Bull. Korean Math. Soc. **46** (2009), No. 4, pp. 771–787
DOI 10.4134/BKMS.2009.46.4.771

VERIFICATION OF A PAILLIER BASED SHUFFLE USING REPRESENTATIONS OF THE SYMMETRIC GROUP

Soojin Cho and Manpyo Hong

Abstract. We use an idea of linear representations of the symmetric group to reduce the number of communication rounds in the verification protocol, proposed in Crypto 2005 by Peng et al., of a shuffling. We assume Paillier encryption scheme with which we can apply some known zero-knowledge proofs following the same line of approaches of Peng et al. Incidence matrices of 1-subsets and 2-subsets of a finite set is intensively used for the implementation, and the idea of λ-designs is employed for the improvement of the computational complexity.

1. Introduction

There have been enormous amount of research and great improvement on mix-net since the scheme of mix-net [3] was proposed by D. Chaum for the anonymous communication in 1981: New encryption schemes were employed [26, 24, 16], weaknesses pointed out through many analyses [30, 29, 21] of the early construction of mix-net has produced more advanced and securer mix networks [24, 7]. For many different purposes for anonymity, various systems were developed; for web services, real time systems were developed [10], and for mailing services non real time systems like babel, mixmaster and mixminion were cultivated [6]. Network topology and mixing mechanism are some of other concerns in constructing mix-nets [5, 9]. Measuring the anonymity of mix-nets is another important fundamental work [32, 8].

One of the most important matters is on the proof of correctness of the mix net. Roughly speaking, there are two kinds of proof system of mix-nets; one is optimistic and the other is verifiable proof system. The correctness

Received December 15, 2008; Revised December 31, 2008.
2000 *Mathematics Subject Classification.* Primary 94A60; Secondary 11T71.
Key words and phrases. proof of shuffle, mix-net, representation of symmetric group, homomorphic encryption, λ-designs.
The first author was supported by the grant No.20073020 from Ajou University.
The second author was supported by the Ubiquitous Computing and Network(UCN) Project, Knowledge and Economy Frontier R&D Program of the Ministry of Knowledge Economy(MKE) in Korea and a result of subproject UCN 09C1-C5-20S.
The preliminary version of this paper appeared in the proceedings of the International Conference on Information Security and Cryptology (ICISC) 2008.

of the shuffling of the whole mix-net is verified after the mix-net outputs the shuffling results in plain texts in *optimistic* proof system [17], while in *verifiable* proof system each mix server provides proofs of correctness of the shuffling [28, 14, 1, 18, 22, 23, 27, 33].

Verification techniques for mix-nets vary according to the mix-net implementation and vice versa. Permutation network is the framework of the verification protocol proposed by Abe [1], ElGamal re-encryption scheme and Paillier re-encryption scheme are used for the proof system of Furukawa-Sako [14] and Nguyen et al. [23] respectively. A more general scheme than ElGamal is allowed to apply the verification by Neff [22]. In the proof system by Groth [18], additively homomorphic re-encryption scheme is necessary. In sender verifiable mix-net based on ElGamal encryption by Wikström, no re-encryption is needed. In [28], Peng et al. applied the idea of Abe [1] to design a proof system for mix-nets employing ElGamal re-encryption system and a very carefully designed proof system by Peng et al. [27] assumes the additively homomorphic (re-)encryption schemes. A recent work by Groth and Lu [19] is based on the homomorphic encryption scheme also, in which computational complexity has been dramatically reduced.

Our main concern is to implement a verification for a shuffle in a re-encryption mix-net implemented with Paillier encryption scheme: We extend the idea of [27] for the verification of a shuffle by considering pairs of messages rather than single messages. This enables us to do the verification in reduced number of communication rounds than four rounds as in [27]. Computationally improved protocol is also suggested using the idea of λ-designs.

In Section 2 we introduce the related works, explain the motivated idea in Section 3. In Section 4, the basics for our mix-net and shuffling is set up and some basic necessary results are presented. In Section 5, the protocol for the proof is given, while the proofs of its correctness and privacy is given in Section 6 using incidence matrices. In Section 7, we describe an (computationally) improved protocol by applying λ-designs and do a comparison with Peng et al.'s. We conclude with some final remarks.

2. Related works: Furukawa-Sako and Peng et al.

In [14], Furukawa-Sako describes an equivalent condition for a matrix to be a permutation matrix which involves quadratic and cubic relations among the entries of a given matrix. The proof of the shuffle for ElGamal encryption scheme and Paillier encryption scheme have been implemented in [14] and [23] respectively.

Peng et al. in [28], restrict the set of available permutations and employ batch verifications of knowledge to reduce the computational cost of the proof. Recently, Groth and Lu suggested very efficient schemes extending the idea of previously known works on the verification [19].

In [27], Peng et al. proposed a very carefully designed, efficient proof system on mix-nets with additively homomorphic re-encryption scheme: Let $E(m, r)$ be the encryption of the message m with randomizer r and $D(c)$ be the decryption of ciphertext c. When $c_1, c_2, \ldots, c_n$ is the list of cipher texts a shuffling party receives from the previous shuffling party, let $c'_1, c'_2, \ldots, c'_n$ be the list of outputs of the current shuffling party that is supposed to be passed to the next shuffling party. Then, when N is the modulus of the message space, the basic idea of the proof is to let the shuffling party do the zero-knowledge proof of the followings; for random integers s_i, s'_i, $i = 1, 2, \ldots, n$, (s)he knows t_i, t'_i, $i = 1, 2, \ldots, n$, such that

$$\sum_{i=1}^{n} s_i D(c_i) = \sum_{i=1}^{n} t_i D(c'_i) \pmod{N}, \tag{1}$$

$$\sum_{i=1}^{n} s'_i D(c_i) = \sum_{i=1}^{n} t'_i D(c'_i) \pmod{N}, \tag{2}$$

$$\sum_{i=1}^{n} s_i s'_i D(c_i) = \sum_{i=1}^{n} t_i t'_i D(c'_i) \pmod{N}. \tag{3}$$

For the implementation of (1), (2) and (3), they adopted known zero-knowledge proofs of knowledge of logarithm, equality of logarithms and knowledge of root, and has shown that at least four communication rounds are necessary.

3. Motivation: Representation of the symmetric group

We briefly introduce the theory of group representations, especially of the symmetric groups, which has motivated the current work. We refer to [31] and [13] for more detailed argument on the theory of representations.

Definition. A d-dimensional *representation of a group* G over a field $\mathbb{F}$ is a d-dimensional $\mathbb{F}$-vector space V equipped with a group homomorphism

$$\rho : G \to GL(V),$$

where $GL(V)$ is the group of invertible linear transformations from V to V.

Group representations, once a basis of the vector space V is fixed, represents group elements as matrices so that the structure of the given group is preserved. Moreover, $\rho : G \to GL(V)$ defines a G-action on V: For $g \in G$ and $v \in V$,

$$g \cdot v = (\rho(g))(v).$$

A representation V of a group G is *irreducible* if there is no nontrivial invariant subspace W of V under the action of G.

Proposition 3.1 (Maschke's Theorem)**.** *Let V be a representation of a finite group G over a field $\mathbb{F}$. If the characteristic of $\mathbb{F}$ is either 0 or relatively prime to $|G|$, then V is a direct sum of irreducible representations of G.*

The *symmetric group* S_n is a group of $n!$ permutations of n objects equipped with the composition. A *partition* of a positive integer n is a non-increasing finite sequence of positive integers, whose sum is n. For example, $(4,3,3,1)$ is a partition of 11. For a given partition $\nu = (\nu_1, \ldots, \nu_l)$ of n, a *standard ν-tableau* is an (left justified) array of ν_i integers in the ith row so that each column and row are increasing from top to bottom and from left to right respectively, where the set of integers used is exactly $\{1, 2, \ldots, n\}$. We let f^ν denote the number of standard ν-tableaux.

For our case, n (therefore $n!$) can be assumed to be relatively prime to the modulus in use that is a product of big primes. Hence, we can apply Maschke's Theorem to the representations of S_n of our concern.

Proposition 3.2. *There is an irreducible representation S^ν of S_n of dimension f^ν for each partition ν of n. Moreover they form the complete list of irreducible representations of S_n and every representation of S_n is a direct sum of S^ν's.*

To represent every permutation in S_n as an $n \times n$ permutation matrix as Furukawa-Sako did in [14] is a representation of S_n called *defining representation*. Defining representation is basically an action of S_n on the set $\{1, 2, \ldots, n\}$, and is a direct sum of the 1-dimensional trivial representation $S^{(n)}$ and $(n-1)$-dimensional representation $S^{(n-1,1)}$. From this observation, extending the work of Furukawa-Sako, we may consider some natural ways to represent the given permutation in S_n as a matrix of various dimensions. Since the representation Furukawa-Sako used is basically corresponding to the partition $(n-1, 1)$, we may consider the partition $(n-2, 2)$ as a second simplest case: S_n-action on the set of 2-subsets of $\{1, 2, \ldots, n\}$ rather than on the set of 1-subsets of $\{1, 2, \ldots, n\}$. If we call this representation $M^{(n-2,2)}$ and the defining representation $M^{(n-1,1)}$, it is known that

$$M^{(n-2,2)} \cong M^{(n-1,1)} \oplus S^{(n-2,2)} \cong S^{(n)} \oplus S^{(n-1,1)} \oplus S^{(n-2,2)}\,.$$

Note that a permutation in S_n is represented as an $\binom{n}{2} \times \binom{n}{2}$ matrix on the representation $M^{(n-2,2)}$ since it is $\binom{n}{2}$-dimensional (and $\dim S^{(n-2,2)} = \binom{n}{2} - \binom{n}{1}$). Note also that $M^{(n-2,2)}$ contains the defining representation $M^{(n-1,1)}$ as well as the irreducible representation $S^{(n-2,2)}$ which makes our new choice of S_n representation special.

We expect that using (linear) representation $M^{(n-2,2)}$ of S_n instead of $M^{(n-1,1)}$ would let us explain non-linear relations that are indispensable to prove that a given matrix is a permutation matrix in *zero-knowledge* manner: Furukawa-Sako [14] used quadratic and cubic relations of the entries of a given matrix, and Peng et al. used a quadratic relation of the entries of a transformation matrix between two sets of decrypted messages, that a shuffling party used as in the Equation (3).

Using the idea explained above and by extending the work by Peng et al. [27], we can implement a proof system for a shuffling with reduced number of communication rounds than the one in [27] since our proof system does not

need to ask the shuffling party to prove non linear equations. We have to pay more computational cost for the verification in general instead. We, however, can control the level of computation cost by adopting the idea of λ-*designs*.

Our method of proof is to let each shuffling party prove the followings, instead of equations (1), (2) and (3).

For given random integers s_{ij}, $1 \le i \ne j \le n$, he knows t_{ij}'s, such that

$$\sum_{i<j} s_{ij}(D(c_i) + D(c_j)) = \sum_{i<j} t_{ij}(D(c'_i) + D(c'_j)) \pmod N,$$

$$\sum_{i<j} s_{ij} D(c_i) D(c_j) = \sum_{i<j} t_{ij} D(c'_i) D(c'_j) \pmod N.$$

4. Paillier based shuffling

As the proof system of Peng et al. [27] does, our proof system relies on the property of homomorphic encryption. We take Paillier encryption scheme for the underlying encryption and decryption, which is additively homomorphic.

4.1. Paillier encryption scheme

`Key generation:` The modulus of the message space is $N = p\,q$ for two large primes $p < q$. A base $g \in \mathbb{Z}^*_{N^2}$ is an element of order $N\ell$ for some $\ell \in \{1, \ldots, \mu\}$, where μ is the least common multiple of $p-1$ and $q-1$ and $Z^*_{N^2}$ is the multiplicative group of invertible elements in Z_{N^2}.

`Encryption:` For a message $m \in Z_N$, select a random $r \in \mathbb{Z}^*_N$. Then the ciphertext is computed by

$$c = g^m \cdot r^N \pmod{N^2}.$$

`Decryption:` For a given ciphertext $c \in \mathbb{Z}^*_{N^2}$, the plaintext m is computed by

$$m = \frac{L(c^\mu \bmod N^2)}{L(g^\mu \bmod N^2)} \pmod N,$$

where L is the function defined as $L(u) = \frac{u-1}{N}$ for $u \in \mathbb{Z}^*_{N^2}$ such that $u = 1 \pmod N$.

As usual, p and q are one of the security factors of the scheme, so the probability of knowing a factorization of N is negligible and p is comparatively larger than the number of messages. We let α be an integer such that $2^\alpha < p$, as Peng et al. did in [27], that is necessary for the arguments of the protocol security, since we use composite integers for the modulus of the message space.

We let $E(m, r)$ denote the encrypted message of m with randomizer r and $D(c)$ denote the decryption of a cipher text c. We also use $E(m)$ for the encryption of m with some random number when there is no need to specify the random number for the encryption. The re-encryption of a cipher-text c

is denoted by $RE(c,r) = cE(0,r)$. It is well known that Paillier encryption is (additively) homomorphic: We have

$$E(m_1 + m_2, r') = E(m_1, r_1)E(m_2, r_2)\,. \tag{4}$$

For the implementation of the main idea, we let $\widetilde{E}$ be another Paillier encryption defined on the same message space and the ciphertext space but with different bases of the ciphertext space. Then the corresponding decryption $\widetilde{D}$ is multiplicatively homomorphic:

$$\widetilde{D}(e_1e_2) = \widetilde{D}(e_1) + \widetilde{D}(e_2)\,. \tag{5}$$

Following proposition is very basic but useful for us to prove the correctness of our protocol.

Proposition 4.1. *When $0 \in \mathbb{Z}_N$ is the additive identity and $1 \in \mathbb{Z}_{N^2}^*$ is the multiplicative identity, followings are satisfied*:

1. $\widetilde{D}(1) = 0$.
2. $\widetilde{D}(c_1c_2^{-1}) = \widetilde{D}(c_1) - \widetilde{D}(c_2)$ *for all* $c_1, c_2 \in \mathbb{Z}_{N^2}^*$.

Proof. Since $\widetilde{E}(0,r) = 1$ when $r = 1 \in \mathbb{Z}_N^*$ and the encryption is one-to-one, $\widetilde{D}(1) = 0$ must hold. For any $c \in \mathbb{Z}_{N^2}^*$, $\widetilde{D}(cc^{-1}) = \widetilde{D}(1) = 0$ by the first part. On the other hand, $\widetilde{D}(cc^{-1}) = \widetilde{D}(c) + \widetilde{D}(c^{-1})$ by (5). Therefore, we have $\widetilde{D}(c^{-1}) = -\widetilde{D}(c)$ and the second part follows. □

Useful properties of Paillier encryption scheme is stated in the following proposition and corollary.

Proposition 4.2 (Lemma 5 in [25])**.** *The decrypted message of $c \in \mathbb{Z}_{N^2}^*$ is $0 \in \mathbb{Z}_N$ if and only if c is an Nth residue modulo N^2, that is $c = x^N \pmod{N^2}$ for some $x \in \mathbb{Z}_{N^2}^*$.*

Corollary 4.3. *For any choice of bases for E and $\widetilde{D}$, following is satisfied*:

$$\widetilde{D}(E(0,r)) = 0 \text{ for any } r\,.$$

Proof. By Proposition 4.2, it is sufficient to show that $E(0,r) = x^N \pmod{N^2}$ for some $x \in \mathbb{Z}_{N^2}^*$ and this is immediate from the definition of the Paillier scheme. □

4.2. Shuffle

We let $\{m_1, m_2, \ldots, m_n\}$ be the set of original messages and $\{c_1, c_2, \ldots, c_n\}$, $c_i = E(m_i)$, $1 \le i \le n$, be the set of encrypted messages. Then a mix-net contains many rounds of shufflings defined as follows.

Shuffling party receives a set $\{c_1, c_2, \ldots, c_n\}$ of encrypted messages from the previous shuffling party and outputs another set of encrypted messages $\{c'_1, c'_2, \ldots, c'_n\}$ that is obtained as follows: for any i, $c'_i = RE(c_{\pi(i)}, r_i)$ for some permutation $\pi \in S_n$ and randomizers r_i.

A *verification of a shuffle* is a process to verify, without revealing any information, that a shuffling party did the shuffling in an honest way; that is *there is a permutation* $\pi \in S_n$ *such that* $D(c'_i) = D(c_{\pi(i)})$.

4.3. Assumption

We make an assumption that the *linear ignorance condition* for the set $\{m_1, \ldots, m_n\}$ of messages and the set $\{m_i m_j \mid 1 \leq i < j \leq n\}$ of products of messages are satisfied, where the linear ignorance condition is defined as follows. Linear ignorance condition is assumed in the first protocol of [27]:

Definition. A set of messages $\{m_1, m_2, \ldots, m_n\}$ satisfies the *linear ignorance condition* if given a set of cipher-texts $\{c_1, c_2, \ldots, c_n\}$ of $\{m_1, m_2, \ldots, m_n\}$, the possibility for an adversary to find a non-trivial linear relation of $\{m_1, \ldots, m_n\}$ is negligible.

Remark 4.4. We may drop the assumption on the linear ignorance condition if we use the method of the second protocol of Peng et al. in [27]. We do not deal with that matter in the present article though.

4.4. A theorem by Peng et al.

The following proposition is proved in [27], and we state it for our later use. (See Lemmas 1, 2, 3 and 4 in [27].) For a given matrix A, we use A^t for the *transpose matrix of* A.

Proposition 4.5. *Suppose that* $\{m_1, m_2, \ldots, m_n\}$ *satisfies the linear ignorance condition, and let* $\{c'_1, c'_2, \ldots, c'_n\}$ *be the corresponding output of* $\{c_1, c_2, \ldots, c_n\}$ *by a shuffling party. For random numbers* $s_1, s_2, \ldots, s_n$ *from* $\mathbb{Z}_N$*, if the shuffling party can find* $t_1, t_2, \ldots, t_n$ *in* $\mathbb{Z}_N$ *such that*

$$\sum_i s_i D(c_i) = \sum_i t_i D(c'_i) \pmod{N} \tag{6}$$

with a probability larger than $2^{-\alpha}$*, then the shuffling party can find an* $n \times n$ *invertible matrix* P *such that*

$$[D(c'_1), D(c'_2), \ldots, D(c'_n)]^t = P\,[D(c_1), D(c_2), \ldots, D(c_n)]^t \pmod{N}, \tag{7}$$

$$[t_1, t_2, \ldots, t_n]^t = P^{-1}\,[s_1, s_2, \ldots, s_n]^t \pmod{N}. \tag{8}$$

Corollary 4.6. *The matrix* P *in Proposition 4.5 is unique.*

Proof. If there are two different matrices satisfying Equation (7), then one can find a non-trivial linear relation of $\{m_1, m_2, \ldots, m_n\}$. □

5. Verification protocol

In this section, we describe our protocol and prove that an honest shuffling party always can pass the verification.

Suppose that a shuffling party receives $\{c_1, c_2, \ldots, c_n\}$ and the corresponding output is $\{c'_1, c'_2, \ldots, c'_n\}$. We let $m_i = D(c_i)$ and $m'_i = D(c'_i)$ for $i = 1, 2, \ldots, n$.

We also suppose that $d_i = \widetilde{D}(c_i)$ and $d'_i = \widetilde{D}(c'_i)$ are published by an authorized party. The chosen basis for $\widetilde{E}$ and $\widetilde{D}$ does not have to be published since $\widetilde{D}$ is used only for the implementation of the protocol and we just need its multiplicatively homomorphic property:

5.1. Protocol

1. The verifier randomly chooses s_{ij}, $1 \le i < j \le n$, from $\{0, 1, \ldots, 2^\alpha - 1\}$ and publishes them.
2. The shuffling party shows, in a zero knowledge manner, that he knows t_{ij} for $1 \le i < j \le n$ and r_i, $i = 1, 2, \ldots, n$, such that

$$\prod_i c'_j = \prod_i c_i E(0, r_i) \pmod{N^2}, \tag{9}$$

$$\prod_{i<j} (c'_i c'_j)^{t_{ij}} = \prod_{i<j} (c_i c_j)^{s_{ij}} (E(0, r_i) E(0, r_j))^{t_{ij}} \pmod{N^2}, \tag{10}$$

$$\prod_{i<j} (c'^{d'_j}_i c'^{d'_i}_j)^{t_{ij}} = \prod_{i<j} (c^{d_j}_i c^{d_i}_j)^{s_{ij}} (E(0, r_i)^{d'_j} E(0, r_j)^{d'_i})^{t_{ij}} \pmod{N^2}. \tag{11}$$

5.1.1. *Implementation.* The same zero-knowledge implementation used in [27] (see Section 3) can be adopted for the implementation of our protocol due to the fact that Equations (10), (11) are essentially the same as the equations proved in [27] except the number of terms in each product.

Lemma 5.1. *If the shuffling party is honest, then he can pass the verification.*

Proof. Suppose $c'_i = RE(c_{\pi(i)}, r_i)$ are obtained using a permutation $\pi \in S_n$. Then by taking $t_{ij} = s_{\pi(i)\pi(j)}$ ($s_{\pi(j)\pi(i)}$ if $\pi(i) > \pi(j)$) the shuffling party can pass the verification: It is easy to check Equation (10) and we only check Equation (11). Observe first that $d'_i = d_{\pi(i)}$ since

$$d'_i = \widetilde{D}(c'_i) = \widetilde{D}(c_{\pi(i)} E(0, r_i)) = \widetilde{D}(c_{\pi(i)}) + \widetilde{D}(E(0, r_i)) = \widetilde{D}(c_{\pi(i)}) = d_{\pi(i)},$$

where the second last equality is valid because of Corollary 4.3. Now we finish the proof;

$$\begin{aligned} \prod_{i<j} \left(c'^{d'_j}_i c'^{d'_i}_j \right)^{t_{ij}} &= \prod_{i<j} \left(c^{d_{\pi(j)}}_{\pi(i)} c^{d_{\pi(i)}}_{\pi(j)} \right)^{s_{\pi(i)\pi(j)}} \left(E(0, r_i)^{d'_j} E(0, r_j)^{d'_i} \right)^{t_{ij}} \\ &= \prod_{i<j} \left(c^{d_j}_i c^{d_i}_j \right)^{s_{ij}} \left(E(0, r_i)^{d'_j} E(0, r_j)^{d'_i} \right)^{t_{ij}}. \end{aligned}$$

□

6. Proof of the correctness

In this section, we prove that the proposed protocol is a correct verification. Incidence matrix plays an important role for the proof.

6.1. Incidence matrices

We let $[n] = \{1, 2, \ldots, n\}$ be the set of integers from 1 to n. Let $W(n)$ be the *incidence matrix* between 1-subsets and 2-subsets of $[n]$; an $\binom{n}{1} \times \binom{n}{2}$ matrix of 0's and 1's, where the rows and columns of which are indexed by 1-subsets and 2-subsets of $[n]$ respectively and

$$(W(n))_{IJ} = \begin{cases} 1 & \text{if } I \subset J\,, \\ 0 & \text{otherwise}\,. \end{cases}$$

Throughout the rest of this article, let us fix an order on the set of 2-subsets of $[n]$ so that the first n of them are $\{1,2\}, \{2,3\}, \ldots, \{n-1,n\}$ and $\{n-2,n\}$. We always use the fixed order on the set of 1-subsets of $[n]$: $\{1\}, \{2\}, \ldots, \{n\}$.

The following proposition is the main tool that enables us to translate results on the collection of messages to the collection of pairs of messages and vice versa.

Proposition 6.1. *There is an n by n invertible matrix R over $\mathbb{Z}_N$ such that $RW(n) = [I_n \,|\, B] \pmod N$, where I_n is the $n \times n$ identity matrix, and B is an $n \times \left(\binom{n}{2} - \binom{n}{1}\right)$ matrix over $\mathbb{Z}_N$. That is $W(n)$ has rank n over $\mathbb{Z}_N$.*

Proof. Suppose that the rows and columns are indexed in the orders we fixed. Consider the series of row operations that changes the rows of $W(n)$ into $\mathbf{r}_1$, $\mathbf{r}_2 - \mathbf{r}_1$, $\mathbf{r}_3 - \mathbf{r}_2 + \mathbf{r}_1$, $\ldots$, $\mathbf{r}_n - \mathbf{r}_{n-1} + \cdots + (-1)^{n-1}\mathbf{r}_1$, where $\mathbf{r}_i$ is the ith row of $W(n)$. It is easy to see that for every $i = 1, \ldots, n-1$, the $\{i, i+1\}$th column of the new matrix contains only one 1 at the ith row. Moreover, the $\{n-2, n\}$th column is $[0, \ldots, 0, 1, -1, 2]^t$. Note that 2 is a unit in $\mathbb{Z}_N$ since N is a product of powers of odd primes. This enables us to multiply the inverse of 2 to the last row and add it and its negative to the $(n-1)$st row and the $(n-2)$nd row, respectively. The composition of series of row operations we used makes R in the theorem. □

Remark 6.2. $W(n)$ is a special case of well known *incidence matrices* of t-subsets and k-subsets of $[n]$, where $t = 1$, $k = 2$. The rank of incidence matrices are known over $\mathbb{Z}_p$ for a prime p (see [34, 12]): Our $W(n)$ has full rank n over $\mathbb{Z}_p$, $p \neq 2$. However, we use a composite number $N = p\,q$ for the modulus and that is why we need to give a proof of Proposition 6.1.

6.2. The proof

We first start with a key lemma for the proof of the correctness.

Lemma 6.3. *Equations* (9), (10) *and* (11) *imply the following equations, respectively*:

$$\sum_i m_i \;=\; \sum_i m'_i \pmod N\,, \tag{12}$$

$$\sum_{i<j} s_{ij}(m_i + m_j) \;=\; \sum_{i<j} t_{ij}(m'_i + m'_j) \pmod N\,, \tag{13}$$

$$\sum_{i<j} s_{ij}(m_i m_j) \quad = \quad \sum_{i<j} t_{ij}(m'_i m'_j) \pmod N . \tag{14}$$

Proof. The modulus for the equations in the proof vary depending on which space we are working. The modulus is N if we are working on the space of messages, while the modulus must be N^2 if we are working on the space of ciphertexts; and we omit the modulus in the proof.

Equation (12) and (13) are immediate from the additively homomorphic property of our encryption scheme, and we only give a careful proof of Equation (14): Assume Equation (11) is true. Since E is additively homomorphic and $c_i = E(m_i)$, $c'_i = E(m'_i)$ for $i = 1, \ldots, n$, we can rewrite Equation (11) as follows.

$$E\left(\sum_{i<j} s_{ij}(d_i m_j + d_j m_i)\right) = E\left(\sum_{i<j} t_{ij}(d'_i m'_j + d'_j m'_i)\right) .$$

Since encryption is always one-to-one, we obtain

$$\sum_{i<j} s_{ij}(d_i m_j + d_j m_i) = \sum_{i<j} t_{ij}(d'_i m'_j + d'_j m'_i) . \tag{15}$$

Moreover, since $d_i = \widetilde{D}(c_i)$, $d'_i = \widetilde{D}(c'_i)$ for $i = 1, \ldots, n$, and $\widetilde{D}$ is multiplicatively homomorphic, we can rewrite Equation (15) as follows;

$$\widetilde{D}\left(\prod_{i<j} c_i^{s_{ij}m_j} c_j^{s_{ij}m_i}\right) = \widetilde{D}\left(\prod_{i<j} {c'}_i^{t_{ij}m'_j} {c'}_j^{t_{ij}m'_i}\right) .$$

Proposition 4.1 now implies that

$$\widetilde{D}\left(\frac{\prod_{i<j} c_i^{s_{ij}m_j} c_j^{s_{ij}m_i}}{\prod_{i<j} {c'}_i^{t_{ij}m'_j} {c'}_j^{t_{ij}m'_i}}\right) = 0 ,$$

and we obtain

$$\frac{\prod_{i<j} c_i^{s_{ij}m_j} c_j^{s_{ij}m_i}}{\prod_{i<j} {c'}_i^{t_{ij}m'_j} {c'}_j^{t_{ij}m'_i}} = x^N \text{ for some } x \in \mathbb{Z}^*_{N^2}$$

by Proposition 4.2. Let $x = E(y)$ for $y \in \mathbb{Z}_N$ then, again by the additively homomorphic property of E, we have

$$\sum_{i<j} s_{ij}(m_j m_i + m_i m_j) = \sum_{i<j} t_{ij}(m'_j m'_i + m'_i m'_j) + N \cdot y .$$

Note that $N \cdot y = 0 \pmod N$. We finally show that Equation (14) is true. □

Remark 6.4. In the proof of Lemma 6.3, special properties of Paillier scheme (Proposition 4.2 and Corollary 4.3) are essentially used.

In the rest of the article, every equation is modulo N, the modulus of the message space. In the following theorems and lemmas, we suppose that a shuffling party can provide proofs of Equations (9), (10) and (11) with a probability larger than $2^{-\alpha}$.

The following theorem is immediate from Proposition 4.5 since we are assuming the linear ignorance condition on $\{m_i m_j \,|\, 1 \le i < j \le n\}$. Remember that we fix an order of 2-subsets of $[n]$, and we follow the same order for s_{ij}'s, t_{ij}'s, $m_i m_j$'s and $m'_i m'_j$'s.

Theorem 6.5. *There is an invertible $\binom{n}{2} \times \binom{n}{2}$ matrix $\widetilde{P}$ over $\mathbb{Z}_N$ such that*

$$\begin{aligned}[m'_1 m'_2, m'_2 m'_3, \dots]^t &= \widetilde{P}\,[m_1 m_2, m_2 m_3, \dots]^t \text{ and} \\ [t_{1\,2}, t_{2\,3}, \dots]^t &= (\widetilde{P})^{-1}\,[s_{1\,2}, s_{2\,3}, \dots]^t\,.\end{aligned}$$

Theorem 6.6. *The shuffling party knows an invertible $n \times n$ matrix P over $\mathbb{Z}_N$ such that*

$$[m'_1, m'_2, \dots, m'_n]^t = P\,[m_1, m_2, \dots, m_n]^t\,.$$

Proof. By Proposition 6.1, for random $s_1, s_2, \dots, s_n$, one always can find s_{ij}'s in $\mathbb{Z}_N$ satisfying $\sum_{k \in \{i,j\}} s_{ij} = s_k$ for each k. We just have to solve $W(n)\,X = [s_1, \dots, s_n]^t$ for $X = [x_{12}, x_{23}, \dots]^t$. Therefore, the existence of P in the theorem is guaranteed by Proposition 4.5, since we assume the linear ignorance condition on $\{m_1, m_2, \dots, m_n\}$. □

We state two Lemmas whose proofs can be done by comparing coefficients of m_i's or $m_i m_j$'s in appropriate equations.

Lemma 6.7. *For any $1 \le i < j \le n$ and $1 \le k < l \le n$,*

$$\widetilde{P}_{\{ij\}\{kl\}} = P_{ik} P_{jl} + P_{il} P_{jk}\,. \tag{16}$$

Proof. We have $m'_i m'_j = \sum_{\alpha<\beta} \widetilde{P}_{\{ij\}\{\alpha\beta\}} m_\alpha m_\beta = (\sum_\alpha P_{i\alpha} m_\alpha)(\sum_\beta P_{j\beta} m_\beta)\,,$ where the first and the second equations are from Theorem 6.5 and Theorem 6.6 respectively. We now compare the coefficients of $m_i m_j$'s to finish the proof. Remember that we are assuming the linear ignorance condition on the set $\{m_i m_j \,|\, i < j\}$ and this makes comparing the coefficients of $m_i m_j$'s be a valid argument. □

Lemma 6.8. *When t_{ij}'s are the numbers provided by the shuffling party for given s_{ij}'s, the following equations are satisfied for any i,j and k.*

$$\sum_{k \in \{\alpha,\beta\}} s_{\alpha\beta} = \sum_{\alpha<\beta} (P_{\alpha k} + P_{\beta k}) t_{\alpha\beta}\,, \tag{17}$$

$$(W(n)\widetilde{P})_{\{k\}\{i\,j\}} = P_{ik} + P_{jk}\,. \tag{18}$$

Proof. If we rewrite Equation (13) using matrix P, then we have the following:

$$\sum_{\alpha<\beta} s_{\alpha\beta}(m_\alpha + m_\beta) = \sum_{\alpha<\beta} t_{\alpha\beta} \left(\sum_{\gamma=1}^{n} P_{\alpha\gamma} m_\gamma + \sum_{\gamma=1}^{n} P_{\beta\gamma} m_\gamma \right).$$

Equation (17) can be obtained by comparing the coefficients of each m_k's. Note here that we are assuming the linear ignorance condition on $\{m_1, \ldots, m_n\}$ and this makes comparing the coefficients of m_k's be a valid argument.

We have $t_{\alpha\beta} = (\widetilde{P})^{-1}[s_{12}, s_{23}, \ldots]^t$ because of Theorem 6.5. Hence, the right hand side of Equation (17) is the kth row of $A(\widetilde{P})^{-1}[s_{12}, s_{23}, \ldots]^t$, where A is an $n \times \binom{n}{2}$ matrix (with the same index sets as $W(n)$'s), whose $(k, \{\alpha\beta\})$-entry is given by $P_{\alpha k} + P_{\beta k}$. The left hand side of Equation (17) is the kth row of $W(n)[s_{12}, s_{23}, \ldots]^t$. We, therefore, can conclude that $W(n)\widetilde{P} = A$ since s_{ij} are randomly chosen numbers by the verifier. □

Theorem 6.9. *The matrix P given in Theorem 6.6 satisfies the following equations, and is a permutation matrix with high probability as a consequence.*

$$\text{For all } i,\ \sum_{\alpha=1}^{n} P_{i\alpha} = 1\,, \tag{19}$$

$$\text{for all } i < j \text{ and } k,\ P_{ik} + P_{jk} = P_{ik} \sum_{\alpha \neq k} P_{j\alpha} + P_{jk} \sum_{\alpha \neq k} P_{i\alpha}\,. \tag{20}$$

Proof. Equation (19) is immediate from Equation (12) and Theorem 6.6:

$$m_1 + \cdots + m_n = m'_1 + \cdots + m'_n = (\sum_{\alpha} P_{1\alpha})m_1 + \cdots + (\sum_{\alpha} P_{n\alpha})m_n\,.$$

For any $i < j$ and k, since

$$P_{ik} + P_{jk} = P_{ik} \sum_{\alpha \neq k} P_{j\alpha} + P_{jk} \sum_{\alpha \neq k} P_{i\alpha} = P_{ik}(1 - P_{jk}) + P_{jk}(1 - P_{ik})\,,$$

we have $2P_{ik}P_{jk} = 0$. In our case, this implies that either $P_{ik} = 0$ or $P_{jk} = 0$ since the factorization of modulus N is assumed to be very hard and, it is not known to the shuffling parties or N is a prime. Therefore, there can be at most one non-zero entry in each column of P. Since P is invertible, P can not have a zero column and each column must have exactly one non-zero entry, that is there are exactly n non-zero entries in P. Since there can be no zero row in P there must be exactly one non-zero entry in each row, and Equation (19) proves that each non-zero entry must be 1. □

Remark 6.10. Two conditions in Theorem 6.9 are sufficient for a matrix P to be a permutation matrix if the modulus is a prime as in the case of modified ElGamal. Therefore, our work can also be thought as an extension of the work by Furukawa-Sako in [14].

Through the arguments of Theorem 6.5, Theorem 6.6 and Theorem 6.9, we have shown the following.

Theorem 6.11. *Suppose that a shuffling party can provide proofs of Equations* (9), (10) *and* (11) *with a probability larger than $2^{-\alpha}$, and assume the*

linear ignorance condition for the sets $\{m_1, m_2, \ldots, m_n\}$ *and* $\{m_i m_j \mid i < j\}$. *Then there is a permutation matrix* P *such that*

$$[m'_1, m'_2, \ldots, m'_n]^t = P\,[m_1, m_2, \ldots, m_n]^t\,.$$

7. Computationally improved protocol

In the previous sections, we proved that the proposed protocol is a valid verification that can be implemented with reduced number of communication rounds between the shuffling party and the verifier than in the protocol by Peng et al. [27]. However, the proposed protocol has a significant draw-back in computation complexity. The complexity of our protocol is $O(n^2)$ while the one by Peng et al. has $O(n)$ as its computational complexity. This is because we use all $\binom{n}{2}$ 2-subsets of $[n]$ for the verification. We, however, can restrict the number of nonzero s_{ij}'s so that we obtain a linear complexity: We must be careful to choose nonzero s_{ij}'s so that we do not lose the balance of i's, though.

7.1. Designs

A good method to choose $\{i, j\}$'s for nonzero s_{ij}'s is to use λ-*designs*. We refer [2, 20] for more detailed argument on design theory.

Definition. For integers $0 \le t < k < n$ and $0 < \lambda$, a t-(n, k, λ) *design* is a collection $\mathcal{B}$ of k-subsets of $[n]$ called *blocks*, with the property that every t-subset of $[n]$ is contained in precisely λ blocks.

The following proposition gives a necessary condition for the existence of a t-(n, k, λ) design.

Proposition 7.1 (Theorem 19.2 in [20])**.** *If* $\mathcal{B}$ *is a* t-(n, k, λ) *design, then the number of blocks is*

$$|\mathcal{B}| = \frac{\lambda\binom{n}{t}}{\binom{k}{t}}\,.$$

What we need is a collection of 2-subsets of $[n]$ in which 1-subsets of $[n]$ are evenly distributed, that is a 1-$(n, 2, \lambda)$ design for some λ. We will simply call a 1-$(n, 2, \lambda)$ design a λ-*design*.

Corollary 7.2. *If* $\mathcal{B}$ *is a* λ*-design, then either* λ *or* n *is an even integer.*

Proof. If $\mathcal{B}$ is a λ-design, then $|\mathcal{B}| = \frac{\lambda n}{2}$ must be an integer by Proposition 7.1. □

Suppose that n is even, then by pairing elements in $[n]$ in λ different ways we can easily construct a λ-design. When n is odd and λ is even, we can consider λ-copies of $[n]$ and in each copy make $\frac{n-1}{2}$ with one element left. We can assume that λ elements left from each copy are all distinct and can pair them up to make a λ-design. This proves the following easy theorem.

Theorem 7.3. *A λ-design exists if and only if either λ or n is an even integer.*

In conclusion, when n is even we easily can construct a λ-design for any choice of λ and the number of blocks is $\frac{\lambda n}{2}$, and when n is odd we can construct a λ-design for even λ.

7.1.1. *Improved protocol.* If the number n of messages is odd, then we always can add one more dummy message as the last one and we may assume that n is even.

1. The verifier construct a λ-design $\mathcal{B}$, randomly chooses s_{ij}, $\{i,j\} \in \mathcal{B}$, from $\{0,1,\ldots,2^{\alpha}-1\}$ and publishes them.
2. The shuffling party shows, in a zero knowledge manner, that he knows t_{ij} for $1 \le i < j \le n$ and r_i, $i = 1,2,\ldots,n$, such that Equations (9), (10) and (11) are satisfied.

By employing the idea of λ-designs, we still can keep the main idea of our protocol and get a reasonable computation cost with a complexity in the class $O(n)$ also.

7.2. Comparison

The computation cost for a verification process depends on the method of implementation of zero-knowledge proofs and the encryption scheme. Since our protocol is based on the Pillier encryption and the equations to be proved by the shuffling party are essentially the same as the ones in [27], we can directly employ the implementation of zero-knowledge proofs proposed in [27] for the proofs of (9), (10) and (11). Moreover, we can estimate the computation cost for the verification by employing the same method Peng et al. used in [27] and compare the cost of our protocol with Peng et al.'s.

There are more recent works other than [27], where better efficiencies has been achieved ([19, 33]) in computation. In [19], the comparison in efficiency has been made among verification protocols including the one by Peng et al. [27] and the one by Groth and Lu [19]. We, hence, compare the efficiency of our protocol with Peng et al.'s only.

As in [27], we assume that the cost of exponentiation with x-bit exponent is $1.5x$, and the cost of the product of n exponentiations with x-bit exponent is at most $n + 0.5nx$.

The cost of computing d_i's and d_i''s is about $\left(\frac{4n}{3} + \frac{8\gamma}{3}\right)$ full length exponentiations, where $\gamma = \frac{n}{|N|}$, since they are decryption processes in Paillier scheme.

Assuming that $\alpha = 20$ and N is a 1024 bit number ($|N| = 1024$), A rough (not sharp) upper bound for the cost of verification is $\frac{10}{3}\lambda n$ (full length exponentiations). Therefore, the overall cost for the verification is about $(\frac{4}{3} + \frac{10}{3}\lambda)n$ if a λ-design is employed. When $\lambda = 1$, the computational cost is not much worse than that of Peng et al., where the number of communication round is reduced. We did not do careful analysis on the security matter raised from the

choice of λ, but believe that $\lambda = 1$ (or a small λ) will serve as a good parameter. In the following table we compare our verification with the first protocol of Peng et al. in [27] in terms of number of full exponentiations, where we do not include the cost for the shuffling.

	Communication rounds	Cost for verification $\lambda = 1$	Cost for verification $\lambda = 2$
Peng et al.	4	$< 4n$	$< 4n$
Our protocol	2	$< 14n/3$	$< 8n$

8. Final remarks

Two conditions in Theorem 6.9 are sufficient for a matrix P to be a permutation matrix if the modulus is a prime as in the case of modified ElGamal. But, when the modulus is a composite integer they do not give sufficient conditions for P to be a permutation matrix. We, in Theorem 6.9, conclude P is a permutation matrix due to hardness of factorization of N. The following example shows that P does not have to be a permutation matrix without this assumption on the modulus.

Example 8.1. When $N = 1453 \cdot 3019 = 4386607$, the following non-permutation matrix satisfies the conditions in Theorem 6.9:

$$P = \begin{bmatrix} 271711 & 4114897 & 0 & 0 \\ 4114897 & 271711 & 0 & 0 \\ 0 & 0 & 271711 & 4114897 \\ 0 & 0 & 4114897 & 271711 \end{bmatrix}.$$

By defining $\widetilde{P}$ by $\widetilde{P}_{\{ij\}\{kl\}} = P_{ik}P_{jl} + P_{il}P_{jk}$, as in Equation (16) and providing t_{ij}'s calculated by $(\widetilde{P})^{-1}[s_{12}, s_{23}, \dots]^t$, a shuffling party can pass the verification while passing contaminated messages: $[m'_1, \dots, m'_n]^t = P[m_1, \dots, m_n]^t$.

Remark 8.2. The matrix P in Example 8.1 can also pass all the verification of Peng et al. in [27]. Peng et al., however, did not give a correct explanation how this can happen in [27]: In their proof of the main theorem, they made a wrong reasoning by overlooking the fact that the message space may have composite modulus (the third paragraph in p. 197 of [27]). This can be treated though by taking the same argument we use for the proof of the correctness of our protocol.

We proposed a verification protocol with reduced number of rounds of communications than the one in [27] where Paillier encryption is adopted. We believe that our argument can be extended to other homomorphic encryption schemes, like newly proposed (doubly homomorphic) encryption by [15].

References

[1] M. Abe, *Mix-networks on permutation networks*, Advances in cryptology—ASIACRYPT '99 (Singapore), 258–273, Lecture Notes in Comput. Sci., 1716, Springer, Berlin, 1999.

[2] P. J. Cameron and J. H. van Lint, *Designs, Graphs, Codes and Their Links*, London Mathematical Society Student Texts, 22. Cambridge University Press, Cambridge, 1991.

[3] D. Chaum, *Untraceable electronic mail, return addresses, and digital pseudonyms*, Commun. ACM **24** (1981), no. 2, 84–88.

[4] S. Cho and M. Hong, *Proving a shuffle using representations of the symmetric group*, ICISC 2008 (P. J. Lee and J. H. Cheon, eds.), 354–367, Lecture Notes in Computer Science, vol. 5461, Springer, 2009.

[5] G. Danezis, *Mix-networks with restricted routes*, Privacy Enhancing Technologies (Roger Dingledine, ed.), 1–17, Lecture Notes in Computer Science, vol. 2760, Springer, 2003.

[6] G. Danezis, R. Dingledine, and N. Mathewson, *Mixminion: Design of a type iii anonymous remailer protocol*, IEEE Symposium on Security and Privacy, 2–15, IEEE Computer Society, 2003.

[7] Y. Desmedt and K. Kurosawa, *How to break a practical mix and design a new one*, EUROCRYPT, 557–572, 2000.

[8] C. Díaz, S. Seys, J. Claessens, and B. Preneel, *Towards measuring anonymity*, in Dingledine and Syverson [11], pp. 54–68.

[9] R. Dingledine, M. J. Freedman, D. Hopwood, and D. Molnar, *A reputation system to increase mix-net reliability*, Information Hiding (Ira S. Moskowitz, ed.), 126–141, Lecture Notes in Computer Science, vol. 2137, Springer, 2001.

[10] R. Dingledine, N. Mathewson, and P. F. Syverson, *Tor: The second-generation onion router*, USENIX Security Symposium, 303–320, USENIX, 2004.

[11] R. Dingledine and P. F. Syverson (eds.), *Privacy enhancing technologies*, Second international workshop, pet 2002, san francisco, ca, usa, april 14-15, 2002, revised papers, Lecture Notes in Computer Science, vol. 2482, Springer, 2003.

[12] P. Frankl, *Intersection theorems and mod p rank of inclusion matrices*, J. Combin. Theory Ser. A **54** (1990), no. 1, 85–94.

[13] W. Fulton and J. Harris, *Representation Theory, A First Course*, Graduate Texts in Mathematics 129, Springer 1991.

[14] J. Furukawa and K. Sako, *An efficient scheme for proving a shuffle*, Advances in cryptology—CRYPTO 2001 (Santa Barbara, CA), 368–387, Lecture Notes in Comput. Sci., 2139, Springer, Berlin, 2001.

[15] E.-J. Goh, *Encryption schemes from bilinear maps*, Ph. D. thesis, Department of Computer Science, Stanford University, Sep. 2007.

[16] P. Golle, M. Jakobsson, A. Juels, and P. F. Syverson, *Universal re-encryption for mixnets*, Topics in cryptology—CT-RSA 2004, 163–178, Lecture Notes in Comput. Sci., 2964, Springer, Berlin, 2004.

[17] P. Golle, S. Zhong, D. Boneh, M. Jakobsson, and A. Juels, *Optimistic mixing for exit-polls*, Advances in cryptology—ASIACRYPT 2002, 451–465, Lecture Notes in Comput. Sci., 2501, Springer, Berlin, 2002.

[18] J. Groth, *A verifiable secret shuffle of homomorphic encryptions*, Public key cryptography—PKC 2003, 145–160, Lecture Notes in Comput. Sci., 2567, Springer, Berlin, 2002.

[19] J. Groth and S. Lu, *Verifiable shuffle of large size ciphertexts*, Public key cryptography—PKC 2007, 377–392, Lecture Notes in Comput. Sci., 4450, Springer, Berlin, 2007.

[20] J. H. van Lint and R. M. Wilson, *A Course in Combinatorics*, Cambridge University Press, Cambridge, 1992.

[21] M. Mitomo and K. Kurosawa, *Attack for flash MIX*, Advances in cryptology—ASIACRYPT 2000 (Kyoto), 192–204, Lecture Notes in Comput. Sci., 1976, Springer, Berlin, 2000.

[22] C. A. Neff, *A verifiable secret shuffle and its application to e-voting*, ACM Conference on Computer and Communications Security, 116–125, 2001.
[23] L. Nguyen, R. Safavi-Naini, and K. Kurosawa, *Verifiable shuffles: A formal model and a Paillier-based efficient construction with provable security*, ACNS (Markus Jakobsson, Moti Yung, and Jianying Zhou, eds.), 61–75, Lecture Notes in Computer Science, vol. 3089, Springer, 2004.
[24] W Ogata, K Kurosawa, K Sako, and K Takatani, *Fault tolerant anonymous channel*, Proc. ICICS '97, 440–444, Lecture Notes in Comput. Sci., 1334, Springer-Verlag, 1997.
[25] P. Paillier, *Public-key cryptosystems based on composite degree residuosity classes*, Advances in cryptology—EUROCRYPT '99 (Prague), 223–238, Lecture Notes in Comput. Sci., 1592, Springer, Berlin, 1999.
[26] C. Park, K. Itoh, and K. Kurosawa, *Efficient anonymous channel and all/nothing election scheme*, Advances in cryptology—EUROCRYPT '93 (Lofthus, 1993), 248–259, Lecture Notes in Comput. Sci., 765, Springer, Berlin, 1994.
[27] K. Peng, C. Boyd, and E. Dawson, *Simple and efficient shuffling with provable correctness and ZK privacy*, Advances in cryptology—CRYPTO 2005, 188–204, Lecture Notes in Comput. Sci., 3621, Springer, Berlin, 2005.
[28] K. Peng, C. Boyd, E. Dawson, and K. Viswanathan, *A correct, private, and efficient mix network*, Public key cryptography—PKC 2004, 439–454, Lecture Notes in Comput. Sci., 2947, Springer, Berlin, 2004.
[29] B. Pfitzmann and A. Pfitzmann, *How to break the direct RSA-implementation of mixes*, EUROCRYPT, 373–381, 1989.
[30] B. Pfitzmann, M. Schunter, and M. Waidner, *How to break another provably secure payment system*, EUROCRYPT, 121–132, 1995.
[31] B. E. Sagan, *The symmetric group. Representations, combinatorial algorithms, and symmetric functions*, The Wadsworth & Brooks/Cole Mathematics Series. Wadsworth & Brooks/Cole Advanced Books & Software, Pacific Grove, CA, 1991.
[32] A. Serjantov and G. Danezis, *Towards an information theoretic metric for anonymity*, in Dingledine and Syverson [11], pp. 41–53.
[33] D. Wikström, *A sender verifiable mix-net and a new proof of a shuffle*, Advances in cryptology—ASIACRYPT 2005, 273–292, Lecture Notes in Comput. Sci., 3788, Springer, Berlin, 2005.
[34] R. M. Wilson, *A diagonal form for the incidence matrices of t-subsets vs. k-subsets*, European J. Combin. **11** (1990), no. 6, 609–615.

Soojin Cho
Department of Mathematics
Ajou University
Suwon 443-749, Korea
E-mail address: chosj@ajou.ac.kr

Manpyo Hong
Department of Information and Computer Engineering
Ajou University
Suwon 443-749, Korea
E-mail address: mphong@ajou.ac.kr