Verifiable Shuffle of Large Size Ciphertexts

Jens Groth[1]* and Steve Lu[2]

[1] UCLA, Computer Science Department
jg@cs.ucla.edu
[2] UCLA, Math Department
stevelu@math.ucla.edu

Abstract. A shuffle is a permutation and rerandomization of a set of ciphertexts. Among other things, it can be used to construct mix-nets that are used in anonymization protocols and voting schemes. While shuffling is easy, it is hard for an outsider to verify that a shuffle has been performed correctly. We suggest two efficient honest verifier zero-knowledge (HVZK) arguments for correctness of a shuffle. Our goal is to minimize round-complexity and at the same time have low communicational and computational complexity.
The two schemes we suggest are both 3-move HVZK arguments for correctness of a shuffle. We first suggest a HVZK argument based on homomorphic integer commitments, and improve both on round complexity, communication complexity and computational complexity in comparison with state of the art. The second HVZK argument is based on homomorphic commitments over finite fields. Here we improve on the computational complexity and communication complexity when shuffling large ciphertexts.

Keywords: Shuffle, homomorphic commitment, homomorphic encryption, mix-net, honest verifier zero-knowledge.

1 Introduction

The main motivating example for shuffling is mix-nets. Parties can encrypt messages and send them to the mix-net; the mix-net then permutes, decrypts and outputs the messages. This allows parties to submit messages anonymously, which for instance is very useful in voting.

One approach to construct a mix-net is the following. The authorities, one by one, permute and rerandomize the ciphertexts. When all authorities have done this, they run a threshold decryption protocol to get out the messages. The central operation here is the permutation and rerandomization of a set of ciphertexts, a *shuffle*.

Obviously, it may be problematic if a dishonest authority replaces some of the ciphertexts, or cheats in some other way. If the cryptosystem is semantically secure, we cannot detect the cheating directly. We therefore need to add verifiability to the shuffle. One option is to request the shuffling authority to create a zero-knowledge argument for correctness of the shuffle. The goal of this paper is to present new honest verifier zero-knowledge arguments for correctness of a shuffle.

* Supported by NSF grant No. 0456717, and NSF Cybertrust grant.

RELATED WORK. Due to the direct applicability of proofs for the correctness of a shuffle, several researchers have investigated the problem and suggested schemes. Proving the correctness of a shuffle is a complicated matter, and as a consequence the most efficient schemes are also very complex. We will mention the more recent and efficient schemes here.

Abe and Hoshino [Abe99, AH01] proposed a 3-move proof for correctness of a shuffle of size $\mathcal{O}(kn \log n)$ bits, where k is the security parameter and n is the number of ciphertexts. Neff [Nef01] suggested an honest verifier zero-knowledge proof for correctness of a n ElGamal ciphertext shuffle based on the invariance of polynomials under permutation of the roots. While giving an efficient proof of size $\mathcal{O}(kn)$ bits, the drawback of this scheme is that it is a 7-move proof. Groth [Gro03, Gro05b] generalized Neff's scheme to work with a large class of homomorphic cryptosystems.

Furukawa and Sako [FS01], later improved by Furukawa [Fur05], proposed a 3-move argument for correctness of a shuffle. This method is based on committing to a permutation matrix and proving that the ciphertexts have been shuffled according to this permutation. They focus on the verifiability of an ElGamal ciphertext shuffle. Subsequent work by Nguyen et al. [NSNK04, NSNK05] and Onodera and Tanaka [OT04] have used the permutation matrix approach to construct correctness arguments for shuffles of Paillier ciphertexts. Peng at al. [PBD05] also investigate shuffling of Paillier ciphertexts, but use different techniques.

Yet another method for proving the correctness of a shuffle has been suggested by Wikström [Wik05a] based on unique factorization of integers. Unlike the other schemes that use commitments over $\mathbb{Z}_q$ for a prime q, he uses a homomorphic integer commitment scheme as a central building block. In some instances, this is actually desirable, for instance in [WG06]. One drawback of this scheme is that it uses 5 rounds.

OUR CONTRIBUTION. We suggest honest verifier zero-knowledge arguments for correctness of a shuffle. Since shuffles are typically used for anonymization, and since anonymization works best when individuals or groups can hide among a large set of other people, it is possible that we need to shuffle a huge number of ciphertexts. As an example, a voting scheme may have thousands or even millions of voters casting ballots. This implies that communication complexity and computational complexity are both of high importance. Furthermore, in a mix-net the authorities shuffle the ciphertexts one at a time and cooperate to generate the challenges for the honest verifier zero-knowledge argument. In order to minimize this work, we want to have as low round complexity as possible.

Our first scheme uses homomorphic integer commitments as the central building block. By working with integers, instead of working over $\mathbb{Z}_q$ as [FS01, Fur05], we show a much simpler way to demonstrate that indeed we have committed to a permutation matrix. The relevant comparison for this scheme is Wikström's argument for correctness of a shuffle [Wik05b] that is also based on integer commitments. Our scheme is better on all performance parameters, a detailed comparison can be found in Section 5.

Our second scheme uses homomorphic commitments over a message space $\mathbb{Z}_q$ for a prime q, just like [FS01, Fur05]. We combine Furukawa's [Fur05] scheme with techniques from [Gro05b] to obtain a 3-move argument for correctness of a shuffle. This generalization of Furukawa's scheme permits shuffling of almost any homomorphic

cryptosystem. If we look at the case of shuffling ElGamal ciphertexts, with the plaintexts belonging to a subgroup of relatively small order, our scheme is almost identical to Furukawa's scheme. However, a scenario with a large message space is perhaps more realistic. For instance, if we are looking at a voting scheme, we may want to permit write-in votes. If we are looking at a scheme for anonymous broadcast, senders may want to post large messages. For this setting, the most relevant comparison of our scheme is with the papers dealing with a shuffle of Paillier ciphertexts. Our scheme, has the same round complexity and is better on the other performance parameters. We refer to Section 5 for a detailed comparison with these schemes.

2 Preliminaries

We shuffle homomorphic ciphertexts and we use homomorphic commitments to shuffle them. For completeness, we will describe them here. We also recap the notion of an honest verifier zero-knowledge argument.

SPECIAL HONEST VERIFIER ZERO-KNOWLEDGE (SHVZK) ARGUMENT. We will describe 3-move public-coin arguments of knowledge with the special honest verifier zero-knowledge [CDS94] property. To explain this, consider a prover and a verifier. They both have access to a common reference string, in the paper it will consist of a public key for the commitment scheme and a public key for the cryptosystem. They also both have access to a statement x. In our case, this statement will consist of two sets of ciphertexts and a claim that one set is a shuffle of the other set. The prover sends an initial message a, the verifier selects a random challenge t, and the prover provides an answer z. The verifier can now evaluate (a, t, z) and decide whether to accept the truth of the statement.

That the protocol is public coin simply means that the challenge t is a random string. In the present paper the challenge will actually be n strings of bit-length ℓ_t. A possible choice is $\ell_t = 80$. If we wish to make the argument non-interactive, i.e., let the prover compute the challenges as a hash-value of x, a, then $\ell_t = 160$ would be suitable to account for the adversary being able to search many combinations of initial messages and hash-values offline.

The protocol must be complete, i.e., given a witness for the statement it should be easy for the prover to convince an honest verifier. It must be sound, i.e., it is infeasible to convince an honest verifier about a false statement. Moreover, the protocol will be an argument of knowledge in the following sense. If an adversary can produce a statement x and has non-negligible[3] probability ε of convincing the verifier, then with overwhelming probability it should be possible to extract a witness in expected polynomial time divided by ϵ. Finally, the protocols we present will have special honest verifier zero-knowledge (SHVZK). Given an arbitrary challenge t, we can simulate the argument (a, t, z).

Well-known examples of 3-move public coin SHVZK arguments of knowledge are Schnorr's [Sch91] and Guillou-Quisquater's [GQ88] identification protocols.

[3] A non-negligible function is the inverse of some polynomial of the security parameter.

HOMOMORPHIC ENCRYPTION. The public key of our cryptosystem specifies a message space, a randomizer space, and a ciphertext space that are abelian groups. The encryption algorithm E takes as input a message and a randomizer and outputs a ciphertext. The homomorphic property is

$$E(m \oplus m'; r \odot r') = E(m; r) \otimes E(m'; r'),$$

where $\oplus, \odot, \otimes$ are the binary operations for messages, randomizers and ciphertexts respectively. For notational convenience, we will in the rest of the paper use $+$ for the messages and randomizers, and $\cdot$ for the ciphertexts.

For the purpose of proving knowledge we assume the cryptosystem has the following root extraction property: Suppose an adversary produces a ciphertext E, an exponent e that is coprime with the order of the message space, and a message and randomizer so $E^e = E(M; R)$. Then we can efficiently extract m, r so $E = E(m; r)$. Examples of homomorphic cryptosystems with the root extraction property are ElGamal [ElG84], Okamoto-Uchiyama [OU98] and Paillier [Pai99].

We need an order of the message space that does not have any prime factors smaller than 2^{ℓ_t}. When specifying the protocols we will for simplicity assume that the randomizer space is $\mathbb{Z}$, and we encrypt M by choosing $R \leftarrow \{0,1\}^{\ell_R}$ and setting $E = E(M; R)$. This choice is purely out of notational convenience, the protocols work just as fine with other types of randomizer spaces.

HOMOMORPHIC COMMITMENT. The public key of the commitment scheme specifies a randomizer space and a commitment space that are abelian groups or abelian semigroups. We allow commitment to multiple elements at once. The homomorphic property is

$$\mathrm{com}(m_1 \oplus m'_1, \ldots, m_n \oplus m'_n; r \odot r') = \mathrm{com}(m_1, \ldots, m_n; r) \otimes \mathrm{com}(m'_1, \ldots, m'_n; r').$$

Again, for notational convenience we will in the rest of the paper use $+$ for the messages and randomizers, and $\cdot$ for the commitments.

In addition, the commitment scheme has a root extraction property which will be used for proving soundness. If an adversary produces a commitment c, and exponent $e \neq 0$ and a randomizer R and messages $M_1, \ldots, M_n$ so $c^e = \mathrm{com}(M_1, \ldots, M_n; R)$, then we can find $m_1, \ldots, m_n, r$ so $c = \mathrm{com}(m_1, \ldots, m_n; r)$.

The two shuffles we will propose make use of two different types of commitments: one will make use of integer commitments and the other will make use of commitments over a finite field $\mathbb{Z}_q$.

An example of a homomorphic commitment scheme over $\mathbb{Z}_q$ is the following variant of the Pedersen commitment [Ped91]. The public key consists of primes q, p with $q|p-1$, and random generators $g_1, \ldots, g_n, h$ of the order-q subgroup of $\mathbb{Z}_p^*$. To commit to n messages $m_1, \ldots, m_n$ using randomness $(u, r) \in \mathbb{Z}_p^* \times \mathbb{Z}_q$ so $u^{\frac{p-1}{q}} = 1 \bmod p$, we compute the commitment $c = u g_1^{m_1} \cdots g_n^{m_n} h^r \bmod p$. Typically, we pick randomness $u = 1$ and $r \leftarrow \mathbb{Z}_q$ uniformly at random. Observe, any $0 < c < p$ is a valid commitment, so it is straightforward to check that a commitment is well-formed. Note also that the commitment scheme is perfectly hiding.

Examples of homomorphic integer commitment schemes can be found in [FO97], later revised in [DF02], and [Gro05a]. We present the latter homomorphic integer commitment scheme that is the most efficient one. The public key consists of an RSA modulus $N = pq$, where $p = 2p'r_p + 1, q = 2q'r_q + 1$ and p', q' are primes. We work in the unique subgroup G of order $p'q'$. Let $g_1, \ldots, g_n, h$ be randomly chosen generators of G. To commit to a set of integers $m_1, \ldots, m_k$ using randomness $(u, e > 0, r)$ so $u^e = 1 \bmod n$, we use

$$c = \mathrm{com}(m_1, \ldots, m_k; (u, e, r)) = u g_1^{m_1} \cdots g_k^{m_k} h^r \bmod N.$$

To open it we reveal $m_1, \ldots, m_k, (u, e, r)$. When selecting the randomness the usual choice is $u = 1, e = 1, r \leftarrow \{0,1\}^{\ell_r + \ell_s}$, where $\ell_r = |G|$ and ℓ_s is a small security parameter. It is of course straightforward to test whether c is a valid commitment, we simply test $c \in \mathbb{Z}_N^*$. This commitment scheme is statistically hiding.

3 Verifiable Secret Shuffle Based on Integer Commitment

A shuffle of input ciphertexts $e_1, \ldots, e_n$ consists of output ciphertexts $E_1, \ldots, E_n$ so there exists a permutation π and randomizers $R_1, \ldots, R_n$ so $E_k = e_{\pi(k)} E(0; R_k)$. E_i is then the encryption of message $M_i = m_{\pi(i)}$. In this section, we suggest a SHVZK argument of knowledge of correctness of a shuffle based on homomorphic integer commitments.

The permutation defines a permutation matrix in the following way. Let A have entries $a_{\pi(i)i} = 1$ and all other entries 0. We can visualize relating the messages $(m_1, \ldots, m_n)$ with the permuted ones $(M_1, \ldots, M_n) = (m_{\pi(1)}, \ldots, m_{\pi(n)})$ by a multiplication by the permutation matrix A:

$$\begin{pmatrix} m_1 \\ m_2 \\ \vdots \\ m_n \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & \ldots & a_{1n} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \ldots & a_{nn} \end{pmatrix} \begin{pmatrix} m_{\pi(1)} \\ m_{\pi(2)} \\ \vdots \\ m_{\pi(n)} \end{pmatrix}$$

The idea in the shuffle argument is the following. We commit to the rows of A, $c_i \leftarrow \mathrm{com}(a_{i1}, \ldots, a_{in})$ for $i = 1, \ldots, n$. The verifier selects random challenges $t_1, \ldots, t_n \leftarrow \{0,1\}^{\ell_t}$ and we argue knowledge of the contents of $\prod_{i=1}^n c_i^{t_i}$. As we shall see this implies knowledge of the contents of each commitment c_i, i.e., knowledge of the matrix A.

The content of $\prod_{i=1}^n c_i^{t_i}$ is $(\sum_{i=1}^n a_{i1} t_i, \ldots, \sum_{i=1}^n a_{in} t_i)$. We will show that $\sum_{j=1}^n (\sum_{i=1}^n a_{ij} t_i) = \sum_{i=1}^n t_i$ for randomly chosen t_i's. Looking at each coefficient of the multi-variate polynomial, this means that with overwhelming probability we have $\sum_{j=1}^n a_{ij} = 1$ for $i = 1, \ldots, n$. In other words, each row of A sums to 1.

We also show that $\sum_{k=1}^n (\sum_{i=1}^n a_{ik} t_i)^2 = \sum_{i=1}^n t_i^2$ for randomly chosen t_i's. This gives us

$$0 = \sum_{k=1}^n \left(\sum_{i=1}^n a_{ik} t_i \right)^2 - \sum_{i=1}^n t_i^2 = \sum_{k=1}^n \sum_{i=1}^n \sum_{j=1}^n a_{ik} t_i a_{jk} t_j - \sum_{i=1}^n \sum_{j=1}^n \delta_{ik} t_i t_j$$

$$= \sum_{i=1}^{n} \sum_{j=1}^{n} \left[\left(\sum_{k=1}^{n} a_{ik} a_{jk} \right) - \delta_{ij} \right] t_i t_j.$$

Looking at coefficients of each pair $t_i t_j$ we see that $\sum_{k=1}^{n} a_{ik} a_{jk} = \delta_{ij}$, where $\delta_{ij} = 1$ if $i = j$ and 0 if $i \neq j$. I.e., the rows are orthogonal and have norm 1, so $AA^T = I$. Lemma 1 now shows that A is a permutation matrix defining some permutation π.

Finally, we have to connect the matrix A with the ciphertexts. We use the values $\sum_{i=1}^{n} a_{ij} t_i = t_{\pi(j)}$ that we have from the commitments. We show that

$$\prod_{i=1}^{n} E_i^{t_{\pi(i)}} = \prod_{i=1}^{n} e_i^{t_i} E\left(0; \sum_{i=1}^{n} t_{\pi(i)} R_i\right),$$

which implies

$$\prod_{i=1}^{n} (E_i e_{\pi(i)}^{-1})^{t_{\pi(i)}} = E\left(0; \sum_{i=1}^{n} t_{\pi(i)} R_i\right).$$

Since the t_i's are chosen at random this shows that with overwhelming probability E_i and $e_{\pi(i)}$ have the same message for any i. We shall see later that for cryptosystems with the root extraction property, we obtain a proof of knowledge, where we can extract randomizers R_i so $E_i = e_{\pi(i)} E(0; R_i)$.

These are the main ideas for obtaining soundness. What remains, is the problem of achieving zero-knowledge. We add some disguising values d_j to the sums we get out, i.e., we work with $d_j + \sum_{i=1}^{n} a_{ij} t_i$, where the d_j's are large random numbers. More precisely, $d_j \leftarrow \{0,1\}^{\ell_t + \ell_s}$, where ℓ_s is a small security parameter, for instance $\ell_s = 80$. This way the actual value of $\sum_{i=1}^{n} a_{ij} t_i$ is hidden throughout the argument. This modification entails a few other modifications to the protocol. The resulting argument is described in Figure 1.

Lemma 1. *Consider an $n \times n$ integer matrix A with entries a_{ij}. If $AA^T = I$ and $\sum_{j=1}^{n} a_{ij} = 1$ for all i then A is a permutation matrix.*

Proof. The condition $AA^T = I$ shows us that all rows have norm 1. In other words, each row has $n-1$ entries that are 0, and one single entry that is ± 1. Then $\sum_{j=1}^{n} a_{ij} = 1$ for all i shows us that these entries must be $+1$. Since A is invertible, the n 1-entries must be spread over all columns and all rows. In other words, A is a permutation matrix. □

Theorem 1. *The protocol in Figure 1 is a 3-move public coin SHVZK argument of knowledge of a correct shuffle. If the commitment scheme is statistically hiding, then the argument is statistical SHVZK.*

Proof. Completeness follows from direct algebraic manipulations. Left is to argue SHVZK and soundness and knowledge.

SHVZK. Given arbitrary challenges $t_1, \ldots, t_n \in \{0,1\}^{\ell_t}$ we have to simulate an argument. The simulation will mimic the real argument and we will highlight the main differences with a bar over the variable.

Shuffle Argument $\mathcal{SHUF}_{\mathbb{Z}}$

Common input: Ciphertexts $e_1, \ldots, e_n, E_1, \ldots, E_n$ and public keys.
Prover's input: Permutation $\pi \in \Sigma_n$ and randomizers $R_1, \ldots, R_n$ so $E_i = e_{\pi(i)} E(0; R_i)$.

Initial message $(\mathcal{P} \longrightarrow \mathcal{V})$**:** Choose randomness $r_i \leftarrow \{0,1\}^{\ell_r}, r_d \leftarrow \{0,1\}^{\ell_r + \log n + \ell_t + \ell_s}, d_j \leftarrow \{0,1\}^{\ell_t + \ell_s}, R_R \leftarrow \{0,1\}^{\ell_R + \log n + \ell_t + \ell_s}$ and set $d_n := -\sum_{j=1}^{n-1} d_j$. Set $E_R := E(0; -R_R) \prod_{i=1}^{n} E_i^{d_i}$. Generate commitments

$$
\begin{aligned}
c_1 &\leftarrow \mathrm{com}(\ 0\ \ 1_{\pi^{-1}(1)}\ \ 0\ \ \ldots\ \ 0,\ \ 2d_{\pi^{-1}(1)}\ ; r_1) \\
c_2 &\leftarrow \mathrm{com}(\ 0\ \ 0\ \ 0\ \ \ldots\ \ 1_{\pi^{-1}(2)},\ 2d_{\pi^{-1}(2)}\ ; r_2) \\
&\vdots \\
c_n &\leftarrow \mathrm{com}(\ 0\ \ \ldots\ \ 0\ \ 1_{\pi^{-1}(n)}\ \ 0,\ \ 2d_{\pi^{-1}(n)}\ ; r_n) \\
\text{and} & \\
c_d &\leftarrow \mathrm{com}(\ d_1,\ \ d_2,\ \ \ldots\ , d_{n-1},\ \ d_n,\ \ \textstyle\sum_{j=1}^{n} d_j^2\ ; r_d)
\end{aligned}
$$

Send $(c_1, \ldots, c_n, c_d, E_R)$ to the verifier.
Challenge $(\mathcal{P} \longleftarrow \mathcal{V})$**:** $t_1, \ldots, t_n \leftarrow \{0,1\}^{\ell_t}$.
Answer $(\mathcal{P} \longrightarrow \mathcal{V})$**:** Set $f_j := t_{\pi(j)} + d_j$, $z := \sum_{i=1}^{n} t_i r_i + r_d$ and $Z := R_R + \sum_{i=1}^{n} t_{\pi(i)} R_i$.
Send $(f_1, \ldots, f_n, z, Z)$ to the verifier.
Verification: Check that $c_1, \ldots, c_n, c_d$ are valid commitments and E_R is a valid ciphertext.
Set $f_\Delta := \sum_{j=1}^{n} f_j^2 - \sum_{i=1}^{n} t_i^2$. Verify

$$
\begin{aligned}
\textstyle\sum_{j=1}^{n} f_j &\stackrel{?}{=} \textstyle\sum_{i=1}^{n} t_i \\
c_d \textstyle\prod_{i=1}^{n} c_i^{t_i} &\stackrel{?}{=} \mathrm{com}(f_1, \ldots, f_n, f_\Delta; z) \\
\textstyle\prod_{i=1}^{n} E_i^{f_i} &\stackrel{?}{=} E(0; Z) E_R \textstyle\prod_{i=1}^{n} e_i^{t_i}
\end{aligned}
$$

Fig. 1. SHVZK Argument of Correct Shuffle Based on Integer Commitment

Initial message: Choose randomness r_i, r_d Choose random $\overline{f_j}$ so that $\sum_{j=1}^{n} \overline{f_j} = \sum_{i=1}^{n} t_i$ and set $f_\Delta := \sum_{j=1}^{n} \overline{f_j}^2 - \sum_{i=1}^{n} t_i^2$. Set $c_i \leftarrow \mathrm{com}(0, \ldots, 0)$. Choose random $\overline{z}$ and set $c_d \leftarrow \mathrm{com}(\overline{f_1}, \ldots, \overline{f_n}, f_\Delta; \overline{z}) \prod_{i=1}^{n} c_i^{-t_i}$. Choose random $\overline{Z}$ and set $E_R := E(0; -\overline{Z}) \prod_{i=1}^{n} E_i^{\overline{f_i}} e_i^{-t_i}$
Write $(c_1, \ldots, c_n, c_d, E_R)$ to the transcript.
Challenge: Write the $t_1, \ldots, t_n$ received as input to the transcript.
Answer: Send $(\overline{f_1}, \ldots, \overline{f_n}, \overline{z}, \overline{Z})$ to the verifier.

The simulated argument is $(c_1, \ldots, c_n, c_d, E_R, t_1, \ldots, t_n, \overline{f_1}, \ldots, \overline{f_n}, \overline{z}, \overline{Z})$.

To see that this is a good simulation, consider the following hybrid argument. We proceed exactly as in the simulation except when forming $c_1, \ldots, c_n$. Here we set $d_j := \overline{f_j} - t_{\pi(j)}$. We set $c_i \leftarrow \mathrm{com}(0, \ldots, 1_{\pi^{-1}(i)}, \ldots, 0, 2d_{\pi^{-1}(i)})$. Proceed with the rest of simulation as described above.

The hybrid argument is statistically indistinguishable from a real argument as the randomness chosen in the hybrid is linearly related to the randomness in the real argument, thus it retains the same distribution. On the other hand, the only thing that differs

from the simulation is the way we form the c_i's. The hiding property of the commitment scheme therefore gives us indistinguishability between the hybrid argument and the simulated argument. If the commitment scheme is statistically hiding, then we have statistical indistinguishability between the hybrid argument and the simulated argument.

SOUNDNESS AND KNOWLEDGE. Consider an adversary that has already sent the initial message $(c_1, \ldots, c_n, c_d, E_R)$ to the verifier and has non-negligible probability ε of answering the challenge. We store the state of this prover and now wish to extract a witness for correctness of the shuffle.

We select at random challenges $t_1, \ldots, t_n$ and run the adversarial prover until we have $n+1$ acceptable answers. We use an expected number of $(n+1)/\varepsilon$ tries to do this. Call the challenges $t_1^{(j)}, \ldots, t_n^{(j)}$ for $j = 0, \ldots, n$ and the corresponding answers $f_1^{(j)}, \ldots, f_n^{(j)}, z^{(j)}, Z^{(j)}$. Since $c_d \prod_{i=1}^n c_i^{t_i^{(j)}} = \mathrm{com}(f_1^{(j)}, \ldots, f_n^{(j)}, f_\Delta^{(j)}; z^{(j)})$ we have

$$\prod_{i=1}^n c_i^{t_i^{(0)} - t_i^{(j)}} = \mathrm{com}(f_1^{(0)} - f_1^{(j)}, \ldots, f_n^{(0)} - f_n^{(j)}, f_\Delta^{(0)} - f_\Delta^{(j)}; z^{(0)} - z^{(j)}).$$

Consider the $n \times n$ matrix T with entries $t_{ij} = t_i^{(0)} - t_i^{(j)}$. With overwhelming probability over the choices of $t_i^{(j)}$ the columns are linearly independent. We can in polynomial time find the transpose of the cofactor matrix C^T so $TC^T = |T|I$, where $|T|$ is the determinant of T.

Call the entries of C^T as v_{jk}, then we have $|T| = \sum_{j=1}^n t_{kj} v_{jk}$ and $0 = \sum_{j=1}^n t_{ij} v_{jk}$ for $k \neq i$. So

$$c_k^{|T|} = c_k^{\sum_{j=1}^n t_{kj} v_{jk}} = \prod_{i=1}^n c_i^{\sum_{j=1}^n t_{ij} v_{jk}} = \prod_{i=1}^n \prod_{j=1}^n c_i^{t_{ij} v_{jk}} = \prod_{j=1}^n \left(\prod_{i=1}^n c_i^{t_i^{(0)} - t_i^{(j)}} \right)^{v_{jk}}.$$

This means

$$c_k^{|T|} = \mathrm{com}\left(\sum_{j=1}^n v_{jk}(f_1^{(0)} - f_1^{(j)}), \ldots, \sum_{j=1}^n v_{jk}(f_n^{(0)} - f_n^{(j)}), \sum_{j=1}^n v_{jk}(f_\Delta^{(0)} - f_\Delta^{(j)}); \sum_{j=1}^n v_{jk}(z^{(0)} - z^{(j)}) \right).$$

By the root extraction property, we can open c_k. We call the opening $(a_{k1}, \ldots, a_{kn}, a_{k\Delta}, r_k)$. Since $c_d = \mathrm{com}(f_1^{(0)}, \ldots, f_n^{(0)}, f_\Delta^{(0)}; z^{(0)}) \prod_{i=1}^n c_i^{-t_i^{(0)}}$, having openings of $c_1, \ldots, c_n$ means that we can find an opening $(d_1, \ldots, d_n, d_\Delta, r_d)$ of c_d.

The adversary, having noticeable probability of answering the challenge $t_1, \ldots, t_n$, is forced to use $f_j = d_j + \sum_{i=1}^n a_{ij} t_i$ and $f_\Delta = d_\Delta + \sum_{i=1}^n a_{i\Delta} t_i$. The equation $f_\Delta = \sum_{j=1}^n f_j^2 - \sum_{i=1}^n t_i^2$ implies

$$\sum_{i=1}^n a_{i\Delta} t_i + d_\Delta = \sum_{j=1}^n \left(\sum_{i=1}^n a_{ij} t_i + d_j \right)^2 - \sum_{i=1}^n t_i^2$$

$$= \sum_{j=1}^{n} \left(\sum_{i=1}^{n} \sum_{k=1}^{n} a_{ij} a_{kj} t_i t_k + 2 d_j \sum_{i=1}^{n} a_{ij} t_i + d_j^2 \right) - \sum_{i=1}^{n} \sum_{k=1}^{n} \delta_{ik} t_i t_k$$
$$= \sum_{i=1}^{n} \sum_{k=1}^{n} \left(\sum_{j=1}^{n} a_{ij} a_{kj} - \delta_{ik} \right) t_i t_k + \sum_{i=1}^{n} \left(2 \sum_{j=1}^{n} d_j a_{ij} \right) t_i + \sum_{j=1}^{n} d_j^2.$$

With overwhelming probability over $t_1, \ldots, t_n$ this can only happen if $\sum_{j=1}^{n} a_{ij} a_{kj} = \delta_{ik}$ for all i, k. Let A be the matrix with entries a_{ij}. Then the equation corresponds to saying $AA^T = I$.

We also have

$$0 = \sum_{j=1}^{n} f_j - \sum_{i=1}^{n} t_i = \sum_{j=1}^{n} \left(\sum_{i=1}^{n} a_{ij} t_i + d_j \right) - \sum_{i=1}^{n} t_i = \sum_{i=1}^{n} \left(\sum_{j=1}^{n} a_{ij} - 1 \right) t_i + \sum_{j=1}^{n} d_j.$$

With overwhelming probability over the t_i's this can only be the case if $\sum_{j=1}^{n} a_{ij} = 1$ for all i.

Lemma 1 tells us that A is a permutation matrix. This means, there exists a permutation π so $a_{\pi(i)i} = 1$ and all other entries are 0.

Look now at the ciphertext equations, $\prod_{i=1}^{n} E_i^{f_i^{(j)}} = E_R E(0; Z^{(j)}) \prod_{i=1}^{n} e_i^{t_i^{(j)}}$ giving us

$$\prod_{i=1}^{n} E_i^{t_{\pi(i)}^{(0)} - t_{\pi(i)}^{(j)}} = \prod_{i=1}^{n} E_i^{f_i^{(0)} - f_i^{(j)}} = E(0; Z^{(0)} - Z^{(j)}) \prod_{i=1}^{n} e_i^{t_i^{(0)} - t_i^{(j)}}.$$

Since $\sum_{j=1}^{n} t_{ij} v_{jk} = |T| \delta_{ik}$ we have

$$(E_k e_{\pi(k)}^{-1})^{|T|} = (E_k e_{\pi(k)}^{-1})^{\sum_{j=1}^{n} t_{\pi(k)j} v_{j\pi(k)}} = \prod_{i=1}^{n} (E_i e_{\pi(i)}^{-1})^{\sum_{j=1}^{n} t_{\pi(i)j} v_{j\pi(k)}}$$
$$= \prod_{j=1}^{n} \left(\prod_{i=1}^{n} (E_i e_{\pi(i)}^{-1})^{t_{\pi(i)}^{(0)} - t_{\pi(i)}^{(j)}} \right)^{v_{j\pi(k)}} = \prod_{j=1}^{n} \left(\prod_{i=1}^{n} E_i^{t_{\pi(i)}^{(0)} - t_{\pi(i)}^{(j)}} \prod_{i=1}^{n} e_i^{t_i^{(0)} - t_i^{(j)}} \right)^{v_{j\pi(k)}}$$
$$= E\left(0; \sum_{j=1}^{n} v_{j\pi(k)} (Z^{(0)} - Z^{(j)}) \right).$$

By the root extraction property we can find an opening $(0, R_k)$ of $E_k e_{\pi(k)}^{-1}$. Doing so for $k = 1, \ldots, n$ means we have found openings $R_1, \ldots, R_n$ so $E_1 = e_{\pi(1)} E(0; R_1), \ldots, E_n = e_{\pi(n)} E(0; R_n)$. □

4 Verifiable Secret Shuffle Based on Commitments Over $\mathbb{Z}_q$

The ideas presented above also apply to the case of homomorphic commitment schemes over $\mathbb{Z}_q$. In this section, we suggest a SHVZK argument of knowledge of correctness of

a shuffle based on homomorphic commitments in $\mathbb{Z}_q$ where $q \equiv 2 \bmod 3$. This shuffle will be a slight modification of the one in the previous section to accommodate the fact that Lemma 1 no longer applies in $\mathbb{Z}_q$. The scheme is more complicated, but the advantage is that it may be easier to set up a scheme with prime order groups instead of using composite order groups. In case of ElGamal encryption with the message space being a small subgroup, the scheme is almost identical to Furukawa's scheme [Fur05]. However, for large message spaces or large ciphertexts we gain much in comparison with the state of the art.

The idea in the shuffle argument is similar to the preceding section. Let A have entries $a_{\pi(i)i} = 1$ in $\mathbb{Z}_q$ and all other entries 0. We commit to the rows of A, $c_i \leftarrow \mathrm{com}(a_{i1}, \ldots, a_{in})$ for $i = 1, \ldots, n$. The verifier selects random challenges $t_1, \ldots, t_n$ and we argue knowledge of the contents of $\prod_{i=1}^{n} c_i^{t_i}$. Just as before, we shall see this implies knowledge of the contents of each commitment c_i, i.e., knowledge of the matrix A.

In the case of commitments in $\mathbb{Z}_q$ we have a similar lemma (Theorem 2 [Fur05]) to identify when a matrix is a permutation matrix. We show that $\sum_{h=1}^{n}(\sum_{i=1}^{n} a_{ih}t_i)^3 = \sum_{i=1}^{n} t_i^3$. This gives us

$$0 = \sum_{h=1}^{n}(\sum_{i=1}^{n} a_{ih}t_i)^3 - \sum_{i=1}^{n} t_i^3 = \sum_{h=1}^{n}\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{n} a_{ih}t_i a_{jh}t_j a_{kh}t_k - \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{n}\delta_{ijk}t_i t_j t_k$$

$$= \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{n}\left[\left(\sum_{h=1}^{n} a_{ih}a_{jh}a_{kh}\right) - \delta_{ijk}\right] t_i t_j t_k.$$

Looking at coefficients of each triple $t_i t_j t_k$ we see that $\sum_{h=1}^{n} a_{ih}a_{jh}a_{kh} = \delta_{ijk}$, where $\delta_{ijk} = 1$ if $i = j = k$ and 0 otherwise. Lemma 2 now shows that A is a permutation matrix defining some permutation π.

Finally, we have to connect the matrix A with the ciphertexts. We use the values $\sum_{i=1}^{n} a_{ij}t_i = t_{\pi(j)}$ that we have from the commitments. We show that $\prod_{i=1}^{n} E_i^{t_{\pi(i)}} = \prod_{i=1}^{n} e_i^{t_i} E(0; \sum_{i=1}^{n} t_{\pi(i)}R_i)$, i.e., $\prod_{i=1}^{n}(E_i e_{\pi(i)}^{-1})^{t_{\pi(i)}} = E(0; \sum_{i=1}^{n} t_{\pi(i)}R_i)$. Since the t_i's are chosen at random this indicates that E_i and $e_{\pi(i)}$ have the same message for any i. Just as before, we add blinding factors to these values to ensure zero-knowledge. The resulting argument is described in Figure 2. If we let ℓ_s be an additional security parameter, we need to choose the d_i's from $\{0,1\}^{\ell_t+\ell_s}$. Because we are working with large ciphertexts, yet are performing all of the operations modulo q, to ensure the check on the ciphertexts still holds true we need to ensure that the equations $f_j = t_{\pi j} + d_j$ do not overflow. For this reason we require that $\ell_t + \ell_s < |q|$. The remaining random variables are only for verifying the commitments modulo q. Therefore, all of the prover's random variables may be reduced modulo q.

Lemma 2 (Theorem 2 [Fur05]). *Consider an $n \times n$ integer matrix A with entries a_{ij} in $\mathbb{Z}_q$ where $q \equiv 2 \bmod 3$. We have that*

$$\sum_{h=1}^{n} a_{ih}a_{jh}a_{kh} = \delta_{ijk} \text{ for all } i, j, k \tag{1}$$

if and only if A is a permutation matrix.

Proof. ($\Leftarrow$) is trivial.

($\Rightarrow$): Let R_i denote the i-th row vector of A. First we show the matrix A has full rank, i.e. the rows form a basis for $\mathbb{Z}_q^n$. If there is a linear combination $\mathbf{0} = \sum_{i=1}^n b_i R_i$ we have that $0 = \sum_{i=1}^n b_i a_{ih}$ for all h. Observe now that for any choice of j, we may multiply $a_{jh}a_{jh}$ to each of these equations, so $0 = \sum_{i=1}^n b_i a_{ih} a_{jh} a_{jh}$. Summing over all h we obtain $0 = \sum_{h=1}^n \sum_{i=1}^n b_i a_{ih} a_{jh} a_{jh} = \sum_{i=1}^n b_i \sum_{h=1}^n a_{ih} a_{jh} a_{jh}$ which by assumption is equal to $\sum_{i=1}^n b_i \delta_{ijj} = b_j$ and hence $b_j = 0$. This shows that the rows are linearly independent in $\mathbb{Z}_q^n$ and hence form a basis for $\mathbb{Z}_q^n$. Next, we show that there is at most one non-zero entry in each column.

If $v = (v_1, \ldots, v_n)$ and $w = (w_1, \ldots, w_n)$ are vectors in $\mathbb{Z}_q^n$, define $\langle v, w \rangle = \sum_{i=1}^n v_i w_i$ to be the dot product of v and w and define $v \odot w = (v_1 w_1, \ldots, v_n w_n)$ to be a vector resulting in the component-wise multiplication of v and w. Notice that $\langle R_i \odot R_j, R_k \rangle = \sum_{h=1}^n a_{ih} a_{jh} a_{kh}$ which is equal to δ_{ijk} by assumption. Observe that if $i \neq j$ then $\langle R_i \odot R_j, R_k \rangle = 0$ for all k, and since the R_k's span all of $\mathbb{Z}_q^n$, we have that $R_i \odot R_j = \mathbf{0}$. Since the choice of i, j was arbitrary, this means between each pair of entries in a column, at most one of them is non-zero; therefore at most one entry is non-zero. The matrix is of full rank, so indeed there is exactly one non-zero entry in each column (and hence in each row). This entry must be a cube root of 1, and $q = 2 \bmod 3$ implies there is a unique cube root, namely 1. Thus A is a permutation matrix over $\mathbb{Z}_q$. □

Theorem 2. *The protocol in Figure 2 is a 3-move public coin SHVZK argument of knowledge of a correct shuffle. If the commitment scheme is statistically hiding, then the argument is statistical SHVZK.*

Proof. Completeness follows from direct algebraic manipulations. Left is to argue SHVZK and soundness.

SHVZK. Given challenges $t_1, \ldots, t_n \in \{0,1\}^{\ell_t}$ we have to simulate an argument. The simulation will mimic the real argument and we will highlight the main differences with a bar over the variable. Simulation input: Challenges $t_1, \ldots, t_n$. Ciphertexts $e_1, \ldots, e_n, E_1, \ldots, E_n$ and public keys.

Initial message: Pick $r_1, \ldots, r_n$, $\overline{f_1}, \ldots, \overline{f_n}$, $\overline{F_1}, \ldots, \overline{F_n}$, r_d, r_D, $\overline{y_d}, \overline{y_D}$, $\overline{Z}$ and $\overline{f_d}$ at random. Set $f_D := \sum_{j=1}^n \overline{f_j}^3 - \sum_{i=1}^n t_i^3 - \overline{f_d}$. Using the challenges, compute $z_d := \sum_{j=1}^n t_j r_j + r_d$ and $z_D := \sum_{j=1}^n t_j^2 r_j + r_D$. Generate commitments $c_i \leftarrow \mathrm{com}(0, \ldots, 0; r_i)$ and $c_d \leftarrow \mathrm{com}(\overline{f_1}, \ldots, \overline{f_n}, \overline{y_d}, \overline{f_d}; z_d) \prod_{i=1}^n c_i^{-t_i}$ and $c_D \leftarrow \mathrm{com}(\overline{F_1}, \ldots, \overline{F_n}, f_D, \overline{y_D}; z_D) \prod_{i=1}^n c_i^{-t_i^2}$.
Set $E_R := E(0; -\overline{Z}) \prod_{i=1}^n E_i^{\overline{f_i}} e_i^{-t_i}$.
Write $(c_1, \ldots, c_n, c_d, c_D, E_R)$ to the transcript.

Challenge: Write the $t_1, \ldots, t_n$ received as input to the transcript.

Answer: Everything we need has already computed in an earlier phase. Thus we can immediately write $(\overline{f_1}, \ldots, \overline{f_n}, \overline{f_d}, \overline{y_d}, z_d, \overline{F_1}, \ldots, \overline{F_n}, \overline{y_D}, z_D, \overline{Z})$ to the transcript.

The simulated argument is

$$(c_1, \ldots, c_n, c_d, c_D, E_R, t_1, \ldots, t_n, \overline{f_1}, \ldots, \overline{f_n}, \overline{f_d}, \overline{y_d}, z_d, \overline{F_1}, \ldots, \overline{F_n}, \overline{y_D}, z_D, \overline{Z})$$

Shuffle Argument $\mathcal{SHUF}_{\mathbb{Z}_q}$

Common input: Ciphertexts $e_1, \ldots, e_n, E_1, \ldots, E_n$ and public keys.
Prover's input: Permutation $\pi \in \Sigma_n$ and randomizers $R_1, \ldots, R_n$ so $E_i = e_{\pi(i)} E(0; R_i)$.

Initial message $(\mathcal{P} \longrightarrow \mathcal{V})$**:** Choose randomness $d_1, \ldots, d_n \leftarrow \{0,1\}^{\ell_t + \ell_s}$ and $r_1, \ldots, r_n$, $D_1, \ldots, D_n, r_d, r_D, s_d, s_D, \Delta \leftarrow \mathbb{Z}_q$. Choose randomness $R_R \leftarrow \{0,1\}^{\ell_R + \log n + \ell_t + \ell_s}$. Set $E_R := E(0; -R_R) \prod_{i=1}^n E_i^{d_i}$. Generate commitments

$$
\begin{aligned}
c_1 &\leftarrow \mathrm{com}(\ 0 \ \ 1_{\pi^{-1}(1)} \ \ 0 \ \ \ldots \ \ 0, \ \ 3d_{\pi^{-1}(1)}, \ \ 3d^2_{\pi^{-1}(1)} \ \ ; r_1) \\
c_2 &\leftarrow \mathrm{com}(\ 0 \ \ 0 \ \ 0 \ \ \ldots \ \ 1_{\pi^{-1}(2)}, \ 3d_{\pi^{-1}(2)}, \ \ 3d^2_{\pi^{-1}(2)} \ \ ; r_2) \\
&\vdots \\
c_n &\leftarrow \mathrm{com}(\ 0 \ \ \ldots \ \ 0 \ \ 1_{\pi^{-1}(n)} \ \ 0, \ \ 3d_{\pi^{-1}(n)}, \ \ 3d^2_{\pi^{-1}(n)} \ \ ; r_n)
\end{aligned}
$$

and

$$
\begin{aligned}
c_d &\leftarrow \mathrm{com}(\ d_1, \ \ d_2, \ \ \ldots, d_{n-1}, \ \ d_n, \ \ s_d, \ \ \textstyle\sum_{j=1}^n d_j^3 - \Delta \ ; r_d) \\
c_D &\leftarrow \mathrm{com}(\ D_1, \ \ D_2, \ \ \ldots, D_{n-1}, \ \ D_n, \ \ \Delta, \ \ s_D \ \ ; r_D)
\end{aligned}
$$

Send $(c_1, \ldots, c_n, c_d, c_D, E_R)$ to the verifier.

Challenge $(\mathcal{P} \longleftarrow \mathcal{V})$**:** $t_1, \ldots, t_n \leftarrow \{0,1\}^{\ell_t}$.

Answer $(\mathcal{P} \longrightarrow \mathcal{V})$**:** Set $f_j := t_{\pi(j)} + d_j$ and $F_j := t^2_{\pi(j)} + D_j$ for $j = 1, \ldots, n$. Also set
$f_d := 3\sum_{j=1}^n t_j d^2_{\pi^{-1}(j)} + \sum_{j=1}^n d_j^3 - \Delta$
$y_d := 3\sum_{j=1}^n t_j d_{\pi^{-1}(j)} + s_d$ and $y_D := 3\sum_{j=1}^n t_j^2 d^2_{\pi^{-1}(j)} + s_D$
$z_d := \sum_{j=1}^n t_j r_j + r_d$ and $z_D := \sum_{j=1}^n t_j^2 r_j + r_D$
Set $Z := R_R + \sum_{i=1}^n t_{\pi(i)} R_i$.
Send $(f_1, \ldots, f_n, f_d, y_d, z_d, F_1, \ldots, F_n, y_D, z_D, Z)$ to the verifier.

Verification: Check that $c_1, \ldots, c_n, c_d, c_D$ are valid commitments and E_R is a valid ciphertext and $f_j > 2^{\ell_t}$ for $j = 1, \ldots, n$. Set $f_D := \sum_{j=1}^n f_j^3 - \sum_{i=1}^n t_i^3 - f_d$. Verify

$$
\begin{aligned}
c_d \textstyle\prod_{i=1}^n c_i^{t_i} &\stackrel{?}{=} \mathrm{com}(f_1, \ldots, f_n, y_d, f_d; z_d) \\
c_D \textstyle\prod_{i=1}^n c_i^{t_i^2} &\stackrel{?}{=} \mathrm{com}(F_1, \ldots, F_n, f_D, y_D; z_D) \\
\textstyle\prod_{i=1}^n E_i^{f_i} &\stackrel{?}{=} E(0; Z) E_R \textstyle\prod_{i=1}^n e_i^{t_i}
\end{aligned}
$$

Fig. 2. SHVZK Argument of Correct Shuffle Based on Commitment over $\mathbb{Z}_q$.

To see that this is a good simulation, consider the following hybrid argument. We proceed exactly as in the simulation except when forming $c_1, \ldots, c_n$. Here we solve for the valid $d_1, \ldots, d_n$, i.e. set $d_j := \overline{f_j} - t_{\pi(j)}$. We similarly set the variables that differ between the simulation and the real argument, namely the appropriate $D_1, \ldots, D_n$, s_d, s_D, R_R and Δ. Observe that the relationship between the variables generated randomly in the simulation and the variables generated randomly in the real argument are governed by linear equations; hence this endows the hybrid argument with the same distribution of variables as a real argument. We generate $c_i \leftarrow \mathrm{com}(0, \ldots, 0, 1_{\pi^{-1}(i)}, 0, \ldots, 0, 3d_{\pi^{-1}(i)}, 3d^2_{\pi^{-1}(i)}; r_i)$. Proceed with the rest of simulation as described above.

The hybrid argument is statistically indistinguishable from a real argument. On the other hand, the only thing that differs from the simulation is the way we form the c_i's.

The hiding property of the commitment scheme therefore gives us indistinguishability between the hybrid argument and the simulated argument. If the commitment scheme is statistically hiding, then we have statistical indistinguishability between the hybrid argument and the simulated argument.

SOUNDNESS AND KNOWLEDGE. Consider an adversary that has already sent the initial message $(c_1, \ldots, c_n, c_d, c_D, E_R)$ to the verifier and has non-negligible probability ε of answering the challenge. We store the state of this prover and now wish to extract a witness for correctness of the shuffle.

We select at random challenges $t_1, \ldots, t_n$ and run the adversarial prover until we have $n+1$ acceptable answers. We use an expected number of $(n+1)/\varepsilon$ tries to do this. Call the challenges $t_1^{(j)}, \ldots, t_n^{(j)}$ for $j = 0, \ldots, n$ and the corresponding answers $f_1^{(j)}, \ldots, f_n^{(j)}, f_d^{(j)}, y_d^{(j)}, z_d^{(j)}, F_1^{(j)}, \ldots, F_n^{(j)}, y_D^{(j)}, z_D^{(j)}, Z^{(j)}$. Since $c_d \prod_{i=1}^n c_i^{t_i^{(j)}} = \mathrm{com}(f_1^{(j)}, \ldots, f_n^{(j)}, y_d^{(j)}, f_d^{(j)}; z_d^{(j)})$ we have

$$\prod_{i=1}^n c_i^{t_i^{(0)} - t_i^{(j)}} = \mathrm{com}(f_1^{(0)} - f_1^{(j)}, \ldots, f_n^{(0)} - f_n^{(j)}, y_d^{(0)} - y_d^{(j)}, f_d^{(0)} - f_d^{(j)}; z_d^{(0)} - z_d^{(j)}).$$

Consider the $n \times n$ matrix T with entries $t_{ij} = t_i^{(0)} - t_i^{(j)}$. With overwhelming probability over the choices of $t_i^{(j)}$ the columns are linearly independent. We can in polynomial time find the inverse matrix T^{-1} so $TT^{-1} = I$.

Call the entries of T^{-1} as v_{jk}, then we have $\sum_{j=1}^n t_{ij} v_{jk} = \delta_{ik}$. So

$$c_k = c_k^{\sum_{j=1}^n t_{kj} v_{jk}} = \prod_{i=1}^n c_i^{\sum_{j=1}^n t_{ij} v_{jk}} = \prod_{i=1}^n \prod_{j=1}^n c_i^{t_{ij} v_{jk}} = \prod_{j=1}^n \left(\prod_{i=1}^n c_i^{t_i^{(0)} - t_i^{(j)}} \right)^{v_{jk}}.$$

This means

$$c_k = \mathrm{com}(\sum_{j=1}^n v_{jk}(f_1^{(0)} - f_1^{(j)}), \ldots, \sum_{j=1}^n v_{jk}(f_n^{(0)} - f_n^{(j)}),$$

$$\sum_{j=1}^n v_{jk}(y_d^{(0)} - y_d^{(j)}), \sum_{j=1}^n v_{jk}(f_d^{(0)} - f_d^{(j)}); \sum_{j=1}^n v_{jk}(z_d^{(0)} - z_d^{(j)})).$$

By the root extraction property, we can open c_k. We call the opening $(a_{k1}, \ldots, a_{kn}, a_{kD}, a_{kd}, r_k)$. Since $c_d = \mathrm{com}(f_1^{(0)}, \ldots, f_n^{(0)}, y_d^{(0)}, f_d^{(0)}; z_d^{(0)}) \prod_{i=1}^n c_i^{-t_i^{(0)}}$, having openings of $c_1, \ldots, c_n$ means that we can find an opening $(d_1, \ldots, d_n, s_d, \Delta_d, r_d)$ of c_d. Similarly, we can find an opening $(D_1, \ldots, D_n, s_D, \Delta_D, r_D)$ of c_D.

The adversary, having noticeable probability of answering the challenge $t_1, \ldots, t_n$, is forced to use $f_j = d_j + \sum_{i=1}^n a_{ij} t_i$ and $f_d = \Delta_d + \sum_{i=1}^n a_{id} t_i$ and $f_D = \Delta_D + \sum_{i=1}^n a_{iD} t_i$. The equation $f_D = \sum_{j=1}^n f_j^3 - \sum_{i=1}^n t_i^3 - f_d$ implies

$$0 = \sum_{j=1}^n f_j^3 - \sum_{i=1}^n t_i^3 - f_d - f_D = \sum_{j=1}^n f_j^3 - \sum_{i=1}^n t_i^3 - \Delta_d - \sum_{i=1}^n a_{id} t_i - \Delta_D - \sum_{i=1}^n a_{iD} t_i$$

$$= \sum_{j=1}^{n}(d_j + \sum_{i=1}^{n} a_{ij}t_i)^3 - \sum_{i=1}^{n} t_i^3 - \Delta_d - \sum_{i=1}^{n} a_{id}t_i - \Delta_D - \sum_{i=1}^{n} a_{iD}t_i$$

With overwhelming probability over $t_1, \ldots, t_n$ this can only happen if every coefficient is zero (considering this as a multivariate polynomial in the t_i's). Indeed, the coefficient for $t_i t_j t_k$ is $\sum_{h=1}^{n} a_{ih}a_{jh}a_{kh} - \delta_{ijk}$ for all i, j, k. Then lemma 2 tells us that A is a permutation matrix. This means, there exists a permutation π so $a_{\pi(i)i} = 1$ and all other entries are 0.

For the ciphertext equations, we make use of a cofactor matrix as in the proof of the integer scheme. Because the f_j's are greater than 2^{ℓ_t}, we know an overflow did not occur $\bmod q$ and thus the equations $f_j = t_{\pi(j)} + d_j$ hold over $\mathbb{Z}$. Then the proof proceeds the same way as in the integer case, and then by the root extraction property we can find an opening $(0, R_k)$ of $E_k e_{\pi(k)}^{-1}$. Doing so for $k = 1, \ldots, n$ means we have found openings $R_1, \ldots, R_n$ so $E_1 = e_{\pi(1)}E(0; R_1), \ldots, E_n = e_{\pi(n)}E(0; R_n)$. □

5 Comparison

As we mentioned in the introduction, there are many efficient shuffle arguments on different encryption schemes. While our shuffle argument can be used with many different homomorphic cryptosystems, its main advantage is when we look at cryptosystems with large message spaces or large ciphertexts. It is therefore natural to compare it to the shuffle arguments that have been proposed for Paillier encryption.

We compare the efficiency of our shuffle arguments with integer commitments ($\mathcal{SHUF}_{\mathbb{Z}}$) and with commitments over $\mathbb{Z}_q$ ($\mathcal{SHUF}_{\mathbb{Z}_q}$) to those of Nguyen et. al. [NSNK05], Peng et. al. [PBD05], and Wikström (Appendix G) [Wik05b]. We consider all schemes running on a 1024-bit Paillier modulus (giving ciphertexts of size $|N^2| = 2048$ bits) and 80-bit challenges. The reader may download a spreadsheet [GL07] to see compare the schemes for other parameter choices.

For the homomorphic integer commitment, we use a 1024-bit safe prime RSA-modulus as in [DF02], which corresponds to the choice in [Wik05a]. Both his and our scheme become faster if one uses the homomorphic integer commitment from [Gro05a]. Our choice of parameters for [Wik05b] (Appendix G) is $K_1 = 240, K_2 = 1024, K_3 = K_4 = K_5 = 80$, whereas for our scheme it is $\ell_t = 80, \ell_s = 80, \ell_r = 1024$.

For our shuffle over $\mathbb{Z}_q$, we use Pedersen commitments with $|q| = 240, |p| = 1024$, giving us parameters $\ell_t = 80, \ell_s = 80, \ell_r = 240$.

For [NSNK05] we chose $\ell_\eta = 1022, \ell_N = 1024, |N| = 1024, |M| = 592$ in their setup. This corresponds to working with a safe prime Paillier modulus. We do point out that their scheme can also be used for a variant of Paillier encryption that uses a smaller randomizer space. Both their scheme and our schemes are more efficient when used with this variant of Paillier encryption.

The argument in [PBD05] (Protocol 1) relies on a non-standard assumption, the linear ignorance assumption. They have a less efficient protocol 2 that does not rely on this assumption. Other than that their scheme just relies on the semantic security of Paillier encryption, and as in the other schemes we measure its performance on 80-bit challenges (L=80).

The table 1 list the number of exponentiations required for the prover and the verifier, the communication bits, the number of rounds, and the security assumptions. The exponentiations listed are the number of full-length (2048-bit modulus, 1024-bit exponent) exponentiations where we scale for a factor of 3 for doubling the length of the modulus and a factor of 2 for doubling the length of the exponent. We compare all schemes without using multi-exponentiation techniques, since it is situation dependent which techniques work best. Also, we compare all schemes for a deterministic verifier. Using batching techniques it is possible to speed up the verification process in all schemes. The table contains the cost of making the shuffle arguments, it does not include the cost of the shuffle itself.

	[NSNK05]	[PBD05][4]	$\mathcal{SHUF}_{\mathbb{Z}_q}$	[Wik05b]	$\mathcal{SHUF}_{\mathbb{Z}}$
Prover (expo.)	3.4n	5.5n	0.5n	2.3n	0.6n
Verifier (expo.)	5.4n	4.3n	0.4n	1.5n	0.3n
Communication (bits)	9376n	9376n	1504n	6080n	1264n
Rounds	3	4	3	5	3
Privacy	Perm. Hiding	Perm. Hiding	SHVZK	SHVZK	SHVZK

Table 1. Comparison of shuffle arguments with Paillier encryption

6 Acknowledgement

The first author would like to thank Douglas Wikström for helpful discussions.

References

[Abe99] Masayuki Abe. Mix-networks on permutation networks. In *proceedings of ASIACRYPT '99*, pages 258–273, 1999.

[AH01] Masayuki Abe and Fumitaka Hoshino. Remarks on mix-network based on permutation networks. In *proceedings of PKC '01, LNCS series, volume 1992*, pages 317–324, 2001.

[CDS94] Ronald Cramer, Ivan Damgård, and Berry Schoenmakers. Proofs of partial knowledge and simplified design of witness hiding protocols. In *proceedings of CRYPTO '94, LNCS series, volume 893*, pages 174–187, 1994.

[DF02] Ivan Damgård and Eiichiro Fujisaki. A statistically-hiding integer commitment scheme based on groups with hidden order. In *proceedings of ASIACRYPT '02, LNCS series, volume 2501*, pages 125–142, 2002.

[ElG84] Taher ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. In *proceedings of CRYPTO '84, LNCS series, volume 196*, pages 10–18, 1984.

[4] Our numbers deviate from their own estimates in [PBD05] since we compare all schemes without multi-exponentiation or randomization of the verifier and do not count the prize of shuffling itself. Also, we do not use hashing and the random oracle model and thus their protocol becomes a 4 round protocol. Finally, it has been pointed out to us that their protocol is not SHVZK [Wik06], but we guess that it is permutation hiding as defined in [NSNK05].

[FO97] Eiichiro Fujisaki and Tatsuaki Okamoto. Statistical zero knowledge protocols to prove modular polynomial relations. In *proceedings of CRYPTO '97, LNCS series, volume 1294*, pages 16–30, 1997.

[FS01] Jun Furukawa and Kazue Sako. An efficient scheme for proving a shuffle. In *proceedings of CRYPTO '01, LNCS series, volume 2139*, pages 368–387, 2001.

[Fur05] Jun Furukawa. Efficient and verifiable shuffling and shuffle-decryption. *IEICE Transactions*, 88-A(1):172–188, 2005.

[GL07] Jens Groth and Steve Lu. Comparison of shuffle arguments. `http://www.brics.dk/~jg/ShuffleComparisons.xls`, 2007.

[GQ88] Louis C. Guillou and Jean-Jacques Quisquater. A practical zero-knowledge protocol fitted to security microprocessor minimizing both trasmission and memory. In *proceedings of EUROCRYPT '88, LNCS series, volume 330*, pages 123–128, 1988.

[Gro03] Jens Groth. A verifiable secret shuffle of homomorphic encryptions. In *proceedings of PKC '03, LNCS series, volume 2567*, pages 145–160, 2003.

[Gro05a] Jens Groth. Cryptography in subgroups of $\mathbb{Z}_n^*$. In *proceedings of TCC '05, LNCS series, volume 3378*, pages 50–65, 2005.

[Gro05b] Jens Groth. A verifiable secret shuffle of homomorphic encryptions. Cryptology ePrint Archive, Report 2005/246, 2005. http://eprint.iacr.org/.

[Nef01] Andrew C. Neff. A verifiable secret shuffle and its application to e-voting. In *CCS '01*, pages 116–125, 2001. Full paper available at `http://www.votehere.net/vhti/documentation/egshuf.pdf`.

[NSNK04] Lan Nguyen, Reihaneh Safavi-Naini, and Kaoru Kurosawa. Verifiable shuffles: A formal model and a paillier-based efficient construction with provable security. In *proceedings of ACNS '04, LNCS series, volume 3089*, pages 61–75, 2004.

[NSNK05] Lan Nguyen, Reihaneh Safavi-Naini, and Kaoru Kurosawa. A provably secure and effcient verifiable shuffle based on a variant of the paillier cryptosystem. *Journal of Universal Computer Science*, 11(6):986–1010, 2005.

[OT04] Takao Onodera and Keisuke Tanaka. A verifiable secret shuffle of paillier's encryption scheme, 2004. Tokyo Institute of Technology, research report C-193.

[OU98] Tatsuaki Okamoto and Shigenori Uchiyama. A new public-key cryptosystem as secure as factoring. In *proceedings of EUROCRYPT '98, LNCS series, volume 1403*, pages 308–318, 1998.

[Pai99] Pascal Paillier. Public-key cryptosystems based on composite residuosity classes. In *proceedings of EUROCRYPT '99, LNCS series, volume 1592*, pages 223–239, 1999.

[PBD05] Kun Peng, Colin Boyd, and Ed Dawson. Simple and efficient shuffling with provable correctness and zk privacy. In *proceedings of CRYPTO '05, LNCS series, volume 3621*, pages 188–204, 2005.

[Ped91] Torben P. Pedersen. Non-interactive and information-theoretic secure verifiable secret sharing. In *proceedings of CRYPTO '91, LNCS series, volume 576*, pages 129–140, 1991.

[Sch91] Claus-Peter Schnorr. Efficient signature generation by smart cards. *J. Cryptology*, 4(3):161–174, 1991.

[WG06] Douglas Wikström and Jens Groth. An adaptively secure mix-net without erasures. In *proceedings of ICALP '06, LNCS series, volume 4052*, pages 276–287, 2006.

[Wik05a] Douglas Wikström. A sender verifiable mix-net and a new proof of a shuffle. In *proceedings of ASIACRYPT '05, LNCS series, volume 3788*, pages 273–292, 2005.

[Wik05b] Douglas Wikström. A sender verifiable mix-net and a new proof of a shuffle. Cryptology ePrint Archive, Report 2005/137, 2005. http://eprint.iacr.org/.

[Wik06] Douglas Wikström. Private Communication, 2006.