Efficient Anonymous Channel and All/Nothing Election Scheme

Choonsik PARK, Kazutomo ITOH and Kaoru KUROSAWA

Department of Electrical and Electronic Engineering,
Faculty of Engineering, Tokyo Institute of Technology
2-12-1 O-okayama, Meguro-ku, Tokyo 152 Japan
parkcs@ss.titech.ac.jp
kkurosaw@ss.titech.ac.jp

Abstract. The contribution of this paper are twofold. First, we present an efficient computationally secure anonymous channel which has no problem of ciphertext length expansion. The length is irrelevant to the number of MIXes (control centers). It improves the efficiency of Chaum's election scheme based on the MIX net automatically. Second, we show an election scheme which satisfies fairness. That is, if some vote is disrupted, no one obtains any information about all the other votes. Each voter sends $O(nk)$ bits so that the probability of the fairness is $1 - 2^{-k}$, where n is the bit length of the ciphertext.

1 Introduction

Chaum showed a computationally secure anonymous channel called a MIX net [1]. It hides even the traffic pattern, that is, who sends whom. The MIX net consists of a series of control centers called MIXes. However, the length of the ciphertext which each sender sends is very large. It grows proportionally to the number of MIXes.

Anonymous channels and election schemes are closely related to each other. An anonymous channel hides the correspondences between the senders and the receivers. An election scheme hides the correspondences between the voters and the content of each vote. From this point of view, Chaum proposed an election scheme based on the MIX net [1]. However, the election scheme based on the MIX net provides very low level of correctness. It doesn't satisfy even fairness. That is, suppose that only one vote is disrupted. Still, everyone can know all the other votes in his election scheme. Then, this information will influence the re-election greatly.

Chaum showed another anonymous channel called a DC net [3], and an election scheme based on the DC net [2]. While the DC net is unconditionally secure, the participants must share random numbers beforehand. It also has a problem of message collision. The election scheme based on the DC net has the same problems.

Benaloh showed a totally different yes/no election scheme which is based on zero knowledge interactive proof systems (ZKIP) and secret sharing schemes [4].

Benaloh's scheme provides very high level of correctness, that is, fault tolerancy. The total number of yes votes is successfully obtained even if less than a half of control centers (corresponding to MIXes) are dishonest. However, the disadvantage of Benaloh's scheme is efficiency. Let p_i be the cheating probability of voter i. To obtain that $p_i \leq 2^{-k}$, each voter has to send $O(nkN)$ bits, where n = the size of each ciphertext and N = the number of the control centers.

The contribution of this paper are twofold. First, we present an efficient computationally secure anonymous channel which has no problem of ciphertext length expansion. The length is irrelevant to the number of MIXes. It improves the efficiency of Chaum's election scheme based on the MIX net automatically. Second, we show an election scheme which satisfies the fairness. That is, if some vote is disrupted, no one obtains any information about all the other votes. Each voter sends $O(nk)$ bits so that the probability of the fairness is $1 - 2^{-k}$, where n is the bit length of the ciphertext.

2 Chaum's Work

2.1 Basic Usage of Public Key

Let E_A be a public key and E_A^{-1} be a secret key of Alice. We assume that, for any X,

$$E_A^{-1}E_A(X) = E_A E_A^{-1}(X) = X. \tag{1}$$

Let M_i be a plaintext and C_i be the ciphertext $(1 \leq i \leq n)$. Suppose that M_i and C_i are made public. Also suppose that n is small enough. When we want to hide the correspondence between M_i and C_i, each M_i should be encrypted as follows.

$$C_i = E_A(M_i \circ R_i),$$

where R_i is a random number. If R_i is not attached, it is easy to find the correspondence between M_i and C_i.

The digital signature for a random number M can be given by

$$D = E_A^{-1}(M \circ 0^l).$$

Everyone can verify the validity of the signature by forming

$$E_A(D) = M \circ 0^l$$

and by checking 0^l, where l is a sufficiently large number.

2.2 Anonymous MIX Channel

Chaum showed a scheme which hides even the traffic pattern. The model is as follows. There are n senders, $A_1, \ldots, A_n$. Each A_i wants to send a message m_i to a receiver B_i in such a way that the correspondence between A_i and B_i is kept secret. It is assumed that there exists a shuffle machine agent S_1 (called a MIX). Let the public key of B_i be E_{B_i} and the public key of S_1 be E_1.

An anonymous channel is realized by the following protocol.

[Simple MIX Anonymous Channel]

Step 1. Each A_i chooses a random number R and writes

$$C_i = E_1(R \circ B_i \circ E_{B_i}(m_i)) \tag{2}$$

on the public board.

Step 2. S_1 decrypts it, throws away R, and writes $\{B_i \circ E_{B_i}(m_i)\}$ on the public board in a lexicographical order.

In this protocol, anyone except for S_1 cannot see the correspondence between $\{A_i\}$ and $\{B_i\}$. To hide the correspondence even from S_1, k MIXes $S_1, \ldots, S_k$ are used. The protocol is as follows. Let E_i be the public key of S_i.

[k MIXes Anonymous Channel]

Step 1. Each A_i chooses random numbers $R_1, \ldots, R_k$ and writes

$$E_1(R_1 \circ E_2(R_2 \cdots E_k(R_k \circ B_i \circ E_{B_i}(m_i)) \cdots))$$

on the public board. (We say that A_i sends $B_i \circ E_{B_i}(m_i)$ to the k MIXes anonymous channel.)

Step 2. S_1 writes

$$E_2(R_2 \cdots E_k(R_k \circ B_i \circ E_{B_i}(m_i)) \cdots)$$

on the public board in a lexicographical order.

Step 3. $S_2, S_3, \ldots,$ and S_{k-1} execute the same job as Step 2 in sequence.

Step 4. Finally, S_k writes $B_i \circ E_{B_i}(m_i)$ on the public board in a lexicographical order.

In this protocol, if at least one MIX is honest, the correspondence between $\{A_i\}$ and $\{B_i\}$ is kept secret even from the MIXes.

2.3 Election Scheme

Chaum proposed an election scheme based on the k MIXes anonymous channel. In the k MIXes anonymous channel, if S_k is dishonest, S_k may write something other than $B_i \circ E_{B_i}(m_i)$ on the public board. A_i can detect this error. However, if A_i claims, S_k can know the correspondence between A_i and B_i because S_k knows B_i. This is a serious problem if the anonymous channel is used for an election scheme. To overcome this problem, Chaum proposed the following election scheme.

Let P_i be a voter and V_i be his vote.

(Registration phase)

Step 1. Each P_i chooses (K_i, K_i^{-1}), where K_i is a public key and K_i^{-1} is the secret key. P_i writes

$$E_1(R_1 \circ E_2(R_2 \cdots E_k(R_k \circ K_i) \cdots))$$

on the public board with his digital signature. (P_i sends K_i to the k MIXes anonymous channel. In step 1 of the k MIXes anonymous channel, $B_i \circ E_{Bi}(m_i)$ is replaced by K_i.)

Step 2. The k MIXes anonymous channel shuffles $\{K_i\}$ in secret. (Step 2 and 3 of the k MIXes anonymous channel are executed.)

Step 3. S_k writes K_i on the public board in the lexicographical order.

Let the list be $(\hat{K}_1, \hat{K}_2, \ldots)$.
(Claiming phase)

Step 4. Each P_i checks that his K_i is in the list on the public board. If not, P_i claims and the election stops. If there are no claims in some period, goto the next phase.

(Voting phase)

Step 5. Each P_i writes

$$E_1(R_1 \circ E_2(R_2 \cdots E_k(R_k \circ (K_i \circ K_i^{-1}(V_i \circ 0^l))) \cdots))$$

on the public board with his digital signature. (P_i sends $K_i \circ K_i^{-1}(V_i \circ 0^l$) to the k MIXes anonymous channel.)

Step 6. After the deadline of the voting period, the k MIXes anonymous channel shuffles $K_i \circ K_i^{-1}(V_i \circ 0^l)$ in secret.

Step 7. S_k writes $K_i \circ K_i^{-1}(V_i \circ 0^l)$ on the public board in the lexicographical order. Let the list be $(u_1 \circ v_1), (u_2 \circ v_2), \ldots$.

Step 8. Everyone checks that $u_i = \hat{K}_i$, and $u_i(v_i) = * \cdots * 0^l$ for each i. If the check fails, stop.

Step 9. It is easy for everyone to obtain $\{V_1, \ldots, V_n\}$.

Remark. At Step 1 and Step 5, digital signatures are necessary to check the voters' identities.

3 Proposed Anonymous Channels

The problem of the k MIXes anonymous channel shown in 2.2 is that each sender A_i has to send a very long message at step 1. The length of $E_1(R_1 \circ E_2(R_2 \cdots E_k(R_k \circ B_i \circ E_{B_i}(m_i)) \cdots))$ is proportional to k, which is the number of the MIXes.

In this section, we will present an anonymous channel which has no problem of such ciphertext length expansion.

[Proposed Anonymous Channel]

The proposed scheme makes use of ElGamal cryptosystem. The authority publishes (q, g, c), where

- q is a large prime number.
- g is a primitive element of $GF(q)$.
- c is the factorization of $q-1$. (Everyone can check that g is a primitive element by using c.)

(Secret key of S_i) $X_i \in \{1, \ldots, q-1\}$
(Public key of S_i) Y_i $(= g^{X_i} \bmod q)$
(S_i chooses X_i and publicizes Y_i.)

Step 1. Each sender A_i chooses a random number R and computes

$$(C_{0i}, C_{1i}) \triangleq (g^R, (B_i \circ E_{B_i}(m_i)) \times (Y_1 \cdots Y_k)^R)$$

A_i writes (C_{0i}, C_{1i}) on the public board. Define $f_j(t, u, r)$ as

$$f_j(t,u,r) \triangleq \begin{cases} (t \times g^r, u \times (Y_{j+1} \cdots Y_k)^r / t^{X_j}) & \text{if } 1 \le j \le k-1 \\ u / t^{X_k} & \text{if j = k} \end{cases}$$

For $i = 1, \ldots, k$, do the following.

Step 2. Let the latest list on the public board be

$$(t_1, u_1), (t_2, u_2), \ldots, (t_n, u_n).$$

S_i chooses random numbers $r_1, \ldots, r_n$ and computes $f_i(t_j, u_j, r_j)$ for each j.

Step 3. S_i writes $\{f_i(t_j, u_j, r_j)\}$ $(j = 1, \ldots, n)$ on the public board in a lexicographical order.

Finally, we have a list of $\{B_i \circ E_{B_i}(m_i)\}$ in a lexicographical order on the public board.

In this protocol, (C_{0i}, C_{1i}) changes as follows for some random numbers $R_1, \ldots, R_{k-1}$.

$$\begin{aligned} (C_{0i}, C_{1i}) &= (g^R, (B_i \circ E_{B_i}(m_i)) \times (Y_1 \cdots Y_k)^R) \\ &\rightarrow (g^{R_1}, (B_i \circ E_{B_i}(m_i)) \times (Y_2 \cdots Y_k)^{R_1}) \\ &\;\;\vdots \\ &\rightarrow (g^{R_{k-1}}, (B_i \circ E_{B_i}(m_i)) \times Y_k^{R_{k-1}}) \\ &\rightarrow B_i \circ E_{B_i}(m_i) \end{aligned}$$

Note that $|(C_{0i}, C_{1i})| = 2 \times |q|$. Thus, the proposed anonymous channel has no problem of the ciphertext length expansion. It is also easy to see that, if there exists at least one honest S_i, the correspondence between A_i and B_i is kept secret from any adversary.

4 Proposed Election Scheme

The proposed anonymous channel of Sect. 3 can be directly applied to Chaum's election scheme in subsection 2.3. Then, the communication complexity is improved automatically.

However, the Chaum's election scheme has a problem of fairness as mentioned in the Introduction. That is, suppose that only V_1 is disturbed by S_k. Then, from the final list on the public board, everyone knows that some vote has been disrupted. However, at the same time, everyone knows $\{V_2, \ldots, V_n\}$. This information (for example, the number of yes votes and that of no votes) will affect the re-election greatly.

Let's study this problem more in detail. For simplicity, suppose that each voter P_i is honest. (It is clear that P_i cannot vote more than one vote in Chaum's scheme.) Consider the following two events.(We assume that there are some undisrupted votes.)

Event 1 : Some vote cannot be recovered.

Event 2 : Some undisrupted vote is made public.

Define P_d as follows.

$$P_d \triangleq P_r[\text{ Event } 2 \mid \text{ Event } 1 \text{ }].$$

In the Chaum's scheme, if S_k behaves as above, then always $P_d = 1$.

This section will present an election scheme such that P_d is negligibly small. We gives a high level description of the proposed election scheme in this section. The details will be given in the next section. The proposed election scheme consists of three phases as Chaum's scheme of 2.3 does. Our registration phase and claiming phase are the same as those of Chaum's scheme. In what follows, we will show our voting phase protocol. In addition to $S_1, \ldots, S_k$, we use S_0 whose role is to flip a coin. (Instead of S_0, we can use a collective coin flipping protocol. Such S_0 or a coin flipping protocol is also necessary in Benaloh's election scheme [4].) In this protocol, we use a variation of the anonymous channel proposed in Sect. 3.

4.1 Proposed Voting Phase Protocol (1)

First, we will present our voting phase protocol which achieves that $P_d \leq 1/2$.

Step 1. Each P_i chooses two random numbers R_{i1} and R_{i2} such that

$$V_i = R_{i1} \oplus R_{i2}, \tag{3}$$

where $\oplus$ denotes bitwise exclusive OR.

Step 2. Each P_i sends the ciphertexts of $R_{i1} \circ 0^l$ and $R_{i2} \circ 0^l$ to $S_1 \sim S_k$. (A group public key cryptosystem given in 5.1 is used.)

Step 3. After the deadline of the voting period, $S_1 \sim S_k$ shuffles the ciphertexts of ($(R_{i1} \circ 0^l), (R_{i2} \circ 0^l)$) in secret.

Step 4. At this moment, we have a secretly shuffled list of ciphertexts of $((\hat{R}_{11} \circ 0^l), (\hat{R}_{12} \circ 0^l)), ((\hat{R}_{21} \circ 0^l), (\hat{R}_{22} \circ 0^l)), \ldots$.
For each i, one of $\hat{R}_{i1} \circ 0^l$ and $\hat{R}_{i2} \circ 0^l$ is randomly chosen and made open. More precisely, S_0 flips a coin for each i. If the coin is head, $S_1 \sim S_k$ decrypt the ciphertext of $\hat{R}_{i1} \circ 0^l$ and make it open. Otherwise, $\hat{R}_{i2} \circ 0^l$ is made open.

Step 5. Everyone checks the form of 0^l of the decrypted pieces (in the same way as step 8 of the protocol in 2.3). If some disruption is detected, the protocol stops.

Step 6. Otherwise, for each i, the remained pieces are made open. Then, the form of 0^l is checked. (The same check as step 8 in 2.3 is done.)

Step 7. For each i such that no disruption is detected for both pieces, V_i is obtained from $R_{i1} \circ 0^l$ and $R_{i2} \circ 0^l$ by using eq. (3).

Remark. Voter's identity checking is done in the same way as in Chaum's election scheme by using digital signatures.

Example 1. Let the number of voters be 3.
[Step 1 and 2.] (Voting)

$$\begin{array}{l} \textbf{voter } 1\ (R_{11}, R_{12}) \Rightarrow \textit{anonymous channel} \\ \textbf{voter } 2\ (R_{21}, R_{22}) \Rightarrow \textit{anonymous channel} \\ \textbf{voter } 3\ (R_{31}, R_{32}) \Rightarrow \textit{anonymous channel} \end{array}$$

[Step 3.] (Shuffling)

$$(\boxed{R_{31}}, \boxed{R_{32}}), (\boxed{R_{11}}, \boxed{R_{12}}), (\boxed{R_{21}}, \boxed{R_{22}})$$

[Step 4.] (Cut and Choose)

$$(R_{31}, \boxed{R_{32}}), (\boxed{R_{11}}, R_{12}), (R_{21}, \boxed{R_{22}})$$

R_{31}, R_{12} and R_{21} are made open.

[Step 6.] (Opening)

$$(R_{31}, R_{32}), (R_{11}, R_{12}), (R_{21}, R_{22})$$

R_{32}, R_{11} and R_{22} are made open.

[Step 7.] (Reconstruction)

$$\hat{V}_1 = R_{31} \oplus R_{32}$$

$$\hat{V}_2 = R_{11} \oplus R_{12}$$

$$\hat{V}_3 = R_{21} \oplus R_{22}$$

Theorem 1. *In the above protocol,* $P_{d1} \leq 1/2$.

Proof. Note that
$P_d = P_r\{$ No disruption is detected at Step 5 | Event 1 $\}$.
Event 1 occurs if some dishonest S_j has rewritten at least one element of $\{R_{i1} \circ 0^l\} \cup \{R_{i2} \circ 0^l\}$. Suppose that one element of $\{R_{i1} \circ 0^l\} \cup \{R_{i2} \circ 0^l\}$ is disrupted. Then, this cheating is detected at Step 4 and Step 5 with probability 1/2. □

4.2 Proposed Voting Phase Protocol (2)

Next, we will show our voting phase protocol which achieves that $P_d \leq 1/2^h$, where h is a security parameter.

Step 1. Each P_i chooses h pairs of random numbers $(R_{11}, R_{21}), \ldots, (R_{1h}, R_{2h})$ such that

$$V_i = R_{11} \oplus R_{21} = \cdots = R_{1h} \oplus R_{2h}, \tag{4}$$

where $\oplus$ denotes bitwise exclusive OR.

Step 2. Each P_i sends the ciphertexts of

$$((R^i_{11} \circ 0^l, R^i_{21} \circ 0^l), \ldots, (R^i_{1h} \circ 0^l, R^i_{2h} \circ 0^l))$$

to $S_1 \sim S_k$.

Step 3. The anonymous channel shuffles

$$\{(R^i_{11} \circ 0^l, R^i_{21} \circ 0^l), \ldots, (R^i_{1h} \circ 0^l, R^i_{2h} \circ 0^l)\}$$

in secret.

Step 4. For each j, one of $R^i_{1j} \circ 0^l$ and $R^i_{2j} \circ 0^l$ is randomly chosen and made open (for $\forall i$).

Step 5. Check the form of 0^l of the opened pieces as Step 8 in 2.3. If some disruption is detected, stop.

Step 6. Open all of $R^i_{1j} \circ 0^l \cup R^i_{2j} \circ 0^l$. Check the form of 0^l.

Step 7. Let

$$G(i) \triangleq \{j \mid \text{No disruption is detected both for } R^i_{1j} \circ 0^l \text{ and } R^i_{2j} \circ 0^l\}.$$

$$J(i) \triangleq \mathbf{min} G(i) \text{ if } \mid G(i) \mid \geq 1.$$

V_i is reconstructed as $R^i_{1J(j)} \oplus R^i_{2J(i)}$.

Example 2. [Step 1 and 2.] (Voting)

voter 1 $(R^1_{11}, R^1_{21}), \ldots\ldots, (R^1_{1h}, R^1_{2h}) \Rightarrow$ *anonymous channel*
voter 2 $(R^2_{11}, R^2_{21}), \ldots\ldots, (R^2_{1h}, R^2_{2h}) \Rightarrow$ *anonymous channel*
voter 3 $(R^3_{11}, R^3_{21}), \ldots\ldots, (R^3_{1h}, R^3_{2h}) \Rightarrow$ *anonymous channel*

[Step 3.] (Shuffling)

$$\begin{array}{l}
(\boxed{R^3_{11}}, \boxed{R^3_{21}}), \ldots\ldots\ldots, (\boxed{R^3_{1h}}, \boxed{R^3_{2h}}) \\
(\boxed{R^1_{11}}, \boxed{R^1_{21}}), \ldots\ldots\ldots, (\boxed{R^1_{1h}}, \boxed{R^1_{2h}}) \\
(\boxed{R^2_{11}}, \boxed{R^2_{21}}), \ldots\ldots\ldots, (\boxed{R^2_{1h}}, \boxed{R^2_{2h}})
\end{array}$$

[Step 4.] (Cut and Choose)

$$(R^3_{11}, \boxed{R^3_{21}}), \ldots\ldots\ldots, (\boxed{R^3_{1h}}, R^3_{2h})$$
$$(\boxed{R^1_{11}}, R^1_{21}), \ldots\ldots\ldots, (R^1_{1h}, \boxed{R^1_{2h}})$$
$$(\boxed{R^2_{11}}, R^2_{21}), \ldots\ldots\ldots, (\boxed{R^2_{1h}}, R^2_{2h})$$

[Step 6.] (Opening)

$$(R^3_{11}, R^3_{21})$$
$$(\mathbf{Error}, R^1_{21}) \Rightarrow (R^1_{12}, \mathbf{Error}) \Rightarrow (R^1_{13}, R^1_{23})$$
$$(\mathbf{Error}, R^2_{21}) \Rightarrow (R^2_{12}, R^2_{22})$$

Error means that some disruption is detected.

[Step 7.] (Reconstruction)

$$\hat{V}_1 = R^3_{11} \oplus R^3_{21}$$
$$\hat{V}_2 = R^1_{13} \oplus R^1_{23}$$
$$\hat{V}_3 = R^2_{12} \oplus R^2_{22}$$

Theorem 2. *In the above protocol,* $P_d \leq 1/2^h$.

Proof. Note that

$P_d = P_r\{$ no disruption is detected at Step 5 | there exists V_a such that both or one of R^a_{1i} and R^a_{2i} is disrupted for $1 \leq \forall i \leq h\}$.

Suppose that there exists V_a such that one of R^a_{1i} and R^a_{2i} is disrupted for $1 \leq \forall i \leq h$. This disruption is detected at Step 5 with probability $1/2^h$. □

5 Full Description of the Proposed Election Scheme

The proposed election scheme uses a modification of the anonymous channel given in Sect.3. The modified anonymous channel makes use of a group public key cryptosystem [5].

5.1 Group Public Key Cryptosystem

Remember that we have used

(Common public information) p, g, c
(Secret key of S_i) $X_i \in \{1, \ldots, q-1\}$
(Public key of S_i) $Y_i\ (= g^{X_i} \bmod q)$

in Sect.3. This setting is the same as the group public key cryptosystem in [5]. The public key of the group is $Y_1 \cdots Y_k$. All S_i have to cooperate to decrypt ciphertexts.

Let m be a plaintext. The ciphertext of the group public key cryptosystem is given by

$$E(m,r) \triangleq (g^r, m_i \times (Y_1 \cdots Y_k)^r) \bmod q,$$

where r is a random number. The decryption protocol is given as follows. Let

$$a \triangleq g^r \bmod q,$$
$$b \triangleq m \times (Y_1 \cdots Y_k)^r \bmod q.$$

[Decryption Protocol]

Step 1. Each S_i computes $Z_i = a^{X_i} (= (g^r)^{X_i} = Y_i^r \bmod q)$ and makes Z_i open.
Step 2. Everyone computes
$b/(Z_1 \cdots Z_k) = m \times (Y_1 \cdots Y_k)^r/(Z_1 \cdots Z_k) = m.$

5.2 One more tool

Let

$$h(a,b,e) \triangleq (a \times g^e, b \times (Y_1 \cdots Y_k)^e) \bmod q.$$

Lemma 3. *If* $(a,b) = E(m,r)$, *then* $h(a,b,e) = E(m, r+e)$.

The proof is immediate.

From this Lemma 3, we see that applying h to $E(m,r)$ successively several times yields $E(m,x)$ for some x.

5.3 Modified Anonymous Channel

We show a modification of the anonymous channel shown in Sect. 3, which will be used in the next subsection.

Step 1. Each sender A_i writes $E(B_i \circ E_{B_i}(m_i), r_i)$ on the public board, where r_i is a random number.
Step 2. S_1 chooses random numbers $e_1, e_2, \ldots,$ and computes

$$h(E(B_i \circ E_{B_i}(m_i), r_i), e_i) = E(B_i \circ E_{B_i}(m_i), r_i + e_i)$$

for each i. S_1 writes $\{E(B_i \circ E_{B_i}(m_i), r_i + e_i)\}$ on the public board in a lexicographical order.
Step 3. $S_2 \sim S_k$ do the same job in sequence. Then, we have a list of $\{E(B_i \circ E_{B_i}(m_i), x_i)\}$ in a lexicographical order on the public board, where x_i is a random number.
Step 4. $S_1 \sim S_k$ obtain $\{B_i \circ E_{B_i}(m_i)\}$ by executing the decryption algorithm in 5.1.
If at least one S_j is honest, nobody knows the correspondence between A_i and B_i.

5.4 Details of the Election Scheme in 4.1

We show the details of the election scheme shown in 4.1. The details of the protocol of 4.2 will be obtained similarly.

Step 1. Each voter P_i chooses two random numbers R_{i1} and R_{i2} such that

$$V_i = R_{i1} \oplus R_{i2}.$$

Step 2. Each P_i chooses r_{i1} and r_{i2} randomly. He computes

$$(a_{i1}, b_{i1}) = E(K_i \circ K_i^{-1}(R_{i1} \circ 0^l), r_{i1})$$

$$(a_{i2}, b_{i2}) = E(K_i \circ K_i^{-1}(R_{i2} \circ 0^l), r_{i2})$$

and writes them on the public board.
At this moment, there is a list on the public board such that

$$((a_{11}, b_{11}), (a_{12}, b_{12})), ((a_{21}, b_{21}), (a_{22}, b_{22})), \ldots.$$

Step 3. For $i = 1, \ldots, k$, do the following in sequence.
Let the latest list on the public board be

$$((\alpha_{11}, \beta_{11}), (\alpha_{12}, \beta_{12})), ((\alpha_{21}, \beta_{21}), (\alpha_{22}, \beta_{22})), \ldots.$$

S_i computes

$$(\hat{\alpha}_{j1}, \hat{\beta}_{j1}) = h(\alpha_{j1}, \beta_{j1}, e_{j1})$$

$$(\hat{\alpha}_{j2}, \hat{\beta}_{j2}) = h(\alpha_{j2}, \beta_{j2}, e_{j2})$$

for each j, where e_{j1} and e_{j2} are random numbers. S_i writes

$$\{((\hat{\alpha}_{j1}, \hat{\beta}_{j1}), (\hat{\alpha}_{j2}, \hat{\beta}_{j2}))\}$$

on the public board in a lexicographical order.
Step 4. Let the list on the public board at this moment be

$$((\hat{\alpha}_{11}, \hat{\beta}_{11}), (\hat{\alpha}_{12}, \hat{\beta}_{12})), ((\hat{\alpha}_{21}, \hat{\beta}_{21}), (\hat{\alpha}_{22}, \hat{\beta}_{22})), \ldots.$$

S_0 chooses a random bit d_i for each i. By using the decryption protocol given in 5.1,

$$S_1, \ldots, S_k \text{ decrypt } (\hat{\alpha}_{i1}, \hat{\beta}_{i1})(= E(\hat{K}_i \circ \hat{K}_i^{-1}(\hat{R}_{i1} \circ 0^l), x_{i1})), \text{ if } d_i = 0$$

$S_1, \ldots, S_k$ decrypt $(\hat{\alpha}_{i2}, \hat{\beta}_{i2})(= E(\hat{K}_i \circ \hat{K}_i^{-1}(R_{i2} \circ 0^l), x_{i2})$, if $d_i = 1$.

Step 5. Everyone checks the form of 0^l of the decrypted pieces (in the same way as Step 8 of the protocol in 2.3). If some disruption is detected, the protocol stops.

Step 6. Otherwise, for each i, the remained pieces are made open. Then, the form of 0^l is checked. (The same check as Step 8 in 2.3 is done.)

Step 7. For each i such that no disruption is detected for both pieces, V_i is obtained from $R_{i1} \circ 0^l$ and $R_{i2} \circ 0^l$ by using eq. (3).

6 Conclusion

First, we have presented an efficient computationally secure anonymous channel which has no problem of ciphertext length expansion. The length is irrelevant to the number of MIXes. It improves the efficiency of Chaum's election scheme based on the MIX net automatically. Second, we have shown an election scheme which satisfies the fairness. That is, if some vote is disrupted, no one obtains any information about all the other votes. Each voter sends $O(nk)$ bits so that the probability of the fairness is $1 - 2^{-k}$, where n is the bit length of the ciphertext.

References

1. Chaum, D.L.: Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms. Communications of the ACM, Vol. 24, No.2, (1981), 84–88
2. Chaum, D.L.: Elections with Unconditionally-Secret Ballots and Disruption Equivalent to Breaking RSA. Advance in Cryptology — EUROCRYPT'89, (1989), 177–182
3. Chaum, D.L. : The Dining Cryptographers Problem: Unconditional sender and Recipient Untraceability. Journal of Cryptology, Vol.1, No.1, (1988), 65–75
4. Benaloh, J.C. : Secret Sharing Homomorphisms: Keeping Shares of a Secret Secret. Advance in Cryptology — CRYPTO'86, (1986), 251–260
5. Desmedt, Y., Frankel, Y. : Threshold cryptosystems. Advance in Cryptology — CRYPTO'89, (1990), 307–315