Peikert 2019 for recent estimates on post-quantum security of CSIDH and CSI-FiSh.) Since we have several lattice-based signatures, e.g., Ducas et al. (2018), Fouque et al. (2017), Akleylek et al. (2017), we also have lattice-based AGKE from our lattice GKE.

1.3 Key Techniques

Hereafter, the user indices are taken in a cycle: for example, $h_{n+1} := h_1$ and $h_0 := h_n$. We first review the BD GKE protocol briefly. It is defined on a cyclic group $\mathbb{G}$ of a prime order q and a generator $g \in \mathbb{G}$ as follows:

Round-1. Each user i generates $a_i \leftarrow_R \mathbb{Z}/q\mathbb{Z}$, $h_i := g^{a_i}$ and broadcasts h_i.

Round-2. Each user i calculates $J_{i-1,i} := (h_{i-1})^{a_i}$, $J_{i,i+1} := (h_{i+1})^{a_i}$ and $u_i := J_{i,i+1} \cdot J_{i-1,i}^{-1}$. User i broadcasts u_i.

KeyComp. User i calculates $K_i := J_{i-1,i}^{n} \cdot u_i^{n-1} \cdot u_{i+1}^{n-2} \cdots u_{i-2}$. Then, $K := K_i = J_{1,2} \cdot J_{2,3} \cdots J_{n,1}$ is the shared key among the n users.

In the (tight) security proof of the BD key exchange protocol from DDH on $\mathbb{G}$, we should simulate broadcast values $(h_i, u_i)_{i\in[n]}$ as well as embed the DDH challenge element into the challenge shared key K.

The SIBD protocol (Furukawa et al. 2018) is obtained from the above BD GKE by replacing (h_i, J_i) with invariants of supersingular elliptic curves. Since the invariants are given by elements in finite fields, we also have

$$u_i := J_{i,i+1} \cdot J_{i-1,i}^{-1}, \quad K := K_i := J_{i-1,i}^{n} \cdot u_i^{n-1} \cdot u_{i+1}^{n-2} \cdots u_{i-2}. \tag{1}$$

We revisit the JV construction (Just and Vaudenay 1996), whose original description was sketchy and the security proof was not given there. Hence, we first give a security proof for JV carefully. Based on the proof, we present our isogeny-based GKE from newly proposed assumptions. Then, as is shown in the proof of Theorem 3, if $J_{i-1,i}$'s are uniformly and independently distributed in $\mathbb{G}$, the n elements $K, u_1, \ldots, u_{i-1}, u_{i+1}, \ldots, u_n$ are also uniformly and independently distributed in $\mathbb{G}$ for $i \in [n]$ (and u_i is given as $u_i = (u_1 \cdots u_{i-1} \cdot u_{i+1} \cdots u_n)^{-1}$). It means that if $J_{i-1,i}$'s are distributed uniformly and independently, the target shared key K is changed to a random one *just by using an information-theoretic game transformation*. This is a key lemma on the BD-type encoding (Lemma 6).

However, for the SIBD protocol (Furukawa et al. 2018), since $J_{i-1,i}$ are given by supersingular j-invariants, we have an efficient algorithm for distinguishing between $J_{i-1,i}$ and a uniformly random element in the finite field (see Sutherland 2012). Hence, for fixing the situation, we introduce new decisional assumptions called d-DSJP and d-DSMP ones. For simplicity, here we just show the 2-DSJP assumption, in which a product of two j-invariants, $J_{i-1,i}^{(1)}$ and $J_{i-1,i}^{(2)}$, that is, $J_{i-1,i}^{(1)} \cdot J_{i-1,i}^{(2)}$, should be indistinguishable from a uniformly random variable. At present, we have *no* efficient algorithm for the problems, and considered them as plausible assumptions.

According to the above ideas, in Sect. 4.1, we give a JV-type generic transformation from KE to GKE based on the BD-type encoding of (u_i) and K from $(J_{i-1,i})$ given in Eq. (1). We then consider the following two approaches for obtaining uniformly random $J_{i-1,i}$'s:

1. Using a secure $\mathrm{KDF}_{\mathbb{G}}$ function φ to obtain random $J_{i-1,i} := \varphi(\kappa_{i-1,i})$ where $\kappa_{i-1,i}$'s are shared keys by secure two-party KE: By this approach, we obtain a new GKE from the "static" R-LWE assumption (Sect. 4.2). We also obtain new GKE protocols from SI-DDH and CSI-DDH assumptions.
2. Using new assumptions on supersingular invariants: By using new DSJP and DSMP assumptions, the local outputs, $(J_{i-1,i})$ and $(M_{i-1,i})$, from two-party key exchange can be computationally changed to random ones, and we obtain new GKE from these post-quantum assumptions (Sects. 4.3 and 4.4) without $\mathrm{KDF}_{\mathbb{G}}$.

1.4 Organization

In Sect. 2, we introduce several preliminary facts: definition of group key exchange, supersingular invariants and underlying assumptions for SIDH and CSIDH. In Sect. 3,

our new assumptions on supersingular invariants are presented. In Sect. 4, we propose new PQ GKE, i.e., lattice-based and isogeny-based GKE from static assumptions.

Notations. When A is a set (resp. a random variable), $y \leftarrow_R A$ denotes that y is uniformly generated from A (resp. randomly generated from A according to its distribution). We denote the finite field of order q by $\mathbb{F}_q$. We denote the set $\{1, \ldots, n\}$ by $[n]$.

2 Preliminaries

2.1 Group Key Exchange

We give definitions of group key exchange, its correctness and security.

Definition 1 (*Group Key Exchange (GKE)*) An algorithm $\Pi := \Pi_{r,n}(\lambda)$ is called as a r-round n-party key exchange protocol if it is composed of probabilistic polynomial-time algorithms (Setup, $(\mathsf{Round}\text{-}r')_{r'=1}^{r}$, $\mathsf{KeyComp}$), where Setup takes a security parameter λ as input, and outputs public parameters params_Π, $\mathsf{Round}\text{-}r'$ for each user i takes previous all public variables and his/her own secrets and outputs (broadcasts) the r'th his/her public values, and $\mathsf{KeyComp}$ for each user i takes all public variables and his/her own secrets and outputs the shared secret value K_i.

We call Π is correct if all (shared) keys $K_1, \ldots, K_n$ are the same values, i.e., $K := K_1 = \cdots = K_n$. The key space (or key set) is denoted by $\mathbb{K} := \mathbb{K}(\lambda)$ whose cardinality $\#\mathbb{K}$ is exponentially large in λ (or has enough entropy).

For a GKE protocol Π, we let $\mathsf{Exec}_\Pi(\lambda)$ denote an execution of the protocol, resulting in a transcript Ψ of all messages sent during the course of that execution, along with the shared key K computed by the parties. We let $\mathsf{Adv}^\Pi_{\mathcal{A}}(\lambda)$ denote the advantage of a polynomial-time quantum adversary $\mathcal{A}$ in distinguishing between the following two distribution ensembles:

$$\{\ (\Psi, K)\ :\ (\Psi, K) \leftarrow_R \mathsf{Exec}_\Pi(\lambda)\ \}_{\lambda\in\mathbb{N}} \quad \text{and}$$
$$\{\ (\Psi, K')\ :\ (\Psi, K) \leftarrow_R \mathsf{Exec}_\Pi(\lambda),\ K' \leftarrow_R \mathbb{K}\ \}_{\lambda\in\mathbb{N}}.$$

Protocol Π is post-quantumly secure if $\mathsf{Adv}^\Pi_{\mathcal{A}}(\lambda)$ is negligible in λ for any polynomial-time quantum $\mathcal{A}$.

2.2 SIDH and CSIDH Key Exchange

In this section, we introduce two efficient Diffie–Hellman-type key exchange protocols using isogenies of supersingular elliptic curves: SIDH (Feo et al. 2014) and CSIDH (Castryck et al. 2018).

2.2.1 Supersingular Isogenies and Invariants

We summarize facts about elliptic curves. For details, see Washington (2008), for example.

Let p be a prime greater than 3 and $\mathbb{F}_p$ be the finite field with p elements. Let $\overline{\mathbb{F}}_p$ be its algebraic closure. Here, an elliptic curve E over $\overline{\mathbb{F}}_p$ is given by the Montgomery normal form

$$E : \delta y^2 = x^3 + mx^2 + x \tag{2}$$

for m and $\delta \in \overline{\mathbb{F}}_p$, where the discriminant of the RHS of Eq. (2) and δ are nonzero. We denote the point at infinity on E by O_E. Elliptic curves are endowed with a unique algebraic group structure, with O_E as a neutral element. The j-invariant and Montgomery coefficient of E are given as $j(E) := \frac{256(m^2-3)^3}{m^2-4}$, $m(E) := m$. Two elliptic curves over $\overline{\mathbb{F}}_p$ are isomorphic if and only if they have the same j-invariant. For $j \in \overline{\mathbb{F}}_p$, $E(j)$ denotes an elliptic curve whose j-invariant is j. For $N \in \mathbb{Z}_{>0}$, the N-torsion points is $E[N] := \{P \in E(\overline{\mathbb{F}}_p) \mid NP = O_E\}$.

Given two elliptic curves E and E' over $\overline{\mathbb{F}}_p$, a homomorphism $\phi : E \to E'$ is a morphism of algebraic curves that sends O_E to $O_{E'}$. A nonzero homomorphism is called an isogeny, and a separable isogeny with the cardinality ℓ of the kernel is called ℓ-isogeny. We consider only separable isogenies in this paper. We compute the ℓ-isogeny by using Vélu's formulas (Vélu 1971) for a small prime $\ell = 2, 3, \ldots$. For explicit formulas, see Jao et al. (2017) for SIDH and see Castryck et al. (2018) for CSIDH.

An elliptic curve E over $\overline{\mathbb{F}}_p$ is called supersingular if there are no points of order p, i.e., $E[p] = \{O_E\}$. The j-invariants of supersingular elliptic curves lie in $\mathbb{F}_{p^2}$. We define two sets as below, for SI-DDH and CSI-DDH assumptions.

$$\mathbb{J}_{p^2} := \{j\text{-invariants of supersingular elliptic curves over } \mathbb{F}_{p^2}\}, \tag{3}$$

$$\mathbb{M}_p := \{\text{Montgomery coefficients of supersingular elliptic curves over } \mathbb{F}_p\}. \tag{4}$$

2.2.2 SIDH Key Exchange and SI-DDH Assumption (Feo et al. 2014)

The detailed description of SIDH key exchange, i.e., $\Pi := \mathsf{SIDH}$, is given in Appendix 3.1. Here, we summarize necessary facts on SIDH for later sections. Public parameters are given as $\mathsf{params}_{\mathsf{SIDH}} := (p, E; P_A, Q_A, P_B, Q_B)$. All the messages during an execution are also given as transcript $\Psi_{AB} := (\mathsf{params}_{\mathsf{SIDH}}, E_A, \phi_A(P_B), \phi_A(Q_B), E_B, \phi_B(P_A), \phi_B(Q_A))$. Alice's and Bob's shared keys, i.e., $K_A := j(E_{AB})$ and $K_B := j(E_{BA})$, are equal, and the value is denoted by K.

Definition 2 (*Supersingular Isogeny Decision Diffie–Hellman (SI-DDH) assumption* Feo et al. 2014; Fujioka et al. 2018) Let $(\Psi_{AB}, j(E_{AB})) \leftarrow_R \mathsf{Exec}_{\mathsf{SIDH}}(\lambda)$,

where $\Psi_{AB} := (\mathsf{params}_{\mathsf{SIDH}},\ E_A, \phi_A(P_B), \phi_A(Q_B), E_B, \phi_B(P_A), \phi_B(Q_A))$. An SI-DDH problem instance is given as (Ψ_{AB}, J_β), where

$$J_0 := j(E_{AB}), \qquad J_1 \leftarrow_R \mathbb{J}_{p^2}, \tag{5}$$

$\beta \leftarrow_R \{0, 1\}$, and $\mathbb{J}_{p^2}$ is defined in Eq. (3). If $|\Pr[\mathcal{A}(\Psi_{AB}, J_0) = 1] - \Pr[\mathcal{A}(\Psi_{AB}, J_1) = 1]| < \mathsf{negl}(\lambda)$ holds for any polynomial-time quantum algorithm $\mathcal{A}$, we say that the SI-DDH assumption holds.

Theorem 1 (Feo et al. 2014) *The SIDH key exchange is post-quantumly secure under the SI-DDH assumption.*

2.2.3 CSIDH Key Exchange and CSI-DDH Assumption (Castryck et al. 2018)

The detailed description of CSIDH key exchange, i.e., $\Pi := \mathsf{CSIDH}$, is given in Appendix 3.2. Here, we summarize necessary facts on CSIDH. Public parameters are given as $\mathsf{params} := (p, E)$. All the messages during a execution are also given as transcript $\Psi_{AB} := (\mathsf{params}_{\mathsf{CSIDH}}, [\mathfrak{a}]E, [\mathfrak{b}]E)$. Alice's and Bob's shared keys, i.e., $K_A := m([\mathfrak{a}][\mathfrak{b}]E)$ and $K_B := m([\mathfrak{b}][\mathfrak{a}]E)$, are equal, and the value is denoted by K.

Definition 3 (*Commutative Supersingular Isogeny Decisional Diffie–Hellman (CSI-DDH) assumption*) Let $(\Psi_{AB}, m([\mathfrak{a}][\mathfrak{b}]E)) \leftarrow_R \mathsf{Exec}_{\mathsf{CSIDH}}(\lambda)$ where $\Psi_{AB} := (\mathsf{params}_{\mathsf{CSIDH}}, [\mathfrak{a}]E, [\mathfrak{b}]E)$. A CSI-DDH problem instance is given as (Ψ_{AB}, M_β), where

$$M_0 := m([\mathfrak{a}][\mathfrak{b}]E), \qquad M_1 \leftarrow_R \mathbb{M}_p,$$

$\beta \leftarrow_R \{0, 1\}$, and $\mathbb{M}_p$ is defined in Eq. (4). If $|\Pr[\mathcal{A}(\Psi_{AB}, M_0) = 1] - \Pr[\mathcal{A}(\Psi_{AB}, M_1) = 1]| < \mathsf{negl}(\lambda)$ holds for any polynomial-time quantum algorithm $\mathcal{A}$, we say that the CSI-DDH assumption holds.

Theorem 2 (Castryck et al. 2018) *The CSIDH key exchange is post-quantumly secure under the CSI-DDH assumption.*

3 New Assumptions on Supersingular Invariants

3.1 New Assumptions on Supersingular j-Invariants

Definition 4 (*Decisional Supersingular j-Invariants Product (d-DSJP) Assumption*) Let $\left(\Psi_{AB}^{(\mu)}, j\left(E_{AB}^{(\mu)}\right)\right)_{\mu\in[d]}$ be transcripts of d-time executions of SIDH with the same $\mathsf{params}_{\mathsf{SIDH}}$, where $\Psi_{AB}^{(\mu)} := \left(\mathsf{params}_{\mathsf{SIDH}}, \left(E_A^{(\mu)}, \phi_A^{(\mu)}(P_B), \phi_A^{(\mu)}(Q_B), E_B^{(\mu)},\right.\right.$

$\phi_B^{(\mu)}(P_A), \phi_B^{(\mu)}(Q_A)\Big)\Big)$ and $\Psi_{AB} := \left(\Psi_{AB}^{(\mu)}\right)_{\mu\in[d]}$. A d-DSJP problem instance is given as (Ψ_{AB}, J_β), where

$$J_0 := \prod_{\mu=1}^{d} j\left(E_{AB}^{(\mu)}\right), \qquad J_1 \leftarrow_R \mathbb{F}_{p^2} \tag{6}$$

and $\beta \leftarrow_R \{0,1\}$. For any adversary $\mathcal{B}$, the advantage of $\mathcal{B}$ is defined as $\mathsf{Adv}_{\mathcal{B}}^{d\text{-DSJP}}(\lambda) := |\Pr[\mathcal{B}(\Psi_{AB}, J_0) = 1] - \Pr[\mathcal{B}(\Psi_{AB}, J_1) = 1]|$, and the d-DSJP assumption holds if $\mathsf{Adv}_{\mathcal{B}}^{d\text{-DSJP}}(\lambda)$ is negligible in λ for any polynomial-time quantum adversary $\mathcal{B}$.[2]

3.1.1 Progressive Weakness Among d-DSJP Assumptions

The next lemma shows that the $(d+1)$-DSJP assumption is weaker than the d-DSJP one. In other words, a security proof from the $(d+1)$-DSJP assumption is considered better than that from the d-DSJP one.

Lemma 1 *The d-DSJP assumption is reduced to the $(d+1)$-DSJP assumption.*

For any adversary $\mathcal{A}$, there is a probabilistic machine $\mathcal{B}$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter λ, $\mathsf{Adv}_{\mathcal{A}}^{(d+1)\text{-DSJP}}(\lambda) \leq \mathsf{Adv}_{\mathcal{B}}^{d\text{-DSJP}}(\lambda)$.

Proof $\mathcal{B}$ receives a d-DSJP tuple (Ψ_{AB}, J_β), where Ψ_{AB} is defined as in Definition 4. J_β is $\prod_{\mu=1}^{d} j\left(E_{AB}^{(\mu)}\right)$ when $\beta = 0$ or a random element in $\mathbb{F}_{p^2}$ when $\beta = 1$. $\mathcal{B}$ generates a new SIDH public key pair $\left(E_A^{(d+1)}, \phi_A^{(d+1)}(P_B), \phi_A^{(d+1)}(Q_B)\right)$, $\left(E_B^{(d+1)}, \phi_B^{(d+1)}(P_A), \phi_B^{(d+1)}(Q_A)\right)$ and SIDH shared key $j\left(E_{AB}^{(d+1)}\right)$, then constructs a new tuple $\Psi'_{AB} := \left(\mathsf{params}, \left(\left(E_A^{(\mu)}, \phi_A^{(\mu)}(P_B), \phi_A^{(\mu)}(Q_B)\right), \left(E_B^{(\mu)}, \phi_B^{(\mu)}(P_A), \phi_B^{(\mu)}(Q_A)\right)\right)_{\mu\in[d+1]}\right)$, and $J'_\beta := J_\beta \cdot j\left(E_{AB}^{(d+1)}\right)$. $\mathcal{B}$ gives a $(d+1)$-DSJP tuple (Ψ'_{AB}, J'_β) to $\mathcal{A}$, and outputs β' when $\mathcal{A}$ outputs β'. □

In fact, we show the 1-DSJP problem is efficiently solved (Lemma 2 in Sect. 3.1.2) and the 2-DSJP problem has a specific approach for solving it via modular polynomials (Sect. 3.1.3).

3.1.2 Case $d = 1$: Relation Between SI-DDH and 1-DSJP Assumptions

While the value of J_0 for SI-DDH in Eq. (5) is the same as that of the 1-DSJP assumption in Eq. (6), the other J_1's in the two assumptions are distributed in different

[2] Its "sum" version (instead of "product"), Decisional Supersingular j-invariants Sum (d-DSJS) assumption, seems to be reasonable for $d \geq 2$, and can be used in security proofs for the "sum" version SI-SBD GKE scheme of SI-PBD GKE in Sect. 4.3. This footnote comment is also applied to the d-DSMP assumption and CSI-PBD GKE in Sect. 4.4 in a similar manner.

manners. Namely, the first (resp. the second) is the uniform distribution over $\mathbb{J}_{p^2} (\subsetneq \mathbb{F}_{p^2})$ (resp. $\mathbb{F}_{p^2}$). As is shown below, the difference is important.

Lemma 2 *The* 1*-DSJP problem can be solved in (deterministic) polynomial time except with a negligible error probability.*

Proof In the 1-DSJP problem, J_0 (resp. J_1) is uniformly distributed in $\mathbb{J}_{p^2}$ (resp. $\mathbb{F}_{p^2}$). Therefore, by applying supersingular identifying algorithm, e.g., Sutherland (2012), we can solve the problem. □

From the above fact, the direct assumption, decisional (1, 1)-SI-PBD assumption in Definition 6 picks up the target key κ_1 ($\beta = 1$ instance) from a uniform distribution in $\mathbb{J}_{p^2}$ instead of $\mathbb{F}_{p^2}$.

3.1.3 Case $d = 2$: An Approach for 2-DSJP via Modular Polynomials

Lemma 1 shows the 2-DSJP assumption is the strongest among the d-DSJP assumptions for $d \geq 2$. In fact, we have some possible approaches for solving the problem as indicated below. But, the attack is not yet effective at present.

Here, we introduce modular polynomials $\Phi_N(X, Y) := \sum c_{ik} X^i Y^k$, which satisfy that $\Phi_N(j, j') = 0$ for two j-invariants j and j' such that there exists an N-isogeny between the associated elliptic curves $E(j)$ and $E(j')$. From the above defining property, it holds that $\Phi_N(X, Y)$ are symmetric polynomials w.r.t. X and Y. Hence, if we set $S := X + Y$ and $T := XY$, $\Phi_N(X, Y)$ are given as $\Phi_N(X, Y) = \Xi_N(S, T) := \sum \gamma_{ik} S^i T^k$ for a two-variable polynomial Ξ_N.

The output J_0 of the 2-DSJP problem is given by the product of two supersingular j-invariants, i.e., $\tau := j\left(E^{(1)}\right) j\left(E^{(2)}\right)$. We substitute $T := \tau$ into $\Xi_N(S, T)$, which we obtain a one-variable polynomial equation $\Xi_N(S, \tau) = 0$. If $E^{(1)}$ and $E^{(2)}$ are N-isogenous, then $\sigma := j\left(E^{(1)}\right) + j\left(E^{(2)}\right)$ satisfies the equation, i.e., $\Xi_N(\sigma, \tau) = 0$.

Based on this fact, we obtain a possible cryptanalysis for the 2-DSJP problem given as below. The input of the algorithm is a 2-DSJP instance (Ψ_{AB}, J_β).

1. Set a set of (small) integers $\mathbb{I} := \{N_1, \ldots, N_t\}$.
2. For each $N \in \mathbb{I}$, solve a one-variable polynomial equation $\xi_N(S) := \Xi_N(S, J_\beta) = 0$, and the set of zero points of ξ_N in $\mathbb{F}_{p^2}$ is denoted by $\mathcal{Z} \subset \mathbb{F}_{p^2}$.
   For each $z \in \mathcal{Z}$, solve the quadratic equation $W^2 - zW + J_\beta = 0$.
   a. If the roots $w_1 \notin \mathbb{F}_{p^2}$ or $w_2 \notin \mathbb{F}_{p^2}$, quit this loop.
   b. Check whether both of w_1 and w_2 are supersingular j-invariants or not. If yes, output $\beta' := 0$.
3. Output $\beta' := 1$.

The degree of isogenous curves $E^{(1)}$ and $E^{(2)}$ above is usually large, therefore, if the security parameter λ is set large, the attack is ineffective. But, the above scenario shows some possible approach to this problem using a specific property on modular polynomials when $d = 2$.

3.2 New Assumptions on Supersingular Montgomery Coefficients

Definition 5 (*Decisional Supersingular Montgomery Coefficients Product (d-DSMP) Assumption*) Let $\left(\Psi_{AB}^{(\mu)}, m\left(E_{AB}^{(\mu)}\right)\right)_{\mu\in[d]}$ be transcripts of d-time executions of CSIDH with the same $\mathsf{params}_{\mathsf{CSIDH}}$, where $\Psi_{AB}^{(\mu)} := \left(\mathsf{params}_{\mathsf{CSIDH}}, \left(E_A^{(\mu)}, E_B^{(\mu)}\right)\right)$ and $\Psi_{AB} := \left(\Psi_{AB}^{(\mu)}\right)_{\mu\in[d]}$, where $E_A^{(\mu)} := \left[\mathfrak{a}^{(\mu)}\right]E$, $E_B^{(\mu)} := \left[\mathfrak{b}^{(\mu)}\right]E$ and $E_{AB}^{(\mu)} := \left[\mathfrak{a}^{(\mu)}\right]\left[\mathfrak{b}^{(\mu)}\right]E$. A d-DSMP problem instance is given as (Ψ_{AB}, M_β), where

$$M_0 := \prod_{\mu=1}^{d} m\left(E_{AB}^{(\mu)}\right), \qquad M_1 \leftarrow_R \mathbb{F}_p,$$

and $\beta \leftarrow_R \{0,1\}$. For any adversary $\mathcal{B}$, the advantage of $\mathcal{B}$ is defined as $\mathsf{Adv}_{\mathcal{B}}^{d\text{-DSMP}}(\lambda) := |\Pr[\mathcal{B}(\Psi_{AB}, M_0) = 1] - \Pr[\mathcal{B}(\Psi_{AB}, M_1) = 1]|$, and the d-DSMP assumption holds if $\mathsf{Adv}_{\mathcal{B}}^{d\text{-DSMP}}(\lambda)$ is negligible in λ for any polynomial-time quantum adversary $\mathcal{B}$.

For the DSMP assumptions, we have similar results for the DSJP. In particular, we have the following lemmas.

Lemma 3 *The d-DSMP assumption is reduced to the $(d+1)$-DSMP assumption.*

Lemma 4 *The 1-DSMP problem can be solved in (deterministic) polynomial time except with a negligible error probability.*

4 Proposed Post-Quantum Group Key Exchange (GKE)

4.1 A Generic JV-Type Compiler for GKE from Two-Party KE (Just and Vaudenay 1996)

We describe a generic BD-type GKE compiler from a two-party KE protocol Π, and the obtained GKE protocol is denoted as Π^{BD}. Such a generic compiler was first proposed by Just and Vaudenay (1996), Boyd and Mathuria (2003), but, no formal proof was attached yet. By describing the security proof carefully, we also give a security proof for our proposal in Sects. 4.3 and 4.4, and we found a condition for the compiler to work correctly. The number of group members is assumed to be $n \geq 3$. Assume that we have two-party key exchange Π with shared keyspace $\mathbb{K}$. We need a map $\varphi : \mathbb{K} \to \mathbb{G}$ (called $\mathbb{G}$-embedding map), where $\mathbb{G}$ is a cyclic group of order q in the BD-type Encoding (BDEnc) as indicated below. We assume that $\gcd(n, q) = 1$ for the number of group members n and the cyclic group order q. (Note that we do not assume the intractability of discrete log in $\mathbb{G}$.)

Exec-Π. Each user i runs the protocol Π with users $i-1$ and $i+1$, respectively, and obtains keys $\kappa_{i-1,i}$ and $\kappa_{i,i+1}$.

BDEnc. User i sets $J_{i-1,i} := \varphi(\kappa_{i-1,i})$ and $J_{i,i+1} := \varphi(\kappa_{i,i+1})$, and broadcasts $u_i := J_{i,i+1} \cdot J_{i-1,i}^{-1} \in \mathbb{G}$.

KeyComp. User i calculates $K_i := J_{i-1,i}^{n} \cdot u_i^{n-1} \cdot u_{i+1}^{n-2} \cdots u_{i-2}$. Then, $K := K_i = J_{1,2} \cdot J_{2,3} \cdots J_{n,1}$ is the shared key among the n users.

The correctness is shown as the same as the original BD key exchange. The security depends on the map φ. Below, we show that it is proven secure assuming that φ is a secure KDF (see Appendix 2 for its definition) and the underlying protocol Π is secure.

Theorem 3 *The GKE protocol Π^{BD} is (post-quantumly) secure if Π is (post-quantumly) secure, φ is a (post-quantumly) secure KDF and* $\gcd(n, q) = 1$ *where q is the order of $\mathbb{G}$.*

For any (quantum) adversary $\mathcal{A}$, there exist (quantum) machines $\mathcal{B}_l$ and $\mathcal{C}_l$, whose running times are essentially the same as that of $\mathcal{A}$, such that $\mathsf{Adv}_{\mathcal{A}}^{\Pi^{\mathsf{BD}}}(\lambda) \leq \sum_{l\in[2n]} \left(\mathsf{Adv}_{\mathcal{B}_l}^{\Pi}(\lambda) + \mathsf{Adv}_{\mathcal{C}_l}^{\mathsf{KDF}}(\lambda)\right) + \varepsilon(\lambda)$, *where $\varepsilon(\lambda)$ is a negligible function in λ.*

Proof The view of $\mathcal{A}$ consists of $(u_1, \ldots, u_n, K)$. To prove Theorem 3, we consider the following $2n + 2$ games. An underlined part indicates a variable that is changed in a game from the previous one.

Game 0: Original game, which is the same as the first case in Definition 1. The values of $J_{i-1,i}, u_i, K$ are given as $J_{i-1,i} := \varphi(\kappa_{i-1,i})$,

$$u_i := J_{i,i+1} \cdot J_{i-1,i}^{-1} \text{ for } i \in [n], \quad K := J_{1,2} \cdot J_{2,3} \cdots J_{n-1,n} \cdot J_{n,1}, \tag{7}$$

where $\kappa_{i-1,i}$ is a shared key by running Π between users $i - 1$ and i.

Game l ($l \in [n]$): The lth output of φ is $\underline{J_{l-1,l} \leftarrow_R \mathbb{G}}$ (for both of users $l - 1$ and l), all the other $J_{i-1,i}$'s for $i \neq l$ are generated as in Game $l - 1$, and the view of $\mathcal{A}$, i.e., $(u_1, \ldots, u_n, K)$, are generated as in Eq. (7) from all the $J_{i-1,i}$'s for $i \in [n]$.

Game $n + 1$: Same as Game n except that the shared key is $\underline{K \leftarrow_R \mathbb{G}}$, and all the other variables are generated as in Game n. Note that K is independent of all the other variables.

Game $n + 1 + l$ ($l \in [n]$): The lth output of φ is $\underline{J_{l-1,l} := \varphi(\kappa_{l-1,l})}$ (for both of users $l - 1$ and l), all the other $J_{i-1,i}$'s for $i \neq l$ are generated as in Game $n + l$, and $(u_1, \ldots, u_n)$ are generated as in Eq. (7) from all the $J_{i-1,i}$'s for $i \in [n]$ and $K \leftarrow_R \mathbb{G}$. Here, note that Game $2n + 1$ is the same as the second case in Definition 1.

Let $\mathsf{Adv}_{\mathcal{A}}^{(l)}(\lambda)$ be the advantage of $\mathcal{A}$ in Game l, respectively.

We will show three lemmas (Lemmas 5–7) that evaluate the gaps between pairs of the advantages in Game 0, …, Game $2n + 1$. From these lemmas, we obtain $\mathsf{Adv}_{\mathcal{A}}^{\Pi^{\mathsf{BD}}}(\lambda) \leq \sum_{l\in[2n+1]} \left|\mathsf{Adv}_{\mathcal{A}}^{(l-1)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(l)}(\lambda)\right| \leq \sum_{l\in[2n]} \left(\mathsf{Adv}_{\mathcal{B}_l}^{\Pi}(\lambda) + \mathsf{Adv}_{\mathcal{C}_l}^{\mathsf{KDF}}(\lambda)\right) + \varepsilon(\lambda)$ where $\varepsilon(\lambda) := \sum_{l\in[2n]} \varepsilon_l(\lambda)$ is a negligible function. This completes the proof of Theorem 3. □

Lemma 5 *For any (quantum) adversary $\mathcal{A}$, there exist (quantum) machines $\mathcal{B}_l$ and $\mathcal{C}_l$, whose running times are essentially the same as that of $\mathcal{A}$, such that $|\mathsf{Adv}^{(l-1)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(l)}_{\mathcal{A}}(\lambda)| \leq \mathsf{Adv}^{\Pi}_{\mathcal{B}_l}(\lambda) + \mathsf{Adv}^{\mathsf{KDF}}_{\mathcal{C}_l}(\lambda) + \varepsilon_l(\lambda)$ for $l \in [n]$, where $\varepsilon_l(\lambda)$ are negligible functions.*

Proof For the proof, we define an intermediate game, i.e., Game $l - 1/2$, between Games $l-1$ and l. In Game $l-1/2$, $\kappa_{l-1,l} \leftarrow_R \mathbb{K}$ and $J_{l-1,l} := \varphi(\kappa_{l-1,l})$, and the rest of variables are all generated in the same manner as in Game $l-1$.

By the definition of two-party KE, the difference of the advantages of Games $l-1$ and $l-1/2$ is bounded by the advantage against the KE protocol Π, i.e., $\mathsf{Adv}^{\Pi}_{\mathcal{B}_l}(\lambda)$ (except with negligible probability). Since the keyspace $\mathbb{K}$ has enough entropy, by the definition of KDF, the difference of the advantages of Games $l-1/2$ and l is bounded by the advantage against KDF, i.e., $\mathsf{Adv}^{\mathsf{KDF}}_{\mathcal{C}_l}(\lambda)$ (except with negligible probability). This completes the proof of Lemma 5. □

Lemma 6 (BDEnc Information-Theoretic Security) *For any (quantum) adversary $\mathcal{A}$, for any security parameter λ, $\mathsf{Adv}^{(n+1)}_{\mathcal{A}}(\lambda) = \mathsf{Adv}^{(n)}_{\mathcal{A}}(\lambda)$.*

Proof We can set $J_{i-1,i} := g^{\alpha_{i-1}}$ for $i \in [n]$, where $g \in \mathbb{G}$ is a generator and $\alpha_i \leftarrow_R \mathbb{Z}/q\mathbb{Z}$ (which are independent from each other). Then, $u_i := J_{i,i+1} \cdot J^{-1}_{i-1,i} = g^{\alpha_i - \alpha_{i-1}}$. First, we see that n elements $(\ \alpha_1,\ \alpha_2 - \alpha_1,\ \alpha_3 - \alpha_2, \ldots, \alpha_n - \alpha_{n-1}\ )$ are uniformly and independently distributed. Since $\alpha_1 + \cdots + \alpha_n = n\alpha_1 + (n-1)(\alpha_2 - \alpha_1) + (n-2)(\alpha_3 - \alpha_2) + \cdots + (\alpha_n - \alpha_{n-1})$ and n mod q has an inverse element (from the assumption $\gcd(n, q) = 1$), n elements $(\ \alpha_1 + \cdots + \alpha_n,\ \alpha_2 - \alpha_1,\ \alpha_3 - \alpha_2, \ldots, \alpha_n - \alpha_{n-1}\ )$ are also uniformly and independently distributed. Since $K = g^{\alpha_1 + \cdots + \alpha_n}$, K is independent of all the other variables, i.e., h_i, u_i. This completes the proof of Lemma 6. □

Lemma 7 *For any (quantum) adversary $\mathcal{A}$, there exists (quantum) machines $\mathcal{B}_{n+l}$ and $\mathcal{C}_{n+l}$, whose running times are essentially the same as that of $\mathcal{A}$, such that for any security parameter λ, $|\mathsf{Adv}^{(n+l)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(n+l+1)}_{\mathcal{A}}(\lambda)| \leq \mathsf{Adv}^{\Pi}_{\mathcal{B}_{n+l}}(\lambda) + \mathsf{Adv}^{\mathsf{KDF}}_{\mathcal{C}_{n+l}}(\lambda) + \varepsilon_{n+l}(\lambda)$ for $l \in [n]$, where $\varepsilon_{n+l}(\lambda)$ are negligible functions.*

Lemma 7 is proven in a similar manner to Lemma 5.

4.2 Constant-Round GKE from Static Standard Assumptions

We instantiate the above generic GKE by Apon et al.'s ring LWE based GKE (Apon et al. 2019) by using a two-party KE Π and some SHA-2 (or SHA-3) based KDF φ, whose range is $\mathbb{G} := \mathbb{F}^*$ for some finite field $\mathbb{F}$. Therefore, we have the following corollary.

Corollary 1 *There exists a post-quantum constant-round GKE from two-party KE Π in Apon et al. (2019) and some standard KDF function φ under the* static *ring LWE assumption.*

Apon et al.'s original GKE is based on the "non-static" or "dynamic" R-LWE assumption. That is, the noise size depends on the number of group members n, then the scheme itself gets to large sizes.

Corollary 2 *There exists a post-quantum constant-round GKE from two-party KE SIDH (resp. CSIDH) and some standard KDF function φ under the SI-DDH (resp. CSI-DDH) assumption.*

4.3 Two-Round Product-BD (PBD) GKE from d-DSJP Assumption

We modify the SIBD Group Key Exchange proposed in Furukawa et al. (2018) to a provably secure one, called Supersingular Isogeny Product-BD ((n, d)-SI-PBD) protocol for n-parties. In other words, our general (n, d)-SI-PBD protocol is obtained via our generic compiler (in Sect. 4.1) from two-party $(2, d)$-SI-PBD protocol, where a $\mathbb{G}$-embedding map φ is given by the identity map $\varphi := \mathsf{id}_{\mathbb{G}} : \mathbb{G} \to \mathbb{G}$.

4.3.1 Construction

We consider n-party key exchange. Each user is indexed by $1, 2, \ldots, n$, where n is supposed to be even for simplicity. Note that we can easily obtain the protocol for odd n. The user indices are taken in a cycle: so $R_{n+1} := R_1$ and $R_0 := R_n$. We introduce the map $\iota(i) := i \bmod 2$ and we will simply write ι instead of writing $\iota(i)$.

Setup. Takes a security parameter λ and the number of users n. The algorithm outputs $\mathsf{params}_{\mathsf{SIDH}} := (p(= f\ell_0^{e_0}\ell_1^{e_1} \pm 1), E, \{P_0, Q_0\}, \{P_1, Q_1\})$ for SIDH.

Round-1. Takes the user index i and params as input. User i randomly chooses $k_i^{(\mu)} \in \mathbb{Z}/\ell_\iota^{e_\iota}\mathbb{Z}$ and computes $R_i^{(\mu)} := P_\iota + k_i^{(\mu)} Q_\iota$. User i then computes the isogeny $\phi_i^{(\mu)}$ and elliptic curve $E_i^{(\mu)} := E/\langle R_i^{(\mu)}\rangle$ such that $\phi_i^{(\mu)} : E \to E_i^{(\mu)}$, where $\ker(\phi_i^{(\mu)}) = \langle R_i^{(\mu)}\rangle$. The user i then sets $\mathsf{pk}_i^1 = \left(E_i^{(\mu)}, \phi_i^{(\mu)}(P_{1-\iota}), \phi_i^{(\mu)}(Q_{1-\iota})\right)_{\mu\in[d]}$ and $\mathsf{sk}_i^1 := \left(k_i^{(\mu)}\right)_{\mu\in[d]}$. Finally, the user i broadcasts pk_i^1 to the other users.

Round-2. Takes the user index i, $\mathsf{params}_{\mathsf{SIDH}}$, $\left(\mathsf{pk}_{i-1}^1, \mathsf{pk}_{i+1}^1\right)$, and sk_i^1. User i executes SIDH key exchange with users $i-1$ and $i+1$ to obtain elliptic curves $E_{i-1,i}^{(\mu)}$ and $E_{i,i+1}^{(\mu)}$, respectively, and then computes

$$J_{i-1,i} := \prod_{\mu=1}^{d} j\left(E_{i-1,i}^{(\mu)}\right) \quad \text{and} \quad J_{i,i+1} := \prod_{\mu=1}^{d} j\left(E_{i,i+1}^{(\mu)}\right).$$

The user then computes $u_i := J_{i,i+1} \cdot J_{i-1,i}^{-1}$ and set $\mathsf{pk}_i^2 := u_i$. Finally, the user i broadcasts pk_i^2 to the other users.

KeyComp. User i collects $\left(\mathsf{pk}_{i'}^2\right)_{i'\in[n]}$ and sk_i^1 and computes $K_i := J_{i-1,i}^n \cdot u_i^{n-1} \cdot u_{i+1}^{n-2} \cdot \cdots \cdot u_{i-3}^2 \cdot u_{i-2}$.

We can easily verify that $K_i = J_{1,2} \cdot J_{2,3} \cdots J_{n-1,n} \cdot J_{n,1}$ holds for any i.

4.3.2 Warm-Up: Security from a Nonstatic Assumption

We rephrase security of the (n, d)-SI-PBD protocol based on Definition 1 as a form of the following assumption (see Lemma 8).

Definition 6 (*Decisional SI-PBD ((n,d)-SI-PBD) Assumption*) Let $(\Psi_{n,d}, K) \leftarrow_R \mathsf{Exec}_{(n,d)\text{-SI-PBD}}(\lambda)$, where $J_{i-1,i} := \prod_{\mu=1}^{d} j\left(E_{i-1,i}^{(\mu)}\right)$, $J_{i,i+1} := \prod_{\mu=1}^{d} j\left(E_{i,i+1}^{(\mu)}\right)$, $u_i := J_{i,i+1} \cdot J_{i-1,i}^{-1}$, $\Psi_{n,d} := \left(\mathsf{params}_{\mathsf{SIDH}}, \left(\left(E_i^{(\mu)}, \phi_i^{(\mu)}(P_{1-\iota}), \phi_i^{(\mu)}(Q_{1-\iota})\right), u_i\right)_{i\in[n],\mu\in[d]}\right)$, and $K := \prod_{i=1}^{n} J_{i,i+1}$. An (n, d)-SI-PBD problem instance is given as $(\Psi_{n,d}, \kappa_\beta)$, where

$$\kappa_0 := K, \qquad \kappa_1 \leftarrow_R \mathbb{F}_{p^2},$$

and $\beta \leftarrow_R \{0, 1\}$. For any quantum algorithm $\mathcal{B}$, the advantage of $\mathcal{B}$ is defined as $\mathsf{Adv}_{\mathcal{B}}^{(n,d)\text{-SI-PBD}}(\lambda) := |\Pr[\mathcal{B}(\Psi_{n,d}, \kappa_0) = 1] - \Pr[\mathcal{B}(\Psi_{n,d}, \kappa_1) = 1]|$, and the (n, d)-SI-PBD assumption holds if $\mathsf{Adv}_{\mathcal{B}}^{(n,d)\text{-SI-PBD}}(\lambda)$is negligible in λ for any polynomial-time quantum adversary $\mathcal{B}$.

Remark 1 We have better security proofs when $d \geq 2$ for the (n, d)-SI-PBD GKE (Theorem 4). However, the above gives only security proofs for the $d = 1$ case, which is based on nonstatic assumptions. Note that since $n \geq 3$ and the key K is a n-time product of j-invariants, then we have no efficient distinguishing algorithm between κ_0 and κ_1.

Lemma 8 *The (n, d)-SI-PBD key exchange among n-parties is post-quantumly secure under the (n, d)-SI-PBD assumption.*

Proof Lemma 8 is trivially obtained from Definitions 1 and 6. □

If the (n, d)-SI-PBD problem is quantum resistantly hard, the SI-PBD key exchange among n-parties is also quantum resistant. Therefore, we should investigate the post-quantum security of the (n, d)-SI-PBD assumption in the next section.

Moreover, as is shown in Lemma 1 for the d-DSJP assumptions, the family of (n, d)-SI-PBD assumptions also has natural sequential reductions among them.

Lemma 9 *The (n, d)-SI-PBD assumption is reduced to the $(n, d + 1)$-SI-PBD assumption.*

For any adversary $\mathcal{A}$, there is a (quantum) machine $\mathcal{B}$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter λ, $\mathsf{Adv}_{\mathcal{A}}^{(n,d+1)\text{-SI-PBD}}(\lambda) \leq \mathsf{Adv}_{\mathcal{B}}^{(n,d)\text{-SI-PBD}}(\lambda)$.

Proof The proof of Lemma 9 is similarly given to that of Lemma 1. □

Lemma 9 shows that $(n, d + 1)$-SI-PBD group key exchange is more secure than (n, d)-SI-PBD one while the former is less efficient than the latter in terms of data sizes and execution times.

4.3.3 Security from d-DSJP Assumption for $d \geq 2$

Theorem 4 *The (n, d)-SI-PBD key exchange among n-parties is post-quantumly secure under the d-DSJP assumption when $d \geq 2$ and $\gcd(n, p^2 - 1) = 1$. (Note that $p^2 - 1$ is the order of cyclic group $\mathbb{G} := \mathbb{F}^*_{p^2}$.)*

For any quantum adversary $\mathcal{A}$, there exist quantum machines $\mathcal{B}_l$, whose running times are essentially the same as that of $\mathcal{A}$, such that $\mathsf{Adv}^{(n,d)\text{-SI-PBD}}_{\mathcal{A}}(\lambda) \leq \sum_{l\in[2n]} \mathsf{Adv}^{d\text{-DSJP}}_{\mathcal{B}_l}(\lambda)$ when $d \geq 2$.

Proof The view of $\mathcal{A}$ consists of $(u_1, \ldots, u_n, K)$. To prove Theorem 4, we consider the following $2n + 2$ games. An underlined part indicates a variable that is changed in a game from the previous one.

Game 0: Original game. That is, the values of $J_{i-1,i}, u_i, K$ are given as $J_{i-1,i} := \prod_{\mu=1}^{d} j\left(E^{(\mu)}_{i-1,i}\right)$,

$$u_i := J_{i,i+1} \cdot J^{-1}_{i-1,i} \text{ for } i \in [n], \quad K := J_{1,2} \cdot J_{2,3} \cdots J_{n-1,n} \cdot J_{n,1}. \tag{8}$$

Game l ($l \in [n]$): The lth output of φ is: $\underline{J_{l-1,l} \leftarrow_R \mathbb{F}_{p^2}}$ (for both of users $l - 1$ and l), all the other $J_{i-1,i}$'s for $i \neq l$ are generated as in Game $l - 1$, and the view of $\mathcal{A}$, i.e., $(u_1, \ldots, u_n, K)$, are generated as in Eq. (8) from all the $J_{i-1,i}$'s for $i \in [n]$.

Game $n + 1$: Same as Game n except that the shared key is $\underline{K \leftarrow_R \mathbb{F}_{p^2}}$, and all the other variables are generated as in Game n. Note that K is independent of all the other variables.

Game $n + 1 + l$ ($l \in [n]$): The lth output of φ is: $\underline{J_{l-1,l} := \prod_{\mu=1}^{d} j\left(E^{(\mu)}_{l-1,l}\right)}$ (for both of users $l - 1$ and l), all the other $J_{i-1,i}$'s for $\underline{i \neq l}$ are generated as in Game $n + l$, $(u_1, \ldots, u_n)$, are generated as in Eq. (8) from all the $J_{i-1,i}$'s for $i \in [n]$ and $K \leftarrow_R \mathbb{F}_{p^2}$. Here, note that Game $2n + 1$ is the same as the $\beta = 1$ case in Definition 6.

Let $\mathsf{Adv}^{(l)}_{\mathcal{A}}(\lambda)$ be the advantage of $\mathcal{A}$ in Game i, respectively.

We will show three lemmas (Lemmas 10–12) that evaluate the gaps between pairs of the advantages in Game 0, …, Game $2n + 1$. From these lemmas, we obtain $\mathsf{Adv}^{(n,d)\text{-SI-PBD}}_{\mathcal{A}}(\lambda) \leq \sum_{l\in[2n+1]} \left|\mathsf{Adv}^{(l-1)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(l)}_{\mathcal{A}}(\lambda)\right| \leq \sum_{l\in[2n]} \mathsf{Adv}^{d\text{-DSJP}}_{\mathcal{B}_l}(\lambda)$. This completes the proof of Theorem 4. □

Lemma 10 *For any quantum adversary $\mathcal{A}$, there exists a quantum machine $\mathcal{B}_l$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter λ, $|\mathsf{Adv}^{(l-1)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(l)}_{\mathcal{A}}(\lambda)| \leq \mathsf{Adv}^{d\text{-DSJP}}_{\mathcal{B}_l}(\lambda)$ for $l \in [n]$.*

Proof $\mathcal{B}$ is given a d-DSJP instance (Ψ_{AB}, J_β), where

$$\Psi_{AB} := \left(\mathsf{params}, \left(\left(E^{(\mu)}_A, \phi^{(\mu)}_A(P_B), \phi^{(\mu)}_A(Q_B)\right), \left(E^{(\mu)}_B, \phi^{(\mu)}_B(P_A), \phi^{(\mu)}_B(Q_A)\right)\right)_{\mu\in[d]}\right).$$

$\mathcal{B}$ (implicitly) sets user $l-1$ A and user l B, and their public keys $\left(E_{l-1}^{(\mu)}, \phi_{l-1}^{(\mu)}(P_\iota), \phi_{l-1}^{(\mu)}(Q_\iota)\right)_{\mu\in[d]} := \left(E_A^{(\mu)}, \phi_A^{(\mu)}(P_B), \phi_A^{(\mu)}(Q_B)\right)_{\mu\in[d]}$ and $\left(E_l^{(\mu)}, \phi_l^{(\mu)}(P_{\iota-1}), \phi_l^{(\mu)}(Q_{\iota-1})\right)_{\mu\in[d]} := \left(E_B^{(\mu)}, \phi_B^{(\mu)}(P_A), \phi_B^{(\mu)}(Q_A)\right)_{\mu\in[d]}$, respectively.

$\mathcal{B}$ generates randomly $J_{i-1,i} \leftarrow_R \mathbb{F}_{p^2}$ for $i < l$, and sets $(l-1)$th j-invariants product as $J_{l-1,l} := J_\beta$. $\mathcal{B}$ generates secret keys $k_i^{(\mu)} \leftarrow_R \mathbb{Z}/\ell_\tau^{e_\tau}\mathbb{Z}$ for all $i \in [n] \setminus \{l-1, l\}$ where $\tau := i \mod n$, and then his/her own public keys $\left(E_i^{(\mu)}, \phi_i^{(\mu)}(P_{\tau-1}), \phi_i^{(\mu)}(Q_{\tau-1})\right)_{\mu\in[d]}$. Since $\mathcal{B}$ has all secret keys except for users $l-1, l$, he can compute all correct j-invariant products $J_{i-1,i}$ for $i > l$.

Using $J_{i-1,i}$ for $i \in [n]$ as defined above, $\mathcal{B}$ computes $u_i := J_{i,i+1} \cdot J_{i-1,i}^{-1}$ and $K := \prod_{i\in[n]} J_{i-1,i}$, and then sends $\mathcal{A}$ the public keys, $(u_i)_{i\in[n]}$, and the challenge value K.

If $\mathcal{A}$ outputs β', then $\mathcal{B}$ also outputs β'. We easily see that the distribution generated by $\mathcal{B}$ is that in Game $l-1$ when $\beta = 0$ and that in Game i when $\beta = 1$.

This completes the proof of Lemma 10. □

Lemma 11 *For any (quantum) adversary $\mathcal{A}$, for any security parameter λ,* $\mathsf{Adv}_{\mathcal{A}}^{(n+1)}(\lambda) = \mathsf{Adv}_{\mathcal{A}}^{(n)}(\lambda)$.

Proof The proof of Lemma 11 is the same as that of Lemma 6 (BDEnc Information Theoretic Security Lemma). □

Lemma 12 *For any quantum adversary $\mathcal{A}$, there exists a quantum machine $\mathcal{B} := \mathcal{B}_{n+l}$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter λ,* $|\mathsf{Adv}_{\mathcal{A}}^{(n+l)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(n+l+1)}(\lambda)| \leq \mathsf{Adv}_{\mathcal{B}_{n+l}}^{d\text{-}\mathsf{DSJP}}(\lambda)$ *for $l \in [n]$.*

Lemma 12 is proven in a similar manner to Lemma 10.

4.4 Two-Round PBD GKE from d-DSMP Assumption

Setup. Takes a security parameter λ and the number of users n. The algorithm outputs $\mathsf{params}_{\mathsf{CSIDH}} := (p(= 4 \cdot \ell_1 \cdots \ell_s - 1), E)$.

Round-1. Takes the user index i and $\mathsf{params}_{\mathsf{CSIDH}}$ as input. User i randomly chooses $\mathbf{e}_i^{(\mu)} := \left(e_{i,1}^{(\mu)}, \ldots, e_{i,s}^{(\mu)}\right)$ and defines $\left[\mathfrak{a}_i^{(\mu)}\right] := \left[\mathfrak{l}_1^{e_{i,1}^{(\mu)}} \cdots \mathfrak{l}_s^{e_{i,s}^{(\mu)}}\right]$. User i then computes elliptic curve $E_i^{(\mu)} := \left[\mathfrak{a}_i^{(\mu)}\right] E$ and sets $\mathsf{pk}_i^1 := \left(E_i^{(\mu)}\right)_{\mu\in[d]} := \left(\left[\mathfrak{a}^{(\mu)}\right] E\right)_{\mu\in[d]}$ and $\mathsf{sk}_i^1 := \left(\mathbf{e}^{(\mu)}\right)_{\mu\in[d]}$. Finally, the user i broadcast pk_i^1 to the other users.

Round-2. Takes the user index i, $\mathsf{params}_{\mathsf{CSIDH}}$, $(\mathsf{pk}_{i-1}^1, \mathsf{pk}_{i+1}^1)$, and sk_i^1. User i executes CSIDH key exchange with users $i-1$ and $i+1$ to obtain elliptic curves $E_{i-1,i}^{(\mu)}$ and $E_{i,i+1}^{(\mu)}$, respectively, and then computes

$$M_{i-1,i} := \prod_{\mu=1}^{d} m\left(E_{i-1,i}^{(\mu)}\right) \text{ and } M_{i,i+1} := \prod_{\mu=1}^{d} m\left(E_{i,i+1}^{(\mu)}\right).$$

The user then computes $u_i := M_{i,i+1} \cdot M_{i-1,i}^{-1}$ and set $\mathsf{pk}_i^2 := u_i$. Finally, the user i broadcasts pk_i^2 to the other users.

KeyComp. User i collects $\left(\mathsf{pk}_{i'}^2\right)_{i' \in [n]}$ and sk_i^1 and computes $K_i := M_{i-1,i}^n \cdot u_i^{n-1} \cdot u_{i+1}^{n-2} \cdot \cdots \cdot u_{i-3}^2 \cdot u_{i-2}$.

We can easily verify that $K_i = M_{1,2} \cdot M_{2,3} \cdots M_{n-1,n} \cdot M_{n,1}$ holds for any i. We have the following lemma and theorem as in the case of the SI-PBD key exchange. The (n, d)-CSI-PBD assumption is defined in Definition 7 in Appendix 4.

Lemma 13 *The (n, d)-CSI-PBD key exchange among n-parties is secure under the (n, d)-CSI-PBD assumption.*

Theorem 5 *The (n, d)-CSI-PBD key exchange among n-parties is post-quantumly secure under the d-DSMP assumption when $d \geq 2$ and $\gcd(n, p-1) = 1$. (Note that $p-1$ is the order of cyclic group $\mathbb{G} := \mathbb{F}_p^*$.)*

For any quantum adversary $\mathcal{A}$, there exist quantum machines $\mathcal{B}_i$, whose running times are essentially the same as that of $\mathcal{A}$, such that for any security parameter λ, $\mathsf{Adv}_{\mathcal{A}}^{(n,d)\text{-CSI-PBD}}(\lambda) \leq \sum_{i \in [2n]} \mathsf{Adv}_{\mathcal{B}_i}^{d\text{-DSMP}}(\lambda)$.

Acknowledgements This research was partially supported by JST CREST Grant Number JPMJCR14D6, Japan. The author would like to thank Tatsuaki Okamoto for his valuable comments on the generic GKE construction given in Sect. 4.1.

Appendix 1: BD Group Key Exchange (Burmester and Desmedt 1994)

We describe the BD Key Exchange among n users on a cyclic group $\mathbb{G}$ of a prime order q and a generator g.

Round-1. Each user i generates $a_i \leftarrow_R \mathbb{Z}/q\mathbb{Z}$, $h_i := g^{a_i}$ and broadcasts h_i.

Round-2. Each user i calculates $J_{i-1,i} := (h_{i-1})^{a_i}$, $J_{i,i+1} := (h_{i+1})^{a_i}$ and $u_i := J_{i,i+1} \cdot J_{i-1,i}^{-1}$. User i broadcasts u_i.

KeyComp. User i calculates $K_i := J_{i-1,i}^n \cdot u_i^{n-1} \cdot u_{i+1}^{n-2} \cdots u_{i-2}$. Then, $K_i = J_{1,2} \cdot J_{2,3} \cdots J_{n,1}$ is the shared key among the n users.

Theorem 6 (Burmester and Desmedt 1994; Katz and Yung 2007) *The BD group key exchange is tightly secure under the DDH assumption. For any adversary $\mathcal{A}$, there is a probabilistic machine $\mathcal{B}$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter λ, $\mathsf{Adv}_{\mathcal{A}}^{\mathrm{BD}}(\lambda) \leq \mathsf{Adv}_{\mathcal{B}}^{\mathrm{DDH}}(\lambda)$.*

Proof DDH solver $\mathcal{B}$ uses an attacker $\mathcal{A}$ against the BD protocol. Below, we prove the case n is even for simplicity. $\mathcal{B}$ receives a DDH tuple (g, g^a, g^b, T) where T is g^{ab} or g^c with random c, and should simulate public information $(h_i, u_i)_{i \in [n]}$ and the shared key K. $\mathcal{B}$ implicitly sets $a_1 := a$ and $a_2 := b$, and generates random $\tilde{a}_2, \tilde{a}_3, \ldots, \tilde{a}_{n-1} \leftarrow \mathbb{Z}/q\mathbb{Z}$. $\mathcal{B}$ also implicitly sets relations

$$\tilde{a}_2 = a_2 - a_n,\ \tilde{a}_3 = a_3 - a_1, \ldots, \tilde{a}_{n-2} = a_{n-2} - a_{n-4},\ \tilde{a}_{n-1} = a_{n-1} - a_{n-3}, \quad (9)$$

which determines $a_3, \ldots, a_{n-1}$ as linear combinations of $a(= a_1), b(= a_2), \tilde{a}_3, \ldots, \tilde{a}_{n-1}$, that is, $a_3 := a_1 + \tilde{a}_3, \ldots, a_{n-2} := a_{n-4} + \tilde{a}_{n-2} = b + \tilde{a}_4 + \cdots + \tilde{a}_{n-2}$, $a_{n-1} := a_{n-3} + \tilde{a}_{n-1} = a + \tilde{a}_3 + \cdots + \tilde{a}_{n-1}$, $a_n := a_2 - \tilde{a}_2$.

Therefore, $\mathcal{B}$ simulates h_i as follows: $h_1 := g^a$, $h_2 := g^b$, $h_3 := g^{a_1+\tilde{a}_3} = g^a \cdot g^{\tilde{a}_3}$, $h_4 := g^{a_2+\tilde{a}_4} = g^b \cdot g^{\tilde{a}_4}, \ldots, h_{n-2} := g^{b+\tilde{a}_4+\cdots+\tilde{a}_{n-2}} = g^b \cdot g^{\tilde{a}_4+\cdots+\tilde{a}_{n-2}}$, $h_{n-1} := g^{a+\tilde{a}_3+\cdots+\tilde{a}_{n-1}} = g^a \cdot g^{\tilde{a}_3+\cdots+\tilde{a}_{n-1}}$, $h_n := g^{a_2-\tilde{a}_2} = g^b \cdot g^{-\tilde{a}_2}$, and $\mathcal{B}$ also simulates u_i as follows using relations (9), $u_i := h_i^{\tilde{a}_{i+1}}$ for $i = 1, \ldots, n-2, u_{n-1} := h_{n-1}^{-\sum_{i=1,3,\ldots,n-3} \tilde{a}_{i+1}}$, $u_n := h_n^{-\sum_{i=2,4,\ldots,n-2} \tilde{a}_{i+1}}$, where $a_n - a_{n-2} = (a_2 - \tilde{a}_2) - (a_2 + \tilde{a}_4 + \cdots + \tilde{a}_{n-2}) = -\sum_{i=1,3,\ldots,n-3} \tilde{a}_{i+1}$ and $a_1 - a_{n-1} = -\sum_{i=2,4,\ldots,n-2} \tilde{a}_{i+1}$ hold. Here, $\mathcal{B}$'s simulations of h_i and u_i are perfect.

Since the correct $K = K_2$ is $K_2 = J_{1,2}^n \cdot u_2^{n-1} \cdot u_3^{n-2} \cdots u_n$ with $J_{1,2} = g^{ab}$, $\mathcal{B}$ simulates shared key K as $K := T^n \cdot u_2^{n-1} \cdot u_3^{n-2} \cdots u_n$ where T is given in the DDH instance and u_i are calculated as above, and then $\mathcal{B}$ give it to $\mathcal{A}$. When $\mathcal{A}$ answers to the question whether K is correct or random, $\mathcal{B}$ answers to his problem as the same way as $\mathcal{A}$.

If $T = g^{ab}$, then the simulation is the same as the real game, and if $T = g^c$, then K is uniformly random and independently distributed from other variables. □

Appendix 2: Key Derivation Function (KDF)

Let two-party key exchange denote Π with shared key space $\mathbb{K}$. A map $\varphi : \mathbb{K} \to \mathbb{G}$ is called key derivation function (with a range $\mathbb{G}$) if two distributions $\{ \varphi(\kappa) \mid \kappa \leftarrow_R \mathbb{K} \}$ and $\{ J \leftarrow_R \mathbb{G} \}$ are indistinguishable. Such a KDF function can be obtained from a standard hash function, e.g., SHA-2 or SHA-3. For the details, see Abe et al. (2005), for example.

Appendix 3: SIDH and CSIDH Key Exchange

Appendix 3.1: SIDH Key Exchange (Feo et al. 2014)

A supersingular elliptic curve E and generators of smooth order rank-2 torsion subgroups are taken as pubic parameters. Alice and Bob set random cyclic subgroups as secret keys, respectively, and calculate isogenies whose kernels are the secret keys by using Vélu's formulas. They publish their public keys, range curves of the isogenies, and images of the generators, respectively. Finally, they calculate isogenies from public keys. The range curves of the isogenies are isomorphic; therefore their j-invariants become the same. The detailed protocol is given as follows.

Setup. Let $e_A, e_B \in \mathbb{Z}$, and ℓ_A, ℓ_B be small primes (e.g., 2, 3), where $\ell_A^{e_A}$ and $\ell_B^{e_B}$ are close. Let p be a prime which satisfies that $p = \ell_A^{e_A}\ell_B^{e_B} f \pm 1$ where f is a small positive integer. Let $E\colon \delta y^2 = x^3 + \alpha x^2 + x$ be a supersingular elliptic curve defined over $\mathbb{F}_{p^2}$, where the cardinality of $E(\mathbb{F}_{p^2})$ is $(\ell_A^{e_A}\ell_B^{e_B} f)^2$. Let P_A, Q_A be generators of $E[\ell_A^{e_A}]$, and P_B, Q_B are generators of $E[\ell_B^{e_B}]$. Let public parameters be $\mathsf{params}_{\mathsf{SIDH}} := (p, E, P_A, Q_A, P_B, Q_B)$.

Round-1. Alice chooses random numbers $k_A \in (\mathbb{Z}/\ell_A^{e_A}\mathbb{Z})^\times$, and calculates $R_A = P_A + k_A Q_A$. Here, an order of R_A is $\ell_A^{e_A}$. Alice calculates an $\ell_A^{e_A}$-isogeny $\phi_A\colon E \to E_A\colon = E/\langle R_A\rangle$ and $\phi_A(P_B), \phi_A(Q_B)$ by using Vélu formulas.
Similarly, Bob chooses random numbers $k_B \in (\mathbb{Z}/\ell_B^{e_B}\mathbb{Z})^\times$, and calculates $R_B = P_B + k_B Q_B$. Here, an order of R_B is $\ell_B^{e_B}$. Bob calculates an $\ell_B^{e_B}$-isogeny $\phi_B\colon E \to E_B\colon = E/\langle R_B\rangle$ and $\phi_B(P_A), \phi_B(Q_A)$ by using Vélu formulas.
Alice sends $E_A, \phi_A(P_B), \phi_A(Q_B)$ to Bob, and Bob sends $E_B, \phi_B(P_A), \phi_B(Q_A)$ to Alice.

KeyComp. Alice calculates $R'_A = \phi_B(P_A) + k_A\phi_B(Q_A)$. Here, an order of R'_A is $\ell_A^{e_A}$. Alice calculates an $\ell_A^{e_A}$-isogeny $\phi'_A\colon E_B \to E_{AB}\colon = E_B/\langle R'_A\rangle$ and $K_A = j(E_{AB})$ by using Vélu formulas.
Bob calculates $R'_B = \phi_A(P_B) + k_B\phi_A(Q_B)$. Here, an order of R'_B is $\ell_B^{e_B}$. Bob calculates an $\ell_B^{e_B}$-isogeny $\phi'_B\colon E_A \to E_{BA}\colon = E_A/\langle R'_B\rangle$ and $K_B = j(E_{BA})$ by using Vélu formulas.

It holds that $\ker(\phi'_A \circ \phi_B) = {\phi_B}^{-1}(\langle R'_A\rangle) = \langle R_A\rangle \oplus \langle R_B\rangle$ and $\ker(\phi'_B \circ \phi_A) = {\phi_A}^{-1}(\langle R'_B\rangle) = \langle R_B\rangle \oplus \langle R_A\rangle$. Hence, $K_A = K_B$ holds; therefore, SIDH is correct.
The SI-DDH assumption is defined in Definition 2.

Theorem 1 (Feo et al. 2014) *The SIDH key exchange is post-quantumly secure under the SI-DDH assumption.*

Appendix 3.2: CSIDH Key Exchange (Castryck et al. 2018)

CSIDH (Commutative Supersingular Isogeny Diffie–Hellman) was proposed by Castryck et al. in 2018 (Castryck et al. 2018).

Let a prime $p := 4 \cdot \ell_1 \cdots \ell_s - 1$, where $\ell_1, \ldots, \ell_s$ are small distinct odd primes. Let $\mathcal{O}$ be an order in an imaginary quadratic field, $\pi \in \mathcal{O}$, π_p the pth power Frobenius endomorphism and $\mathcal{E}\ell\ell_p(\mathcal{O}, \pi)$ the set of $\mathbb{F}_p$-isomorphism classes of $\mathbb{F}_p$-rational supersingular elliptic curves whose $\mathbb{F}_p$-endomorphism ring is equal to $\mathcal{O}$ and the Frobenius π_p is given by $\pi \in \mathcal{O}$. For CSIDH, we only consider the case that $\mathcal{O} \cong \mathbb{Z}[\pi_p]$. CSIDH is based on the action of the ideal class group $\mathrm{cl}(\mathcal{O})$ on $\mathcal{E}\ell\ell_p(\mathcal{O}, \pi)$. Alice and Bob generate random elements in $\mathrm{cl}(\mathcal{O})$ for their secret keys, and calculate the actions on $E/\mathbb{F}_p\colon y^2 = x^3 + x$. They publish the obtained elliptic curves as public keys. Finally, they calculate their secret key actions on the public keys, respectively. The obtained elliptic curves are isomorphic over $\mathbb{F}_p$, and the Montgomery coefficients are the same. The detailed protocol is given as follows.

Setup. Let p be a prime as $p = 4 \cdot \ell_1 \cdots \ell_s - 1$, where the $\ell_1, \ldots, \ell_s$ are small distinct odd primes. Let E be the supersingular elliptic curve $y^2 = x^3 + x$ and public parameters $\mathsf{params}_{\mathsf{CSIDH}} := (p, E)$.

Round-1. One randomly chooses an integer vector $(e_1, \ldots, e_s)$ from $\{-\eta, \ldots, \eta\}^s$. Define $[\mathfrak{a}] = \left[\mathfrak{l}_1^{e_1} \cdots \mathfrak{l}_s^{e_s}\right] \in \mathrm{cl}(\mathcal{O})$, where $\mathfrak{l}_i = (\ell_i, \pi_p - 1)$, $\mathfrak{l}_i^{-1} = (\ell_i, \pi_p + 1)$, and η is the smallest integer which satisfies that $2\eta + 1 \geq \sqrt[s]{\#\mathrm{cl}(\mathcal{O})}$. One calculates the action of $[\mathfrak{a}]$ on E and the Montgomery coefficient $m \in \mathbb{F}_p$ of $[\mathfrak{a}]E\colon y^2 = x^3 + mx^2 + x$. Let the integer vector $(e_1, \ldots, e_s)$ (or $[\mathfrak{a}]$) be the secret key, and $m \in \mathbb{F}_p$ be the public key.

KeyComp. Alice (resp. Bob) has her (resp. his) secret key, $[\mathfrak{a}]$ (resp. $[\mathfrak{b}]$). Alice calculates the action $[\mathfrak{a}]E_B = [\mathfrak{a}][\mathfrak{b}]E$, where $E_B\colon y^2 = x^3 + m_B x^2 + x$. Bob calculates the action $[\mathfrak{b}]E_A = [\mathfrak{b}][\mathfrak{a}]E$, where $E_A\colon y^2 = x^3 + m_A x^2 + x$. Define shared keys $K_A := m([\mathfrak{a}][\mathfrak{b}]E)$, and $K_B := m([\mathfrak{b}][\mathfrak{a}]E)$.

By commutativity of $\mathrm{cl}(\mathcal{O})$ and the uniqueness of the Montgomery coefficient, it holds that $K_A = K_B$; therefore, CSIDH is correct.

The CSI-DDH assumption is defined in Definition 3.

Theorem 2 (Castryck et al. 2018) *The CSIDH key exchange is post-quantumly secure under the CSI-DDH assumption.*

Appendix 4: Decisional CSI-PBD ((n, d)-CSI-PBD) Assumption

Definition 7 (*Decisional CSI-PBD ((n, d)-CSI-PBD) Assumption*)
Let $(\Psi_{n,d}, K) \leftarrow_R \mathsf{Exec}_{(n,d)\text{-CSI-PBD}}(\lambda)$, where $M_{i-1,i} := \prod_{\mu=1}^{d} m\left(E_{i-1,i}^{(\mu)}\right)$, $M_{i,i+1} := \prod_{\mu=1}^{d} m\left(E_{i,i+1}^{(\mu)}\right)$, $u_i := M_{i,i+1} \cdot M_{i-1,i}^{-1}$, $\Psi_{n,d} := \left(\mathsf{params}_{\mathsf{CSIDH}}, \left(E_i^{(\mu)}, u_i\right)_{i\in[n],\mu\in[d]}\right)$, and $K := \prod_{i=1}^{n} M_{i,i+1}$. An (n, d)-CSI-PBD problem instance is given as $(\Psi_{n,d}, \kappa_\beta)$ where $\kappa_0 := K$, $\kappa_1 \leftarrow_R \mathbb{F}_p$, and $\beta \leftarrow_R \{0, 1\}$. For any quantum algorithm $\mathcal{B}$, the advantage of $\mathcal{B}$ is defined as $\mathsf{Adv}_{\mathcal{B}}^{(n,d)\text{-CSI-PBD}}(\lambda) := |\Pr[\mathcal{B}(\Psi_{n,d}, \kappa_0) = 1] - \Pr[\mathcal{B}(\Psi_{n,d}, \kappa_1) = 1]|$, and the (n, d)-CSI-PBD assumption holds if $\mathsf{Adv}_{\mathcal{B}}^{(n,d)\text{-CSI-PBD}}(\lambda)$ is negligible in λ for any polynomial-time quantum adversary $\mathcal{B}$.

References

M. Abe, R. Gennaro, K. Kurosawa, V. Shoup, Tag-KEM/DEM: a new framework for hybrid encryption and a new analysis of Kurosawa-Desmedt KEM, in *EUROCRYPT 2005* (2005), pp. 128–146

S. Akleylek, E. Alkim, P.S. Barreto, J. Buchmann, E. Eaton, G. Gutoski, J. Krämer, P. Longa, H. Polat, J.E. Ricardini, G. Zanon, qTESLA. Submission to NIST PQC Standardization (2017)

D. Apon, D. Dachman-Soled, H. Gong, J. Katz, Constant-round group key exchange from the ring-LWE assumption, in *PQCrypto 2019* (2019), pp. 189–205
D.J. Bernstein, T. Lange, C. Martindale, L. Panny, Quantum circuits for the CSIDH: optimizing quantum evaluation of isogenies, in *EUROCRYPT 2019*, Part II (2019), pp. 409–441
W. Beullens, T. Kleinjung, F. Vercauteren, CSI-FiSh: efficient isogeny based signatures through class group computations. IACR Cryptol. ePrint Arch. **2019**, 498 (2019)
D. Boneh, D. Glass, D. Krashen, K. Lauter, S. Sharif, A. Silverberg, M. Tibouchi, M. Zhandry, Multiparty non-interactive key exchange and more from isogenies on elliptic curves, in *MATHCRYPT 2018* (2018), https://eprint.iacr.org/2018/665
C. Boyd, A. Mathuria, *Protocols for Authentication and Key Establishment*, Information Security and Cryptography (Springer, Berlin, 2003)
M. Burmester, Y. Desmedt, A secure and efficient conference key distribution system (extended abstract), in *EUROCRYPT'94* (1994), pp. 275–286
W. Castryck, T. Lange, C. Martindale, L. Panny, J. Renes, CSIDH: an efficient post-quantum commutative group action, in *ASIACRYPT 2018*, Part III (2018), pp. 395–427
L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, P. Schwabe, G. Seiler, D. Stehlé, Crystals-dilithium: a lattice-based digital signature scheme. IACR Trans. Cryptogr. Hardw. Embed. Syst. **2018**(1), 238–268 (2018)
L.D. Feo, D. Jao, J. Plût, Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. J. Math. Cryptol. **8**(3), 209–247 (2014)
P.A. Fouque, J. Hoffstein, P. Kirchner, V. Lyubashevsky, T. Pornin, T. Prest, T. Ricosset, G. Seiler, W. Whyte, Z. Zhang, Falcon: fast-Fourier lattice-based compact signatures over ntru. Submission to NIST PQC Standardization (2017)
A. Fujioka, K. Takashima, S. Terada, K. Yoneyama, Supersingular isogeny Diffie-Hellman authenticated key exchange, in *ICISC 2018* (2018), pp. 177–195
S. Furukawa, N. Kunihiro, K. Takashima, Multi-party key exchange protocols from supersingular isogenies, in *ISITA 2018* (IEEE Xplore, 2018), pp. 208–212
D. Jao, R. Azarderakhsh, M. Campagna, C. Costello, L.D. Feo, B. Hess, A. Jalali, B. Koziel, B. LaMacchia, P. Longa, M.N.J. Renes, V. Soukharev, D. Urbanik, Sike: supersingular isogeny key encapsulation. Submission to NIST PQC Standardization (2017)
M. Just, S. Vaudenay, Authenticated multi-party key agreement, in *ASIACRYPT'96* (1996), pp. 36–49
J. Katz, M. Yung, Scalable protocols for authenticated group key exchange. J. Cryptol. **20**(1), 85–113 (2007)
L. Kowalczyk, H. Wee, Compact adaptively secure ABE for NC^1 from k-lin, in *EUROCRYPT 2019*, Part I (2019), pp. 3–33
T. Okamoto, K. Takashima, Fully secure functional encryption with general relations from the decisional linear assumption, in *CRYPTO 2010* (2010), pp. 191–208. Full version is available as an online first article in J. Cryptol
C. Peikert, He gives c-sieves on the CSIDH. IACR Cryptol. ePrint Arch. **2019**, 725 (2019)
P.W. Shor, Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM J. Comput. **26**(5), 1484–1509 (1997)
A. Sutherland, Identifying supersingular elliptic curves. LMS J. Comput. Math. **15**, 317–325 (2012)
K. Takashima, Expressive attribute-based encryption with constant-size ciphertexts from the decisional linear assumption, in *SCN 2014* (2014), pp. 298–317
J. Vélu, Isogénies entre courbes elliptiques. Comptes Rendus Acad. Sci. Paris, Sér. A. **273**, 238–241 (1971)
L. Washington, *Elliptic Curves: Number Theory and Cryptography*, 2nd edn. (CRC Press, Boca Raton, 2008)