# A Provably Secure Group Signature Scheme from Code-Based Assumptions

Martianus Frederic Ezerman, Hyung Tae Lee, San Ling, Khoa Nguyen(✉), and Huaxiong Wang

Division of Mathematical Sciences, School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore, Singapore
{fredezerman,hyungtaelee,lingsan,khoantt,hxwang}@ntu.edu.sg

**Abstract.** We solve an open question in code-based cryptography by introducing the first provably secure group signature scheme from code-based assumptions. Specifically, the scheme satisfies the CPA-anonymity and traceability requirements in the random oracle model, assuming the hardness of the McEliece problem, the Learning Parity with Noise problem, and a variant of the Syndrome Decoding problem. Our construction produces smaller key and signature sizes than the existing post-quantum group signature schemes from lattices, as long as the cardinality of the underlying group does not exceed the population of the Netherlands ($\approx 2^{24}$ users). The feasibility of the scheme is supported by implementation results. Additionally, the techniques introduced in this work might be of independent interest: a new verifiable encryption protocol for the randomized McEliece encryption and a new approach to design formal security reductions from the Syndrome Decoding problem.

## 1 Introduction

### 1.1 Background and Motivation

Group signature [CvH91] is a fundamental cryptographic primitive with two intriguing features: On the one hand, it allows users of a group to anonymously sign documents on behalf of the whole group (*anonymity*); On the other hand, there is a tracing authority that can tie a given signature to the signer's identity should the need arise (*traceability*). These two properties make group signatures highly useful in various real-life scenarios such as controlled anonymous printing services, digital right management systems, e-bidding and e-voting schemes. Theoretically, designing secure and efficient group signature schemes is of deep interest since doing so typically requires a sophisticated combination of carefully chosen cryptographic ingredients. Numerous constructions of group signatures have been proposed, most of which are based on classical number-theoretic assumptions (e.g., [CS97, ACJT00, BBS04, BW06, LPY12]).

While number-theoretic-based group signatures could be very efficient (e.g., [ACJT00, BBS04]), such schemes would become insecure once the era of scalable quantum computing arrives [Sho97]. The search for post-quantum group

© International Association for Cryptologic Research 2015
T. Iwata and J.H. Cheon (Eds.): ASIACRYPT 2015, Part I, LNCS 9452, pp. 260–285, 2015.
DOI: 10.1007/978-3-662-48797-6_12

signatures, as a preparation for the future, has been quite active recently, with 6 published schemes [GKV10, CNR12, LLLS13, LLNW14, LNW15, NZZ15], all of which are based on computational assumptions from lattices. Despite their theoretical interest, those schemes involve significantly large key and signature sizes, and no implementation result has been given. Our evaluation shows that the lattice-based schemes listed above are indeed very far from being practical (see Sect. 1.2). This somewhat unsatisfactory situation highlights two interesting challenges: First, making post-quantum group signatures one step closer to practice; Second, bringing in more diversity with a scheme from another candidate for post-quantum cryptography (e.g., code-based, hash-based, multivariate-based). For instance, an easy-to-implement and competitively efficient code-based group signature scheme would be highly desirable.

A code-based group signature, in the strongest security model for static groups [BMW03], would typically require the following 3 cryptographic layers:

1. The first layer requires a secure (standard) signature scheme to sign messages[1]. We observe that the existing code-based signatures fall into two categories.
   The "hash-and-sign" category consists of the CFS signature [CFS01] and its modified versions [Dal08, Fin10, MVR12]. The known security proofs for schemes in this category, however, should be viewed with skepticism: the assumption used in [Dal08] was invalidated by distinguishing attacks [FGUO+13], while the new assumption proposed in [MVR12] lies on a rather fragile ground.
   The "Fiat-Shamir" category consists of schemes derived from Stern's identification protocol [Ste96] and its variant [Vér96, CVA10, MGS11] via the Fiat-Shamir transformation [FS86]. Although these schemes produce relatively large signatures (as the underlying protocol has to be repeated many times to make the soundness error negligibly small), their provable security (in the random oracle model) is well-understood.
2. The second layer demands a semantically secure encryption scheme to enable the tracing feature: the signer is constrained to encrypt its identifying information and to send the ciphertext as part of the group signature, so that the tracing authority can decrypt if and when necessary. This ingredient is also available in code-based cryptography, thanks to various CPA-secure and CCA-secure variants of the McEliece [McE78] and the Niederreiter [Nie86] cryptosystems (e.g., [NIKM08, DDMN12, Per12, MVVR12]).
3. The third layer, which is essentially bottleneck in realizing secure code-based group signatures, requires a zero-knowledge (ZK) protocol that connects the first two layers. Specifically, the protocol should demonstrate that a given signature is generated by a certain certified group user who honestly encrypts its identifying information. Constructing such a protocol is quite challenging. There have been ZK protocols involving the CFS and Stern's signatures,

[1] In most schemes in the [BMW03] model, a standard signature is also employed to issue users' secret keys. However, this is not necessarily the case: the scheme constructed in this paper is an illustrative example.

which yield identity-based identification schemes [CGG07, ACM11, YTM+14] and threshold ring signatures [MCG08, MCGL11]. There also have been ZK proofs of plaintext knowledge for the McEliece and the Niederreiter cryptosystems [HMT13]. Yet we are not aware of any efficient ZK protocol that *simultaneously* deals with both code-based signature and encryption schemes in the above sense.

Designing a provably secure group signature scheme, thus, is a long-standing open question in code-based cryptography (see, e.g., [CM10]).

### 1.2 Our Contributions

In this work, we construct a group signature scheme which is provably secure under code-based assumptions. Specifically, the scheme achieves the anonymity and traceability requirements ([BMW03, BBS04]) in the random oracle model, assuming the hardness of the McEliece problem, the Learning Parity with Noise problem, and a variant of the Syndrome Decoding problem.

**Contributions to Code-Based Cryptography.** By introducing the first provably secure code-based group signature scheme, we solve the open problem discussed earlier. Along the way, we introduce two new techniques for code-based cryptography, which might be of independent interest:

1. We design a ZK protocol for the randomized McEliece encryption scheme, that allows the prover to convince the verifier that a given ciphertext is well-formed, and that the hidden plaintext satisfies an additional condition. Such protocols, called *verifiable encryption protocols*, are useful not only in constructing group signatures, but also in much broader contexts [CS03]. It is worth noting that, prior to our work, verifiable encryption protocols for code-based cryptosystems only exist in the very basic form [HMT13] (where the plaintext is publicly given), which seem to have restricted applications.
2. In our security proof of the traceability property, to obtain a reduction from the hardness of the Syndrome Decoding (SD) problem, we come up with an approach that, as far as we know, has not been considered in the literature before. Recall that the (average-case) SD problem with parameters $m, r, \omega$ is as follows: given a *uniformly random* matrix $\widetilde{\mathbf{H}} \in \mathbb{F}_2^{r \times m}$ and a *uniformly random* syndrome $\tilde{\mathbf{y}} \in \mathbb{F}_2^r$, the problem asks to find a vector $\mathbf{s} \in \mathbb{F}_2^m$ that has Hamming weight $\omega$ (denoted by $\mathbf{s} \in \mathsf{B}(m, \omega)$) such that $\widetilde{\mathbf{H}} \cdot \mathbf{s}^\top = \tilde{\mathbf{y}}^\top$. In our scheme, the key generation algorithm produces public key containing matrix $\mathbf{H} \in \mathbb{F}_2^{r \times m}$ and syndromes $\mathbf{y}_j \in \mathbb{F}_2^r$, while users are given secret keys of the form $\mathbf{s}_j \in \mathsf{B}(m, \omega)$ such that $\mathbf{H} \cdot \mathbf{s}_j^\top = \mathbf{y}_j^\top$. In the security proof, since we would like to embed an SD challenge instance $(\widetilde{\mathbf{H}}, \tilde{\mathbf{y}})$ into the public key without being noticed with non-negligible probability by the adversary, we have to require that $\mathbf{H}$ and the $\mathbf{y}_j$'s produced by the key generation are indistinguishable from uniform.

One method to generate these keys is to employ the "hash-and-sign" technique from the CFS signature [CFS01]. Unfortunately, while the syndromes $\mathbf{y}_j$'s could be made uniformly random (as the outputs of the random oracle), the assumption that the CFS matrix $\mathbf{H}$ is *computationally* close to uniform (for practical parameters) is invalidated by distinguishing attacks [FGUO+13].

Another method, pioneered by Stern [Ste96], is to pick $\mathbf{H}$ and the $\mathbf{s}_j$'s uniformly at random. The corresponding syndromes $\mathbf{y}_j$'s could be made *computationally* close to uniform if the parameters are set such that $\omega$ is slightly smaller than the value $\omega_0$ given by the Gilbert-Varshamov bound[2], i.e., $\omega_0$ such that $\binom{m}{\omega_0} \approx 2^r$. However, for these parameters, it is not guaranteed with high probability that a uniformly random $\mathsf{SD}$ instance $(\widetilde{\mathbf{H}}, \tilde{\mathbf{y}})$ has solutions, which would affect the success probability of the reduction algorithm.

In this work, we consider the case when $\omega$ is moderately larger than $\omega_0$, so that two conditions hold: First, the uniform distribution over the set $\mathsf{B}(m, \omega)$ has sufficient min-entropy to apply the left-over hash lemma [GKPV10]; Second, the $\mathsf{SD}$ problem with parameters $(m, r, \omega)$ admits solutions with high probability, yet remains intractable[3] against the best known attacks [FS09, BJMM12]. This gives us a new method to generate uniformly random vectors $\mathbf{s}_j \in \mathsf{B}(m, \omega)$ and matrix $\mathbf{H} \in \mathbb{F}_2^{r \times m}$ so that the syndromes $\mathbf{y}_j$'s corresponding to the $\mathbf{s}_j$'s are *statistically* close to uniform. This approach, which somewhat resembles the technique used in [GPV08] for the Inhomogeneous Small Integer Solution problem, is helpful in our security proof (and generally, in designing formal security reductions from the $\mathsf{SD}$ problem).

**Contributions to Post-Quantum Group Signatures.** Our construction provides the first non-lattice-based alternative to provably secure post-quantum group signatures. The scheme features public key and signature sizes linear in the number of group users $N$, which is asymptotically not as efficient as the recently published lattice-based counterparts ([LLLS13, LLNW14, LNW15, NZZ15]). However, when instantiating with practical parameters, our scheme behaves much more efficiently than the scheme proposed in [NZZ15] (which is arguably the current most efficient lattice-based group signature in the asymptotic sense). Indeed, our estimation shows that our scheme gives public key and signature sizes that are 2300 times and 540 times smaller, respectively, for an average-size group of $N = 2^8$ users. As $N$ grows, the advantage lessens, but our scheme remains more efficient even for a huge group of $N = 2^{24}$ users (which is comparable to the whole population of the Netherlands). The details of our estimation are given in Table 1.

Furthermore, we give implementation results - the first ones for post-quantum group signatures - to support the feasibility of our scheme (see Sect. 5). Our

[2] In this case, the function $f_{\mathbf{H}}(\mathbf{s}_j) = \mathbf{H} \cdot \mathbf{s}_j^\top$ acts as a pseudorandom generator [FS96].

[3] The variant of the $\mathsf{SD}$ problem considered in this work are not widely believed to be the hardest one [Ste96, Meu13], but suitable parameters can be chosen (e.g., see Sect. 5) such that the best known attacks run in exponential time.

**Table 1.** Efficiency comparison between our scheme and [NZZ15].

| | $N$ | Public key size | | Signature size[b] | |
|---|---|---|---|---|---|
| Our scheme | $2^8$ | $5.13 \times 10^6$ bits | (642 KB) | $8.57 \times 10^6$ bits | (1.07 MB) |
| | $2^{16}$ | $4.10 \times 10^7$ bits | (5.13 MB) | $1.77 \times 10^7$ bits | (2.21 MB) |
| | $2^{24}$ | $9.23 \times 10^9$ bits | (1.16 GB) | $2.36 \times 10^9$ bits | (294 MB) |
| [NZZ15][a] | $\leq 2^{24}$ | $1.18 \times 10^{10}$ bits | (1.48 GB) | $4.63 \times 10^9$ bits | (579 MB) |

[a]The parameters of our scheme are set as in Sect. 5. For the [NZZ15] scheme, we choose the commonly used lattice dimension $n = 2^8$, and set parameters $m = 2^9 \times 150$ and $q = 2^{150}$ so that the requirements given in [NZZ15, Section 5.1] are satisfied. Both schemes achieve the CPA-anonymity notion [BBS04] and soundness error $2^{-80}$.

[b]In our implementations presented in Sect. 5, the actual signature sizes could be reduced thanks to an additional technique.

results, while not yielding a truly practical scheme, would certainly help to bring post-quantum group signatures one step closer to practice.

### 1.3 Overview of Our Techniques

Let $m, r, \omega, n, k, t$ and $\ell$ be positive integers. We consider a group of size $N = 2^\ell$, where each user is indexed by an integer $j \in [0, N-1]$. The secret signing key of user $j$ is a vector $\mathbf{s}_j$ chosen uniformly at random from the set $\mathsf{B}(m, \omega)$. A uniformly random matrix $\mathbf{H} \in \mathbb{F}_2^{r \times m}$ and $N$ syndromes $\mathbf{y}_0, \ldots, \mathbf{y}_{N-1} \in \mathbb{F}_2^r$, such that $\mathbf{H} \cdot \mathbf{s}_j^\top = \mathbf{y}_j^\top$, for all $j$, are made public. Let us now explain the development of the 3 ingredients used in our scheme.

**The Signature Layer.** User $j$ can run Stern's ZK protocol [Ste96] to prove the possession of a vector $\mathbf{s} \in \mathsf{B}(m, \omega)$ such that $\mathbf{H} \cdot \mathbf{s}^\top = \mathbf{y}_j^\top$, where the constraint $\mathbf{s} \in \mathsf{B}(m, \omega)$ is proved in ZK by randomly permuting the entries of $\mathbf{s}$ and showing that the permuted vector belongs to $\mathsf{B}(m, \omega)$. The protocol is then transformed into a Fiat-Shamir signature [FS86]. However, such a signature is publicly verifiable only if the index $j$ is given to the verifier.

The user can further hide its index $j$ to achieve unconditional anonymity among all $N$ users (which yields a *ring signature* [RST01] on the way, a la [BS13]), as follows. Let $\mathbf{A} = \left[\mathbf{y}_0^\top | \cdots | \mathbf{y}_j^\top | \cdots | \mathbf{y}_{N-1}^\top\right] \in \mathbb{F}_2^{r \times N}$ and let $\mathbf{x} = \delta_j^N$ - the $N$-dimensional unit vector with entry 1 at the $j$-th position. Observe that $\mathbf{A} \cdot \mathbf{x}^\top = \mathbf{y}_j^\top$, and thus, the equation $\mathbf{H} \cdot \mathbf{s}^\top = \mathbf{y}_j^\top$ can be written as

$$\mathbf{H} \cdot \mathbf{s}^\top \oplus \mathbf{A} \cdot \mathbf{x}^\top = \mathbf{0}, \tag{1}$$

where $\oplus$ denotes addition modulo 2. Stern's framework allows the user to prove in ZK the possession of $(\mathbf{s}, \mathbf{x})$ satisfying this equation, where the condition $\mathbf{x} = \delta_j^N$ can be justified using a random permutation.

**The Encryption Layer.** To enable the tracing capability of the scheme, we let user $j$ encrypt the binary representation of $j$ via the randomized McEliece

encryption scheme [NIKM08]. Specifically, we represent $j$ as vector $\mathsf{I2B}(j) = (j_0, \ldots, j_{\ell-1}) \in \{0,1\}^\ell$, where $\sum_{i=0}^{\ell-1} j_i 2^{\ell-1-i} = j$. Given a public encrypting key $\mathbf{G} \in \mathbb{F}_2^{k \times n}$, a ciphertext of $\mathsf{I2B}(j)$ is of the form:

$$\mathbf{c} = \big(\mathbf{u} \,\|\, \mathsf{I2B}(j)\big) \cdot \mathbf{G} \oplus \mathbf{e} \in \mathbb{F}_2^n, \tag{2}$$

where $(\mathbf{u}, \mathbf{e})$ is the encryption randomness, with $\mathbf{u} \in \mathbb{F}_2^{k-\ell}$, and $\mathbf{e} \in \mathsf{B}(n,t)$ (i.e., $\mathbf{e}$ is a vector in $\mathbb{F}_2^n$, that has weight $t$).

**Connecting the Signature and Encryption Layers.** User $j$ must demonstrate that it does not cheat (e.g., by encrypting some string that does not point to $j$) without revealing $j$. Thus, we need a $\mathsf{ZK}$ protocol that allows the user to prove that the vector $\mathbf{x} = \delta_j^N$ used in (1) and the plaintext hidden in (2) both correspond to the same secret $j \in [0, N-1]$. The crucial challenge is to establish a connection (which is verifiable in $\mathsf{ZK}$) between the "index representation" $\delta_j^N$ and the binary representation $\mathsf{I2B}(j)$. This challenge is well-handled by the following technique.

Instead of working with $\mathsf{I2B}(j) = (j_0, \ldots, j_{\ell-1})$, we consider an extension of $\mathsf{I2B}(j)$, defined as $\mathsf{Encode}(j) = (1-j_0, j_0, \ldots, 1-j_i, j_i, \ldots, 1-j_{\ell-1}, j_{\ell-1}) \in \mathbb{F}_2^{2\ell}$. We then suitably insert $\ell$ zero-rows into matrix $\mathbf{G}$ to obtain matrix $\widehat{\mathbf{G}} \in \mathbb{F}_2^{(k+\ell)\times n}$ such that $\big(\mathbf{u} \,\|\, \mathsf{Encode}(j)\big) \cdot \widehat{\mathbf{G}} = \big(\mathbf{u} \,\|\, \mathsf{I2B}(j)\big) \cdot \mathbf{G}$. Let $\mathbf{f} = \mathsf{Encode}(j)$, then equation (2) can be rewritten as:

$$\mathbf{c} = \big(\mathbf{u} \,\|\, \mathbf{f}\big) \cdot \widehat{\mathbf{G}} \oplus \mathbf{e} \in \mathbb{F}_2^n. \tag{3}$$

Now, let $\mathsf{B2I} : \{0,1\}^\ell \to [0, N-1]$ be the inverse function of $\mathsf{I2B}(\cdot)$. For every $\mathbf{b} \in \{0,1\}^\ell$, we carefully design two classes of permutations $T_{\mathbf{b}} : \mathbb{F}_2^N \to \mathbb{F}_2^N$ and $T'_{\mathbf{b}} : \mathbb{F}_2^{2\ell} \to \mathbb{F}_2^{2\ell}$, such that for any $j \in [0, N-1]$, the following hold:

$$\mathbf{x} = \delta_j^N \iff T_{\mathbf{b}}(\mathbf{x}) = \delta^N_{\mathsf{B2I}(\mathsf{I2B}(j)\oplus\mathbf{b})};$$
$$\mathbf{f} = \mathsf{Encode}(j) \iff T'_{\mathbf{b}}(\mathbf{f}) = \mathsf{Encode}(\mathsf{B2I}(\mathsf{I2B}(j) \oplus \mathbf{b})).$$

Given these equivalences, in the protocol, the user samples a uniformly random vector $\mathbf{b} \in \{0,1\}^\ell$, and sends $\mathbf{b}_1 = \mathsf{I2B}(j) \oplus \mathbf{b}$. The verifier, seeing that $T_{\mathbf{b}}(\mathbf{x}) = \delta^N_{\mathsf{B2I}(\mathbf{b}_1)}$ and $T'_{\mathbf{b}}(\mathbf{f}) = \mathsf{Encode}(\mathsf{B2I}(\mathbf{b}_1))$, should be convinced that $\mathbf{x}$ and $\mathbf{f}$ correspond to the same $j \in [0, N-1]$, yet the value of $j$ is completely hidden from its view, because vector $\mathbf{b}$ essentially acts as a "one-time pad".

The technique extending $\mathsf{I2B}(j)$ into $\mathsf{Encode}(j)$ and then permuting $\mathsf{Encode}(j)$ in a "one-time pad" fashion is inspired by a method originally proposed by Langlois et al. [LLNW14] in a seemingly unrelated context, where the goal is to prove that the message being signed under the Bonsai tree signature [CHKP10] is of the form $\mathsf{I2B}(j)$, for some $j \in [0, N-1]$. Here, we adapt and develop their method to *simultaneously* prove two facts: the plaintext being encrypted under the randomized McEliece encryption is of the form $\mathsf{I2B}(j)$, and the unit vector $\mathbf{x} = \delta_j^N$ is used in the signature layer.

By embedding the above technique into Stern's framework, we obtain an interactive $\mathsf{ZK}$ argument system, in which, given the public input $(\mathbf{H}, \mathbf{A}, \mathbf{G})$, the

user is able to prove the possession of a secret tuple $(j, \mathbf{s}, \mathbf{x}, \mathbf{u}, \mathbf{f}, \mathbf{e})$ satisfying (1) and (3). The protocol is repeated many times to achieve negligible soundness error, and then made non-interactive, resulting in a non-interactive ZK argument of knowledge $\Pi$. The final group signature is of the form $(\mathbf{c}, \Pi)$, where $\mathbf{c}$ is the ciphertext. In the random oracle model, the anonymity of the scheme relies on the zero-knowledge property of $\Pi$ and the CPA-security of the randomized McEliece encryption scheme, while its traceability is based on the hardness of the variant of the SD problem discussed earlier.

### 1.4 Related Works and Open Questions

A group signature scheme based on the security of the ElGamal signature scheme and the hardness of decoding of linear codes was given in [MCK01]. In a concurrent and independent work, Alamélou et al. [ABCG15] also propose a code-based group signature scheme. These two works have yet to provide a *provably secure* group signature scheme based *solely* on code-based assumptions, which we achieve in the present paper.

Our work constitutes a foundational step in code-based group signatures. In the next steps, we will work towards improving the current construction in terms of efficiency (e.g., making the signature size less dependent on the number of group users), as well as functionality (e.g., achieving dynamic enrollment and efficient revocation of users). Another interesting open question is to construct a scheme achieving CCA-anonymity.

## 2 Preliminaries

Notations. We let $\lambda$ denote the security parameter and $\mathsf{negl}(\lambda)$ denote a negligible function in $\lambda$. We denote by $a \xleftarrow{\$} A$ if $a$ is chosen uniformly at random from the finite set $A$. The symmetric group of all permutations of $k$ elements is denoted by $\mathsf{S}_k$. We use bold capital letters, (e.g., $\mathbf{A}$), to denote matrices, and bold lowercase letters, (e.g., $\mathbf{x}$), to denote row vectors. We use $\mathbf{x}^\top$ to denote the transpose of $\mathbf{x}$ and $wt(\mathbf{x})$ to denote the (Hamming) weight of $\mathbf{x}$. We denote by $\mathsf{B}(m, \omega)$ the set of all vectors $\mathbf{x} \in \mathbb{F}_2^m$ such that $wt(\mathbf{x}) = \omega$. Throughout the paper, we define a function I2B which takes a non-negative integer $a$ as an input, and outputs the binary representation $(a_0, \cdots, a_{\ell-1}) \in \{0,1\}^\ell$ of $a$ such that $a = \sum_{i=0}^{\ell-1} a_i 2^{\ell-1-i}$, and a function B2I which takes as an input the binary representation $(a_0, \cdots, a_{\ell-1}) \in \{0,1\}^\ell$ of $a$, and outputs $a$. All logarithms are of base 2.

### 2.1 Background on Code-Based Cryptography

We first recall the Syndrome Decoding problem, which is well-known to be NP-complete [BMvT78], and is widely believed to be intractable in the average case for appropriate choice of parameters [Ste96, Meu13].

**Definition 1 (The Syndrome Decoding problem).** *The* $\mathsf{SD}(m, r, \omega)$ *problem is as follows: given a uniformly random matrix* $\mathbf{H} \in \mathbb{F}_2^{r \times m}$ *and a uniformly random syndrome* $\mathbf{y} \in \mathbb{F}_2^r$*, find a vector* $\mathbf{s} \in \mathsf{B}(m, \omega)$ *such that* $\mathbf{H} \cdot \mathbf{s}^\top = \mathbf{y}^\top$.

*When* $m = m(\lambda), r = r(\lambda), \omega = \omega(\lambda)$*, we say that the* $\mathsf{SD}(m, r, \omega)$ *problem is hard, if the success probability of any* PPT *algorithm in solving the problem is at most* $\mathsf{negl}(\lambda)$.

In our security reduction, the following variant of the left-over hash lemma for matrix multiplication over $\mathbb{F}_2$ is used.

**Lemma 1 (Left-over hash lemma, adapted from [GKPV10]).** *Let* $D$ *be a distribution over* $\mathbb{F}_2^m$ *with min-entropy* $e$*. For* $\epsilon > 0$ *and* $r \leq e - 2\log(1/\epsilon) - \mathcal{O}(1)$*, the statistical distance between the distribution of* $(\mathbf{H}, \mathbf{H} \cdot \mathbf{s}^\top)$*, where* $\mathbf{H} \xleftarrow{\$} \mathbb{F}_2^{r \times m}$ *and* $\mathbf{s} \in \mathbb{F}_2^m$ *is drawn from distribution* $D$*, and the uniform distribution over* $\mathbb{F}_2^{r \times m} \times \mathbb{F}_2^r$ *is at most* $\epsilon$.

*In particular, if* $\omega < m$ *is an integer such that* $r \leq \log \binom{m}{\omega} - 2\lambda - \mathcal{O}(1)$ *and* $D$ *is the uniform distribution over* $\mathsf{B}(m, \omega)$ *(i.e.,* $D$ *has min-entropy* $\log \binom{m}{\omega}$*), then the statistical distance between the distribution of* $(\mathbf{H}, \mathbf{H} \cdot \mathbf{s}^\top)$ *and the uniform distribution over* $\mathbb{F}_2^{r \times m} \times \mathbb{F}_2^r$ *is at most* $2^{-\lambda}$.

**The Randomized McEliece Encryption Scheme.** We employ a randomized variant of the McEliece [McE78] encryption scheme, suggested in [NIKM08], where a uniformly random vector is concatenated to the plaintext. The scheme is described as follows:

- $\mathsf{ME.Setup}(1^\lambda)$: Select parameters $n = n(\lambda), k = k(\lambda), t = t(\lambda)$ for a binary $[n, k, 2t + 1]$ Goppa code. Choose integers $k_1$, $k_2$ such that $k = k_1 + k_2$. Set the plaintext space as $\mathbb{F}_2^{k_2}$.
- $\mathsf{ME.KeyGen}(n, k, t)$: Perform the following steps:
  1. Produce a generator matrix $\mathbf{G}' \in \mathbb{F}_2^{k \times n}$ of a randomly selected $[n, k, 2t+1]$ Goppa code. Choose a random invertible matrix $\mathbf{S} \in \mathbb{F}_2^{k \times k}$ and a random permutation matrix $\mathbf{P} \in \mathbb{F}_2^{n \times n}$. Let $\mathbf{G} = \mathbf{S}\mathbf{G}'\mathbf{P} \in \mathbb{F}_2^{k \times n}$.
  2. Output encrypting key $\mathsf{pk_{ME}} = \mathbf{G}$ and decrypting key $\mathsf{sk_{ME}} = (\mathbf{S}, \mathbf{G}', \mathbf{P})$.
- $\mathsf{ME.Enc}(\mathsf{pk_{ME}}, \mathbf{m})$: To encrypt a message $\mathbf{m} \in \mathbb{F}_2^{k_2}$, sample $\mathbf{u} \xleftarrow{\$} \mathbb{F}_2^{k_1}$ and $\mathbf{e} \xleftarrow{\$} \mathsf{B}(n, t)$, then output the ciphertext $\mathbf{c} = (\mathbf{u} \| \mathbf{m}) \cdot \mathbf{G} \oplus \mathbf{e} \in \mathbb{F}_2^n$.
- $\mathsf{ME.Dec}(\mathsf{sk_{ME}}, \mathbf{c})$: Perform the following steps:
  1. Compute $\mathbf{c} \cdot \mathbf{P}^{-1} = ((\mathbf{u} \| \mathbf{m}) \cdot \mathbf{G} \oplus \mathbf{e}) \cdot \mathbf{P}^{-1}$ and then $\mathbf{m}' \cdot \mathbf{S} = \mathit{Decode}_{\mathbf{G}'}(\mathbf{c} \cdot \mathbf{P}^{-1})$ where *Decode* is an error-correcting algorithm with respect to $\mathbf{G}'$. If *Decode* fails, then return $\perp$.
  2. Compute $\mathbf{m}' = (\mathbf{m}'\mathbf{S}) \cdot \mathbf{S}^{-1}$, parse $\mathbf{m}' = (\mathbf{u} \| \mathbf{m})$, where $\mathbf{u} \in \mathbb{F}_2^{k_1}$ and $\mathbf{m} \in \mathbb{F}_2^{k_2}$, and return $\mathbf{m}$.

The scheme described above is $\mathsf{CPA}$-secure in the standard model assuming the hardness of the $\mathsf{DMcE}(n, k, t)$ problem and the $\mathsf{DLPN}(k_1, n, \mathsf{B}(n, t))$ problem [NIKM08, Döt14]. We now recall these two problems.

**Definition 2 (The Decisional McEliece problem).** *The* $\mathsf{DMcE}(n,k,t)$ *problem is as follows: given a matrix* $\mathbf{G} \in \mathbb{F}_2^{k\times n}$*, distinguish whether* $\mathbf{G}$ *is a uniformly random matrix over* $\mathbb{F}_2^{k\times n}$ *or it is generated by algorithm* $\mathsf{ME.KeyGen}(n,k,t)$ *described above.*

*When* $n = n(\lambda), k = k(\lambda), t = t(\lambda)$*, we say that the* $\mathsf{DMcE}(n,k,t)$ *problem is hard, if the success probability of any* PPT *distinguisher is at most* $1/2 + \mathsf{negl}(\lambda)$*.*

**Definition 3 (The Decisional Learning Parity with (fixed-weight) Noise problem).** *The* $\mathsf{DLPN}(k,n,\mathsf{B}(n,t))$ *problem is as follows: given a pair* $(\mathbf{A},\mathbf{v}) \in \mathbb{F}_2^{k\times n} \times \mathbb{F}_2^n$*, distinguish whether* $(\mathbf{A},\mathbf{v})$ *is a uniformly random pair over* $\mathbb{F}_2^{k\times n} \times \mathbb{F}_2^n$ *or it is obtained by choosing* $\mathbf{A} \xleftarrow{\$} \mathbb{F}_2^{k\times n}$*,* $\mathbf{u} \xleftarrow{\$} \mathbb{F}_2^k$*,* $\mathbf{e} \xleftarrow{\$} \mathsf{B}(n,t)$ *and outputting* $(\mathbf{A}, \mathbf{u}\cdot\mathbf{A} \oplus \mathbf{e})$*.*

*When* $k = k(\lambda), n = n(\lambda), t = t(\lambda)$*, we say that the* $\mathsf{DLPN}(k,n,\mathsf{B}(n,t))$ *problem is hard, if the success probability of any* PPT *distinguisher is at most* $1/2 + \mathsf{negl}(\lambda)$*.*

### 2.2 Group Signatures

We follow the definition of group signatures provided in [BMW03] for the case of static groups.

**Definition 4.** A group signature $\mathcal{GS} = (\mathsf{KeyGen}, \mathsf{Sign}, \mathsf{Verify}, \mathsf{Open})$ is a tuple of four polynomial-time algorithms:

- $\mathsf{KeyGen}$: This randomized algorithm takes as input $(1^\lambda, 1^N)$, where $N \in \mathbb{N}$ is the number of group users, and outputs $(\mathsf{gpk}, \mathsf{gmsk}, \mathsf{gsk})$, where $\mathsf{gpk}$ is the group public key, $\mathsf{gmsk}$ is the group manager's secret key, and $\mathsf{gsk} = \{\mathsf{gsk}[j]\}_{j\in[0,N-1]}$ with $\mathsf{gsk}[j]$ being the secret key for the group user of index $j$.
- $\mathsf{Sign}$: This randomized algorithm takes as input a secret signing key $\mathsf{gsk}[j]$ for some $j \in [0, N-1]$ and a message $M$ and returns a group signature $\Sigma$ on $M$.
- $\mathsf{Verify}$: This deterministic algorithm takes as input the group public key $\mathsf{gpk}$, a message $M$, a signature $\Sigma$ on $M$, and returns either 1 ($\mathsf{Accept}$) or 0 ($\mathsf{Reject}$).
- $\mathsf{Open}$: This deterministic algorithm takes as input the group manager's secret key $\mathsf{gmsk}$, a message $M$, a signature $\Sigma$ on $M$, and returns an index $j \in [0, N-1]$ associated with a particular user, or $\perp$, indicating failure.

$\mathsf{Correctness}$. The correctness of a group signature scheme requires that for all $\lambda, N \in \mathbb{N}$, all $(\mathsf{gpk}, \mathsf{gmsk}, \mathsf{gsk})$ produced by $\mathsf{KeyGen}(1^\lambda, 1^N)$, all $j \in [0, N-1]$, and all messages $M \in \{0,1\}^*$,

$$\mathsf{Verify}\big(\mathsf{gpk}, M, \mathsf{Sign}(\mathsf{gsk}[j], M)\big) = 1; \quad \mathsf{Open}\big(\mathsf{gmsk}, M, \mathsf{Sign}(\mathsf{gsk}[j], M)\big) = j.$$

**Security Notions.** A secure group signature scheme must satisfy two security notions:

- *Traceability* requires that all signatures, even those produced by a coalition of group users and the group manager, can be traced back to a member of the coalition.

– *Anonymity* requires that, signatures generated by two distinct group users are computationally indistinguishable to an adversary who knows all the user secret keys. In Bellare et al.'s model [BMW03], the anonymity adversary is granted access to an opening oracle (CCA-anonymity). Boneh et al. [BBS04] later proposed a relaxed notion, where the adversary cannot query the opening oracle (CPA-anonymity).

Formal definitions of CPA-anonymity and traceability are as follows.

**Definition 5.** *We say that a group signature* $\mathcal{GS} = (\mathsf{KeyGen}, \mathsf{Sign}, \mathsf{Verify}, \mathsf{Open})$ *is* CPA*-anonymous if for all polynomial* $N(\cdot)$ *and any PPT adversaries* $\mathcal{A}$*, the advantage of* $\mathcal{A}$ *in the following experiment is negligible in* $\lambda$*:*

1. *Run* $(\mathsf{gpk}, \mathsf{gmsk}, \mathsf{gsk}) \leftarrow \mathsf{KeyGen}(1^\lambda, 1^N)$ *and send* $(\mathsf{gpk}, \mathsf{gsk})$ *to* $\mathcal{A}$*.*
2. $\mathcal{A}$ *outputs two identities* $j_0, j_1 \in [0, N-1]$ *with a message* $M$*. Choose a random bit* $b$ *and give* $\mathsf{Sign}(\mathsf{gsk}[j_b], M)$ *to* $\mathcal{A}$*. Then,* $\mathcal{A}$ *outputs a bit* $b'$*.*

$\mathcal{A}$ *succeeds if* $b' = b$*, and the advantage of* $\mathcal{A}$ *is defined to* $\left|\Pr[\mathcal{A} \text{ succeeds}] - \frac{1}{2}\right|$.

**Definition 6.** *We say that a group signature* $\mathcal{GS} = (\mathsf{KeyGen}, \mathsf{Sign}, \mathsf{Verify}, \mathsf{Open})$ *is traceable if for all polynomial* $N(\cdot)$ *and any PPT adversaries* $\mathcal{A}$*, the success probability of* $\mathcal{A}$ *in the following experiment is negligible in* $\lambda$*:*

1. *Run* $(\mathsf{gpk}, \mathsf{gmsk}, \mathsf{gsk}) \leftarrow \mathsf{KeyGen}(1^\lambda, 1^N)$ *and send* $(\mathsf{gpk}, \mathsf{gmsk})$ *to* $\mathcal{A}$*.*
2. $\mathcal{A}$ *may query the following oracles adaptively and in any order:*
   - *A* $\mathcal{O}^{\mathsf{Corrupt}}$ *oracle that on input* $j \in [0, N-1]$*, outputs* $\mathsf{gsk}[j]$*.*
   - *A* $\mathcal{O}^{\mathsf{Sign}}$ *oracle that on input* $j$*, a message* $M$*, returns* $\mathsf{Sign}(\mathsf{gsk}[j], M)$*.*

   *Let* $CU$ *be the set of identities queried to* $\mathcal{O}^{\mathsf{Corrupt}}$*.*
3. *Finally,* $\mathcal{A}$ *outputs a message* $M^*$ *and a signature* $\Sigma^*$*.*

$\mathcal{A}$ *succeeds if (1)* $\mathsf{Verify}(\mathsf{gpk}, M^*, \Sigma^*) = 1$ *and (2)* $\mathsf{Sign}(\mathsf{gsk}[j], M^*)$ *was never queried for* $j \notin CU$*, yet (3)* $\mathsf{Open}(\mathsf{gmsk}, M^*, \Sigma^*) \notin CU$*.*

## 3 The Underlying Zero-Knowledge Argument System

Recall that a statistical zero-knowledge argument system is an interactive protocol where the soundness property holds for *computationally bounded* cheating provers, while the zero-knowledge property holds against *any* cheating verifier. In this section we present a statistical zero-knowledge argument system which will serve as a building block in our group signature scheme in Sect. 4.

Before describing the protocol, we first introduce several supporting notations and techniques. Let $\ell$ be a positive integer, and let $N = 2^\ell$.

1. For $\mathbf{x} = (x_0, x_1, \ldots, x_{N-1}) \in \mathbb{F}_2^N$ and for $j \in [0, N-1]$, we denote by $\mathbf{x} = \delta_j^N$ if $x_j = 1$ and $x_i = 0$ for all $i \neq j$.

2. We define an encoding function $\mathsf{Encode} : [0, N-1] \to \mathbb{F}_2^{2\ell}$, that encodes integer $j \in [0, N-1]$, whose binary representation is $\mathsf{I2B}(j) = (j_0, \ldots, j_{\ell-1})$, as vector:

$$\mathsf{Encode}(j) = (1 - j_0, j_0, \ldots, 1 - j_i, j_i, \ldots, 1 - j_{\ell-1}, j_{\ell-1}).$$

3. Given a vector $\mathbf{b} = (b_0, \ldots, b_{\ell-1}) \in \{0,1\}^\ell$, we define the following 2 permutations:
   (a) $T_{\mathbf{b}} : \mathbb{F}_2^N \to \mathbb{F}_2^N$ that transforms $\mathbf{x} = (x_0, \ldots, x_{N-1})$ to $(x'_0, \ldots, x'_{N-1})$, where for each $i \in [0, N-1]$, we have $x_i = x'_{i^*}$, where $i^* = \mathsf{B2I}\big(\mathsf{I2B}(i) \oplus \mathbf{b}\big)$.
   (b) $T'_{\mathbf{b}} : \mathbb{F}_2^{2\ell} \to \mathbb{F}_2^{2\ell}$ that transforms $\mathbf{f} = (f_0, f_1, \ldots, f_{2i}, f_{2i+1}, \ldots, f_{2(\ell-1)}, f_{2(\ell-1)+1})$ to $(f_{b_0}, f_{1-b_0}, \ldots, f_{2i+b_i}, f_{2i+(1-b_i)}, \ldots, f_{2(\ell-1)+b_{\ell-1}}, f_{2(\ell-1)+(1-b_{\ell-1})})$.

Observe that, for any $j \in [0, N-1]$ and any $\mathbf{b} \in \{0,1\}^\ell$, we have:

$$\mathbf{x} = \delta_j^N \iff T_{\mathbf{b}}(\mathbf{x}) = \delta^N_{\mathsf{B2I}(\mathsf{I2B}(j) \oplus \mathbf{b})}; \tag{4}$$

$$\mathbf{f} = \mathsf{Encode}(j) \iff T'_{\mathbf{b}}(\mathbf{f}) = \mathsf{Encode}(\mathsf{B2I}(\mathsf{I2B}(j) \oplus \mathbf{b})). \tag{5}$$

**Example:** Let $N = 2^4$. Let $j = 6$, then $\mathsf{I2B}(j) = (0,1,1,0)$ and $\mathsf{Encode}(j) = (1,0,0,1,0,1,1,0)$. If $\mathbf{b} = (1,0,1,0)$, then $\mathsf{B2I}(\mathsf{I2B}(j) \oplus \mathbf{b}) = \mathsf{B2I}(1,1,0,0) = 12$, and we have:

$$T_{\mathbf{b}}(\delta_6^{16}) = \delta_{12}^{16} \text{ and } T'_{\mathbf{b}}(\mathsf{Encode}(6)) = (0,1,0,1,1,0,1,0) = \mathsf{Encode}(12).$$

### 3.1 The Interactive Protocol

We now present our interactive zero-knowledge argument of knowledge ($\mathsf{ZKAoK}$). Let $n, k, t, m, r, \omega, \ell$ be positive integers, and $N = 2^\ell$. The public input consists of matrices $\mathbf{G} \in \mathbb{F}_2^{k \times n}$, $\mathbf{H} \in \mathbb{F}_2^{r \times m}$; $N$ syndromes $\mathbf{y}_0, \ldots, \mathbf{y}_{N-1} \in \mathbb{F}_2^r$; and a vector $\mathbf{c} \in \mathbb{F}_2^n$. The protocol allows prover $\mathcal{P}$ to *simultaneously* convince verifier $\mathcal{V}$ in zero-knowledge that $\mathcal{P}$ possesses a vector $\mathbf{s} \in \mathsf{B}(m, \omega)$ corresponding to certain syndrome $\mathbf{y}_j \in \{\mathbf{y}_0, \ldots, \mathbf{y}_{N-1}\}$ with hidden index $j$, *and* that $\mathbf{c}$ is a correct encryption of $\mathsf{I2B}(j)$ via the randomized McEliece encryption. Specifically, the secret witness of $\mathcal{P}$ is a tuple $(j, \mathbf{s}, \mathbf{u}, \mathbf{e}) \in [0, N-1] \times \mathbb{F}_2^m \times \mathbb{F}_2^{k-\ell} \times \mathbb{F}_2^n$ satisfying:

$$\begin{cases} \mathbf{H} \cdot \mathbf{s}^\top = \mathbf{y}_j^\top \ \ \wedge \ \ \mathbf{s} \in \mathsf{B}(m, \omega); \\ \big(\mathbf{u} \,\|\, \mathsf{I2B}(j)\big) \cdot \mathbf{G} \oplus \mathbf{e} = \mathbf{c} \ \ \wedge \ \ \mathbf{e} \in \mathsf{B}(n, t). \end{cases} \tag{6}$$

Let $\mathbf{A} = \big[\mathbf{y}_0^\top | \cdots | \mathbf{y}_j^\top | \cdots | \mathbf{y}_{N-1}^\top\big] \in \mathbb{F}_2^{r \times N}$ and $\mathbf{x} = \delta_j^N$. We have $\mathbf{A} \cdot \mathbf{x}^\top = \mathbf{y}_j^\top$, and thus, the equation $\mathbf{H} \cdot \mathbf{s}^\top = \mathbf{y}_j^\top$ can be written as $\mathbf{H} \cdot \mathbf{s}^\top \oplus \mathbf{A} \cdot \mathbf{x}^\top = \mathbf{0}$.

Let $\widehat{\mathbf{G}} \in \mathbb{F}_2^{(k+\ell) \times n}$ be the matrix obtained from $\mathbf{G} \in \mathbb{F}_2^{k \times n}$ by replacing its last $\ell$ rows $\mathbf{g}_{k-\ell+1}, \mathbf{g}_{k-\ell+2}, \ldots, \mathbf{g}_k$ by $2\ell$ rows $\mathbf{0}^n, \mathbf{g}_{k-\ell+1}, \mathbf{0}^n, \mathbf{g}_{k-\ell+2}, \ldots, \mathbf{0}^n, \mathbf{g}_k$. We then observe that $\big(\mathbf{u} \,\|\, \mathsf{I2B}(j)\big) \cdot \mathbf{G} = \big(\mathbf{u} \,\|\, \mathsf{Encode}(j)\big) \cdot \widehat{\mathbf{G}}$.

Let $\mathbf{f} = \mathsf{Encode}(j)$, then (6) can be equivalently rewritten as:

$$\begin{cases} \mathbf{H} \cdot \mathbf{s}^\top \oplus \mathbf{A} \cdot \mathbf{x}^\top = \mathbf{0} \ \wedge \ \mathbf{x} = \delta_j^N \ \wedge \ \mathbf{s} \in \mathsf{B}(m, \omega); \\ \left( \mathbf{u} \,\|\, \mathbf{f} \right) \cdot \widehat{\mathbf{G}} \oplus \mathbf{e} = \mathbf{c} \ \wedge \ \mathbf{f} = \mathsf{Encode}(j) \ \wedge \ \mathbf{e} \in \mathsf{B}(n, t). \end{cases} \tag{7}$$

To obtain a ZKAoK for relation (7) in Stern's framework [Ste96], $\mathcal{P}$ proceeds as follows:

- To prove that $\mathbf{x} = \delta_j^N$ *and* $\mathbf{f} = \mathsf{Encode}(j)$ while keeping $j$ secret, prover $\mathcal{P}$ samples a uniformly random vector $\mathbf{b} \in \{0,1\}^\ell$, sends $\mathbf{b}_1 = \mathsf{I2B}(j) \oplus \mathbf{b}$, and shows that:

$$T_{\mathbf{b}}(\mathbf{x}) = \delta^N_{\mathsf{B2I}(\mathbf{b}_1)} \ \wedge \ T'_{\mathbf{b}}(\mathbf{f}) = \mathsf{Encode}(\mathsf{B2I}(\mathbf{b}_1)).$$

  By the equivalences observed in (4) and (5), the verifier will be convinced about the facts to prove. Furthermore, since $\mathbf{b}$ essentially acts as a "one-time pad", the secret $j$ is perfectly hidden.
- To prove in zero-knowledge that $\mathbf{s} \in \mathsf{B}(m, \omega)$, $\mathcal{P}$ samples a uniformly random permutation $\pi \in \mathsf{S}_m$, and shows that $\pi(\mathbf{s}) \in \mathsf{B}(m, \omega)$. Similarly, to prove in zero-knowledge that $\mathbf{e} \in \mathsf{B}(n, t)$, a uniformly random permutation $\sigma \in \mathsf{S}_n$ is employed.
- Finally, to prove the linear equations in zero-knowledge, $\mathcal{P}$ samples uniformly random "masking" vectors $(\mathbf{r_s}, \mathbf{r_x}, \mathbf{r_u}, \mathbf{r_f}, \mathbf{r_e})$, and shows that:

$$\begin{cases} \mathbf{H} \cdot (\mathbf{s} \oplus \mathbf{r_s})^\top \ \oplus \ \mathbf{A} \cdot (\mathbf{x} \oplus \mathbf{r_x})^\top = \mathbf{H} \cdot \mathbf{r_s}^\top \ \oplus \ \mathbf{A} \cdot \mathbf{r_x}^\top; \\ \left( \mathbf{u} \oplus \mathbf{r_u} \,\|\, \mathbf{f} \oplus \mathbf{r_f} \right) \cdot \widehat{\mathbf{G}} \ \oplus \ (\mathbf{e} \oplus \mathbf{r_e}) \ \oplus \ \mathbf{c} = \left( \mathbf{r_u} \,\|\, \mathbf{r_f} \right) \cdot \widehat{\mathbf{G}} \ \oplus \ \mathbf{r_e}. \end{cases} \tag{8}$$

Now let $\mathrm{COM} : \{0,1\}^* \to \{0,1\}^\lambda$ be a collision-resistant hash function, to be modelled as a random oracle. Prover $\mathcal{P}$ and verifier $\mathcal{V}$ first perform the preparation steps described above, and then interact as described in Fig. 1.

### 3.2 Analysis of the Protocol

The properties of our protocol are summarized in the following theorem.

**Theorem 1.** *The interactive protocol described in Sect. 3.1 has perfect completeness, and has communication cost bounded by* $\beta = (N + 3\log N) + m(\log m + 1) + n(\log n + 1) + k + 5\lambda$ *bits. If* COM *is modelled as a random oracle, then the protocol is statistical zero-knowledge. If* COM *is a collision-resistant hash function, then the protocol is an argument of knowledge.*

**Completeness.** It can be seen that the given interactive protocol is perfectly complete, i.e., if $\mathcal{P}$ possesses a valid witness $(j, \mathbf{s}, \mathbf{u}, \mathbf{e})$ and follows the protocol, then $\mathcal{V}$ always outputs 1. Indeed, given $(j, \mathbf{s}, \mathbf{u}, \mathbf{e})$ satisfying (6), $\mathcal{P}$ can always obtain $(j, \mathbf{s}, \mathbf{x}, \mathbf{u}, \mathbf{f}, \mathbf{e})$ satisfying (7). Then, as discussed above, the following are true:

$$\begin{cases} \forall \pi \in \mathsf{S}_m : \pi(\mathbf{s}) \in \mathsf{B}(m, \omega); \ \ \forall \sigma \in \mathsf{S}_n : \sigma(\mathbf{e}) \in \mathsf{B}(n, t); \\ \forall \mathbf{b} \in \{0,1\}^\ell \colon T_{\mathbf{b}}(\mathbf{x}) = \delta^N_{\mathsf{B2I}(\mathsf{I2B}(j) \oplus \mathbf{b})} = \mathbf{w_x}; \ T'_{\mathbf{b}}(\mathbf{f}) = \mathsf{Encode}(\mathsf{B2I}(\mathsf{I2B}(j) \oplus \mathbf{b})) = \mathbf{w_f}. \end{cases}$$

1. **Commitment:** $\mathcal{P}$ samples the following uniformly random objects:

$$\begin{cases} \mathbf{b} \xleftarrow{\$} \{0,1\}^{\ell}; \;\; \pi \xleftarrow{\$} \mathsf{S}_m; \;\; \sigma \xleftarrow{\$} \mathsf{S}_n; \;\; \rho_1, \rho_2, \rho_3 \xleftarrow{\$} \{0,1\}^{\lambda}; \\ \mathbf{r_s} \xleftarrow{\$} \mathbb{F}_2^m; \;\; \mathbf{r_x} \xleftarrow{\$} \mathbb{F}_2^N; \;\; \mathbf{r_u} \xleftarrow{\$} \mathbb{F}_2^{k-\ell}; \;\; \mathbf{r_f} \xleftarrow{\$} \mathbb{F}_2^{2\ell}; \;\; \mathbf{r_e} \xleftarrow{\$} \mathbb{F}_2^n. \end{cases}$$

   It then sends the commitment $\mathsf{CMT} := (c_1, c_2, c_3)$ to $\mathcal{V}$, where

$$\begin{cases} c_1 = \mathrm{COM}\big(\mathbf{b},\ \pi,\ \sigma,\ \ \mathbf{H} \cdot \mathbf{r_s}^{\top} \oplus \mathbf{A} \cdot \mathbf{r_x}^{\top},\ \ \big(\mathbf{r_u} \,\|\, \mathbf{r_f}\big) \cdot \widehat{\mathbf{G}} \oplus \mathbf{r_e};\ \ \rho_1\big), \\ c_2 = \mathrm{COM}\big(\pi(\mathbf{r_s}), T_{\mathbf{b}}(\mathbf{r_x}),\ T'_{\mathbf{b}}(\mathbf{r_f}),\ \sigma(\mathbf{r_e});\ \ \rho_2\big), \\ c_3 = \mathrm{COM}\big(\pi(\mathbf{s} \oplus \mathbf{r_s}),\ T_{\mathbf{b}}(\mathbf{x} \oplus \mathbf{r_x}),\ T'_{\mathbf{b}}(\mathbf{f} \oplus \mathbf{r_f}),\ \sigma(\mathbf{e} \oplus \mathbf{r_e});\ \ \rho_3\big). \end{cases}$$

2. **Challenge:** Receiving $\mathsf{CMT}$, $\mathcal{V}$ sends a challenge $\mathrm{Ch} \xleftarrow{\$} \{1,2,3\}$ to $\mathcal{P}$.
3. **Response:** $\mathcal{P}$ responds as follows:
   - If $\mathrm{Ch} = 1$: Reveal $c_2$ and $c_3$. Let $\mathbf{b}_1 = \mathsf{I2B}(j) \oplus \mathbf{b}$,

$$\begin{cases} \mathbf{v_s} = \pi(\mathbf{r_s}), \\ \mathbf{w_s} = \pi(\mathbf{s}), \end{cases} \quad \mathbf{v_x} = T_{\mathbf{b}}(\mathbf{r_x}),\ \ \mathbf{v_f} = T'_{\mathbf{b}}(\mathbf{r_f}), \ \text{ and } \ \begin{cases} \mathbf{v_e} = \sigma(\mathbf{r_e}), \\ \mathbf{w_e} = \sigma(\mathbf{e}). \end{cases}$$

     Send $\mathsf{RSP} := \big(\mathbf{b}_1,\ \mathbf{v_s},\ \mathbf{w_s},\ \mathbf{v_x},\ \mathbf{v_f},\ \mathbf{v_e},\ \mathbf{w_e};\ \rho_2, \rho_3\big)$ to $\mathcal{V}$.
   - If $\mathrm{Ch} = 2$: Reveal $c_1$ and $c_3$. Let

$$\begin{cases} \mathbf{b}_2 = \mathbf{b};\ \ \pi_2 = \pi;\ \ \sigma_2 = \sigma; \\ \mathbf{z_s} = \mathbf{s} \oplus \mathbf{r_s};\ \ \mathbf{z_x} = \mathbf{x} \oplus \mathbf{r_x};\ \ \mathbf{z_u} = \mathbf{u} \oplus \mathbf{r_u};\ \ \mathbf{z_f} = \mathbf{f} \oplus \mathbf{r_f};\ \ \mathbf{z_e} = \mathbf{e} \oplus \mathbf{r_e}. \end{cases}$$

     Send $\mathsf{RSP} := \big(\mathbf{b}_2,\ \pi_2,\ \sigma_2,\ \mathbf{z_s},\ \mathbf{z_x},\ \mathbf{z_u},\ \mathbf{z_f},\ \mathbf{z_e};\ \rho_1, \rho_3\big)$ to $\mathcal{V}$.
   - If $\mathrm{Ch} = 3$: Reveal $c_1$ and $c_2$. Let

$$\mathbf{b}_3 = \mathbf{b};\ \ \pi_3 = \pi;\ \ \sigma_3 = \sigma;\ \ \mathbf{y_s} = \mathbf{r_s};\ \ \mathbf{y_x} = \mathbf{r_x};\ \ \mathbf{y_u} = \mathbf{r_u};\ \ \mathbf{y_f} = \mathbf{r_f};\ \ \mathbf{y_e} = \mathbf{r_e}.$$

     Send $\mathsf{RSP} := \big(\mathbf{b}_3,\ \pi_3,\ \sigma_3,\ \mathbf{y_s},\ \mathbf{y_x},\ \mathbf{y_u},\ \mathbf{y_f},\ \mathbf{y_e};\ \rho_1, \rho_2\big)$ to $\mathcal{V}$.

**Verification:** Receiving $\mathsf{RSP}$, $\mathcal{V}$ proceeds as follows:

- If $\mathrm{Ch} = 1$: Let $\mathbf{w_x} = \delta^N_{\mathsf{B2I}(\mathbf{b}_1)} \in \mathbb{F}_2^N$ and $\mathbf{w_f} = \mathsf{Encode}(\mathsf{B2I}(\mathbf{b}_1)) \in \mathbb{F}_2^{2\ell}$. Check that $\mathbf{w_s} \in \mathsf{B}(m, \omega)$, $\mathbf{w_e} \in \mathsf{B}(n, t)$, and that:

$$\begin{cases} c_2 = \mathrm{COM}\big(\mathbf{v_s},\ \mathbf{v_x},\ \mathbf{v_f},\ \mathbf{v_e};\ \ \rho_2\big), \\ c_3 = \mathrm{COM}\big(\mathbf{v_s} \oplus \mathbf{w_s},\ \ \mathbf{v_x} \oplus \mathbf{w_x},\ \ \mathbf{v_f} \oplus \mathbf{w_f},\ \ \mathbf{v_e} \oplus \mathbf{w_e};\ \ \rho_3\big). \end{cases}$$

- If $\mathrm{Ch} = 2$: Check that

$$\begin{cases} c_1 = \mathrm{COM}\big(\mathbf{b}_2,\ \ \pi_2,\ \ \sigma_2,\ \ \mathbf{H} \cdot \mathbf{z_s}^{\top} \oplus \mathbf{A} \cdot \mathbf{z_x}^{\top},\ \ \big(\mathbf{z_u} \,\|\, \mathbf{z_f}\big) \cdot \widehat{\mathbf{G}} \oplus \mathbf{z_e} \oplus \mathbf{c};\ \ \rho_1\big), \\ c_3 = \mathrm{COM}\big(\pi_2(\mathbf{z_s}),\ \ T_{\mathbf{b}_2}(\mathbf{z_x}),\ \ T'_{\mathbf{b}_2}(\mathbf{z_f}),\ \ \sigma_2(\mathbf{z_e});\ \ \rho_3\big). \end{cases}$$

- If $\mathrm{Ch} = 3$: Check that

$$\begin{cases} c_1 = \mathrm{COM}\big(\mathbf{b}_3,\ \ \pi_3,\ \ \sigma_3,\ \ \mathbf{H} \cdot \mathbf{y_s}^{\top} \oplus \mathbf{A} \cdot \mathbf{y_x}^{\top},\ \ \big(\mathbf{y_u} \,\|\, \mathbf{y_f}\big) \cdot \widehat{\mathbf{G}} \oplus \mathbf{y_e};\ \ \rho_1\big), \\ c_2 = \mathrm{COM}\big(\pi_3(\mathbf{y_s}),\ \ T_{\mathbf{b}_3}(\mathbf{y_x}),\ \ T'_{\mathbf{b}_3}(\mathbf{y_f}),\ \ \sigma_3(\mathbf{y_e});\ \ \rho_2\big). \end{cases}$$

In each case, $\mathcal{V}$ outputs 1 if and only if all the conditions hold. Otherwise, $\mathcal{V}$ outputs 0.

**Fig. 1.** The underlying zero-knowledge argument system of our group signature scheme.

As a result, $\mathcal{P}$ should always pass $\mathcal{V}$'s checks in the case $\mathrm{Ch} = 1$. In the case $\mathrm{Ch} = 2$, since the linear equations in (8) hold true, $\mathcal{P}$ should also pass the verification. Finally, in the case $\mathrm{Ch} = 3$, it suffices to note that $\mathcal{V}$ simply checks for honest computations of $c_1$ and $c_2$.

**Communication Cost.** The commitment $\mathsf{CMT}$ has bit-size $3\lambda$. If $\mathrm{Ch} = 1$, then the response $\mathsf{RSP}$ has bit-size $3\ell + N + 2(m+n+\lambda)$. In each of the cases $\mathrm{Ch} = 2$ and $\mathrm{Ch} = 3$, $\mathsf{RSP}$ has bit-size $2\ell + N + m(\log m + 1) + n(\log n + 1) + k + 2\lambda$. Therefore, the total communication cost (in bits) of the protocol is less than the bound $\beta$ specified in Theorem 1.

**Zero-Knowledge Property.** The following lemma says that our interactive protocol is statistically zero-knowledge if COM is modelled as a random oracle.

**Lemma 2.** *In the random oracle model, there exists an efficient simulator $\mathcal{S}$ interacting with a (possibly cheating) verifier $\widehat{\mathcal{V}}$, such that, given only the public input of the protocol, $\mathcal{S}$ outputs with probability negligibly close to $2/3$ a simulated transcript that is statistically close to the one produced by the honest prover in the real interaction.*

**Argument of Knowledge Property.** The next lemma states that our protocol satisfies the special soundness property of $\Sigma$-protocols, which implies that it is an argument of knowledge [Gro04].

**Lemma 3.** *Let* COM *be a collision-resistant hash function. Given the public input of the protocol, a commitment* $\mathsf{CMT}$ *and* 3 *valid responses* $\mathsf{RSP}_1, \mathsf{RSP}_2, \mathsf{RSP}_3$ *to all* 3 *possible values of the challenge* Ch*, one can efficiently construct a knowledge extractor $\mathcal{E}$ that outputs a tuple $(j', \mathbf{s}', \mathbf{u}', \mathbf{e}') \in [0, N-1] \times \mathbb{F}_2^m \times \mathbb{F}_2^{k-\ell} \times \mathbb{F}_2^n$ such that:*

$$\begin{cases} \mathbf{H} \cdot \mathbf{s}'^{\top} = \mathbf{y}_{j'}^{\top} \ \wedge \ \mathbf{s}' \in \mathsf{B}(m, \omega); \\ \big(\mathbf{u}' \,\|\, \mathsf{I2B}(j')\big) \cdot \mathbf{G} \oplus \mathbf{e}' = \mathbf{c} \ \wedge \ \mathbf{e}' \in \mathsf{B}(n, t). \end{cases}$$

The proofs of Lemmas 2 and 3 employ the standard simulation and extraction techniques for Stern-type protocols (e.g., [Ste96, KTX08, LNSW13]). These proofs are omitted here due to space constraints. They can be found in the full version of this paper [ELL+15].

## 4 Our Code-Based Group Signature Scheme

### 4.1 Description of the Scheme

Our group signature scheme is described as follows:

**KeyGen**$(1^\lambda, 1^N)$: On input a security parameter $\lambda$ and an expected number of group users $N = 2^\ell \in \mathsf{poly}(\lambda)$, for some positive integer $\ell$, this algorithm first selects the following:

–Parameters $n = n(\lambda), k = k(\lambda), t = t(\lambda)$ for a binary $[n, k, 2t+1]$ Goppa code.
–Parameters $m = m(\lambda), r = r(\lambda), \omega = \omega(\lambda)$ for the Syndrome Decoding problem, such that

$$r \leq \log \binom{m}{w} - 2\lambda - \mathcal{O}(1). \tag{9}$$

– Two collision-resistant hash functions, to be modelled as random oracles:
1. $\mathrm{COM} : \{0,1\}^* \to \{0,1\}^\lambda$, to be used for generating zero-knowledge arguments.
2. $\mathcal{H} : \{0,1\}^* \to \{1,2,3\}^\kappa$ (where $\kappa = \omega(\log \lambda)$), to be used in the Fiat-Shamir transformation.

The algorithm then performs the following steps:
1. Run $\mathsf{ME.KeyGen}(n, k, t)$ to obtain a key pair $\left(\mathsf{pk_{ME}} = \mathbf{G} \in \mathbb{F}_2^{k\times n} \ ; \ \mathsf{sk_{ME}}\right)$ for the randomized McEliece encryption scheme with respect to a binary $[n, k, 2t+1]$ Goppa code. The plaintext space is $\mathbb{F}_2^\ell$.
2. Choose a matrix $\mathbf{H} \xleftarrow{\$} \mathbb{F}_2^{r\times m}$.
3. For each $j \in [0, N-1]$, pick $\mathbf{s}_j \xleftarrow{\$} \mathsf{B}(m, \omega)$, and let $\mathbf{y}_j \in \mathbb{F}_2^r$ be its syndrome, i.e., $\mathbf{y}_j^\top = \mathbf{H} \cdot \mathbf{s}_j^\top$.
*Remark 1.* We note that, for parameters $m, r, \omega$ satisfying condition (9), the distribution of syndrome $\mathbf{y}_j$, for all $j \in [0, N-1]$, is statistically close to the uniform distribution over $\mathbb{F}_2^r$ (by Lemma 1).
4. Output

$$\big(\mathsf{gpk} = (\mathbf{G}, \mathbf{H}, \mathbf{y}_0, \ldots, \mathbf{y}_{N-1}),\ \mathsf{gmsk} = \mathsf{sk_{ME}},\ \mathsf{gsk} = (\mathbf{s}_0, \ldots, \mathbf{s}_{N-1})\big). \tag{10}$$

**Sign**$(\mathsf{gsk}[j], M)$: To sign a message $M \in \{0,1\}^*$ under $\mathsf{gpk}$, the group user of index $j$, who possesses secret key $\mathbf{s} = \mathsf{gsk}[j]$, performs the following steps:
1. Encrypt the binary representation of $j$, i.e., vector $\mathsf{I2B}(j) \in \mathbb{F}_2^\ell$, under the randomized McEliece encrypting key $\mathbf{G}$. This is done by sampling $(\mathbf{u} \xleftarrow{\$} \mathbb{F}_2^{k-\ell}, \mathbf{e} \xleftarrow{\$} \mathsf{B}(n, t))$ and outputting the ciphertext:

$$\mathbf{c} = \big(\mathbf{u} \,\|\, \mathsf{I2B}(j)\big) \cdot \mathbf{G} \oplus \mathbf{e} \in \mathbb{F}_2^n.$$

2. Generate a $\mathsf{NIZKAoK}$ $\Pi$ to simultaneously prove in zero-knowledge the possession of a vector $\mathbf{s} \in \mathsf{B}(m, \omega)$ corresponding to a certain syndrome $\mathbf{y}_j \in \{\mathbf{y}_0, \ldots, \mathbf{y}_{N-1}\}$ with hidden index $j$, and that $\mathbf{c}$ is a correct McEliece encryption of $\mathsf{I2B}(j)$. This is done by employing the interactive argument system in Sect. 3 with public input $(\mathbf{G}, \mathbf{H}, \mathbf{y}_0, \ldots, \mathbf{y}_{N-1}, \mathbf{c})$, and prover's witness $(j, \mathbf{s}, \mathbf{u}, \mathbf{e})$ that satisfies:

$$\begin{cases} \mathbf{H} \cdot \mathbf{s}^\top = \mathbf{y}_j^\top \ \ \wedge \ \ \mathbf{s} \in \mathsf{B}(m, \omega); \\ \big(\mathbf{u} \,\|\, \mathsf{I2B}(j)\big) \cdot \mathbf{G} \oplus \mathbf{e} = \mathbf{c} \ \ \wedge \ \ \mathbf{e} \in \mathsf{B}(n, t). \end{cases} \tag{11}$$

The protocol is repeated $\kappa = \omega(\log \lambda)$ times to achieve negligible soundness error, and then made non-interactive using the Fiat-Shamir heuristic. Namely, we have

$$\Pi = \big(\mathsf{CMT}^{(1)}, \ldots, \mathsf{CMT}^{(\kappa)};\ (\mathsf{Ch}^{(1)}, \ldots, \mathsf{Ch}^{(\kappa)});\ \mathsf{RSP}^{(1)}, \ldots, \mathsf{RSP}^{(\kappa)}\big), \tag{12}$$

where $(\mathsf{Ch}^{(1)}, \ldots, \mathsf{Ch}^{(\kappa)}) = \mathcal{H}\big(M; \mathsf{CMT}^{(1)}, \ldots, \mathsf{CMT}^{(\kappa)}; \mathsf{gpk}, \mathbf{c}\big) \in \{1, 2, 3\}^{\kappa}$.

3. Output the group signature $\Sigma = (\mathbf{c}, \Pi)$.

**Verify**$(\mathsf{gpk}, M, \Sigma)$: Parse $\Sigma$ as $(\mathbf{c}, \Pi)$ and parse $\Pi$ as in (12). Then proceed as follows:

1. If $(\mathsf{Ch}^{(1)}, \ldots, \mathsf{Ch}^{(\kappa)}) \neq \mathcal{H}\big(M; \mathsf{CMT}^{(1)}, \ldots, \mathsf{CMT}^{(\kappa)}; \mathsf{gpk}, \mathbf{c}\big)$, then return 0.
2. For $i = 1$ to $\kappa$, run the verification step of the interactive protocol in Sect. 3 with public input $(\mathbf{G}, \mathbf{H}, \mathbf{y}_0, \ldots, \mathbf{y}_{N-1}, \mathbf{c})$ to check the validity of $\mathsf{RSP}^{(i)}$ with respect to $\mathsf{CMT}^{(i)}$ and $\mathsf{Ch}^{(i)}$. If any of the verification conditions does not hold, then return 0.
3. Return 1.

**Open**$(\mathsf{gmsk}, M, \Sigma)$: Parse $\Sigma$ as $(\mathbf{c}, \Pi)$ and run $\mathsf{ME.Dec}(\mathsf{gmsk}, \mathbf{c})$ to decrypt $\mathbf{c}$. If decryption fails, then return $\perp$. If decryption outputs $\mathbf{g} \in \mathbb{F}_2^{\ell}$, then return $j = \mathsf{B2I}(\mathbf{g}) \in [0, N-1]$.

The efficiency, correctness, and security aspects of the above group signature scheme are summarized in the following theorem.

**Theorem 2.** *The given group signature scheme is correct. The public key has size* $nk + (m+N)r$ *bits, and signatures have bit-size bounded by* $\big((N + 3\log N) + m(\log m + 1) + n(\log n + 1) + k + 5\lambda\big)\kappa + n$. *Furthermore, in the random oracle model:*

- *If the Decisional McEliece problem* $\mathsf{DMcE}(n, k, t)$ *and the Decisional Learning Parity with fixed-weight Noise problem* $\mathsf{DLPN}(k - \ell, n, \mathsf{B}(n, t))$ *are hard, then the scheme is* CPA*-anonymous.*
- *If the Syndrome Decoding problem* $\mathsf{SD}(m, r, \omega)$ *is hard, then the scheme is traceable.*

### 4.2 Efficiency and Correctness

**Efficiency.** It is clear from (10) that $\mathsf{gpk}$ has bit-size $nk + (m + N)r$. The length of the NIZKAoK $\Pi$ is $\kappa$ times the communication cost of the underlying interactive protocol. Thus, by Theorem 1, $\Sigma = (\mathbf{c}, \Pi)$ has bit-size bounded by $\big((N + 3\log N) + m(\log m + 1) + n(\log n + 1) + k + 5\lambda\big)\kappa + n$.

**Correctness.** To see that the given group signature scheme is correct, first observe that the honest user with index $j$, for any $j \in [0, N-1]$, can always obtain a tuple $(j, \mathbf{s}, \mathbf{u}, \mathbf{e})$ satisfying (11). Then, since the underlying interactive protocol is perfectly complete, $\Pi$ is a valid NIZKAoK and algorithm $\mathsf{Verify}(\mathsf{gpk}, M, \Sigma)$ always outputs 1, for any message $M \in \{0, 1\}^*$.

Regarding the correctness of algorithm $\mathsf{Open}$, it suffices to note that, if the ciphertext $\mathbf{c}$ is of the form $\mathbf{c} = \big(\mathbf{u} \,\|\, \mathsf{I2B}(j)\big) \cdot \mathbf{G} \oplus \mathbf{e}$, where $\mathbf{e} \in \mathsf{B}(n,t)$, then, by the correctness of the randomized McEliece encryption scheme, algorithm $\mathsf{ME.Dec}(\mathsf{gmsk}, \mathbf{c})$ will output $\mathsf{I2B}(j)$.

### 4.3 Anonymity

Let $\mathcal{A}$ be any PPT adversary attacking the $\mathsf{CPA}$-anonymity of the scheme with advantage $\epsilon$. We will prove that $\epsilon = \mathsf{negl}(\lambda)$ based on the $\mathsf{ZK}$ property of the underlying argument system, and the assumed hardness of the $\mathsf{DMcE}(n,k,t)$ and the $\mathsf{DLPN}(k-\ell, n, \mathsf{B}(n,t))$ problems. Specifically, we consider the following sequence of hybrid experiments $G_0^{(b)}, G_1^{(b)}, G_2^{(b)}, G_3^{(b)}$ and $G_4$.

**Experiment** $G_0^{(b)}$. This is the real $\mathsf{CPA}$-anonymity game. The challenger runs $\mathsf{KeyGen}(1^\lambda, 1^N)$ to obtain

$$\big(\mathsf{gpk} = (\mathbf{G}, \mathbf{H}, \mathbf{y}_0, \ldots, \mathbf{y}_{N-1}),\ \ \mathsf{gmsk} = \mathsf{sk}_{\mathsf{ME}},\ \ \mathsf{gsk} = (\mathsf{gsk}[0], \ldots, \mathsf{gsk}[N-1])\big),$$

and then gives $\mathsf{gpk}$ and $\{\mathsf{gsk}[j]\}_{j\in[0,N-1]}$ to $\mathcal{A}$. In the challenge phase, $\mathcal{A}$ outputs a message $M^*$ together with two indices $j_0, j_1 \in [0, N-1]$. The challenger sends back a challenge signature $\Sigma^* = (\mathbf{c}^*, \Pi^*) \leftarrow \mathsf{Sign}(\mathsf{gpk}, \mathsf{gsk}[j_b])$, where $\mathbf{c}^* = \big(\mathbf{u} \,\|\, \mathsf{I2B}(j_b)\big) \cdot \mathbf{G} \oplus \mathbf{e}$, with $\mathbf{u} \xleftarrow{\$} \mathbb{F}_2^{k-\ell}$ and $\mathbf{e} \xleftarrow{\$} \mathsf{B}(n,t)$. The adversary then outputs $b$ with probability $1/2 + \epsilon$.

**Experiment** $G_1^{(b)}$. In this experiment, we introduce the following modification in the challenge phase: instead of faithfully generating the $\mathsf{NIZKAoK}$ $\Pi^*$, the challenger simulates it as follows:

1. Compute $\mathbf{c}^* \in \mathbb{F}_2^n$ as in experiment $G_0^{(b)}$.
2. Run the simulator of the underlying interactive protocol in Sect. 3 $t = \omega(\log \lambda)$ times on input $(\mathbf{G}, \mathbf{H}, \mathbf{y}_0, \ldots, \mathbf{y}_{N-1}, \mathbf{c}^*)$, and then program the random oracle $\mathcal{H}$ accordingly.
3. Output the simulated $\mathsf{NIZKAoK}$ $\Pi^*$.

Since the underlying argument system is statistically zero-knowledge, $\Pi^*$ is statistically close to the real $\mathsf{NIZKAoK}$. As a result, the simulated signature $\Sigma^* = \big(\mathbf{c}^*, \Pi^*\big)$ is statistically close to the one in experiment $G_0^{(b)}$. It then follows that $G_0^{(b)}$ and $G_1^{(b)}$ are indistinguishable from $\mathcal{A}$'s view.

**Experiment** $G_2^{(b)}$. In this experiment, we make the following change with respect to $G_1^{(b)}$: the encrypting key $\mathbf{G}$ obtained from $\mathsf{ME.KeyGen}(n,k,t)$ is replaced by a uniformly random matrix $\mathbf{G} \xleftarrow{\$} \mathbb{F}_2^{k\times n}$. We will demonstrate in Lemma 4 that experiments $G_1^{(b)}$ and $G_2^{(b)}$ are computationally indistinguishable based on the assumed hardness of the $\mathsf{DMcE}(n,k,t)$ problem.

**Lemma 4.** *If $\mathcal{A}$ can distinguish experiments $G_1^{(b)}$ and $G_2^{(b)}$ with probability non-negligibly larger than $1/2$, then there exists an efficient distinguisher $\mathcal{D}_1$ solving the $\mathsf{DMcE}(n,k,t)$ problem with the same probability.*

*Proof.* An instance of the $\mathsf{DMcE}(n,k,t)$ problem is a matrix $\mathbf{G}^* \in \mathbb{F}_2^{k\times n}$ which can either be uniformly random, or be generated by $\mathsf{ME.KeyGen}(n,k,t)$. Distinguisher $\mathcal{D}_1$ receives a challenge instance $\mathbf{G}^*$ and uses $\mathcal{A}$ to distinguish between the two. It interacts with $\mathcal{A}$ as follows.

– $\mathsf{Setup}$. Generate $(\mathbf{H}, \mathbf{y}_0, \ldots, \mathbf{y}_{N-1})$ and $(\mathsf{gsk}[0], \ldots, \mathsf{gsk}[N-1])$ as in the real scheme. Then, send the following to $\mathcal{A}$:

$$\big(\ \mathsf{gpk}^* = (\mathbf{G}^*, \mathbf{H}, \mathbf{y}_0, \ldots, \mathbf{y}_{N-1}),\ \ \mathsf{gsk} = (\mathsf{gsk}[0], \ldots, \mathsf{gsk}[N-1])\ \big).$$

– $\mathsf{Challenge}$. Receiving the challenge $(M^*, j_0, j_1)$, $\mathcal{D}_1$ proceeds as follows:

1. Pick $b \xleftarrow{\$} \{0,1\}$, and compute $\mathbf{c}^* = \big(\mathbf{u} \,\|\, \mathsf{I2B}(j_b)\big)\cdot\mathbf{G}^* \oplus \mathbf{e}$, where $\mathbf{u} \xleftarrow{\$} \mathbb{F}_2^{k-\ell}$ and $\mathbf{e} \xleftarrow{\$} \mathsf{B}(n,t)$.
2. Simulate the $\mathsf{NIZKAoK}$ $\Pi^*$ on input $(\mathbf{G}^*, \mathbf{H}, \mathbf{y}_0, \ldots, \mathbf{y}_{N-1}, \mathbf{c}^*)$, and output $\Sigma^* = \big(\mathbf{c}^*, \Pi^*\big)$.

We observe that if $\mathbf{G}^*$ is generated by $\mathsf{ME.KeyGen}(n,k,t)$ then the view of $\mathcal{A}$ in the interaction with $\mathcal{D}_1$ is statistically close to its view in experiment $G_1^{(b)}$ with the challenger. On the other hand, if $\mathbf{G}^*$ is uniformly random, then $\mathcal{A}$'s view is statistically close to its view in experiment $G_2^{(b)}$. Therefore, if $\mathcal{A}$ can guess whether it is interacting with the challenger in $G_1^{(b)}$ or $G_2^{(b)}$ with probability non-negligibly larger than $1/2$, then $\mathcal{D}_1$ can use $\mathcal{A}$'s guess to solve the challenge instance $\mathbf{G}^*$ of the $\mathsf{DMcE}(n,k,t)$ problem, with the same probability. □

**Experiment** $G_3^{(b)}$. Recall that in experiment $G_2^{(b)}$, we have

$$\mathbf{c}^* = \big(\mathbf{u} \,\|\, \mathsf{I2B}(j_b)\big)\cdot\mathbf{G} \oplus \mathbf{e} = (\mathbf{u}\cdot\mathbf{G}_1 \oplus \mathbf{e}) \oplus \mathsf{I2B}(j_b)\cdot\mathbf{G}_2,$$

where $\mathbf{G}_1 \in \mathbb{F}_2^{(k-\ell)\times n}$, $\mathbf{G}_2 \in \mathbb{F}_2^{\ell\times n}$ such that $\begin{bmatrix}\mathbf{G}_1\\ \mathbf{G}_2\end{bmatrix} = \mathbf{G}$; and $\mathbf{u} \xleftarrow{\$} \mathbb{F}_2^{k-\ell}$, $\mathbf{e} \xleftarrow{\$} \mathsf{B}(n,t)$.

In experiment $G_3^{(b)}$, the generation of $\mathbf{c}^*$ is modified as follows: we instead let $\mathbf{c}^* = \mathbf{v} \oplus \mathsf{I2B}(j_b)\cdot\mathbf{G}_2$, where $\mathbf{v} \xleftarrow{\$} \mathbb{F}_2^n$. Experiments $G_2^{(b)}$ and $G_3^{(b)}$ are computationally indistinguishable based on the assumed hardness of the $\mathsf{DLPN}(k-\ell, n, \mathsf{B}(n,t))$ problem, as shown in Lemma 5.

**Lemma 5.** *If $\mathcal{A}$ can distinguish experiments $G_2^{(b)}$ and $G_3^{(b)}$ with probability non-negligibly larger than $1/2$, then there exists an efficient distinguisher $\mathcal{D}_2$ solving the $\mathsf{DLPN}(k-\ell, n, \mathsf{B}(n,t))$ problem with the same probability.*

*Proof.* An instance of the $\mathsf{DLPN}(k-\ell, n, \mathsf{B}(n,t))$ problem is a pair $(\mathbf{B}, \mathbf{v}) \in \mathbb{F}_2^{(k-\ell)\times n} \times \mathbb{F}_2^n$, where $\mathbf{B}$ is uniformly random, and $\mathbf{v}$ is either uniformly random or of the form $\mathbf{v} = \mathbf{u}\cdot\mathbf{B} \oplus \mathbf{e}$, for $(\mathbf{u} \xleftarrow{\$} \mathbb{F}_2^{k-\ell}; \mathbf{e} \xleftarrow{\$} \mathsf{B}(n,t))$. Distinguisher $\mathcal{D}_2$ receives a challenge instance $(\mathbf{B}, \mathbf{v})$ and uses $\mathcal{A}$ to distinguish between the two. It interacts with $\mathcal{A}$ as follows.

– Setup. Pick $\mathbf{G}_2 \xleftarrow{\$} \mathbb{F}_2^{\ell \times n}$ and let $\mathbf{G}^* = \left[\frac{\mathbf{B}}{\mathbf{G}_2}\right]$. Generate $(\mathbf{H}, \mathbf{y}_0, \ldots, \mathbf{y}_{N-1})$ and $(\mathsf{gsk}[0], \ldots, \mathsf{gsk}[N-1])$ as in the real scheme, and send the following to $\mathcal{A}$:

$$\big(\ \mathsf{gpk}^* = (\mathbf{G}^*, \mathbf{H}, \mathbf{y}_0, \ldots, \mathbf{y}_{N-1}),\ \ \mathsf{gsk} = (\mathsf{gsk}[0], \ldots, \mathsf{gsk}[N-1])\ \big).$$

– Challenge. Receiving the challenge $(M^*, j_0, j_1)$, $\mathcal{D}_2$ proceeds as follows:
  1. Pick $b \xleftarrow{\$} \{0,1\}$, and let $\mathbf{c}^* = \mathbf{v} \oplus \mathsf{I2B}(j_b) \cdot \mathbf{G}_2$, where $\mathbf{v}$ comes from the challenge DLPN instance.
  2. Simulate the NIZKAoK $\Pi^*$ on input $(\mathbf{G}^*, \mathbf{H}, \mathbf{y}_0, \ldots, \mathbf{y}_{N-1}, \mathbf{c}^*)$, and output $\Sigma^* = \big(\mathbf{c}^*, \Pi^*\big)$.

We observe that if $\mathcal{D}_2$'s input pair $(\mathbf{B}, \mathbf{v})$ is of the form $(\mathbf{B}, \mathbf{v} = \mathbf{u} \cdot \mathbf{B} \oplus \mathbf{e})$, where $\mathbf{u} \xleftarrow{\$} \mathbb{F}_2^{k-\ell}$ and $\mathbf{e} \xleftarrow{\$} \mathsf{B}(n,t)$, then the view of $\mathcal{A}$ in the interaction with $\mathcal{D}_2$ is statistically close to its view in experiment $G_2^{(b)}$ with the challenger. On the other hand, if the pair $(\mathbf{B}, \mathbf{v})$ is uniformly random, then $\mathcal{A}$'s view is statistically close to its view in experiment $G_3^{(b)}$. Therefore, if $\mathcal{A}$ can guess whether it is interacting with the challenger in $G_2^{(b)}$ or $G_3^{(b)}$ with probability non-negligibly larger than 1/2, then $\mathcal{D}_2$ can use $\mathcal{A}$'s guess to solve the challenge instance of the $\mathsf{DLPN}(k-\ell, \mathsf{B}(n,t))$ problem with the same probability. □

**Experiment** $G_4$. In this experiment, we employ the following modification with respect to $G_3^{(b)}$: the ciphertext $\mathbf{c}^*$ is now set as $\mathbf{c}^* = \mathbf{r} \xleftarrow{\$} \mathbb{F}_2^n$. Clearly, the distributions of $\mathbf{c}^*$ in experiments $G_3^{(b)}$ and $G_4$ are identical. As a result, $G_4$ and $G_3^{(b)}$ are statistically indistinguishable. We note that $G_4$ no longer depends on the challenger's bit $b$, and thus, $\mathcal{A}$'s advantage in this experiment is 0.

The above discussion shows that experiments $G_0^{(b)}, G_1^{(b)}, G_2^{(b)}, G_3^{(b)}, G_4$ are indistinguishable, and that $\mathbf{Adv}_{\mathcal{A}}(G_4) = 0$. It then follows that the advantage of $\mathcal{A}$ in attacking the CPA-anonymity of the scheme, i.e., in experiment $G_0^{(b)}$, is negligible. This concludes the proof of the CPA-anonymity property.

### 4.4 Traceability

Let $\mathcal{A}$ be a PPT traceability adversary against our group signature scheme, that has success probability $\epsilon$. We construct a PPT algorithm $\mathcal{F}$ that solves the $\mathsf{SD}(m, r, \omega)$ problem with success probability polynomially related to $\epsilon$.

Algorithm $\mathcal{F}$ receives a challenge $\mathsf{SD}(m, r, \omega)$ instance, that is, a uniformly random matrix-syndrome pair $(\widetilde{\mathbf{H}}, \tilde{\mathbf{y}}) \in \mathbb{F}_2^{r \times m} \times \mathbb{F}_2^r$. The goal of $\mathcal{F}$ is to find a vector $\mathbf{s} \in \mathsf{B}(m, \omega)$ such that $\widetilde{\mathbf{H}} \cdot \mathbf{s}^\top = \tilde{\mathbf{y}}^\top$. It then proceeds as follows:

1. Pick a guess $j^* \xleftarrow{\$} [0, N-1]$ and set $\mathbf{y}_{j^*} = \tilde{\mathbf{y}}$.
2. Set $\mathbf{H} = \widetilde{\mathbf{H}}$. For each $j \in [0, N-1]$ such that $j \neq j^*$, sample $\mathbf{s}_j \xleftarrow{\$} \mathsf{B}(m, \omega)$ and set $\mathbf{y}_j \in \mathbb{F}_2^r$ be its syndrome, i.e., $\mathbf{y}_j^\top = \mathbf{H} \cdot \mathbf{s}_j^\top$.
3. Run $\mathsf{ME.KeyGen}(n, k, t)$ to obtain a key pair $\big(\mathsf{pk}_{\mathsf{ME}} = \mathbf{G} \in \mathbb{F}_2^{k \times n}\ ;\ \mathsf{sk}_{\mathsf{ME}}\big)$.
4. Send $\mathsf{gpk} = \big(\mathbf{G}, \mathbf{H}, \mathbf{y}_0, \ldots, \mathbf{y}_{N-1}\big)$ and $\mathsf{gmsk} = \mathsf{sk}_{\mathsf{ME}}$ to $\mathcal{A}$.

We note that, since the parameters $m, r, \omega$ were chosen such that $r \leq \log \binom{m}{w} - 2\lambda - \mathcal{O}(1)$, by Lemma 1, the distribution of syndrome $\mathbf{y}_j$, for all $j \neq j^*$, is statistically close to the uniform distribution over $\mathbb{F}_2^r$. In addition, the syndrome $\mathbf{y}_{j^*} = \tilde{\mathbf{y}}$ is truly uniform over $\mathbb{F}_2^r$. It then follows that the distribution of $(\mathbf{y}_0, \ldots, \mathbf{y}_{N-1})$ is statistically close to that in the real scheme (see Remark 1). As a result, the distribution of $(\mathsf{gpk}, \mathsf{gmsk})$ is statistically close to the distribution expected by $\mathcal{A}$.

The forger $\mathcal{F}$ then initializes a set $CU = \emptyset$ and handles the queries from $\mathcal{A}$ as follows:

- Queries to the random oracle $\mathcal{H}$ are handled by consistently returning uniformly random values in $\{1, 2, 3\}^\kappa$. Suppose that $\mathcal{A}$ makes $Q_\mathcal{H}$ queries, then for each $\eta \leq Q_\mathcal{H}$, we let $r_\eta$ denote the answer to the $\eta$-th query.
- $\mathcal{O}^{\mathsf{Corrupt}}(j)$, for any $j \in [0, N-1]$: If $j \neq j^*$, then $\mathcal{F}$ sets $CU := CU \cup \{j\}$ and gives $\mathbf{s}_j$ to $\mathcal{A}$; If $j = j^*$, then $\mathcal{F}$ aborts.
- $\mathcal{O}^{\mathsf{Sign}}(j, M)$, for any $j \in [0, N-1]$ and any message $M$:
  - If $j \neq j^*$, then $\mathcal{F}$ honestly computes a signature, since it has the secret key $\mathbf{s}_j$.
  - If $j = j^*$, then $\mathcal{F}$ returns a simulated signature $\Sigma^*$ computed as in Sect. 4.3 (see Experiment $G_1^{(b)}$ in the proof of anonymity).

At some point, $\mathcal{A}$ outputs a forged group signature $\Sigma^*$ on some message $M^*$, where

$$\Sigma^* = \big(\mathbf{c}^*, \big(\mathsf{CMT}^{(1)}, \ldots, \mathsf{CMT}^{(\kappa)};\ \mathsf{Ch}^{(1)}, \ldots, \mathsf{Ch}^{(\kappa)};\ \mathsf{RSP}^{(1)}, \ldots, \mathsf{RSP}^{(\kappa)}\big)\big).$$

By the requirements of the traceability experiment, one has $\mathsf{Verify}(\mathsf{gpk}, M^*, \Sigma^*) = 1$, and for all $j \in CU$, signatures of user $j$ on $M^*$ were never queried. Now $\mathcal{F}$ uses $\mathsf{sk}_{\mathsf{ME}}$ to open $\Sigma^*$, and aborts if the opening algorithm does not output $j^*$. It can be checked that $\mathcal{F}$ aborts with probability at most $(N-1)/N + (2/3)^\kappa$, because the choice of $j^* \in [0, N-1]$ is completely hidden from $\mathcal{A}$'s view, and $\mathcal{A}$ can violate the soundness of the argument system with probability at most $(2/3)^\kappa$. Thus, with probability at least $1/N - (2/3)^\kappa$, it holds that

$$\mathsf{Verify}(\mathsf{gpk}, M^*, \Sigma^*) = 1 \ \wedge \ \mathsf{Open}(\mathsf{sk}_{\mathsf{ME}}, M^*, \Sigma^*) = j^*. \tag{13}$$

Suppose that (13) holds. Algorithm $\mathcal{F}$ then exploits the forgery as follows. Denote by $\Delta$ the tuple $\big(M^*; \mathsf{CMT}^{(1)}, \ldots, \mathsf{CMT}^{(\kappa)}; \mathbf{G}, \mathbf{H}, \mathbf{y}_0, \ldots, \mathbf{y}_{N-1}, \mathbf{c}^*\big)$. Observe that if $\mathcal{A}$ has never queried the random oracle $\mathcal{H}$ on input $\Delta$, then

$$\Pr\big[\big(\mathsf{Ch}^{(1)}, \ldots, \mathsf{Ch}^{(\kappa)}\big) = \mathcal{H}(\Delta)\big] \leq 3^{-\kappa}.$$

Therefore, with probability at least $\epsilon - 3^{-\kappa}$, there exists certain $\eta^* \leq Q_\mathcal{H}$ such that $\Delta$ was the input of the $\eta^*$-th query. Next, $\mathcal{F}$ picks $\eta^*$ as the target forking point and replays $\mathcal{A}$ many times with the same random tape and input as in the original run. In each rerun, for the first $\eta^* - 1$ queries, $\mathcal{A}$ is given the same answers $r_1, \ldots, r_{\eta^*-1}$ as in the initial run, but from the $\eta^*$-th query onwards,

$\mathcal{F}$ replies with fresh random values $r'_{\eta^*}, \ldots, r'_{q_{\mathcal{H}}} \xleftarrow{\$} \{1,2,3\}^{\kappa}$. The Improved Forking Lemma of Pointcheval and Vaudenay [PV97, Lemma 7] implies that, with probability larger than $1/2$ and within less than $32 \cdot Q_{\mathcal{H}}/(\epsilon - 3^{-\kappa})$ executions of $\mathcal{A}$, algorithm $\mathcal{F}$ can obtain a 3-fork involving the tuple $\Delta$. Now, let the answers of $\mathcal{F}$ with respect to the 3-fork branches be

$$r_{1,\eta^*} = (\mathsf{Ch}_1^{(1)}, \ldots, \mathsf{Ch}_1^{(\kappa)});\ r_{2,\eta^*} = (\mathsf{Ch}_2^{(1)}, \ldots, \mathsf{Ch}_2^{(\kappa)});\ r_{3,\kappa^*} = (\mathsf{Ch}_3^{(1)}, \ldots, \mathsf{Ch}_3^{(\kappa)}).$$

Then, by a simple calculation, one has:

$$\Pr\left[\exists i \in \{1, \ldots, \kappa\} :\ \{\mathsf{Ch}_1^{(i)}, \mathsf{Ch}_2^{(i)}, \mathsf{Ch}_3^{(i)}\} = \{1,2,3\}\right] = 1 - (7/9)^{\kappa}.$$

Conditioned on the existence of such index $i$, one parses the 3 forgeries corresponding to the fork branches to obtain $\left(\mathsf{RSP}_1^{(i)}, \mathsf{RSP}_2^{(i)}, \mathsf{RSP}_3^{(i)}\right)$. They turn out to be 3 *valid* responses with respect to 3 different challenges for the same commitment $\mathsf{CMT}^{(i)}$. Then, by using the knowledge extractor of the underlying interactive argument system (see Lemma 3), one can efficiently extract a tuple $(j', \mathbf{s}', \mathbf{u}', \mathbf{e}') \in [0, N-1] \times \mathbb{F}_2^m \times \mathbb{F}_2^{k-\ell} \times \mathbb{F}_2^n$ such that:

$$\begin{cases} \mathbf{H} \cdot \mathbf{s}'^{\top} = \mathbf{y}_{j'}^{\top} \ \wedge \ \mathbf{s}' \in \mathsf{B}(m, \omega); \\ \left(\mathbf{u}' \,\|\, \mathsf{I2B}(j')\right) \cdot \mathbf{G} \oplus \mathbf{e}' = \mathbf{c}^* \ \wedge \ \mathbf{e}' \in \mathsf{B}(n, t). \end{cases}$$

Since the given group signature scheme is correct, the equation $\left(\mathbf{u}' \,\|\, \mathsf{I2B}(j')\right) \cdot \mathbf{G} \oplus \mathbf{e}' = \mathbf{c}^*$ implies that $\mathsf{Open}(\mathsf{sk}_{\mathsf{ME}}, M^*, \Sigma^*) = j'$. On the other hand, we have $\mathsf{Open}(\mathsf{sk}_{\mathsf{ME}}, M^*, \Sigma^*) = j^*$, which leads to $j' = j^*$. Therefore, it holds that $\widetilde{\mathbf{H}} \cdot \mathbf{s}'^{\top} = \mathbf{H} \cdot \mathbf{s}'^{\top} = \mathbf{y}_{j^*}^{\top} = \tilde{\mathbf{y}}^{\top}$, and that $\mathbf{s}' \in \mathsf{B}(m, \omega)$. In other words, $\mathbf{s}'$ is a valid solution to the challenge $\mathsf{SD}(m, r, \omega)$ instance $(\widetilde{\mathbf{H}}, \tilde{\mathbf{y}})$.

Finally, the above analysis shows that, if $\mathcal{A}$ has success probability $\epsilon$ and running time $T$ in attacking the traceability of our group signature scheme, then $\mathcal{F}$ has success probability at least $1/2\big(1/N - (2/3)^{\kappa}\big)\big(1 - (7/9)^{\kappa}\big)$ and running time at most $32 \cdot T \cdot Q_{\mathcal{H}}/(\epsilon - 3^{-\kappa}) + \mathsf{poly}(\lambda, N)$. This concludes the proof of the traceability property.

## 5 Implementation Results

This section presents our basic implementation results of the proposed code-based group signature to demonstrate its feasibility. The testing platform was a modern PC running at 3.5 GHz CPU with 16 GB RAM. We employed the NTL library [NTL] and the gf2x library [GF2] for efficient polynomial operations over a field of characteristic 2. To decode binary Goppa codes, the Paterson algorithm [Pat75] was used in our implementation of the McEliece encryption. We employed SHA-3 with various output sizes to realize several hash functions. To achieve 80-bit security, we chose the parameters as follows:

- The McEliece parameters were set to $(n, k, t) = (2^{11}, 1696, 32)$, as in [BS08].
- The parameters for Syndrome Decoding were set to $(m, r, \omega) = (2756, 550, 121)$ so that the distribution of $\mathbf{y}_0, \ldots, \mathbf{y}_{N-1}$ is $2^{-80}$-close to the uniform distribution over $\mathbb{F}_2^r$ (by Lemma 1), and that the $\mathsf{SD}(m, r, \omega)$ problem is intractable with respect to the best known attacks. In particular, these parameters ensure that:
  1. The Information Set Decoding algorithm proposed in [BJMM12] has work factor more than $2^{80}$. (See also [Sen14, Slide 3] for an evaluation formula.)
  2. The birthday attacks presented in [FS09] have work factors more than $2^{80}$.
- The number of protocol repetitions $\kappa$ was set to 140 to obtain soundness $1 - 2^{-80}$.

**Table 2.** Implementation results and sizes

| $N$ | PK size | Average signature size | Message | KeyGen | Sign | Verify | Open |
|---|---|---|---|---|---|---|---|
| $2^4$ (=16) | 625 KB | 111 KB | 1 B | 14.020 | 0.045 | 0.034 | 0.155 |
| | | | 1 GB | | 5.473 | 5.450 | |
| $2^8$ (=256) | 642 KB | 114 KB | 1 B | 14.128 | 0.046 | 0.036 | 0.155 |
| | | | 1 GB | | 5.459 | 5.450 | |
| $2^{12}$ (=4,096) | 906 KB | 159 KB | 1 B | 14.255 | 0.059 | 0.044 | 0.155 |
| | | | 1 GB | | 5.474 | 5.462 | |
| $2^{16}$ (=65,536) | 5.13 MB | 876 KB | 1 B | 16.302 | 0.269 | 0.193 | 0.161 |
| | | | 1 GB | | 5.704 | 5.630 | |
| $2^{20}$ (=1,048,576) | 72.8 MB | 12.4 MB | 1 B | 52.084 | 3.734 | 2.605 | 0.155 |
| | | | 1 GB | | 9.196 | 8.055 | |
| $2^{24}$ (=16,777,216) | 1.16 GB | 196 MB | 1 B | 636.511 | 58.535 | 40.801 | 0.154 |
| | | | 1 GB | | 64.047 | 46.402 | |

Unit for time: second

Table 2 shows our implementation results, together with the public key and signature sizes with respect to various numbers of group users and different message sizes. To reduce the signature size, in the underlying zero-knowledge protocol, we sent a random seed instead of permutations when Ch = 2. Similarly, we sent a random seed instead of the whole response $\mathsf{RSP}$ when Ch = 3. Using this technique, the average signature sizes were reduced to about 159 KB for 4,096 users and 876 KB for 65,536 users, respectively. Our public key and signature sizes are linear in the number of group users $N$, but it does not come to the front while $N$ is less than $2^{12}$ due to the size of parameters $\mathbf{G}$ and $\mathbf{H}$.

Our implementation took about 0.27 and 0.20 seconds for 1 B message and about 5.70 and 5.60 seconds for 1 GB message, respectively, to sign a message and

to verify a generated signature for a group of 65,536 users. In our experiments, it takes about 5.40 seconds to hash 1 GB message and it leads to the differences of signing and verifying times between 1 B and 1 GB messages.

As far as we know, the implementation results presented here are the first ones for post-quantum group signatures. Our results, while not yielding a truly practical scheme, would certainly help to bring post-quantum group signatures one step closer to practice.

**Acknowledgements.** The authors would like to thank Jean-Pierre Tillich, Philippe Gaborit, Ayoub Otmani, Nicolas Sendrier, Nico Döttling, and anonymous reviewers of ASIACRYPT 2015 for helpful comments and discussions. The research was supported by Research Grant TL-9014101684-01 and the Singapore Ministry of Education under Research Grant MOE2013-T2-1-041.

## References

[ABCG15] Alamélou, Q. Blazy, O., Cauchie, S., Gaborit, P.: A code-based group signature scheme. Presented at WCC, April 2015

[ACJT00] Ateniese, G., Camenisch, J.L., Joye, M., Tsudik, G.: A practical and provably secure coalition-resistant group signature scheme. In: Bellare, M. (ed.) CRYPTO 2000. LNCS, vol. 1880, pp. 255–270. Springer, Heidelberg (2000)

[ACM11] El Yousfi Alaoui, S.M., Cayrel, P.-L., Mohammed, M.: Improved identity-based identification and signature schemes using quasi-dyadic Goppa codes. In: Kim, T., Adeli, H., Robles, R.J., Balitanas, M. (eds.) ISA 2011. CCIS, vol. 200, pp. 146–155. Springer, Heidelberg (2011)

[BBS04] Boneh, D., Boyen, X., Shacham, H.: Short group signatures. In: Franklin, M. (ed.) CRYPTO 2004. LNCS, vol. 3152, pp. 41–55. Springer, Heidelberg (2004)

[BJMM12] Becker, A., Joux, A., May, A., Meurer, A.: Decoding random binary linear codes in $2^{n/20}$: how $1 + 1 = 0$ improves information set decoding. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 520–536. Springer, Heidelberg (2012)

[BMvT78] Berlekamp, E., McEliece, R.J., van Tilborg, H.C.A.: On the inherent intractability of certain coding problems. IEEE Trans. Inf. Theor. **24**(3), 384–386 (1978)

[BMW03] Bellare, M., Micciancio, D., Warinschi, B.: Foundations of group signatures: formal definitions, simplified requirements, and a construction based on general assumptions. In: Biham, E. (ed.) EUROCRYPT 2003. LNCS, vol. 2656, pp. 614–629. Springer, Heidelberg (2003)

[BS08] Biswas, B., Sendrier, N.: McEliece cryptosystem implementation: theory and practice. In: Buchmann, J., Ding, J. (eds.) PQCrypto 2008. LNCS, vol. 5299, pp. 47–62. Springer, Heidelberg (2008)

[BS13] Bettaieb, S., Schrek, J.: Improved lattice-based threshold ring signature scheme. In: Gaborit, P. (ed.) PQCrypto 2013. LNCS, vol. 7932, pp. 34–51. Springer, Heidelberg (2013)

[BW06] Boyen, X., Waters, B.: Compact group signatures without random oracles. In: Vaudenay, S. (ed.) EUROCRYPT 2006. LNCS, vol. 4004, pp. 427–444. Springer, Heidelberg (2006)

[CFS01] Courtois, N.T., Finiasz, M., Sendrier, N.: How to achieve a McEliece-based digital signature scheme. In: Boyd, C. (ed.) ASIACRYPT 2001. LNCS, vol. 2248, pp. 157–174. Springer, Heidelberg (2001)

[CGG07] Cayrel, P. L., Gaborit, P., Girault, M.: Identity-based identification and signature schemes using correcting codes. In: WCC, pp. 69–78 (2007)

[CHKP10] Cash, D., Hofheinz, D., Kiltz, E., Peikert, C.: Bonsai trees, or how to delegate a lattice basis. In: Gilbert, H. (ed.) EUROCRYPT 2010. LNCS, vol. 6110, pp. 523–552. Springer, Heidelberg (2010)

[CM10] Cayrel, P.-L., Meziani, M.: Post-quantum cryptography: code-based signatures. In: Kim, T., Adeli, H. (eds.) AST/UCMA/ISA/ACN 2010. LNCS, vol. 6059, pp. 82–99. Springer, Heidelberg (2010)

[CNR12] Camenisch, J., Neven, G., Rückert, M.: Fully anonymous attribute tokens from lattices. In: Visconti, I., De Prisco, R. (eds.) SCN 2012. LNCS, vol. 7485, pp. 57–75. Springer, Heidelberg (2012)

[CS97] Camenisch, J., Stadler, M.A.: Efficient group signature schemes for large groups. In: Kaliski Jr, B.S. (ed.) CRYPTO 1997. LNCS, vol. 1294, pp. 410–424. Springer, Heidelberg (1997)

[CS03] Camenisch, J., Shoup, V.: Practical verifiable encryption and decryption of discrete logarithms. In: Boneh, D. (ed.) CRYPTO 2003. LNCS, vol. 2729, pp. 126–144. Springer, Heidelberg (2003)

[CVA10] Cayrel, P.-L., Véron, P., El Yousfi Alaoui, S.M.: A zero-knowledge identification scheme based on the $q$-ary syndrome decoding problem. In: Biryukov, A., Gong, G., Stinson, D.R. (eds.) SAC 2010. LNCS, vol. 6544, pp. 171–186. Springer, Heidelberg (2011)

[CvH91] Chaum, D., van Heyst, E.: Group signatures. In: Davies, D.W. (ed.) EUROCRYPT 1991. LNCS, vol. 547, pp. 257–265. Springer, Heidelberg (1991)

[Dal08] Dallot, L.: Towards a concrete security proof of Courtois, Finiasz and Sendrier signature scheme. In: Lucks, S., Sadeghi, A.-R., Wolf, C. (eds.) WEWoRC 2007. LNCS, vol. 4945, pp. 65–77. Springer, Heidelberg (2008)

[DDMN12] Döttling, N., Dowsley, R., Müller-Quade, J., Nascimento, A.C.A.: A CCA2 secure variant of the McEliece cryptosystem. IEEE Trans. Inf. Theor. **58**(10), 6672–6680 (2012)

[Döt14] Döttling, N.: Cryptography based on the hardness of decoding. Ph.D. thesis, Karlsruhe Institute of Technology (2014). https://crypto.iti.kit.edu/fileadmin/User/Doettling/thesis.pdf

[ELL+15] Ezerman, M.F., Lee, H.T., Ling, S., Nguyen, K., Wang, H.: A provably secure group signature scheme from code-based assumptions. In: IACR Cryptography ePrint Archive, Report 2015/479 (2015)

[FGUO+13] Faugere, J.-C., Gauthier-Umana, V., Otmani, A., Perret, L., Tillich, J.-P.: A distinguisher for high-rate McEliece cryptosystems. IEEE Trans. Inf. Theor. **59**(10), 6830–6844 (2013)

[Fin10] Finiasz, M.: Parallel-CFS. In: Biryukov, A., Gong, G., Stinson, D.R. (eds.) SAC 2010. LNCS, vol. 6544, pp. 159–170. Springer, Heidelberg (2011)

[FS86] Fiat, A., Shamir, A.: How to prove yourself: practical solutions to identification and signature problems. In: Odlyzko, A.M. (ed.) CRYPTO 1986. LNCS, vol. 263, pp. 186–194. Springer, Heidelberg (1987)

[FS96] Fischer, J.-B., Stern, J.: An efficient pseudo-random generator provably as secure as syndrome decoding. In: Maurer, U.M. (ed.) EUROCRYPT 1996. LNCS, vol. 1070, pp. 245–255. Springer, Heidelberg (1996)

[FS09] Finiasz, M., Sendrier, N.: Security bounds for the design of code-based cryptosystems. In: Matsui, M. (ed.) ASIACRYPT 2009. LNCS, vol. 5912, pp. 88–105. Springer, Heidelberg (2009)

[GF2] gf2x library, ver. 1.1. https://gforge.inria.fr/projects/gf2x/

[GKPV10] Goldwasser, S., Kalai, Y., Peikert, C., Vaikuntanathan, V.: Robustness of the learning with errors assumption. In: ICS, pp. 230–240. Tsinghua University Press (2010)

[GKV10] Gordon, S.D., Katz, J., Vaikuntanathan, V.: A group signature scheme from lattice assumptions. In: Abe, M. (ed.) ASIACRYPT 2010. LNCS, vol. 6477, pp. 395–412. Springer, Heidelberg (2010)

[GPV08] Gentry, C., Peikert, C., Vaikuntanathan, V.: Trapdoors for hard lattices and new cryptographic constructions. In: STOC, pp. 197–206. ACM (2008)

[Gro04] Groth, J.: Evaluating security of voting schemes in the universal composability framework. In: Jakobsson, M., Yung, M., Zhou, J. (eds.) ACNS 2004. LNCS, vol. 3089, pp. 46–60. Springer, Heidelberg (2004)

[HMT13] Hu, R., Morozov, K., Takagi, T.: Proof of plaintext knowledge for code-based public-key encryption revisited. In: ASIA CCS, pp. 535–540. ACM (2013)

[KTX08] Kawachi, A., Tanaka, K., Xagawa, K.: Concurrently secure identification schemes based on the worst-case hardness of lattice problems. In: Pieprzyk, J. (ed.) ASIACRYPT 2008. LNCS, vol. 5350, pp. 372–389. Springer, Heidelberg (2008)

[LLLS13] Laguillaumie, F., Langlois, A., Libert, B., Stehlé, D.: Lattice-based group signatures with logarithmic signature size. In: Sako, K., Sarkar, P. (eds.) ASIACRYPT 2013, Part II. LNCS, vol. 8270, pp. 41–61. Springer, Heidelberg (2013)

[LLNW14] Langlois, A., Ling, S., Nguyen, K., Wang, H.: Lattice-based group signature scheme with verifier-local revocation. In: Krawczyk, H. (ed.) PKC 2014. LNCS, vol. 8383, pp. 345–361. Springer, Heidelberg (2014)

[LNSW13] Ling, S., Nguyen, K., Stehlé, D., Wang, H.: Improved zero-knowledge proofs of knowledge for the ISIS problem, and applications. In: Kurosawa, K., Hanaoka, G. (eds.) PKC 2013. LNCS, vol. 7778, pp. 107–124. Springer, Heidelberg (2013)

[LNW15] Ling, S., Nguyen, K., Wang, H.: Group signatures from lattices: simpler, tighter, shorter, ring-based. In: Katz, J. (ed.) PKC 2015. LNCS, vol. 9020, pp. 427–449. Springer, Heidelberg (2015)

[LPY12] Libert, B., Peters, T., Yung, M.: Scalable group signatures with revocation. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 609–627. Springer, Heidelberg (2012)

[McE78] McEliece, R.J.: A public-key cryptosystem based on algebraic coding theory. Deep Space Network Progress Report, vol. 44, pp. 114–116 (1978)

[MCG08] Melchor, C.A., Cayrel, P.-L., Gaborit, P.: A new efficient threshold ring signature scheme based on coding theory. In: Buchmann, J., Ding, J. (eds.) PQCrypto 2008. LNCS, vol. 5299, pp. 1–16. Springer, Heidelberg (2008)

[MCGL11] Melchor, C.A., Cayrel, P.-L., Gaborit, P., Laguillaumie, F.: A new efficient threshold ring signature scheme based on coding theory. IEEE Trans. Inf. Theor. **57**(7), 4833–4842 (2011)

[MCK01] Ma, J.F., Chiam, T.C., Kot, A.C: A new efficient group signature scheme based on linear codes. In: Networks, pp. 124–129. IEEE (2001)

[Meu13] Meurer, A.: A coding-theoretic approach to cryptanalysis. Ph.D. thesis, Ruhr University Bochum (2013). http://www.cits.rub.de/imperia/md/content/diss.pdf

[MGS11] Melchor, C.A., Gaborit, P., Schrek, J.: A new zero-knowledge code based identification scheme with reduced communication. CoRR, abs/1111.1644 (2011)

[MVR12] Mathew, K.P., Vasant, S., Rangan, C.P.: On provably secure code-based signature and signcryption scheme. In: IACR Cryptography ePrint Archive, Report 2012/585 (2012)

[MVVR12] Mathew, K.P., Vasant, S., Venkatesan, S., Pandu Rangan, C.: An efficient IND-CCA2 secure variant of the Niederreiter encryption scheme in the standard model. In: Susilo, W., Mu, Y., Seberry, J. (eds.) ACISP 2012. LNCS, vol. 7372, pp. 166–179. Springer, Heidelberg (2012)

[Nie86] Niederreiter, H.: Knapsack-type cryptosystems and algebraic coding theory. Probl. Control Inf. Theor. **15**(2), 159–166 (1986)

[NIKM08] Nojima, R., Imai, H., Kobara, K., Morozov, K.: Semantic security for the McEliece cryptosystem without random oracles. Des. Codes Cryptogr. **49**(1–3), 289–305 (2008)

[NTL] NTL: a library for doing number theory version 9.0.2. http://www.shoup.net/ntl/

[NZZ15] Nguyen, P.Q., Zhang, J., Zhang, Z.: Simpler efficient group signatures from lattices. In: Katz, J. (ed.) PKC 2015. LNCS, vol. 9020, pp. 401–426. Springer, Heidelberg (2015)

[Pat75] Patterson, N.J.: The algebraic decoding of Goppa codes. IEEE Trans. Inf. Theor. **21**(2), 203–207 (1975)

[Per12] Persichetti, E.: On a CCA2-secure variant of McEliece in the standard model. In: IACR Cryptography ePrint Archive, Report 2012/268 (2012)

[PV97] Pointcheval, D., Vaudenay, S.: On provable security for digital signature algorithms. Technical report LIENS-96-17, Laboratoire d'Informatique de Ecole Normale Superieure (1997)

[RST01] Rivest, R.L., Shamir, A., Tauman, Y.: How to leak a secret. In: Boyd, C. (ed.) ASIACRYPT 2001. LNCS, vol. 2248, pp. 552–565. Springer, Heidelberg (2001)

[Sen14] Sendrier, N.: QC-MDPC-McEliece: a public-key code-based encryption scheme based on quasi-cyclic moderate density parity check codes. In: Workshop "Post-Quantum Cryptography: Recent Results and Trends", Fukuoka, Japan, November 2014

[Sho97] Shor, P.: Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM J. Comput. **26**(5), 1484–1509 (1997)

[Ste96] Stern, J.: A new paradigm for public key identification. IEEE Trans. Inf. Theor. **42**(6), 1757–1768 (1996)

[Vér96] Véron, P.: Improved identification schemes based on error-correcting codes. Appl. Algebra Eng. Commun. Comput. **8**(1), 57–69 (1996)

[YTM+14] Yang, G., Tan, C.H., Mu, Y., Susilo, W., Wong, D.S.: Identity based identification from algebraic coding theory. Theor. Comput. Sci. **520**, 51–61 (2014)