Anonymous Post-Quantum Cryptocash* (Full Version)

Huang Zhang[1,2], Fangguo Zhang[1,2**], Haibo Tian[1,2], and Man Ho Au[3]

[1] School of Data and Computer Science, Sun Yat-sen University, Guangzhou 510006, China
[2] Guangdong Key Laboratory of Information Security, Guangzhou 510006, China
[3] Department of Computing, The Hong Kong Polytechnic University, Hong Kong, China

Abstract. In this paper, we construct an anonymous and decentralized cryptocash system which is potentially secure against quantum computers. In order to achieve that, a linkable ring signature based on ideal lattices is proposed. The size of a signature in our scheme is $O(\log N)$, where N is the cardinality of the ring. The framework of our cryptocash system follows that of CryptoNote with some modifications. By adopting the short quantum-resistant linkable ring signature scheme, our system is anonymous and efficient. We also introduce how to generate the verifying and signing key pairs of the linkable ring signature temporarily. With these techniques, the privacy of users is protected, even though their transactions are recorded in the public ledger.

1 Introduction

Electronic currencies or cryptocash systems have been proposed for many years. But none of them is prevalent before the Bitcoin system appears. Bitcoin was first described by Satoshi Nakamoto in 2008 [25]. Its success is partially due to its properties of decentralization and anonymity. To prevent "double spending", the system maintains the history of transactions among most nodes in a peer-to-peer network. A consensus mechanism called proof-of-work is used to maintain the history.

Later, researchers find that the public history of Bitcoin causes weaknesses which violate its original designing goals. The latest result states that Bitcoin only addresses the anonymity and unlinkability issues partially [3]. For example, multiple public keys of the same user can potentially be linked when a user sends change back to his wallet. In this case, two or more of a single user's public keys will appear in the same transaction [28]. Recently, there are more discussions about the weak anonymity of Bitcoin [27, 30]. Although this weakness can be overcome by adopting mixing and distributed methods, the solutions have to include a trusted third party which is a violation to the decentralization property.

* The paper is the full version of the paper accepted by FC 2018
** ✉ Fangguo Zhang, isszhfg@mail.sysu.edu.cn

There are some creative works to design a strong anonymous cryptocash system. Miers *et al.* [23] proposed "Zerocoin" that allows users to spend their coins using anonymous proof of ownership instead of explicit public-key based digital signatures. Saberhagen presented two properties, namely, "untraceability" and "unlinkability", that must be possessed in a fully anonymous cryptocash model. Then, they designed CryptoNote system with these properties [31]. Monero is a system based on CryptoNote. In CryptoNote, to provide anonymity, there are two ways for all transactions on the network: 1) hiding the sender's address using ring signatures, 2) hiding the receiver's identity using stealth addresses. Both sending and receiving addresses are verifying keys of a ring signature scheme. A ring signature can also be used in the Zerocoin system [12].

The notion of ring signatures, introduced by Rivest *et al.* [29], permits a user to sign a message on behalf of a group. A verifier is convinced that the real signer is a member of the group, but cannot explicitly identify the real signer. Considering the anonymity of a cryptocash system, a ring signature is obviously more suitable than a standard signature. But there is a cost: the size of the signature and the computational complexity are inherently larger than those of a standard signature. A traditional ring signature scheme usually features a signature size of $O(N)$, where the ring has N participants. To construct a ring signature of $O(\log N)$ or $O(1)$ size was an open problem in this field. Recently, Groth and Kohlweiss proposed a commitment-based scheme with logarithmic signature size [12].

However, a cryptocash system which replaces a standard signature with a ring signature naively suffers from the double spending attack. To fix this problem, it is necessary for the public to determine ring signatures generated by the same key pair. The traceable ring signature [9] is a candidate that enables users to trace the verifying and signing key pair which have been used for signing different messages. But the traceability is redundant for an efficiency-sensitive cryptocash system. CryptoNote and Monero chose to modify the traceable ring signature into a "one-time signature" to reduce the computational cost. Generally speaking, a linkable ring signature [17], which is a variant of the linkable spontaneous anonymous group signature [16], is sufficient enough for cryptocash systems to determine double spending. Even though signatures of these schemes are of size $O(N)$, CryptoNote and Monero do provide better privacy than Bitcoin.

Most cryptocash systems are based on traditional cryptographic schemes. The security of these schemes is based on hard computational problems, such as the factorization and discrete logarithm problem (DLP). However, researchers have proved that a quantum computer is able to solve these problems efficiently so that schemes based on them are not secure under the quantum computing model. One solution is to build schemes on computational problems that remain even hard for quantum computers. Lattice problems have been widely believed as suitable choices to build quantum resistant cryptographic schemes since Ajtai proposed his seminal work [2]. Some post-quantum signature schemes have been proposed recently [18, 7, 10]. Relying on these schemes, it is easy to obtain a post-quantum cryptocash system by replacing the ECDSA signature scheme in Bitcoin. However, the resulting cryptocash system is simply like Bitcoin in which the transactions are still linkable. Even though there are several lattice-based ring signatures [6, 36, 37] including the one with logarithmic size [15], none of them has the linkable or traceable property which is vital to prevent double spending.

In this paper, we aim at designing an anonymous post-quantum cryptocash (APQC) system. In order to achieve this goal, we propose a linkable ring signature based on ideal lattices. The size of a signature in this scheme is $O(\log N)$, where N is the cardinality of the ring. The framework of our cryptocash system follows that of CryptoNote [31], and the ideal-lattice-based signature scheme is inspired by the work of Groth and Kohlweiss [12] with some modifications.

The paper is organized as follows: in Sect. 2, we introduce notations and concepts applied in our work. The model of the ring signature based cryptocash is described in Sect. 3. Section 4 involves the concrete construction of the ideal-lattice-based linkable ring signature. We design the standard transaction of our cryptocash system in Sect. 5. Section 6 is a brief conclusion for this paper.

2 Preliminaries

2.1 Notations

We use $\mathbb{Z}$, $\mathbb{R}$ to denote the set of all integers and the set of all reals, respectively. For any $x \in \mathbb{R}$, $\lceil x \rceil$ denotes the smallest integer that is not smaller than x. A set $\{x_1, \dots, x_n\}$ is denoted by $\{x_i\}_{i=1}^n$. We use $|S|$ to indicate the cardinality of a set S. Vectors are named by lower-case bold letters (e.g., $\mathbf{x}$) and matrices by upper-case bold letters (e.g., $\mathbf{X}$). For a vector $\mathbf{x}$, $\|\mathbf{x}\|_p$ represents its ℓ_p norm, and p is omitted if $p = \infty$. The norm of a polynomial is defined similarly by regarding it as a vector. The ith entry of a vector $\mathbf{x}$ is denoted by x_i. If $\mathbf{x}$ is a vector of polynomials, then $\|\mathbf{x}\| = \max_i \|x_i\|$. A matrix $\mathbf{X}$ is identified with the ordered set $\{\mathbf{x}_i\}_i$ of its column vectors, and its ℓ_p norm is defined as $\|\mathbf{X}\|_p = \max_i \|\mathbf{x}_i\|_p$. If $a \in R$ and $\mathbf{X}$ is a matrix with entries in ring R, $a\mathbf{X}$ denotes the scalar multiplication. $\mathbf{I}$ is the identity matrix whose dimension is known from the context. For an integer i, i_j symbolizes the jth bit of i. $\delta_{i\ell}$ is Kronecker's delta, *i.e.*, $\delta_{\ell\ell} = 1$ and $\delta_{i\ell} = 0$ for $i \neq \ell$. For two strings x_1 and x_2, $x_1 \| x_2$ denotes the concatenation of them. If D is a distribution, $X \leftarrow \mathrm{diag}(D^{m\times m})$ is used to denote a random diagonal matrix sampled from $D^{m\times m}$.

2.2 Lattices and Hard problems

A lattice $\Lambda = \mathcal{L}(\mathbf{B})$ with dimension m and rank n is a subgroup of the linear space $\mathbb{R}^m$. Every element in Λ can be represented as an integral combination of its basis $\mathbf{B} \in \mathbb{R}^{m\times n}$. In our work, we will focus on a specific class of lattices, called ideal lattices, which can be described as ideals of certain polynomial rings.

Definition 1 (Definition 2 in [19]). *An ideal lattice is an integer lattice $\mathcal{L}(\mathbf{B}) \subseteq \mathbb{Z}^n$ such that $\mathcal{L}(\mathbf{B}) = \{g \bmod f : g \in \mathcal{I}\}$ for some monic polynomial f of degree n and ideal $\mathcal{I} \in \mathbb{Z}[x]/\langle f \rangle$.*

The quotient ring $\mathbb{Z}[x]/\langle f\rangle$ is additively isomorphic to the integer lattice $\mathbb{Z}^n$.

To extend the hash function family in previous works [2, 5, 22], Micciancio defined the generalized knapsack function family [20, 21].

Definition 2 (Definition 4.1 in [21]). *For any ring R, subset $D \subset R$ and integer $m \geq 1$, the generalized knapsack function family $\mathcal{K}(R, D, m) = \{f_{\mathbf{a}} : D^m \to R\}_{\mathbf{a}\in R^m}$ is defined by*

$$f_{\mathbf{a}}(\mathbf{x}) = \sum_{i=1}^{m} x_i \cdot a_i,$$

for all $\mathbf{a} \in R^m$ and $\mathbf{x} \in D^m$, where $\sum_i x_i \cdot a_i$ is computed using the ring addition and multiplication operations. the ring addition and multiplication operations.

Besides one-wayness, Micciancio showed that for a special case of the above function family, the distribution of $f_{\mathbf{a}}(\mathbf{x})$ is uniform and independent from $\mathbf{a}$.

Theorem 1 (Theorem 4.2 in [21]). *For any finite field $\mathbb{F}$, subset $S \subset \mathbb{F}$, and integers n, m, the hash function family $\mathcal{K}(\mathbb{F}^n, S^n, m)$ is ϵ-regular for*

$$\epsilon = \frac{1}{2}\sqrt{(1 + |\mathbb{F}|/|S|^m)^n - 1}.$$

In particular, for any $q = n^{O(1)}$, $|S| = n^{\Omega(1)}$ and $m = \omega(1)$, the function ensemble $\mathcal{K}(\mathbb{F}_q^n, S^n, m)$ is almost regular (i.e., ϵ is negligible).

Here, "ϵ-regular" means that the statistical distance between uniform distribution $U((\mathbb{F}^n)^m, \mathbb{F}^n)$ and $\{(\mathbf{a}, f_{\mathbf{a}}(\mathbf{x})) : \mathbf{a} \leftarrow U((\mathbb{F}^n)^m), \mathbf{x} \leftarrow U((S^n)^m)\}$ is at most ϵ. $\mathbb{F}^n$ is a ring of n-dimensional vectors with the usual vector addition operation and convolution product.

Sometimes, one-wayness is not sufficient enough for the design of a cryptographic protocol. Lyubashevsky and Micciancio proved that finding a collision in some instance of the generalized knapsack function family is as hard as solving the worst-case problem in a certain lattice [19].

Definition 3 (Collision Problem). *For any generalized knapsack function family $\mathcal{K}(R, D, m)$, define the collision problem $Col_{\mathcal{K}}(h_{\mathbf{a}})$ as follows: given a function $h_{\mathbf{a}} \in \mathcal{K}$, find $\mathbf{b}, \mathbf{c} \in D^m$ such that $\mathbf{b} \neq \mathbf{c}$ and $h_{\mathbf{a}}(\mathbf{b}) = h_{\mathbf{a}}(\mathbf{c})$.*

If there is no polynomial time algorithm that can solve $\mathrm{Col}_{\mathcal{K}}$ with non-negligible probability when given a function $h_{\mathbf{a}}$ which is distributed uniformly at random in $\mathcal{K}$, then $\mathcal{K}$ is collision resistant family of hash functions.

The expansion factor is a parameter proposed to quantify the quality of modulus f in the ideal lattice [19]. The expansion factor of f is defined as

$$\mathrm{EF}(f, k) = \max_{g\in\mathbb{Z}[x], \deg(g)\leq k(\deg(f)-1)} \|g\|_f / \|g\|_\infty$$

where $\|g\|_f$ is short for $\|g \bmod f\|_\infty$. Moreover, $\mathrm{EF}(x^n + 1, k) \leq k$.

The generalized knapsack function family $\mathcal{K}(R, D, m)$ considered in [19] is instantiated as follows. Let $R = \mathbb{Z}_q[x]/\langle f\rangle$ be a ring for some integer q, where $f \in \mathbb{Z}[x]$ is a monic, irreducible polynomial of degree n with expansion factor $\mathrm{EF}(f, 3) \leq \varepsilon$. Let $D = \{g \in R : \|g\| \leq \beta\}$ for some positive integer β.

Theorem 2 (Theorem 2 in [19]). *Let $\mathcal{K}(R, D, m)$ be a generalized knapsack function family as above with $m \geq \frac{\log q}{\log 2\beta}$ and $q > 2\varepsilon\beta mn^{1.5}\log n$. Then, for $\gamma = 8\varepsilon^2\beta mn\log^2 n$, there is a polynomial time reduction from f-SPP$_\gamma(\mathcal{I})$ for any ideal $\mathcal{I} \in R$ to Col$_\mathcal{K}(h)$ where h is chosen uniformly at random from $\mathcal{K}$.*

If we denote by $\mathcal{I}(f)$ the set of lattices that are isomorphic (as additive groups) to ideals of $\mathbb{Z}[x]/\langle f\rangle$ where f is monic, then there is a straightforward reduction from $\mathcal{I}(f)$-SVP$_\gamma$ to f-SPP$_\gamma$, and the vise versa. It is conjectured that approximating $\mathcal{I}(f)$-SVP$_\gamma$ to within a polynomial factor is a hard problem, although it is not NP-hard [1, 11].

2.3 The Public-key Encryption on Ideal Lattices

The cryptosystem we described here was proposed by Stehlé *et al.* [34]. The ideal-lattice-based encryption scheme is formalized as a collection of efficient procedures $\mathcal{ES}$=(**Setup**, **KGen**, **Enc**, **Dec**).

Setup(1^n): n is the security parameter. Fix $f(X) = X^n + 1 \in \mathbb{Z}[X]$ and $q = \text{poly}(n)$ a prime satisfying $q \equiv 3 \bmod 8$. Set $\sigma = 1$, $r = 1 + \log_3 q$, and $m = (\lceil\log q\rceil + 1)\sigma + r$. Let $R = \mathbb{Z}_q[X]/\langle f\rangle$. All the parameters generated in this procedure are published as the global parameter *pp*.

KGen(pp): On input global parameter *pp*, it runs the trapdoor generation algorithm **Id-Trap** to get a trapdoor function $h_\mathbf{g} : \mathbb{Z}_q^n \times \mathbb{Z}_q^{mn} \to \mathbb{Z}_q^{mn}$ and a trapdoor S, where $\mathbf{g}$ is the function index. The first component of the domain of $h_\mathbf{g}$ can be viewed as a subset of $\mathbb{Z}_2^{\ell_I}$ for $\ell_I = O(n\log q)$. Generate $\mathbf{r} \in \mathbb{Z}_2^{\ell_I+\ell_\mu}$ uniformly and define the Toeplitz matrix $M_{\mathrm{GL}} \in \mathbb{Z}_2^{\ell_\mu\times\ell_i}$ whose ith row is $(r_i, \ldots, r_{\ell_I+i-1})$. It outputs the public key $epk = (\mathbf{g}, \mathbf{r})$ and the secret key $esk = S$.

Enc(pp, epk, μ): Given ℓ_μ bit message μ with $\ell_\mu = n/\log n$ and public key $epk = (\mathbf{g}, \mathbf{r})$, sample $(\mathbf{s}, \mathbf{e})$ with $\mathbf{s} \in \mathbb{Z}_q^n$ uniform and $\mathbf{e}$ sampled from $\bar{\Psi}_{\alpha q}$, where $\bar{\Psi}_{\alpha q}$ is the reduction modulo q of the standard Gaussian distribution with parameter αq. It then evaluates $C_1 = h_\mathbf{g}(\mathbf{s}, \mathbf{e})$ and computes $C_2 = \mu \oplus (M_{\mathrm{GL}} \cdot \mathbf{s})$, where the product $M_{\mathrm{GL}} \cdot \mathbf{s}$ is computed over $\mathbb{Z}_2$, and $\mathbf{s}$ is viewed as a string over $\mathbb{Z}_2^{\ell_I}$. Return the ciphertext $C = (C_1, C_2)$.

Dec(pp, esk, C): Given cyphertext $C = (C_1, C_2)$ and secret key $esk = (S, \mathbf{r})$, invert C_1 to compute $(\mathbf{s}, \mathbf{e})$ such that $h_\mathbf{g}(\mathbf{s}, \mathbf{e}) = C_1$, and return message $\mu = C_2 \oplus (M_{\mathrm{GL}} \cdot \mathbf{s})$.

To see the details of the trapdoor generation algorithm **Id-Trap** and the one-way trapdoor function family $\{h_\mathbf{g} : \mathbb{Z}_q^n \times \mathbb{Z}_q^{mn} \to \mathbb{Z}_q^{mn}\}_{g\in(\mathbb{Z}_q[x]/\langle f\rangle)^m}$, we refer to the literature [34] in which Stehlé *et al.* also proved that the above encryption scheme is IND-CPA secure if the Ideal-LWE$^f_{m,q;\Psi_{\alpha q}}$ problem is hard.

The notion of key privacy is formally defined by Bellare *et al.* [4]. It requires that the receiver of a ciphertext is anonymous from the point of view of the adversary. Fortunately, we can deduce from the observation 1 of [13] that the aforementioned encryption scheme $\mathcal{ES}$ is of key privacy.

3 Anonymous Cryptographic Currency Model Based on Linkable Ring Signatures

Cryptocash system based on linkable ring signatures emerged after researchers found that Bitcoin was not fully anonymous and untraceable. CryptoNote and Monero are two typical instances. We describe here the properties of an anonymous cryptocash system and state the techniques [31] to construct such a system.

In a cryptocash system, there are three parties: a sender, who owns a coin and decides to spend it, a receiver, who is the destination that a coin is delivered to, and a public ledger where all transactions are recorded. An anonymous cryptocash system should satisfy the following properties:

- **Untraceability**: If Tx is a transaction from sender A to receiver B, and Tx has been recorded in the public ledger, no one else can determine the sender with probability significantly larger than $1/N$ by accessing the transcript of Tx, where N is the number of possible senders in a related input of the Tx. Moreover, even receiver B cannot prove that A is the true sender of Tx.
- **Unlinkability**: If Tx_1 is a transaction from sender A to receiver C, Tx_2 is another transaction from sender B to receiver C, and Tx_1, Tx_2 have been recorded in the public ledger, then for any subsequent transactions in the public ledger, no one else can use them to link the outputs of the two transactions to a single user, even for senders A and B.
- **Detecting Double Spending**: If Tx_1 is a transaction which describes that coin c has been sent from sender A to receiver B, and Tx_1 has been recorded in the public ledger, every user of the system could detect another transaction Tx_2 that describes the same coin c. Furthermore, Tx_2 will never be accepted and recorded in the public ledger.

To design a cryptocash protocol which provides all the above properties, the CryptoNote and Monero suggested to adopt the modification of the traceable ring signature [9], which generates a one-time signature on behalf of a temporal group. Since it is a one-time signature with an explicit identification tag about the signing key, it could prevent a coin being double-spent. Besides, since it is a ring signature where the identity of the real signer is hidden within a set of possible signers, it guarantees untraceability. In addition, ring signature supports unlinkability since the inputs in a transaction may be brought from outputs of transactions belonging to other users.

To employ a linkable ring signature in a cryptocash system, the receiver should produce a one-time key pair for each transaction. A sender could obtain the public key of the receiver for the transaction and build a transaction with an output script containing that key's information. The drawback of this trivial method is that a receiver has to maintain a lot of one-time keys. Furthermore, a sender has to contact each receiver for their fresh one-time public key when the sender builds a transaction. Alternatively, CryptoNote suggests another method which enables a receiver to store only a single key pair. A sender could produce a random value to generate a one-time public key for the receiver based on this single public key. The one-time public key is referred to as the destination address. This is a convenient design at the cost of a slightly weakened unlinkability. Specifically, if a user has a single key, a sender could always identify a receiver from the sender's transaction by its random value of the transaction. If two senders collude, and they have sent coins to the same receiver, they could identify the same receiver while the trivial method avoids this. And if a later transaction includes the two senders' outputs at the same time, with a higher probability, the later transaction is made by the receiver. Note that a receiver could still produce another key pair at will as in the Bitcoin system to avoid the small problem.

Finally, let us observe a standard transaction in a linkable ring signature based cryptocash system. In such a system, the value of a coin is bound with a destination address. Suppose A and B are two users in the system. B has a single key pair (pk_B, sk_B). A has the private key sk_1 of a destination address vk_1, which represents a coin, say c, which has been sent to A previously. If A decides to send c to B, he generates a destination address vk_2 and an auxiliary input aux for B; he then chooses a number of transactions from the public ledger such that the delivered value of coin is equivalent to c; he extracts the destination addresses of those transactions and assembles them with vk_1 to form a ring L; he runs a ring signature algorithm to sign transaction Tx, which involves information about (c, aux, vk_2, L), with signing key sk_1 and broadcasts the transaction; If the signature generated by sk_1 is not linkable to any transaction on the ledger, the public ledger will accept this transaction and record it; B uses its private key sk_B to check every passing transaction to determine if transaction Tx is for B and recovers the signing key sk_2 corresponding to vk_2. With sk_2, user B can spend c by signing another transaction. However, even A does not know when and where B spends it due to the functionality of the linkable ring signature.

It is obvious that linkable ring signature is vital for an anonymous cryptocash system. We next detail the lattice-based version of a linkable ring signature.

4 Linkable Ring Signature Based on Ideal-Lattices

The strong similarity in the construction between a lattice-based signature and DLP-based one (see Lyubashevsky's signature [18] and the Schnorr's signature [32, 33]) implies that the latter can help us to design the former, *e.g.*, using the work in [17] or [18], we can easily obtain a linkable ring signature based on lattices with signature size of $O(N)$, where N is the number of participants of the ring. However, such a construction is not efficient enough for a practical cryptocash system. In this section, we will present an ideal-lattice-based linkable ring signature of size $O(\log N)$ using the idea in [12]. We start this section with a brief recall on their work.

4.1 A Brief Recall

In [12], Groth and Kohlweiss proposed an efficient Sigma-protocol, which can be used as an ad-hoc group identification scheme. Their ring signature scheme is a direct transformation of the identification scheme with the Fiat-Shamir heuristic [8]. As the transmission of the identification scheme involves only $O(\log N)$ commitments, the resulting ring signature scheme is of size $O(\log N)$.

Their work starts from homomorphic commitments scheme such as the Pedersen commitment scheme (*i.e.*, $\mathrm{com}(m; r) = h^m g^r$). The first step is to design a Sigma-protocol Σ_1 to prove in zero-knowledge that such a commitment is opened to 0 or 1. Once the subroutine Σ_1 is established, to design a ad-hoc group identification scheme is to construct a Sigma-protocol Σ_2 to show in zero-knowledge that one of N commitments is opened to 0. Here, a commitment to 0 is the public key of a user and the randomness is the corresponding secret key. If the ℓth user of the ad-hoc group $\{\mathrm{user}_0, \cdots, \mathrm{user}_{N-1}\}$ wants to identify himself secretly, Σ_2 first commits the integer ℓ bit by bit and runs Σ_1 to prove in zero-knowledge that those $\log N$ commitments are opened to 0 or 1. Then Σ_2 proves in zero-knowledge that the ℓth user can open the ℓth public key (*i.e.*, a commitment to 0) to 0, with the help of the intermediate parameters used in the foregoing Σ_1's. By replacing the challenge message with the hash value of all initial messages in Σ_2, we obtain a non-interactive zero-knowledge proof system which can be regarded as a ring signature. For the details of the generic construction of Σ_1 and Σ_2, we refer readers to the literature [12].

It is worth mentioning that the underlying homomorphic commitment is the corner stone for both the construction and the security proof of the foregoing ring signature. As a counterpart of their work, our scheme also contains an ideal-lattice-based commitment scheme (*i.e.*, $\mathrm{com}(\mathbf{S}; \mathbf{X}) = \mathbf{HS} + \mathbf{GX}$). The details of the commitment scheme is left to Sect. 4.3.

4.2 Our Construction

To construct an $O(\log N)$ ring signature, Groth and Kohlweiss proposed a technique to compute the coefficients of a polynomial in the indeterminate x over finite field $\mathbb{Z}_q$ in advance, where x is a hash value computed later [12]. We extend their method to handle polynomials with coefficients belonging to a ring of square matrices. The major difference is that the multiplication of matrices is not commutative. This is the reason why we restrict x in our scheme to be a 1×1 matrix. Since the scalar multiplication is commutative, we have the following result.

Let integer ℓ be in the interval $[0, N-1]$, and $M = \lceil \log N \rceil$. Given matrices $\mathbf{B}_j$, set $\mathbf{W}_j = \ell_j x\mathbf{I} + \mathbf{B}_j$. Let $\mathbf{W}_{j,1} = \mathbf{W}_j = \ell_j x\mathbf{I} + \mathbf{B}_j = \delta_{1\ell_j} x\mathbf{I} + \mathbf{B}_j$ and $\mathbf{W}_{j,0} = x\mathbf{I} - \mathbf{W}_j = (1-\ell_j)x\mathbf{I} - \mathbf{B}_j = \delta_{0\ell_j} x\mathbf{I} - \mathbf{B}_j$. Then for each $i \in [0, N-1]$, the product $\prod_{j=1}^{M} \mathbf{W}_{j,i_j}$ is a polynomial in x of the form

$$P_i(x) = \prod_{j=1}^{M} (\delta_{i_j \ell_j} x\mathbf{I}) + \sum_{k=0}^{M-1} \mathbf{P}_{i,k} x^k = \delta_{i\ell} x^M \mathbf{I} + \sum_{k=0}^{M-1} \mathbf{P}_{i,k} x^k, \tag{1}$$

where $\mathbf{P}_{i,k}$ is the coefficient of the kth degree term, and can be efficiently computed if $\{\mathbf{B}_j\}_{j=1}^{M}$, i and ℓ are given.

The linkable ring signature scheme consists of a tuple of efficient procedures $\mathcal{LRS} = (\mathbf{Setup}, \mathbf{KGen}, \mathbf{Sign}, \mathbf{Vfy}, \mathbf{Link})$. Let N be the maximum size of the ring, $M = \lceil \log N \rceil$, and n be a power of 2. The details of those procedures are shown as follows:

Setup(n, N): On input N and security parameter n, the procedure initiates a hash function introduced in [18] as a random oracle $\mathcal{H} : \{0,1\}^* \rightarrow \{v : v \in \{-1,0,1\}^n, \|v\|_1 \leq p\}$, such that $2^p \cdot \binom{n}{p} \geq 2^{100}$. It sets $\varepsilon = 3$, $t = n^{\Omega(1)}$, $m = \omega(1)$, $\beta \geq \max\{\frac{t(p^{M+1}-1)}{(p-1)}, 8t^2pmn + 8t^3mn^2\}$. Pick a prime q such that $(2\beta)^m > q > 2\varepsilon\beta mn^{1.5}\log n$. All operations in this system are done in $R = \mathbb{Z}_q[X]/\langle f \rangle$, for $f = X^n + 1$. Let $Q = \{g \in R : \|g\| \leq t\}$ and $\tilde{Q} = \{g \in R : \|g\| \leq t-1\}$. Relying on those parameters, this procedure samples matrices $\mathbf{G}, \mathbf{H} \in R^{1\times m}$ uniformly at random. Finally it outputs $pp = (n, m, \mathbf{G}, \mathbf{H}, \mathcal{H}, q, t, N)$ as the global parameters.

KGen(pp): For the ith user, this procedure randomly chooses $\mathbf{X}_i \leftarrow Q^{m\times m}$ and computes $\mathbf{Y}_i = \mathbf{G}\mathbf{X}_i$. The ith user's verifying key is $vk_i = \mathbf{Y}_i$ and the singing key is $sk_i = \mathbf{X}_i$.

Sign(pp, sk_ℓ, μ, L): Without loss of generality, let $L = (\mathbf{Y}_0, \mathbf{Y}_1, \ldots, \mathbf{Y}_{N-1})$ be the ensemble of a ring with the largest size. On input a message μ, the ℓth user's signature on behalf of L is generated as follows

- Compute $\mathbf{R}_\ell = \mathbf{H}\mathbf{X}_\ell$.
- For j from 1 to M
  - sample $\mathbf{K}_j, \mathbf{C}_j, \mathbf{D}_j \leftarrow Q^{m\times m}$,
  - if $\ell_j = 0$, randomly pick $\mathbf{B}_j \leftarrow \text{diag}(Q^{m\times m})$,
    else if $\ell_j = 1$, randomly draw $\mathbf{B}_j \leftarrow \text{diag}(\tilde{Q}^{m\times m})$,
  - compute $\mathbf{V}_{\ell_j} = \mathbf{H}(\ell_j\mathbf{I}) + \mathbf{G}\mathbf{K}_j$,
  - compute $\mathbf{V}_{a_j} = \mathbf{H}\mathbf{B}_j + \mathbf{G}\mathbf{C}_j$,
  - compute $\mathbf{V}_{b_j} = \mathbf{H}(\ell_j\mathbf{B}_j) + \mathbf{G}\mathbf{D}_j$.
- For k from 0 to $M-1$
  - sample $\mathbf{E}_k \leftarrow Q^{m\times m}$,
  - compute $\mathbf{V}_{d_k} = (\sum_{i=0}^{N-1} \mathbf{Y}_i\mathbf{P}_{i,k}) + \mathbf{G}\mathbf{E}_k$, where $\mathbf{P}_{i,k}$ is introduced in (1),
  - compute $\mathbf{V}'_{d_k} = \mathbf{H}\mathbf{E}_k$.
- Let $S_1 = \{\mathbf{V}_{\ell_j}, \mathbf{V}_{a_j}, \mathbf{V}_{b_j}, \mathbf{V}_{d_{j-1}}, \mathbf{V}'_{d_{j-1}}\}_{j=1}^M$ and then compute hash value $x = \mathcal{H}(pp, \mu, L, S_1, \mathbf{R}_\ell)$.
- For j from 1 to M, compute
  - $\mathbf{W}_j = \ell_j x\mathbf{I} + \mathbf{B}_j$,
  - $\mathbf{Z}_{a_j} = \mathbf{K}_j(x\mathbf{I}) + \mathbf{C}_j$,
  - $\mathbf{Z}_{b_j} = \mathbf{K}_j(x\mathbf{I} - \mathbf{W}_j) + \mathbf{D}_j$.
- compute $\mathbf{Z}_d = \mathbf{X}_\ell(x^M\mathbf{I}) - \sum_{k=0}^{M-1} \mathbf{E}_k x^k$.
- Let $S_2 = \{\mathbf{W}_j, \mathbf{Z}_{a_j}, \mathbf{Z}_{b_j}\}_{j=1}^M$. Publish the signature $\sigma = \{S_1, S_2, \mathbf{Z}_d, \mathbf{R}_\ell, L\}$.

Vfy(pp, μ, σ): On input signature σ and message μ, this procedure does as follows to test the validity of σ.

1. Compute hash value $x = \mathcal{H}(pp, \mu, L, S_1, \mathbf{R}_\ell)$.
2. For j from 1 to M, consider the following conditions

- $\mathbf{W}_j$ is a diagonal matrix,
- $\|\mathbf{W}_j\| \leq t$,
- $\|\mathbf{Z}_{a_j}\| \leq (p+1)t$,
- $\|\mathbf{Z}_{b_j}\| \leq tp + t^2 n + t$,

If any of them does not hold, output 0 and abort.

3. If $\|\mathbf{Z}_d\| \leq \frac{t(p^{M+1}-1)}{p-1}$ is not satisfied, output 0 and abort.
4. For j from 1 to M, consider the following equations
   - $\mathbf{V}_{\ell_j}(x\mathbf{I}) + \mathbf{V}_{a_j} = \mathbf{H}\mathbf{W}_j + \mathbf{G}\mathbf{Z}_{a_j}$,
   - $\mathbf{V}_{\ell_j}(x\mathbf{I} - \mathbf{W}_j) + \mathbf{V}_{b_j} = \mathbf{G}\mathbf{Z}_{b_j}$.

   If any of the aforementioned equations does not hold, output 0 and abort.
5. If the equation $\mathbf{R}_\ell(x^M\mathbf{I}) + \sum_{k=0}^{M-1} \mathbf{V}'_{d_k}(-x^k) = \mathbf{H}\mathbf{Z}_d$ does not hold, output 0 and abort.
6. Inspect whether

$$\sum_{i=0}^{N-1}(\mathbf{Y}_i \prod_{j=1}^{M} \mathbf{W}_{j,i_j}) + \sum_{k=0}^{M-1} \mathbf{V}_{d_k}(-x^k) = \mathbf{G}\mathbf{Z}_d$$

   satisfies. If not, output 0; otherwise output 1 (accept).

Link(pp, σ_1, σ_2): For two signatures $\sigma_1 = (\ldots, \mathbf{R}_1, L_1)$ and $\sigma_2 = (\ldots, \mathbf{R}_2, L_2)$, if $\mathbf{R}_1 = \mathbf{R}_2$, return 1 (linked) for concluding that they are generated by the same signer; otherwise, return 0 (unlinked).

Correctness: To see that the signature generated by the **Sign** procedure always passes the **Vfy** procedure, we first observe the four equations in the **Vfy** procedure. The equations in step 4 are to prove in zero-knowledge that the signer is the ℓth user (some $\ell \in [0, N-1]$). The correctness of those equations is shown directly through a simple deduction. The equation in step 5 is to prove that the parameter for linking is correct. For a valid signature, it holds since

$$\begin{aligned}
&\mathbf{R}_\ell(x^M\mathbf{I}) + \textstyle\sum_{k=0}^{M-1} \mathbf{V}'_{d_k}(-x^k) \\
&= \mathbf{H}\mathbf{X}_\ell(x^M\mathbf{I}) + \textstyle\sum_{k=0}^{M-1} \mathbf{H}\mathbf{E}_k(-x^k) \\
&= \mathbf{H}(\mathbf{X}_\ell(x^M\mathbf{I}) - \textstyle\sum_{k=0}^{M-1} \mathbf{E}_k x^k) \\
&= \mathbf{H}\mathbf{Z}_d
\end{aligned}.$$

The equation in step 5 is to prove in zero-knowledge that the anonymous signer holds the secret key of the ℓth user. To see the correctness of the last equation, observe that $P_i(x) = \prod_{j=1}^{M} \mathbf{W}_{j,i_j}$ introduced in (1) is a polynomial in x of degree

M, only if $i = \ell$. With this fact in mind, we have

$$
\begin{aligned}
&\textstyle\sum_{i=0}^{N-1}(\mathbf{Y}_i \prod_{j=1}^{M} \mathbf{W}_{j,i_j}) + \sum_{k=0}^{M-1} \mathbf{V}_{d_k}(-x^k) \\
&= \textstyle\sum_{i=0}^{N-1} \mathbf{Y}_i(\delta_{i\ell} x^M \mathbf{I} + \sum_{k=0}^{M-1} \mathbf{P}_{i,k} x^k) + \sum_{k=0}^{M-1}((\sum_{i=0}^{N-1} \mathbf{Y}_i \mathbf{P}_{i,k}) + \mathbf{G}\mathbf{E}_k)(-x^k) \\
&= \textstyle\sum_{i=0}^{N-1} \sum_{k=0}^{M-1}(\mathbf{Y}_i \mathbf{P}_{i,k} x^k - \mathbf{Y}_i \mathbf{P}_{i,k} x^k) + \mathbf{Y}_\ell(\delta_{\ell\ell} x^M \mathbf{I}) + \sum_{k=0}^{M-1} \mathbf{G}\mathbf{E}_k(-x^k) \\
&= \textstyle\mathbf{G}(\mathbf{X}_\ell(x^M \mathbf{I}) - \sum_{k=0}^{M-1} \mathbf{E}_k x^k) \\
&= \mathbf{G}\mathbf{Z}_d
\end{aligned} .
$$

It remains to show that $\{\mathbf{W}_j, \mathbf{Z}_{a_j}, \mathbf{Z}_{b_j}\}_{j=1}^{M}$, and $\mathbf{Z}_d$ are short enough to pass step 2 of the **Vfy** procedure.

Note that for polynomials $a, b \in R$, the norm of their product is bounded by $\|a\| \cdot \|b\| \cdot n$. For $a \in R$ and $b \in \{v : v \in \{-1, 0, 1\}^n, \|v\|_1 \leq p\}$ the norm of $a \cdot b$ is not larger than $\|a\| \cdot \|b\| \cdot p$. Depending on the above two facts and the triangle inequality, the correctness of the inequalities in step 2 can be validated easily. For example, $\|\mathbf{Z}_{b_j}\| = \|\mathbf{K}_j(x\mathbf{I} - \mathbf{W}_j) + \mathbf{D}_j\| \leq \|\mathbf{K}_j x \mathbf{I}\| + \|\mathbf{K}_j(-\mathbf{W}_j)\| + \|\mathbf{D}_j\| \leq tp + t^2 n + t$ and $\|\mathbf{Z}_d\| \leq \|\mathbf{X}_\ell(x^M \mathbf{I})\| + \|\sum_{k=0}^{M-1} \mathbf{E}_k x^p\| \leq tp^M + \|\mathbf{E}_0 x^0\| + \|\mathbf{E}_1 x^1\| + \cdots + \|\mathbf{E}_{M-1} x^{M-1}\| \leq tp^M + t + tp + \cdots + tp^{M-1} = \frac{t(p^{M+1}-1)}{p-1}$.

Even though the foregoing linkable ring signature is designed over ideal lattices, a classic edition of this signature can be built by instead using any cyclic group as long as its underlying DLP is hard. We propose a linkable ring signature based on the ECDLP, and discuss how to implement this signature with ECC in appendix A.

4.3 Security Proof

Groth and Kohlweiss have proved that the generic construction of their ring signature is secure in the random oracle model, if its underlying commitment scheme is perfectly hiding and computationally binding [12]. Since our $\mathcal{LRS}$ is designed over the framework of their generic construction, in order to prove the security of $\mathcal{LRS}$, it is sufficient enough to prove the binding and hiding properties of the commitment scheme applied in $\mathcal{LRS}$.

Theorem 3 (Theorem 4 in [12]). *The generic construction of the ring signature scheme in [12] is perfect anonymity if the underlying commitment scheme is perfectly hiding. It is unforgeable in the random oracle model if the commitment scheme is perfectly hiding and computationally binding.*

A non-interactive commitment scheme allows a sender to construct a commitment to a value. The sender may later open the commitment and reveal the value so that the receiver can verify the opening and check if it is the value that was committed at the beginning. A commitment scheme is said to be hiding, only if it reveals nothing about the committed value. The binding property ensures that a sender cannot open the commitment to two different values.

The non-interactive commitment scheme adopted in our $\mathcal{LRS}$ consists of a pair of efficient algorithms $\mathcal{CMT}$=(**Gen, Com**).

Gen(1^n)**:** As the commitment scheme is a subroutine of $\mathcal{LRS}$, the setup algorithm runs $\mathcal{LRS}.\mathbf{Setup}(1^n)$ to obtain $\mathcal{LRS}.pp$ and picks $pp = (n, m, \mathbf{G}, \mathbf{H}, q, t)$ out of $\mathcal{LRS}.pp$ to be the global parameters of the commitment scheme. Values and randomness are elements in $P_1^{m\times m}$ and $P_2^{m\times m}$, respectively, where $P_1 = \{g \in R : \|g\| \leq 4tp + 4t^2n\}$, $P_2 = \{g \in R : \|g\| \leq 4tp + 4tp^2 + 4t^2pn + 4t^2n\}$. Notice that Q is a subset of P_1 and P_2. All operations are done in R.

Com$(pp, \mathbf{S})$**:** On input a value $\mathbf{S} \in P_1^{m\times m}$, this algorithm samples $\mathbf{X} \leftarrow P_2^{m\times m}$ uniformly at random, and generates a commitment by computing $\mathbf{C} = \mathbf{HS} + \mathbf{GX}$. $\mathbf{C}$ can later be opened by unveiling the short $\mathbf{S}$ and $\mathbf{X}$.

The correctness of the foregoing commitment scheme $\mathcal{CMT}$ is obvious. It remains to prove that $\mathcal{CMT}$ is hiding and biding.

Theorem 4 (Binding and Hiding). *For any committed value $\mathbf{S} \in P_1^{m\times m}$ and uniformly chosen randomness $\mathbf{X} \leftarrow P_2^{m\times m}$, the commitment $\mathbf{C} = \mathbf{HS} + \mathbf{GX}$ reveals nothing about $\mathbf{S}$. Moreover, the sender cannot open $\mathbf{C}$ to $\mathbf{S}' \neq \mathbf{S}$, if the collision problem $Col_{\mathcal{K}}(h_{\mathbf{H}})$ with respect to the generalized knapsack function family $\mathcal{K}(R, D, m)$ is intractable to solve, where $D = \{g \in R : \|g\| \leq \beta\}$.*

Proof. Given a matrix $\mathbf{G} \in R^{1\times m}$ sampled uniformly at random, we obtain a uniformly random instance $f_{\mathbf{G}} : P_2^m \to R$ from the generalized knapsack function family $\mathcal{F}(R, P_1, m)$. Let $\mathbf{x}_i$ symbolize the ith column of the matrix $\mathbf{X} \in P_2^{m\times m}$, so that it is a vector sampled from $P_2^{m\times 1}$ uniformly at random. Note that R can be regarded as $\mathbb{Z}_q^n$ and $\mathbb{Z}_q$ is a finite field. Additionally, we have $q > 2\varepsilon\beta mn^{1.5}\log n$, $4tp + 4tp^2 + 4t^2pn + 4t^2n > t = n^{\Omega(1)}$, $m = \omega(1)$ in our setting. Therefore, according to Theorem 1, the distribution of $f_{\mathbf{G}}(\mathbf{x}_i) = \mathbf{G}\mathbf{x}_i$ is almost uniform over $\mathbb{Z}_q^n$ (namely R), and $f_{\mathbf{G}}(\mathbf{X}) = \mathbf{GX}$ is almost uniform over $R^{1\times m}$. The same result is also suitable for $f_{\mathbf{H}}(\mathbf{S}) = \mathbf{HS}$. Then, for any $\mathbf{S} \in P_1^{m\times m}$, $\mathbf{C} = \mathbf{HS} + \mathbf{GX}$ is almost uniformly distributed over $R^{1\times m}$ and reveals nothing about the $\mathbf{S}$.

We proceed to prove the binding property. From our parameter settings $(2\beta)^m \geq q$, we have $m > \frac{\log q}{\log 2\beta}$. Depending on Theorem 2, to solve the collision problem $\mathrm{Col}_{\mathcal{K}}(h_{\mathbf{H}})$ with respect to the generalized knapsack function family $\mathcal{K}(R, D, m)$ is as hard as to solve the $\mathcal{I}(f)$-SVP$_\gamma$ problem, where $\gamma = 8\varepsilon^2\beta mn\log^2 n$ is a polynomial in security parameter n. Suppose for the sake of contradiction that a PPT adversary $\mathcal{A}$ can break the binding property of $\mathcal{CMT}$. We will design an algorithm $\mathcal{B}$ to solve $\mathrm{Col}_{\mathcal{K}}(h_{\mathbf{H}})$ with respect to $\mathcal{K}(R, D, m)$.

After $\mathcal{B}$ receiving an instance $h_{\mathbf{G}}$ labeled by $\mathbf{G} \in R^{1\times m}$ from $\mathcal{K}(R, D, m)$, it selects $\mathbf{T} \leftarrow Q^{m\times m}$ uniformly at random, and computes $\mathbf{H} = \mathbf{GT}$. Relying on the similar discussion in the proof of hiding property and $t = n^{\Omega(1)}$, $\mathbf{H}$ is uniformly distributed in $R^{1\times m}$. Subsequently, $\mathcal{B}$ simulates a commitment scheme for $\mathcal{A}$ by publishing $pp = (n, m, \mathbf{G}, \mathbf{H}, q, t)$ as the global parameters. Note that the distributions of $\mathbf{G}$ and $\mathbf{H}$ are the same as that in the original scheme. Consequently, by the hypothesis, $\mathcal{A}$ could return $\mathbf{S}, \mathbf{S}' \in P_1^{m\times m}$, $\mathbf{X}, \mathbf{X}' \in P_2^{m\times m}$, such that $\mathbf{S} \neq \mathbf{S}$

and $\mathbf{HS} + \mathbf{GX} = \mathbf{HS}' + \mathbf{GX}'$ in a non-negligible probability. $\mathcal{B}$ considers the two possible cases.

Case 1: If $\mathbf{HS} = \mathbf{HS}'$, then $\mathbf{S}$ and $\mathbf{S}'$ are a pair of collisions with respect to $h_{\mathbf{H}}$, since $\|\mathbf{S}\|, \|\mathbf{S}'\| \leq 4tp + 4t^2n < \beta$.

Case 2: If $\mathbf{HS} \neq \mathbf{HS}'$, then $\mathbf{X} \neq \mathbf{X}'$ and we have $\mathbf{H}(\mathbf{S} - \mathbf{S}') = \mathbf{G}(\mathbf{X}' - \mathbf{X})$. By using $\mathbf{H} = \mathbf{GT}$, we can deduce that $\mathbf{GT}' = \mathbf{G}(\mathbf{X}' - \mathbf{X})$, where $\mathbf{T}' = \mathbf{T}(\mathbf{S} - \mathbf{S}')$. Let $\mathbf{t}_i$ be the ith row of $\mathbf{T}$ and $\mathbf{s}_j - \mathbf{s}'_j$ be the jth column of $\mathbf{S} - \mathbf{S}'$, then we have

$$\begin{aligned}
\|\mathbf{T}'\| = \|\mathbf{T}(\mathbf{S} - \mathbf{S}')\| &= \max_{i,j} \|\langle \mathbf{t}_i, \mathbf{s}_j - \mathbf{s}'_j \rangle\| \\
&\leq m \cdot \max_{i,j,k} \|(t_i)_k \cdot ((s_j)_k - (s'_j)_k)\| \\
&\leq m \cdot \max_{i,j,k} (\|(t_i)_k\| \cdot \|(s_j)_k - (s'_j)_k\| \cdot n) \\
&\leq m \cdot \max_{i,j,k} (\|(t_i)_k\| \cdot (\|(s_j)_k\| + \|(s'_j)_k\|) \cdot n) \\
&= 2mn \cdot \max_{i,j,k} (\|(t_i)_k\| \cdot \|(s_j)_k\|) \\
&= 2mn \cdot t \cdot (4tp + 4t^2n) = 8t^2pmn + 8t^3mn^2 < \beta
\end{aligned} .$$

Additionally, since $\|\mathbf{S} - \mathbf{S}'\| \leq \|\mathbf{S}\| + \|\mathbf{S}'\| \leq 8tp + 8t^2n < \beta$, $\mathbf{S} - \mathbf{S}'$ and $\mathbf{T}'$ are a pair of collisions with respect to $h_{\mathbf{G}}$.

Both **Case 1** and **Case 2** yield a contradiction to the assumption that $\mathrm{Col}_{\mathcal{K}}(h_{\mathbf{G}})$ with respect to the generalized knapsack function family $\mathcal{K}(R, D, m)$ is intractable to solve. Consequently, the commitment scheme is binding. □

Since our underlying commitment scheme $\mathcal{CMT}$ is binding and hiding, the anonymity and unforgeability of the linkable ring signature $\mathcal{LRS}$ can be shown according to Theorem 3. For a complete discussion of the security proof, we refer readers to appendix B. Actually, most of the techniques follows that of [12] and x has the unique multiplicative inverse in R.

The next is to prove that our linkable ring signature is linkable.

Theorem 5 (Linkability). *Our linkable ring signature $\mathcal{LRS}$ is linkable. Formally, given a key pair $(\mathbf{X}, \mathbf{Y} = \mathbf{GX})$, it is impossible to generate a valid signature $\sigma = \{S_1, S_2, \mathbf{Z}_d, \mathbf{R}, L\}$, such that $\mathbf{Y} \in L$ and $\mathbf{R} \neq \mathbf{HX}$.*

Proof. Assume that a user with $(\mathbf{X}, \mathbf{Y})$ generates a signature $\sigma = \{S_1, S_2, \mathbf{Z}_d, \mathbf{R}, L\}$ on behalf of L, such that $\mathbf{Y} \in L$ and $\mathbf{R} \neq \mathbf{HX}$. As σ is a valid signature, from step 5 of the **Vfy** procedure, we have

$$\mathbf{R}(x^M\mathbf{I}) + \sum_{k=0}^{M-1} \mathbf{V}'_{d_k}(-x^k) = \mathbf{HZ}_d \quad . \tag{2}$$

Additionally, since $\mathbf{Z}_d$ can pass step 6 of the **Vfy** procedure, it must be generated by using the signing key $\mathbf{X}$, *i.e.*, $\mathbf{Z}_d = \mathbf{X}(x^M\mathbf{I}) - \sum_{k=0}^{M-1} \mathbf{E}_k x^k$. Otherwise, it yields a contradiction to the unforgeability of the signature scheme. Subsequently, we can deduce from (2) that

$$(\mathbf{HX} - \mathbf{R})(x^M \mathbf{I}) + \sum_{k=0}^{M-1} (\mathbf{V}'_{d_k} - \mathbf{HE}_k)(x^k) = \mathbf{0} \quad . \tag{3}$$

If $\mathbf{R} \neq \mathbf{HX}$, the left side of (3) is a polynomial in x of degree M. Once $\mathbf{R}$, $\mathbf{V}'_{d_k}$, $\mathbf{X}$ and $\mathbf{E}_k$ are given, equation (3) has at most M solutions. However, x is obtained by computing the hash function $\mathcal{H}(pp, \mu, L, \{\mathbf{V}_{\ell_j}, \mathbf{V}_{a_j}, \mathbf{V}_{b_j}, \mathbf{V}_{d_{j-1}}, \mathbf{V}'_{d_{j-1}}\}_{j=1}^{M}, \mathbf{R})$, and involves $2^n \cdot \binom{n}{p} \geq 2^{100}$ possible values. Consequently, the probability that the hash value is the solution of (3) is negligible and the only sensible condition for (3) to be satisfied is $\mathbf{HX} = \mathbf{R}$ and $\mathbf{V}'_{d_k} = \mathbf{HE}_k$. □

5 APQC Based on Linkable Ring Signatures

In CryptoNote, the author suggested using stealth addresses to protect the privacy of receivers in all transactions. A stealth address is a one-time address (a verifying key which is also called a destination key) for a receiver to receive coins. It is generated by the sender of a transaction, and only the real receiver could determine the one-time address and recover the corresponding signing key.

In this section, we will introduce a key-generation protocol to handle stealth addresses. By combining this protocol and the linkable ring signature presented in the previous section, we describe the standard transaction of APQC in detail at last.

5.1 Key-generation Protocol

The key-generation protocol is responsible for three purposes. Firstly, it generates public and private keys for a user that initially joins the cryptocash system. Secondly, if Alice wants to pay coins to Bob, this protocol generates a fresh one-time address for Bob by using the random values chosen by Alice and the public key of Bob. Note that the one-time address is essentially a verifying key of the linkable ring signature scheme. Thirdly, since Alice broadcasts the transaction labeled with the destination address, the key-generation protocol helps Bob to efficiently recognize this transaction and to recover the corresponding signing key.

This protocol is formalized as four efficient procedures $\mathcal{KG}$=(**Setup**, **UKeyGen**, **DKeyGen**, **DKeyRec**) which are short forms for setup, user keys generation, destination keys generation, and destination keys recovery, respectively.

Setup$(1^n, 1^\lambda)$: On input security parameter, this procedure generates global parameters *pp* for the whole cryptocash system which means this procedure also runs $\mathcal{LRS}$.**Setup**(1^n) and $\mathcal{ES}$.**Setup**(1^n) as subroutines so that the signature scheme and encryption scheme are accurately initiated (see Sect. 2.3 and Sect. 4.2 for details). Let $(n, m, \mathbf{G}, \mathbf{H}, \mathcal{H}, q, t, N)$ be the global parameters of the linkable ring signature, and $R = \mathbb{Z}[X]_q / \langle X^n + 1 \rangle$. Besides that, it chooses a cryptographic hash function $hash : \{0,1\}^* \to \{0,1\}^\lambda$. Let $D = \{g \in R : \|g\| \leq t/2\}$.

UKeyGen(pp)**:** When a user wants to join the cryptocash system, he executes this procedure. This procedure first generates the keys for public key encryption scheme $(epk, esk) \leftarrow \mathcal{ES}.\mathbf{KGen}(pp)$. It then generates a partial key pair for the linkable ring signature scheme $\mathbf{X} \leftarrow D^{m\times m}$, $\mathbf{Y} = \mathbf{GX}$. Note that the norm of the partial signing key $\mathbf{X}$ is a little smaller than the original one of the linkable ring signature. $(epk, \mathbf{Y})$ and $(esk, \mathbf{X})$ are the public and private keys held by the user.

DKeyGen$(pp, epk, \mathbf{Y})$**:** If Alice wants to send coins to Bob who holds keys $(epk, \mathbf{Y})$, $(esk, \mathbf{X})$, she runs the procedure with epk and $\mathbf{Y}$. This procedure samples $\mathbf{X}_p \leftarrow D^{m\times m}$ and generates the destination key $\mathbf{Y}_d = \mathbf{GX}_p + \mathbf{Y}$ for Bob. $\mathbf{X}_p$ is a part of the signing key with respect to the destination key $\mathbf{Y}_d$, but no one except Bob can recover the integral signing key. This procedure proceeds to pick an AES secret key k uniformly at random. It then computes $c_1 = \mathcal{ES}.\mathbf{Enc}_{epk}(k)$ with the public key encryption and computes $c_2 = \mathbf{AES}_k(hash(epk)\|\mathbf{X}_p)$ with the AES algorithm. Finally, it outputs the destination key $\mathbf{Y}_d$, and the auxiliary information c_1, c_2. The process of **DkeyGen** procedure is depicted in Fig. 1.

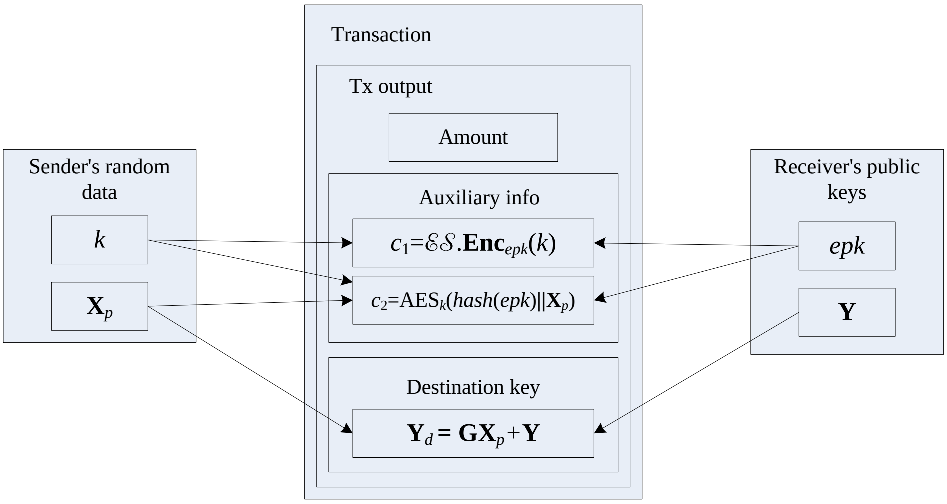

Fig. 1. DKeyGen procedure

DKeyRec$(pp, epk, esk, \mathbf{Y}, \mathbf{X}, (\mathbf{Y}_d, c_1, c_2))$**:** Bob runs this procedure to check $(\mathbf{Y}_d, c_1, c_2)$ of a passing transaction. If it is a transaction with Bob as recipient, it will be that 1) $k = \mathcal{ES}.\mathbf{Dec}_{esk}(c_1)$; 2) $(hash(epk)\|\mathbf{X}_p) = \mathbf{AES}_k(c_2)$. If this procedure finds that the first part of the plaintext of c_2 is not the hash value of Bob's public encryption key epk, then this procedure aborts and outputs 0. Otherwise, Bob computes $\mathbf{X}_d = \mathbf{X}_p + \mathbf{X}$ and $\mathbf{Y}'_d = \mathbf{GX}_d$. If $\mathbf{Y}'_d = \mathbf{Y}_d$, this procedure outputs 1 and admits the validity of the destination key $\mathbf{Y}_d$ and its signing key $\mathbf{X}_d$. Since $\|\mathbf{X}_d\| \le \|\mathbf{X}_p\| + \|\mathbf{X}\| \le t$, $\mathbf{X}_d$ is a valid signing key with correspondence to the destination key $\mathbf{Y}_d$. The process of this procedure is briefly shown in Fig. 2.

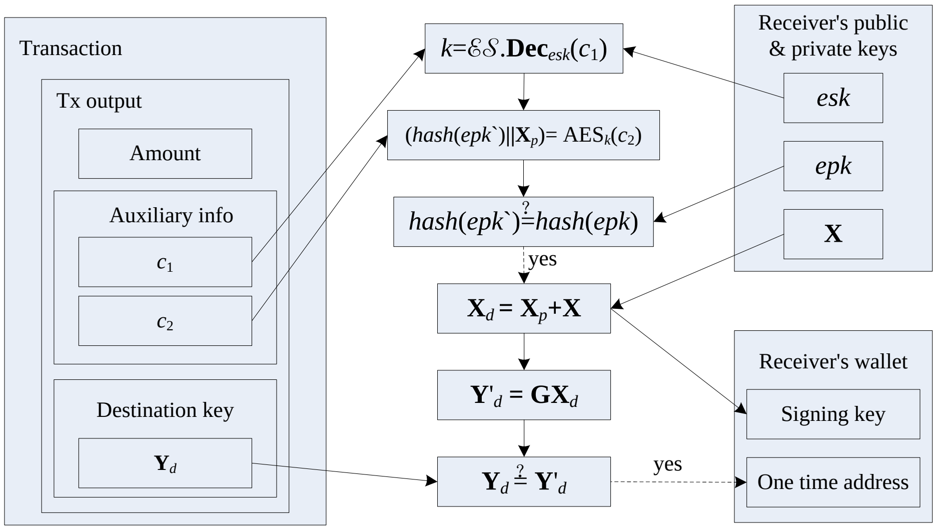

Fig. 2. DKeyRec procedure

5.2 Transactions

We proceed to introduce transactions in APQC. Let Bob and Alice be two users of our APQC. Bob will runs $\mathcal{KG}.\mathbf{UKeyGen}$ to generates his public and private keys $(epk_{\text{Bob}}, \mathbf{Y}_{\text{Bob}})$, $(esk_{\text{Bob}}, \mathbf{X}_{\text{Bob}})$, when he initially joins the system. Similarly, $(epk_{\text{Alice}}, \mathbf{Y}_{\text{Alice}})$, $(esk_{\text{Alice}}, \mathbf{X}_{\text{Alice}})$ are the keys held by Alice. Besides the user keys, Alice and Bob maintain their own wallet addresses, respectively.

Assume that the destination address $\mathbf{Y}_{Bj}$ and its signing key $\mathbf{X}_{Bj}$ are in Alice's wallet, and she wants to send coins of this address to Bob. Alice will specify $N-1$ foreign outputs ($\text{Output}_{B1}, \ldots, \text{Output}_{B(j-1)}, \text{Output}_{B(j+1)}, \ldots, \text{Output}_{BN}$) in which the amount is equivalent to that of Output_{Bj}. She proceeds to find Bob's public key $(epk_{\text{Bob}}, \mathbf{Y}_{\text{Bob}})$ and runs $\mathcal{KG}.\mathbf{DkeyGen}(pp, epk_{\text{Bob}}, \mathbf{Y}_{Bob})$ to generate the destination key $\mathbf{Y}_{Cj}$ and its auxiliary information c_1, c_2 for Bob (see Fig. 1). She then pushes 1) Tx input including $\{\text{Output}_{Bi}\}_{i=1}^{N}$ and the amount she sends to Bob, 2) the destination key $\mathbf{Y}_{Cj}$ and auxiliary information c_1, c_2 she generated for Bob, 3) all previous transactions with output $\{\text{Output}_{Bi}\}_{i=1}^{N}$, into the hash function to obtain a hash digest, μ, of the transaction. Subsequently, she signs the hash digest by running $\sigma \leftarrow \mathcal{LRS}.\mathbf{Sign}(pp, \mathbf{X}_{Bj}, \mu, \mathbf{Y}_{B1}, \ldots, \mathbf{Y}_{BN})$, where $\mathbf{Y}_{Bi}$ is the destination key of Output_{Bi}. Finally she broadcasts the transaction.

Bob checks all passing transactions. For each transaction, he extracts the destination key and auxiliary information $(\mathbf{Y}_d, c_1, c_2)$, and runs the procedure $\mathcal{KG}.\mathbf{DKeyRec}(pp, epk_{Bob}, esk_{Bob}, \mathbf{Y}_{Bob}, \mathbf{X}_{Bob}, (\mathbf{Y}_d, c_1, c_2))$. If this transaction is the one that Alice sent to Bob, the foregoing procedure will return the signing key $\mathbf{X}_{Cj}$ for the destination key $\mathbf{Y}_d = \mathbf{Y}_{Cj}$. If this happens, Bob accepts this transaction and records $\mathbf{X}_{Cj}$, $\mathbf{Y}_d$ into his wallet. Bob can later spend the coin stored in the destination address $\mathbf{Y}_d$ because he has the signing key $\mathbf{X}_{Cj}$.

The standard transaction is also briefly depicted in Fig. 3.

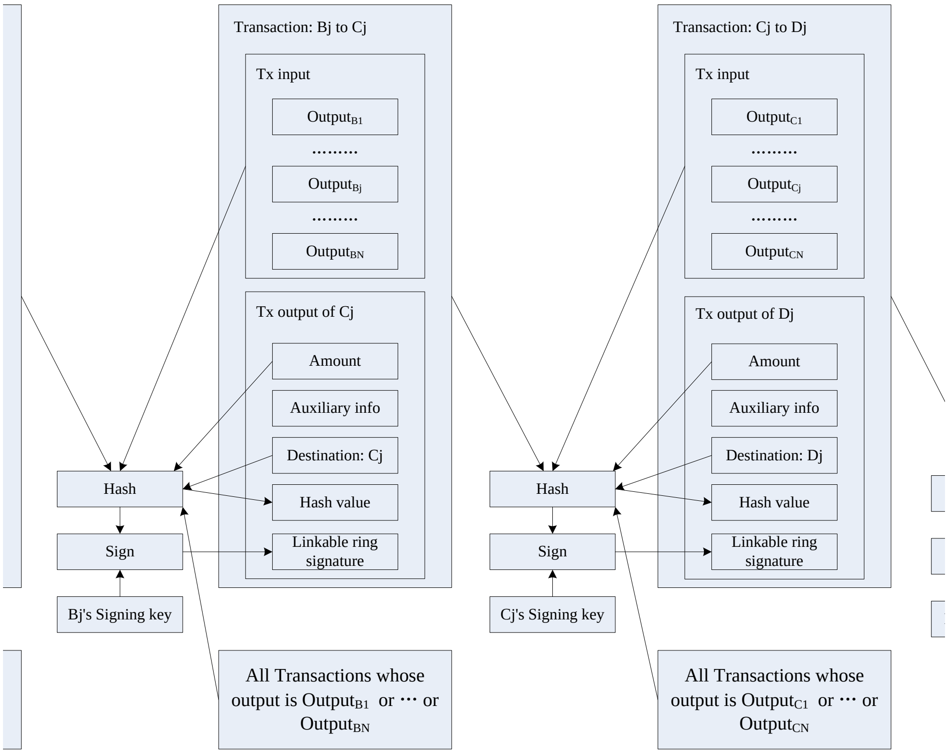

Fig. 3. Transaction chains

6 Conclusions and Future Works

While a lot of lattice-based ring signature and standard signature have been designed recently, linkable ring signature over lattices has not been to the best of our knowledge. The strong similarity in the construction between a lattice-based signature and DLP-based one, *e.g.*, the signature in [18] and the Schnorr's signature

[32, 33], can help us to design the lattice-based counterparts of DLP-based linkable ring signatures. In this paper, using the techniques in [12], we construct a linkable ring signature from ideal-lattices in which the size of a signature, on behalf of a ring with N participants, is $O(\log N)$. Based on the proposed signature scheme, we present an anonymous post-quantum cryptocash system by following the major ideas in CryptoNote and Monero. In order to generate stealth addresses (verifying keys) and recover corresponding signing keys for the linkable ring signature, we provide a key-generation protocol as a subroutine of the cryptocash system. By combining all those techniques together, our cryptocash protocol obtains a new level anonymity comparing to the original Bitcoin system. Furthermore, the new designed cryptocash system has the potential to resist quantum attacks.

Recently, the unlinkability and untraceability of Monero were analyzed by [24] and [14]. Some of them were blamed on the abuses of users, *e.g.* signing a transaction on behalf of a ring with only 1 participant. Besides, there are still a few inherent weakness in Monero, *e.g.* for an overwhelming proportion of input addresses, a user can't find enough addresses with the same value to hide his real address, especially in the early time of the system. Next, we shall trace these problems and discuss what should be done to make our cryptocash system secure under these analyses. A full cryptocash system will be implement to test the communication and computation costs. And if possible, we would like to contribute our system to the cryptocash community for public usage.

Acknowledgements

This work was supported by the National Key R&D Program of China (2017YB0802500), the National Natural Science Foundation of China (No. 61672550) and the Fundamental Research Funds for the Central Universities (No.17lgjc45).

The authors are grateful to the anonymous reviewers for their valuable suggestions and comments on this paper.

References

1. Aharonov, D., Regev, O.: Lattice problems in NP $\cap$ coNP. J. ACM 52(5), 749–765 (sep 2005)
2. Ajtai, M.: Generating hard instances of lattice problems (extended abstract). In: Symposium on Theory of Computing—STOC 1996. pp. 99–108. ACM (1996)
3. Barber, S., Boyen, X., Shi, E., Uzun, E.: Bitter to better—how to make Bitcoin a better currency. In: Financial Cryptography and Data Security—FC 2012. pp. 399–414. Springer Berlin Heidelberg (2012)
4. Bellare, M., Boldyreva, A., Desai, A., Pointcheval, D.: Key-privacy in public-key encryption. In: Advances in Cryptology—ASIACRYPT 2001. pp. 566–582. Springer Berlin Heidelberg (2001)
5. Cai, J.Y., Nerurkar, A.P.: An improved worst-case to average-case connection for lattice problems. In: Symposium on Foundations of Computer Science—FOCS 1997. pp. 468–477. IEEE (Oct 1997)

6. Cayrel, P.L., Lindner, R., Rückert, M., Silva, R.: A lattice-based threshold ring signature scheme. In: Progress in Cryptology—LATINCRYPT 2010. pp. 255–272. Springer Berlin Heidelberg (2010)
7. Ducas, L., Durmus, A., Lepoint, T., Lyubashevsky, V.: Lattice signatures and bimodal Gaussians. In: Advances in Cryptology—CRYPTO 2013. pp. 40–56. Springer Berlin Heidelberg (2013)
8. Fiat, A., Shamir, A.: How to prove yourself: Practical solutions to identification and signature problems. In: Advances in Cryptology—CRYPTO 1986. pp. 186–194. Springer Berlin Heidelberg (1987)
9. Fujisaki, E., Suzuki, K.: Traceable ring signature. In: Public Key Cryptography—PKC 2007. pp. 181–200. Springer Berlin Heidelberg (2007)
10. Gentry, C., Peikert, C., Vaikuntanathan, V.: Trapdoors for hard lattices and new cryptographic constructions. In: Symposium on Theory of Computing—STOC 2008. pp. 197–206. ACM (2008)
11. Goldreich, O., Goldwasser, S.: On the limits of non-approximability of lattice problems. In: Symposium on Theory of Computing—STOC 1998. pp. 1–9. ACM (1998)
12. Groth, J., Kohlweiss, M.: One-out-of-many proofs: Or how to leak a secret and spend a coin. In: Advances in Cryptology—EUROCRYPT 2015. pp. 253–280. Springer Berlin Heidelberg (2015)
13. Halevi, S.: A sufficient condition for key-privacy. Cryptology ePrint Archive, Report 2005/005 (2005), http://eprint.iacr.org/2005/005
14. Kumar, A., Fischer, C., Tople, S., Saxena, P.: A traceability analysis of Monero's blockchain. Cryptology ePrint Archive, Report 2017/338 (2017), http://eprint.iacr.org/2017/338
15. Libert, B., Ling, S., Nguyen, K., Wang, H.: Zero-knowledge arguments for lattice-based accumulators: Logarithmic-size ring signatures and group signatures without trapdoors. In: Advances in Cryptology—EUROCRYPT 2016. pp. 1–31. Springer Berlin Heidelberg (2016)
16. Liu, J.K., Wei, V.K., Wong, D.S.: Linkable spontaneous anonymous group signature for ad hoc groups. In: Information Security and Privacy—ACISP 2004. pp. 325–335. Springer Berlin Heidelberg (2004)
17. Liu, J.K., Wong, D.S.: Linkable ring signatures: Security models and new schemes. In: Computational Science and Its Applications—ICCSA 2005. pp. 614–623. Springer Berlin Heidelberg (2005)
18. Lyubashevsky, V.: Lattice signatures without trapdoors. In: Advances in Cryptology—EUROCRYPT 2012. pp. 738–755. Springer Berlin Heidelberg (2012)
19. Lyubashevsky, V., Micciancio, D.: Generalized compact knapsacks are collision resistant. In: Automata, Languages and Programming—ICALP 2006. pp. 144–155. Springer Berlin Heidelberg (2006)
20. Micciancio, D.: Generalized compact knapsacks, cyclic lattices, and efficient one-way functions from worst-case complexity assumptions. In: Symposium on Foundations of Computer Science—FOCS 2002. pp. 356–365. IEEE (2002)
21. Micciancio, D.: Generalized compact knapsacks, cyclic lattices, and efficient one-way functions. computational complexity 16(4), 365–411 (Dec 2007)
22. Micciancio, D., Regev, O.: Worst-case to average-case reductions based on Gaussian measures. SIAM Journal on Computing 37(1), 267–302 (2007)
23. Miers, I., Garman, C., Green, M., Rubin, A.D.: Zerocoin: Anonymous distributed e-cash from Bitcoin. In: Symposium on Security and Privacy—SP 2013. pp. 397–411 (May 2013)

24. Miller, A., Möser, M., Lee, K., Narayanan, A.: An empirical analysis of linkability in the Monero blockchain. eprint arXiv:1704.04299 (2017), https://arxiv.org/abs/1704.04299
25. Nakamoto, S.: Bitcoin: A peer-to-peer electronic cash system (2012), http://www.bitcoin.org/bitcoin.pdf
26. Noether, S.: Ring signature confidential transactions for Monero. Cryptology ePrint Archive, Report 2015/1098 (2015), https://eprint.iacr.org/2015/1098
27. Ober, M., Katzenbeisser, S., Hamacher, K.: Structure and anonymity of the Bitcoin transaction graph. Future Internet 5(2), 237–250 (2013)
28. Reid, F., Harrigan, M.: An Analysis of Anonymity in the Bitcoin System, pp. 197–223. Springer New York (2013)
29. Rivest, R.L., Shamir, A., Tauman, Y.: How to leak a secret. In: Advances in Cryptology—ASIACRYPT 2001. pp. 552–565. Springer Berlin Heidelberg (2001)
30. Ron, D., Shamir, A.: Quantitative analysis of the full Bitcoin transaction graph. In: Financial Cryptography and Data Security—FC 2013. pp. 6–24. Springer Berlin Heidelberg (2013)
31. Saberhagen, N.v.: Cryptonote v2.0 (2013), https://cryptonote.org/whitepaper.pdf
32. Schnorr, C.P.: Efficient identification and signatures for smart cards. In: Advances in Cryptology—EUROCRYPT 1989. pp. 688–689. Springer Berlin Heidelberg (1990)
33. Schnorr, C.P.: Efficient signature generation by smart cards. Journal of Cryptology 4(3), 161–174 (Jan 1991)
34. Stehlé, D., Steinfeld, R., Tanaka, K., Xagawa, K.: Efficient public key encryption based on ideal lattices. In: Advances in Cryptology—ASIACRYPT 2009. pp. 617–635. Springer Berlin Heidelberg (2009)
35. Sun, S.F., Au, M.H., Liu, J.K., Yuen, T.H.: RingCT 2.0: A compact accumulator-based (linkable ring signature) protocol for blockchain cryptocurrency Monero. In: Computer Security—ESORICS 2017. pp. 456–474. Springer International Publishing (2017)
36. Wang, C., Wang, H.: A new ring signature scheme from NTRU lattice. In: Computational and Information Sciences—ICCIS 2012. pp. 353–356. IEEE (Aug 2012)
37. Wang, J., Sun, B.: Ring signature schemes from lattice basis delegation. In: Information and Communications Security—ICICS 2011. pp. 15–28. Springer Berlin Heidelberg (2011)

Appendix

A Short Linkable Ring Signature Based on ECDLP

Let N be the size of the ring and $n = \log N$. Define $f_{j,1} = f_j = \ell_j e + a_j = \delta_{1\ell_j} e + a_j$, and $f_{j,0} = e - f_j = (1 - \ell_j)e - a_j = \delta_{0\ell_j} e - a_j$. For every $i \in [0, N-1]$ the product $\prod_{j=1}^{n} f_{j,i_j}$ is a polynomial in the indeterminate e of the form

$$p_i(e) = \prod_{j=1}^{n} (\delta_{i_j \ell_j} e) + \sum_{k=0}^{n-1} p_{i,k} e^k = \delta_{i\ell} e^n + \sum_{k=1}^{n-1} p_{i,k} e^k.$$

Here, $p_{i,k}$ is the coefficient of the kth degree term of the polynomial $p_i(e)$, and can be efficiently computed when $\{a_j\}_{j=1}^{n}$, i and ℓ are given.

The linkable ring signature based on ECDLP consists of five efficient procedures (**Setup**, **KGen**, **Sign**, **Vry**, **Link**).

Setup(1^λ): Let E be an elliptic curve defined over a finite field $\mathbb{F}_q$. Let $G \in E$ be a point of prime order p, here $|p| = \lambda$ and let $\mathbb{G}$ be the prime order subgroup of E generated by G. Choose another element $H \in \mathbb{G}$ randomly. Let $\mathcal{H} : \{0,1\}^* \to \mathbb{Z}_p$ be a cryptographic hash function. The output of this procedure is $pp = (\mathbb{G}, G, H, p, q, \mathcal{H})$.

KGen(pp): For the ith user, this procedure chooses the signing key $x_i \in \mathbb{Z}_p$ uniformly at random and computes the verifying key $Y_i = x_i G$. It outputs (x_i, Y_i) as the key pair of the ith user.

Sign(pp, x_ℓ, μ, L): Let $L = (Y_0, Y_1, \ldots, Y_{N-1})$ be the ensemble of the ring. On input the message μ, the ℓth user's signature on behalf of L is generated as follows

- Compute $I_\ell = x_\ell H$.
- For j from 1 to n,
  - choose $r_j, a_j, s_j, t_j, \rho_k \leftarrow \mathbb{Z}_p$ at random.
  - compute $C_{\ell_j} = \ell_j H + r_j G$,
  - compute $C_{a_j} = a_j H + s_j G$,
  - compute $C_{b_j} = a_j \ell_j H + t_j G$,
- For k from 1 to $n-1$
  - choose $\rho_k \leftarrow \mathbb{Z}_p$ at random,
  - compute $C_{d_k} = (\sum_{i=0}^{N-1} p_{i,k} Y_i) + \rho_k G$,
  - compute $C'_{d_k} = \rho_k H$, for $k = j-1$.
- Let $\mathbf{a} = \{C_{\ell_j}, C_{a_j}, C_{b_j}, C_{d_{j-1}}, C'_{d_{j-1}}\}_{j=1}^{n}$ and compute $e = \mathcal{H}(pp, u, L, \mathbf{a}, I_\ell)$
- For j from 1 to n, compute
  - $f_j = e\ell_j + a_j$,
  - $z_{a_j} = er_j + s_j$,
  - $z_{b_j} = (e - f_j) r_j + t_j$,
  - $z_d = e^n x_\ell - \sum_{k=0}^{n-1} e^k \rho_k$.
- Let $\mathbf{b} = \{f_j, z_{a_j}, z_{b_j}\}_{j=1}^{n}$. Publish $\sigma = \{\mathbf{a}, \mathbf{b}, z_d, I_\ell, L\}$ as the signature of the ℓth user.

Vry(pp, μ, σ, L):

- Compute $e = \mathcal{H}(pp, \mu, L, \mathbf{a}, I_\ell)$
- For j from 1 to n, consider the following equalities
  - $eC_{\ell_j} + C_{a_j} = f_j H + z_{a_j} G$,
  - $(e - f_j)C_{\ell_j} + C_{b_j} = z_{b_j} G$,

  If any one of them doesn't hold, output 0 and abort.
- If the equality $e^n I_\ell + \sum_{k=0}^{n-1}(-e^k)C'_{d_k} = z_d H$ doesn't hold, output 0 and abort.
- Inspect whether $\sum_{i=0}^{N-1}(\prod_{j=1}^{n} f_{j,i_j})Y_i + \sum_{k=0}^{n-1}(-e^k)C_{d_k} = z_d G$. If it is not, output 0 and abort; otherwise output 1.

Link(pp, σ, σ'): For any two signatures $\sigma_1 = (\ldots, I_1, L_1)$ and $\sigma_2 = (\ldots, I_2, L_2)$, if $I_1 = I_2$, return 1 (linked) for concluding that they are generated by the same signer; otherwise, return 0 (unlinked). Since H is the global parameter and $I_\ell = x_\ell H$ is an ingredient to verify the valid signature, x_i can only sign one message during the whole system life.

Note that the Pedersen commitment of value 0 can act as a public key of our ECDLP-based linkable ring signature. As a result, the technique of RingCT [26] (later strengthened by Sun *et al.* [35]), which is adopted in Monero to hide the amount of a transaction, is trivially achievable in our settings. Using the above logarithmic size linkable ring signature to replace the linkable ring signature scheme in Monero, we can implement a more efficient Monero system.

B Security Proofs for Underlying Primitives

If we remove the parameters and steps for linking from our linkable ring signature $\mathcal{LRS}$, we obtain an ordinary ring signature. Section. 4.1 shows the three underlying ingredients to construct a ring signature — a commitment scheme and two Sigma-protocols (Σ_1, Σ_2). Our commitment scheme is introduced in Sect. 4.3. The major work of the current section is to describe Σ_1 and Σ_2.

B.1 Definitions [12]

Let R be an efficiently decidable ternary relation. For pairs $(crs, u, w) \in \mathcal{R}$ we call u the statement and w the witness, where crs is a common reference string. Let L be the CRS-dependent language consisting of statements in $\mathcal{R}$. A Σ-protocol (3-move interactive proof system) for relation $\mathcal{R}$ consists of a common reference string generation algorithm $\mathcal{G}$, a prover $\mathcal{P}$ and a verifier $\mathcal{V}$. We require that they all be PPT algorithms. The following run of a Σ-protocol describes the interaction of the algorithms

1. $\mathcal{G}$ produces the common reference string crs of length $\Omega(\lambda)$, on input a security parameter λ.
2. $\mathcal{P}$ takes as input (crs, u, w) and generates an initial message a.
3. $\mathcal{V}$ sends to $\mathcal{P}$ a challenge x chosen uniformly at random.

4. On input x, $\mathcal{P}$ gives a response z to $\mathcal{V}$ in return.
5. Given (crs, u, a, x, z), $\mathcal{V}$ returns 1 if accepting the proof and 0 if rejecting the proof.

The triple $(\mathcal{G}, \mathcal{P}, \mathcal{V})$ is called a Σ-protocol for $\mathcal{R}$ if it provides the properties of completeness, n-special soundness and special honest verifier zero-knowledge.
Perfect completeness. A proof system is complete if an honest prover with a valid witness can convince an honest verifier. Formally we have that for all $\lambda \in \mathbb{N}$, $crs \leftarrow \mathcal{G}(1^\lambda)$ if $(crs, u, w) \in \mathcal{R}$, then

$$\Pr[\mathcal{V}(crs, u, a, x, z) = 1 : a \leftarrow \mathcal{P}(crs, u, w), x \leftarrow \{0,1\}^\lambda, z \leftarrow \mathcal{P}(x)] = 1\,.$$

n-special soundness. A proof system is able to convince an honest verifier, only if the statement is true. In other words, if the statement is false (a statement has no corresponding witness), no one could convince an honest verifier. The n-special soundness says that given responses to a number of different challenges, it is possible to compute a witness for the statement. Formally, there is an efficient extraction algorithm $\mathcal{X}$ such that for all $\lambda \in \mathbb{N}$, $crs \leftarrow \mathcal{G}(1^\lambda)$ and $(crs, u, w) \in \mathcal{R}$, it satisfies

$$\Pr[w \leftarrow \mathcal{X}(crs, u, a, x_1, z_1, \ldots, x_n, z_n) : (u, a, x_1, z_1, \ldots, x_n, z_n) \leftarrow \mathcal{A}(crs)] \approx 1\,,$$

where $\mathcal{A}$ is an efficient algorithm to generate n distinct valid responses for n distinct challenges corresponding to the same initial message.
Special honest verifier zero-knowledge. A Σ-protocol is computational zero-knowledge if the proofs do not reveal any information about the witnesses to a bounded adversary. Instead of the original definition, we consider the special honest verifier zero-knowledge in the sense that if the verifier's challenge is known in advance and the statement is true, then it is possible to simulate the entire proofs without knowing the witness. Formally there exists a PPT simulator $\mathcal{S}$ such that for all $\lambda \in \mathbb{N}$, $crs \leftarrow \mathcal{G}(1^\lambda)$, $(crs, u, w) \in \mathcal{R}$ and PPT adversaries $\mathcal{A}$

$$\begin{aligned}&\Pr[\mathcal{A}(a, x, z) = 1 : a \leftarrow \mathcal{P}(crs, u, w), x \in \{0,1\}^\lambda, z \leftarrow \mathcal{P}(x)]\\ &\quad\approx \Pr[\mathcal{A}(a, x, z) = 1 : x \in \{0,1\}^\lambda, (a, z) \leftarrow \mathcal{S}(crs, u, x)]\,.\end{aligned}$$

B.2 Sigma-Protocol for Commitment to 0 or 1

Let $\mathcal{G}$ be the setup algorithm, $\mathcal{CMT}.\mathbf{Gen}$, for the commitment scheme in Sect. 4.3. We define a polynomial time decidable relation

$$\mathcal{R} = \{(crs, \mathbf{V}_\ell, (\ell, \mathbf{K})) : \mathbf{V}_\ell = \mathbf{H}(\ell\mathbf{I}) + \mathbf{GK} \wedge \ell \in \{0,1\} \wedge \mathbf{K} \in Q^{m\times m}\} \quad ,$$

where $crs \leftarrow \mathcal{G}(1^n)$ is global parameters for the commitment scheme. The corresponding CRS-dependent language is defined as

$$L = \{\mathbf{V}_\ell : \exists(\ell, \mathbf{K}) : (crs, \mathbf{V}_\ell, (\ell, \mathbf{K})) \in \mathcal{R}\} \quad .$$

$\Sigma_1 = \{\mathcal{G}, \mathcal{P}, \mathcal{V}\}$ is a Sigma-protocol for $\mathcal{R}$. The prover, $\mathcal{P}(crs, \ell, \mathbf{K}, \mathbf{V}_\ell)$, aims at proving membership in L to the verifier, $\mathcal{V}(crs, \mathbf{V}_\ell)$. The details of $\mathcal{P}$, $\mathcal{V}$ and their interactions are as follows.

Algorithm $\mathcal{P}$:

– Initial message:
  - if $\ell = 0$, randomly pick $\mathbf{B} \leftarrow \mathrm{diag}(Q^{m\times m})$,
    else if $\ell = 1$, randomly draw $\mathbf{B} \leftarrow \mathrm{diag}(\tilde{Q}^{m\times m})$,
  - sample $\mathbf{C}, \mathbf{D} \leftarrow Q^{m\times m}$,
  - $\mathbf{V}_a = \mathbf{HB} + \mathbf{GC}$,
  - $\mathbf{V}_b = \mathbf{H}(\ell s) + \mathbf{GD}$.

Algorithm $\mathcal{V}$:

– Challenge:
  - $x \leftarrow \{-1, 0, 1\}^n$ such that $\|x\|_1 \le p$ and $2^p \cdot \binom{n}{p} \ge 2^{100}$.

Algorithm $\mathcal{P}$:

– Response:
  - $\mathbf{W} = \ell x \mathbf{I} + \mathbf{B}$,
  - $\mathbf{Z}_a = \mathbf{K}(x\mathbf{I}) + \mathbf{C}$,
  - $\mathbf{Z}_b = \mathbf{K}(x\mathbf{I} - \mathbf{W}) + \mathbf{D}$.

Algorithm $\mathcal{V}$:

– Verification:
  - $\mathbf{W}$ is a diagonal matrix,
  - $\|\mathbf{W}\| \le t$, $\|\mathbf{Z}_a\| \le (p+1)t$, $\|\mathbf{Z}_b\| \le tp + t^2 n + t$,
  - $\mathbf{V}_\ell(x\mathbf{I}) + \mathbf{V}_a = \mathbf{HW} + \mathbf{GZ}_a$,
  - $\mathbf{V}_\ell(x\mathbf{I} - \mathbf{W}) + \mathbf{V}_b = \mathbf{GZ}_b$.

Depending on the discussion in Sect. 4.3, for any value to be committed, its commitment $\mathbf{V}_\ell$ generated by $\mathcal{CMT}$ is uniformly distributed in $R^{1\times m}$. Thus, every element $\mathbf{V}_\ell \in R^{1\times m}$ has the potential to be opened as a value-randomness pair $(\ell, \mathbf{K})$, for $\ell \in \{0, 1\}$, $\mathbf{K} \in P_2^{m\times m}$. Thus, the membership in language, L, is trivially determined. As a result, the proof system has to reveal information about witness, so that valid proofs must be generated by using witness. This fact may be the reason that we do not consider the perfect zero-knowledge in the current proof system.

We proceed to prove completeness, 2-special soundness, and perfect special honest verifier zero-knowledge in Σ_1.

Completeness: From the verification step, we have

$$\mathbf{V}_\ell(x\mathbf{I}) + \mathbf{V}_a = \mathbf{H}(\ell x\mathbf{I} + \mathbf{B}) + \mathbf{G}(\mathbf{K}(x\mathbf{I}) + \mathbf{C}) = \mathbf{HW} + \mathbf{GZ}_a \quad ,$$

$$\mathbf{V}_\ell(x\mathbf{I} - \mathbf{W}) + \mathbf{V}_b = \mathbf{H}(\ell(1-\ell)x\mathbf{I}) + \mathbf{GZ}_b \quad . \tag{4}$$

If $\ell \in \{0, 1\}$, equation (4) equals $\mathbf{GZ}_b$. Completeness is satisfied.

Before we getting start the discussion on 2-special soundness. Note that q is a prime and $f \in \mathbb{Z}_q[X]$ is an irreducible polynomial. This implies that $R = \mathbb{Z}_q[X]/\langle f\rangle$ is a finite field and hence every non-zero element $a \in R$ is invertible. With the modified Euclidean algorithm, we can find $b, s \in \mathbb{Q}[X]$ such that $ab + sf = 1$. Moreover, under the condition $\deg b \le \deg f = n$, $b \in \mathbb{Q}[X]$ is unique. Consequently, the polynomial $[b]_q \in R$, obtained by reducing the coefficients of b modulo q, is the unique inverse of a in R.

2-special soundness: A malicious prover may generate a commitment $\mathbf{V}_\ell = \mathbf{HM} + \mathbf{GK}$, and try to convince an honest verifier of Σ_1, even though $\mathbf{M} \neq \ell\mathbf{I}$, $\ell \in \{0,1\}$ or the malicious prover does not know a witness for $\mathbf{V}_\ell$.

Let $\mathbf{W}$, $\mathbf{Z}_a$, $\mathbf{Z}_b$ and $\mathbf{W}'$, $\mathbf{Z}'_a$, $\mathbf{Z}'_b$ be two valid responses to challenges x and x' on the same initial message $\mathbf{V}_a$, $\mathbf{V}_b$, respectively. We remark that those responses may not be generated following the description of Σ_1, so does the initial message.

Since the responses and initial message could convince an honest verifier, from the first equation in the verification step, we have

$$\begin{aligned}(\mathbf{V}_\ell(x\mathbf{I}) + \mathbf{V}_a) - (\mathbf{V}_\ell(x'\mathbf{I}) + \mathbf{V}_a) &= \mathbf{HW} + \mathbf{GZ}_a - (\mathbf{HW}' + \mathbf{GZ}'_a) \\ &\Downarrow \\ \mathbf{V}_\ell(x - x') &= \mathbf{H}(\mathbf{W} - \mathbf{W}') + \mathbf{G}(\mathbf{Z}_a - \mathbf{Z}'_a) \\ &\Downarrow \\ (\mathbf{HM} + \mathbf{GK})(x - x') &= \mathbf{H}(\mathbf{W} - \mathbf{W}') + \mathbf{G}(\mathbf{Z}_a - \mathbf{Z}'_a)\end{aligned} \quad . \tag{5}$$

Define

$$\mathbf{M}(x - x') = (\mathbf{W} - \mathbf{W}') \quad , \mathbf{K}(x - x') = (\mathbf{Z}_a - \mathbf{Z}'_a) \quad . \tag{6}$$

Then, $\mathbf{M} = (\mathbf{W} - \mathbf{W}')(x - x')^{-1}$ and $\mathbf{K} = (\mathbf{Z}_a - \mathbf{Z}'_a)(x - x')^{-1}$ seem to be an opening to $\mathbf{V}_\ell = \mathbf{HM} + \mathbf{GK}$.

We proceed to observe the second equation in the verification step, we have

$$\begin{aligned}\mathbf{V}_\ell(x\mathbf{I} - \mathbf{W}) + \mathbf{V}_b - (\mathbf{V}_\ell(x'\mathbf{I} - \mathbf{W}') + \mathbf{V}_b) &= \mathbf{GZ}_b - \mathbf{GZ}'_b \\ &\Downarrow \\ \mathbf{V}_\ell((x - x') - (\mathbf{W} - \mathbf{W}')) &= \mathbf{G}(\mathbf{Z}_b - \mathbf{Z}'_b) \\ &\Downarrow \\ &\text{using the result in (5)} \\ &\Downarrow \\ \mathbf{V}_\ell((x - x') - \mathbf{M}(x - x')) &= \mathbf{G}(\mathbf{Z}_b - \mathbf{Z}'_b) \\ &\Downarrow \\ \mathbf{V}_\ell(\mathbf{I} - \mathbf{M})(x - x') &= \mathbf{G}(\mathbf{Z}_b - \mathbf{Z}'_b) \\ &\Downarrow \\ \mathbf{HM}(\mathbf{I} - \mathbf{M})(x - x') + \mathbf{GK}(\mathbf{I} - \mathbf{M})(x - x') &= \mathbf{G}(\mathbf{Z}_b - \mathbf{Z}'_b)\end{aligned} \quad . \tag{7}$$

If $\mathbf{M} \neq \mathbf{I} \wedge \mathbf{M} \neq \mathbf{0}$, by using (6), we rewrite (7) as

$$\begin{aligned}\mathbf{H}(\mathbf{I} - \mathbf{M})(\mathbf{W} - \mathbf{W}') + \mathbf{G}(\mathbf{Z}_a - \mathbf{Z}'_a - \mathbf{K}(\mathbf{W} - \mathbf{W}') = \mathbf{G}(\mathbf{Z}_b - \mathbf{Z}'_b) \\ \Downarrow \\ \text{multiplying } (x - x') \text{ on both sides} \\ \Downarrow \\ \mathbf{H}((x - x')\mathbf{I} - (\mathbf{W} - \mathbf{W}'))(\mathbf{W} - \mathbf{W}') + \mathbf{G}((\mathbf{Z}_a - \mathbf{Z}'_a)((x - x') + (\mathbf{W} - \mathbf{W}')) \\ = \mathbf{H0} + \mathbf{G}(\mathbf{Z}_b - \mathbf{Z}'_b)(x - x') \quad .\end{aligned}$$

Consider the norms of the multiplicands of $\mathbf{G}$ and $\mathbf{H}$ in the foregoing equation.

$$\begin{aligned}&\|((x-x')\mathbf{I}-(\mathbf{W}-\mathbf{W}'))(\mathbf{W}-\mathbf{W}')\| \\ =&\|(\mathbf{W}-\mathbf{W}')(x-x')-(\mathbf{W}-\mathbf{W}')^2\| \\ \le&\|(\mathbf{W}-\mathbf{W}')(x-x')\|+\|(\mathbf{W}-\mathbf{W}')^2\| \\ \le&(\|\mathbf{W}\|+\|\mathbf{W}'\|)2p+(\|\mathbf{W}\|+\|\mathbf{W}'\|)^2 n \\ \le&4tp+4t^2 n\end{aligned} \quad ,$$

$$\begin{aligned}&\|(\mathbf{Z}_a-\mathbf{Z}'_a)((x-x')+(\mathbf{W}-\mathbf{W}'))\| \\ =&\|(\mathbf{Z}_a-\mathbf{Z}'_a)(x-x')-(\mathbf{Z}_a-\mathbf{Z}'_a)(\mathbf{W}-\mathbf{W}')\| \\ \le&\|(\mathbf{Z}_a-\mathbf{Z}'_a)(x-x')\|+\|(\mathbf{Z}_a-\mathbf{Z}'_a)(\mathbf{W}-\mathbf{W}')\| \\ \le&(\|\mathbf{Z}_a\|+\|\mathbf{Z}'_a\|)2p+(\|\mathbf{Z}_a\|+\|\mathbf{Z}'_a\|)(\|\mathbf{W}\|+\|\mathbf{W}'\|)n \\ \le&4(p+1)tp+4(p+1)t^2 n \\ =&4tp+4tp^2+4t^2pn+4t^2n\end{aligned} \quad ,$$

$$\|(\mathbf{Z}_b-\mathbf{Z}'_b)(x-x')\| \le (\|\mathbf{Z}_b\|+\|\mathbf{Z}'_b\|)2p \le (tp+t^2n+t)4p = 4tp+4tp^2+4t^2pn \quad .$$

Since the largest one, $\|(\mathbf{Z}_a-\mathbf{Z}'_a)((x-x')+(\mathbf{W}-\mathbf{W}'))\|$, is not larger than β, $((x-x')\mathbf{I}-(\mathbf{W}-\mathbf{W}'))(\mathbf{W}-\mathbf{W}')$, $(\mathbf{Z}_a-\mathbf{Z}'_a)((x-x')+(\mathbf{W}-\mathbf{W}'))$ and $\mathbf{0}$, $(\mathbf{Z}_b-\mathbf{Z}'_b)(x-x')$ are two pairs of opening for the same commitment. This yields a contradiction that $\mathcal{CMT}$ is binding. Consequently, we have $\mathbf{M}\neq\mathbf{I}\vee\mathbf{M}\neq\mathbf{0}$, i.e., $\mathbf{M}=\ell\mathbf{I}$, $\ell\in\{0,1\}$.

Return to (6), we know that $(\mathbf{W}-\mathbf{W}')(x-x')^{-1}$ and $(\mathbf{Z}_a-\mathbf{Z}'_a)(x-x')^{-1}$ are a witness to prove that $\mathbf{V}_\ell$ is a commitment to 0 or 1.

Special honest verifier zero-knowledge: Given the system parameters, $\mathbf{V}_\ell$ and x, the simulator randomly chooses $\mathbf{W}$ from $\{g\in R:\|g\|\le t\}$, $\mathbf{Z}_a$ from $\{g\in R:\|g\|\le(p-1)t\}$, $\mathbf{Z}_b$ from $\{g\in R:\|g\|\le tp+t^2n+t\}$. The distributions to sample them is equivalent to their distributions in the real protocol. It then computes $\mathbf{V}_a=\mathbf{V}_\ell(-x\mathbf{I})+(\mathbf{HW}+\mathbf{GZ}_a)$, and $\mathbf{V}_b=\mathbf{V}_\ell(\mathbf{W}-x\mathbf{I})+\mathbf{GZ}_b$. Since the distribution $\mathbf{GZ}_a$ and that of $\mathbf{GZ}_b$ are almost uniform over $R^{1\times m}$, $\mathbf{V}_a$ and $\mathbf{V}_b$ are random elements from the uniform distribution over $R^{1\times m}$. This shows that the simulated proofs can also convince an honest verifier.

B.3 Σ-Protocol for One Out of N Commitments Containing 0

Let $\{\mathbf{Y}_i=\mathbf{GX}_i\}_{i=0}^{N-1}$ be N commitments to 0 and the prover knows the opening of the ℓth commitment (namely, $\mathbf{X}_i$). $\Sigma_2=(\mathcal{G},\mathcal{P},\mathcal{V})$ is a Sigma-protocol to prove that one of these N commitments is opened to 0

Algorithm $\mathcal{P}$:

- Initial messages:
  - For j from 1 to M,
    - sample $\mathbf{K}_j,\mathbf{C}_j,\mathbf{D}_j,\mathbf{E}_k\leftarrow Q^{m\times m}$,
    - if $\ell_j=0$, randomly pick $\mathbf{B}_j\leftarrow\mathrm{diag}(Q^{m\times m})$,
      else if $\ell_j=1$, randomly draw $\mathbf{B}_j\leftarrow\mathrm{diag}(\tilde{Q}^{m\times m})$,
    - compute $\mathbf{V}_{\ell_j}=\mathbf{H}(\ell_j\mathbf{I})+\mathbf{GK}_j$, and $\mathbf{V}_{a_j}=\mathbf{HB}_j+\mathbf{GC}_j$,
    - compute $\mathbf{V}_{b_j}=\mathbf{H}(\ell_j\mathbf{B}_j)+\mathbf{GD}_j$,

- For k from 0 to $M-1$
  * compute $\mathbf{V}_{d_k} = (\sum_{i=0}^{N-1} \mathbf{Y}_i \mathbf{P}_{i,k}) + \mathbf{G}\mathbf{E}_k$

– Challenge:

- $x \leftarrow \{-1,0,1\}^n$ such that $\|\mathbf{v}\|_1 \leq p\}$ and $2^p \cdot \binom{n}{p} \geq 2^{100}$.

– Responses:

- For j from 1 to M, compute
  * $\mathbf{W}_j = \ell_j x\mathbf{I} + \mathbf{B}_j$,
  * $\mathbf{Z}_{a_j} = \mathbf{K}_j(x\mathbf{I}) + \mathbf{C}_j$,
  * $\mathbf{Z}_{b_j} = \mathbf{K}_j(x\mathbf{I} - \mathbf{W}_j) + \mathbf{D}_j$,
- Compute $\mathbf{Z}_d = \mathbf{X}_\ell(x^M\mathbf{I}) - \sum_{k=0}^{M-1} \mathbf{E}_k x^k$.

Algorithm $\mathcal{V}$:

– For j from 1 to M, check
- $\|\mathbf{W}_j\| \leq t$,
- $\|\mathbf{Z}_{a_j}\| \leq (p+1)t$,
- $\|\mathbf{Z}_{b_j}\| \leq tp + t^2 n + t$,
- $\|\mathbf{Z}_d\| \leq \frac{t(p^{M+1}-1)}{p-1}$.
- $\mathbf{V}_{\ell_j}(x\mathbf{I}) + \mathbf{V}_{a_j} = \mathbf{H}\mathbf{W}_j + \mathbf{G}\mathbf{Z}_{a_j}$,
- $\mathbf{V}_{\ell_j}(x\mathbf{I} - \mathbf{W}_j) + \mathbf{V}_{b_j} = \mathbf{G}\mathbf{Z}_{b_j}$.

– Check $\sum_{i=0}^{N-1}(\mathbf{Y}_i \prod_{j=1}^{M} \mathbf{W}_{j,i_j}) + \sum_{k=0}^{M-1} \mathbf{V}_{d_k}(-x^k) = \mathbf{G}\mathbf{Z}_d$

We consider the properties of completeness, perfect $(M+1)$-special soundness, special honest verifier zero-knowledge of the protocol.

Completeness: The completeness has been shown in the proof of $\mathcal{LRS}$.

$(M+1)$-special soundness:

Suppose the adversary creates $M+1$ accepting responses $\mathbf{W}_1^{(0)}, \ldots, \mathbf{Z}_d^{(0)}, \ldots, \mathbf{W}_1^{(M)}, \ldots, \mathbf{Z}_d^{(M)}$ to $M+1$ different challenges $x^{(0)}, \ldots, x^{(M)}$ on the same initial message $\mathbf{V}_{\ell_1}, \ldots, \mathbf{V}_{d_0}, \ldots, \mathbf{V}_{\ell_M}, \ldots, \mathbf{V}_{d_{M-1}}$. The 2-special soundness of the Σ-protocol from appendix B.2 gives us opening of $\mathbf{V}_{\ell_1}, \ldots, \mathbf{V}_{\ell_M}$ of the form $\mathbf{V}_{\ell_j} = \mathbf{H}(\ell_j\mathbf{I}) + \mathbf{G}\mathbf{K}_j$ with $\ell_j \in \{0,1\}$. Since for any $u \in [0,M]$, $\mathbf{W}_j^{(u)}$, $\mathbf{Z}_{a_j}^{(u)}$, $x^{(u)}$ and the openings of $\mathbf{V}_{\ell_j}$ are known, we can obtain openings of $\mathbf{V}_{a_j}$ from the verification equation $\mathbf{V}_{\ell_j}(x\mathbf{I}) + \mathbf{V}_{a_j} = \mathbf{H}\mathbf{W}_j^{(u)} + \mathbf{G}\mathbf{Z}_{a_j}^{(u)}$. Consequently, we know the components to combine $\mathbf{W}_j^{(u)} = \ell_j x^{(u)}\mathbf{I} + \mathbf{B}_j$ for all $j \in [M]$ and $u \in [0,M]$. Using $\ell_j x^{(u)}$, $\mathbf{B}_j$ and ℓ_j for $j \in [M]$, we can reconstruct $\mathbf{W}_{j,1}^{(u)} = \ell_j x^{(0)}\mathbf{I} + \mathbf{B}_j$ and $\mathbf{W}_{j,0}^{(u)} = (1-\ell_j)x^{(u)}\mathbf{I} - \mathbf{B}_j$ for all $u \in [0,M]$. Following the last verification equation, we obtain for each $u \in [0,m]$

$$\sum_{i=0}^{N-1}(\mathbf{Y}_i \prod_{j=1}^{M} \mathbf{W}_{j,i_j}^{(u)}) + \sum_{k=0}^{M-1} \mathbf{V}_{d_k}(-(x^{(u)})^k) = \mathbf{G}\mathbf{Z}_d^{(u)} \quad ,$$

and the expression on the left can be ordered as $\mathbf{Y}_\ell(x^{(u)})^M - \sum_{i=0}^{M-1} \mathbf{GE}_i(x^{(u)})^i$. As it state in [12], $(1, (x^{(u)})^1, \cdots, (x^{(u)})^n)$ can be viewed as rows of a Vandermonde matrix. Since $x^{(0)}, \ldots, x^{(M)}$ are all different and x is invertible in R, the equation

$$\begin{pmatrix} 1 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ (x^{(0)})^M & \cdots & (x^{(M)})^M \end{pmatrix} \cdot \begin{pmatrix} \alpha_0 \\ \vdots \\ \alpha_M \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix} \tag{8}$$

has a unique solution for $\alpha_u \in R$, $u \in [0, M]$. Since

$$\sum_{u=0}^{M} \mathbf{GZ}_d^{(u)} \alpha_u = \sum_{u=0}^{M} (\mathbf{Y}_\ell(x^{(u)})^M - \sum_{i=0}^{M-1} \mathbf{GE}_i(x^{(u)})^i)\alpha_u = \mathbf{Y}_\ell = \mathbf{GX}_\ell\,,$$

we obtain $\mathbf{X}'_\ell$, by computing $\sum_{u=0}^{M} \mathbf{Z}_d^{(u)} \alpha_u$. Note that, the valid responses are of the form $\mathbf{Z}_d^{(u)} = \mathbf{X}_\ell((x^{(u)})^M \mathbf{I}) - \sum_{k=0}^{M-1} \mathbf{E}_k(x^{(u)})^k$. Therefore, the resulting matrix $\mathbf{X}'_\ell = \mathbf{X}_\ell$.

Special honest verifier zero-knowledge: Given the system parameters, the challenge x, and $\{\mathbf{Y}_i = \mathbf{GX}_i\}_{i=0}^{N-1}$, the simulator randomly chooses $\|\mathbf{W}_j\| \leq t$, $\|\mathbf{Z}_{a_j}\| \leq (p+1)t$, $\|\mathbf{Z}_{b_j}\| \leq tp + t^2 n + t$, $\|\mathbf{Z}_d\| \leq \frac{t(p^{M+1}-1)}{p-1}$, for $j \in [M]$. It then samples $\mathbf{K}_j \leftarrow Q^{m\times m}$ and $\mathbf{E}_k \leftarrow Q^{m\times m}$ and generates $\mathbf{V}_{\ell_j} = \mathbf{GK}_j$, $\mathbf{V}_{d_k} = \mathbf{GE}_k$ for $j \in [M]$ and $k \in [M-1]$. Subsequently, it computes $\mathbf{V}_{a_j} = \mathbf{V}_{\ell_j}(-x\mathbf{I}) + \mathbf{HW}_j + \mathbf{GZ}_{a_j}$ and $\mathbf{V}_{b_j} = V_{\ell_j}(x\mathbf{I} - W_j) + \mathbf{GZ}_{b_j}$ to finish the simulation of the proofs that $\mathbf{V}_{\ell_1}, \ldots, \mathbf{V}_{\ell_M}$ contain 0. Finally, it sets $\mathbf{V}_{d_0} = \sum_{i=0}^{N-1}(\mathbf{Y}_i \prod_{j=1}^{M} \mathbf{W}_{j,i_j}) + \sum_{k=1}^{M-1} \mathbf{V}_{d_k}(-x^k) - \mathbf{GZ}_d$ so that

$$\sum_{i=0}^{N-1}(\mathbf{Y}_i \prod_{j=1}^{M} \mathbf{W}_{j,i_j}) + \sum_{k=0}^{M-1} \mathbf{V}_{d_k}(-x^k) = \mathbf{GZ}_d$$

which satisfies the last verification equation.